



Department of ELECTRONICS AND TELECOMMUNICATIONS
Master Degree in Mechatronic Engineering

Academic year: 2022-2023

Satellite Navigation Systems - Fabio Dovis

Satellite Navigation Systems Laboratory Experiences

Work team

Pighini Alessandro - 304097
Nazzaro Davide - 296232

Contents

LAB01: Raw GNSS measurements in Android devices	3
Objective	3
Experimental raw measurements analysis	3
Comparison of data taken from other locations	3
Filtered data	6
LAB02: State estimation	9
Number of visible satellites and measured pseudoranges	9
User state estimation	9
Obtained solution comparison	10
Performance comparison between LMS and WLMS results	10
LAB03: Spreading codes	16
GPS codes properties	16
Generation of Galileo codes	17
Generation of local replica code and codes in time and frequency	17
LAB04: State estimation	20
Generation of the GNSS carrier and visualization of signals in time and frequency	20
Correlations	20
Serial acquisition	21
LAB05: Acquisition of real signals	24
Parallel acquisition	24
The effect of noise	24
Real GNSS Signals	26
LAB06: Tracking	29
Correlators and discriminator	29
Multipath	31

LAB01: GNSS in mobile devices

Objective

The goal of this experience is to become confident with the GNSS measurements by means of Android devices. In this laboratory experience a previous collected data will be analyzed through a MATLAB script capable of reading GnssLogger outputs. In a following step the measurements will be taken by us, using our Android devices in order to evaluate and compare them in different conditions.

Experimental raw measurements analysis

In order to describe the results obtained from the provided Matlab script, the data collected in the courtyard of "Classrooms I" at the Polytechnic University of Turin will be used. The environment taken to perform this data collection can be categorized as an urban canyon, although a considerable portion of the sky was available to capture signals that were not reflected.

Pseudoranges as a function of time change, in seconds, are shown in Figure 1. The satellite IDs are labeled with different numbers (visible to the right of the graph). The changes in the pseudoranges from a given initial value are also represented. It is possible to see how some pseudoranges increase, while others decrease; this is due to the fact that some satellites approach the receiver, while others move away from it. At the bottom of the figure is the hardware clock discontinuity.

In figure 2 it can be seen the derivative of the pseudorange over time. It is the rate at which the pseudoranges grow or decrease over time. It is also possible to see the fluctuations due to noise.

In figure 3 several lines are depicted, one for each satellite from which our receiver picked up data, depicting the signal to noise ratio of each satellite. From this graph, it is possible to determine which satellite is more affected by the noise.

Figure 4 depicts the median value and the positioning state, the horizontal velocity (in this case the data were collected at a stationary position, so you can see that the velocity is practically always zero) and the Horizontal Dilution Of Position, affected by the number of visible satellites.

Figure 5 represents a collection of states estimated by the receiver and the clock bias.

Comparison of data taken from other locations

During the laboratory exercise, additional measurements were performed in other environments in order to compare the performance of the receiver due to the change in external conditions. Of particular interest were the measurements taken in a position in which a small part of the sky was visible (staircase leading to classrooms I), in fact from figure 6 it is possible to notice that the number of visible satellites is reduced.

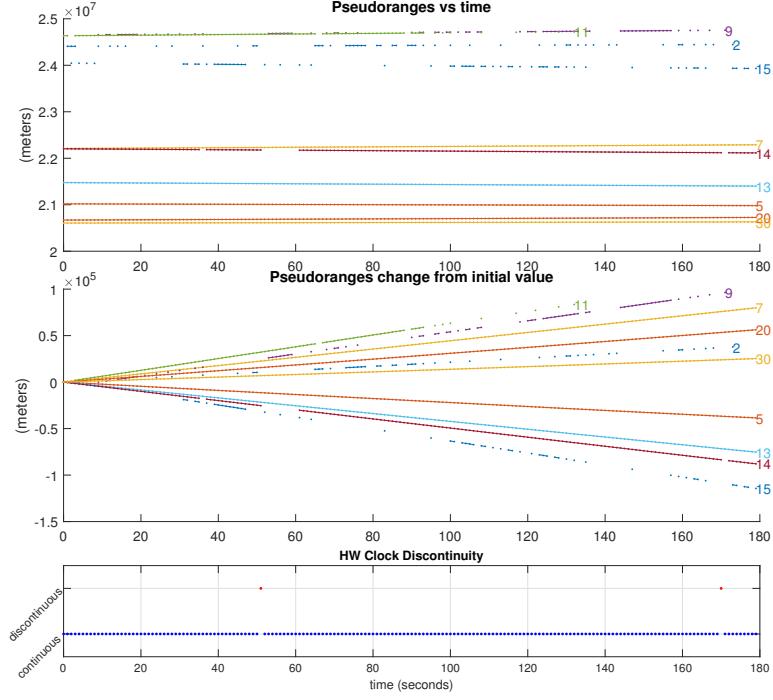


Figure 1. Pseudoranges measurements

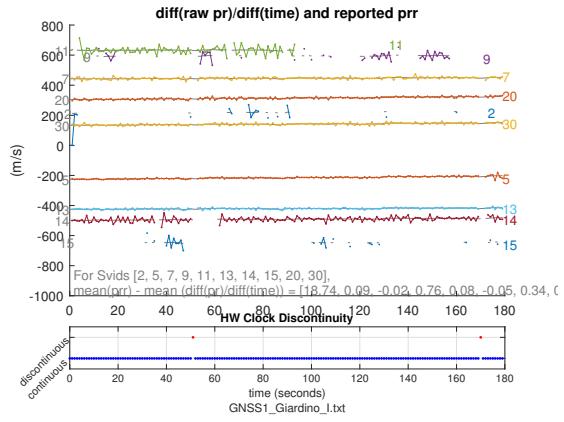


Figure 2. Derivative of the raw pseudorange over time

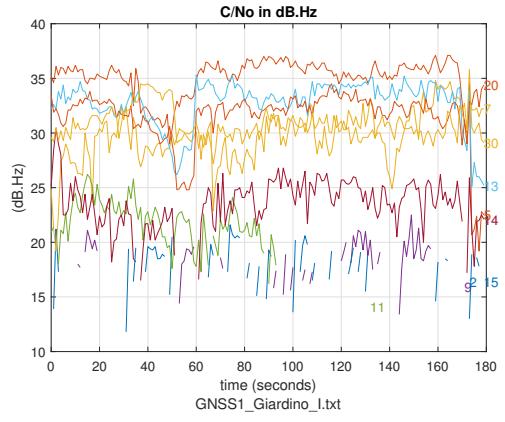


Figure 3. Signal to noise ratio (C/N_0) of every visible satellite

Another place where a different type of measurement was appreciated was the construction site in front of classrooms T, where half of the sky was directly visible from the receiver, while the other half was covered by a reflective surface. In fact, in this last position it is possible, in Figure 7 to notice the higher presence of signals with lower C/N_0 (due to the multipath effects)

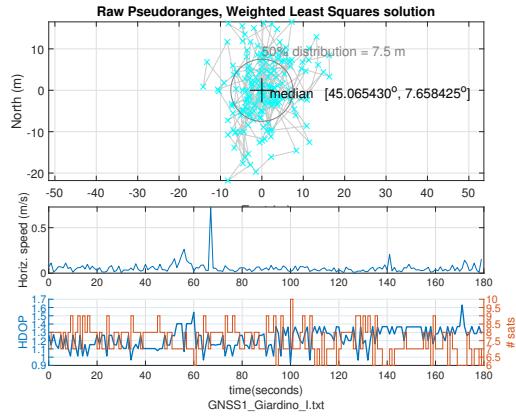


Figure 4. Weighted Least Squares solution

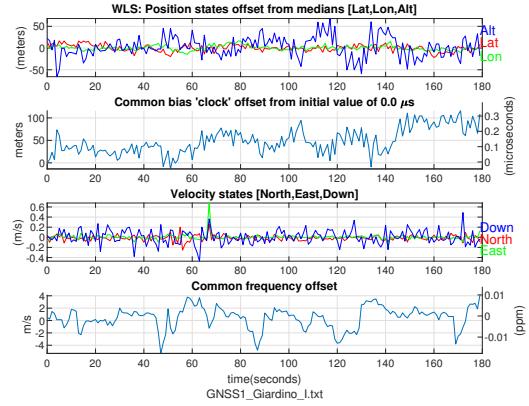


Figure 5. Position and velocity estimated states

with respect to the measurement taken in the courtyard.

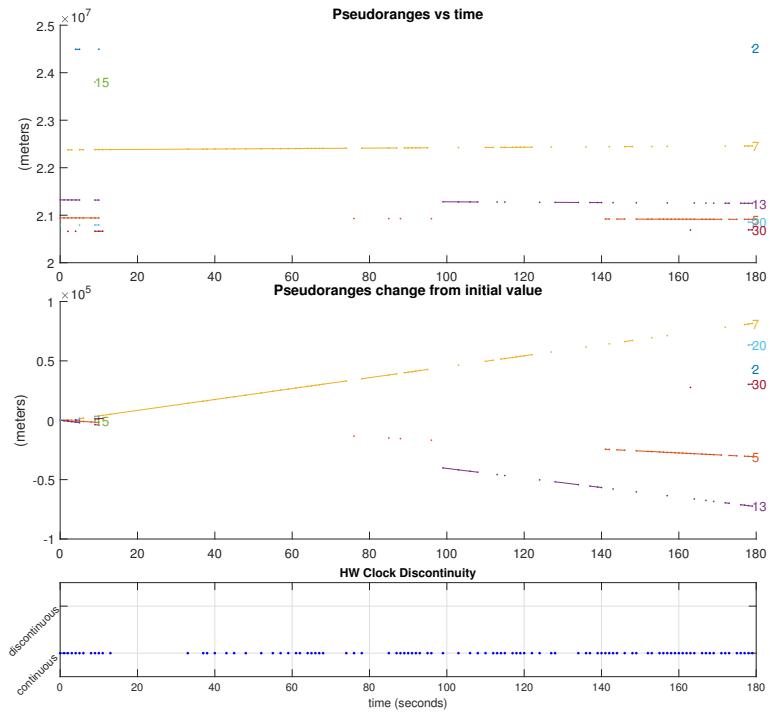


Figure 6. Pseudoranges measurements

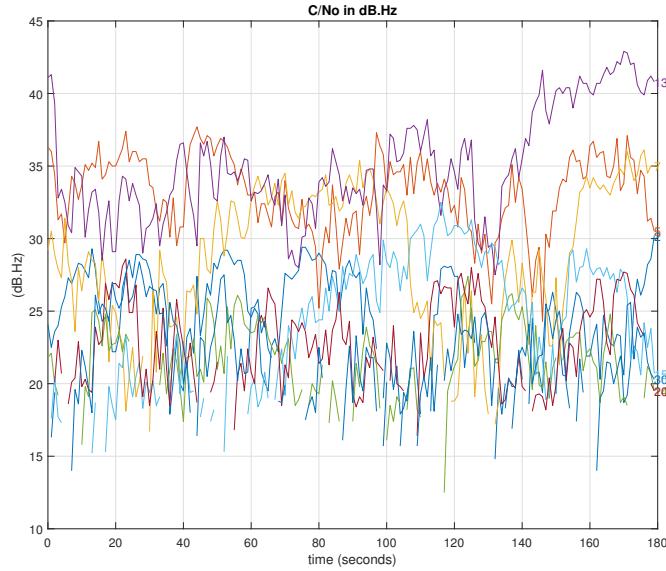


Figure 7. Pseudoranges measurements

Filtered data

Through the provided script it is possible to apply filters to refine the analysis of the measures. In this case filters were applied to the measurements taken in the courtyard of classrooms "I". Figure 8 shows the C/N_0 obtained from the data filtered so as to have $C/N_0 > 30$. It is also possible to select the constellation from which to take the measurements: Figure 9 shows the pseudoranges taken by Galileo. Or it is possible to isolate the data coming from a single satellite (Figure 10). Inserting a multipath filter improves performance because the signals that are filtered out would affect PVT estimation. One filter that could be implemented is one capable of selecting satellites by altitude, so as to eliminate those with higher GDOP. The drawback is to decrease the number of visible satellites.

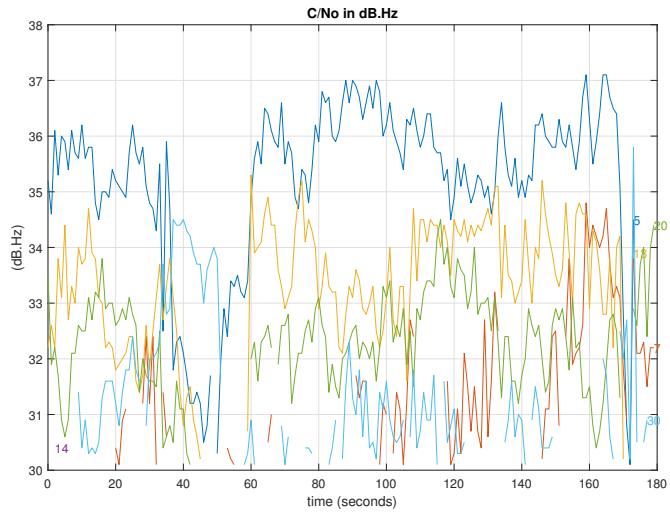


Figure 8. Filtered C/N_0

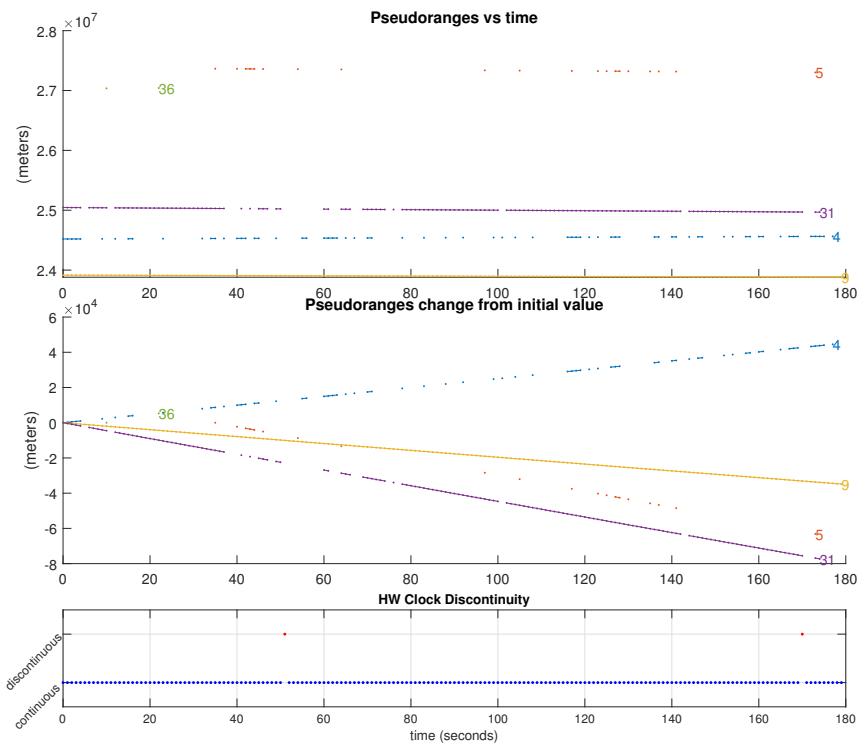


Figure 9. Filtered constellation pseudoranges

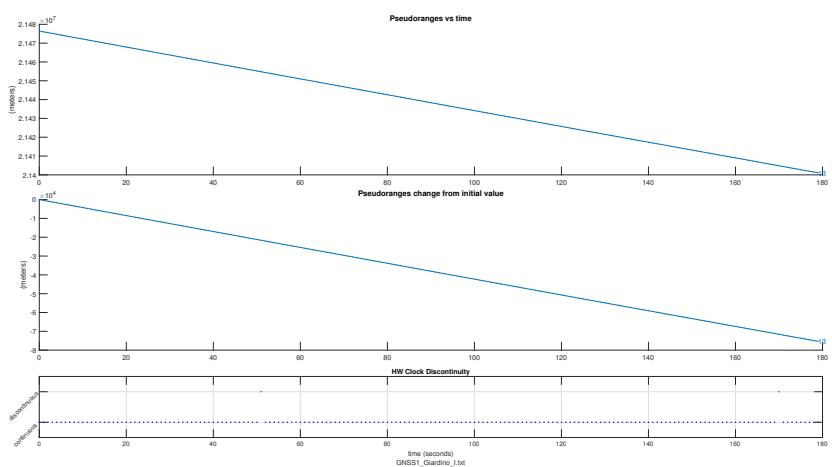


Figure 10. Single satellite pseudoranges

LAB02: PVT evaluation

The objective of this laboratory experience is to perform a state estimation, from a provided dataset, by means of a Matlab script.

Number of visible satellites and measured pseudoranges

The first task of the script is to plot the number of visible satellites and the measurements of the pseudoranges over time for every constellation. In Figure 11, 13, 15, 17 the number of visible satellites along the time are plotted for every constellation. Figure 12, 14, 16, 18 instead, show the measured pseudoranges along the time.

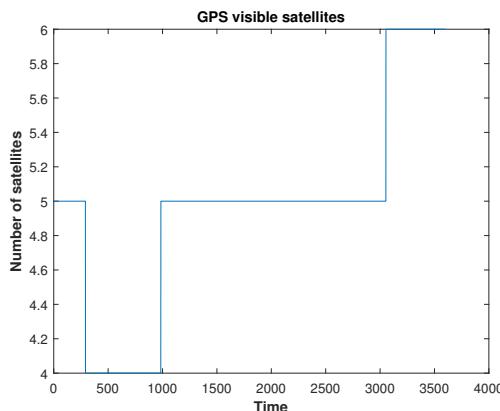


Figure 11. GPS constellation visible satellites

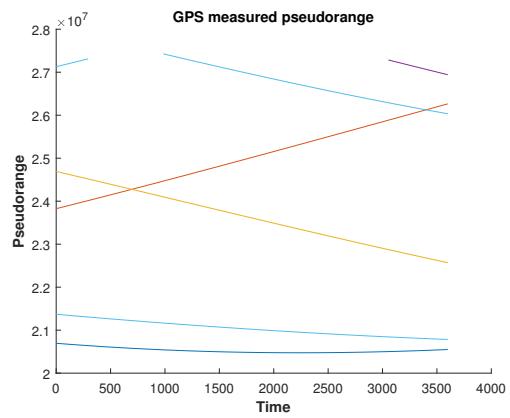


Figure 12. GPS pseudoranges measurement

User state estimation

The second task consists in providing a Least Mean Square algorithm in order to estimate user position. The first provided dataset has been chosen and, in order to perform the analysis, the GALILEO constellation has been selected.

In Figures 20 and 21 the position estimation has been plotted in a two-dimensional and three-dimensional plots. Converting the estimate solution the city of cape town came out (Figure 19).

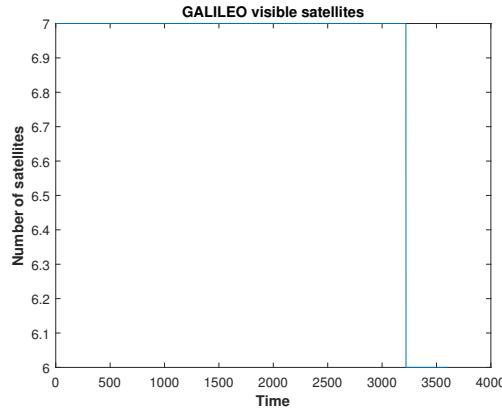


Figure 13. GALILEO constellation visible satellites

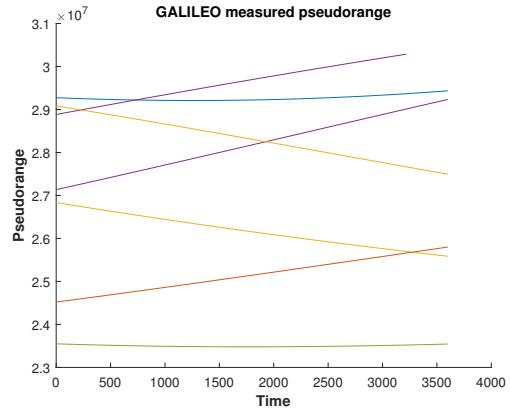


Figure 14. GALILEO pseudoranges measurement

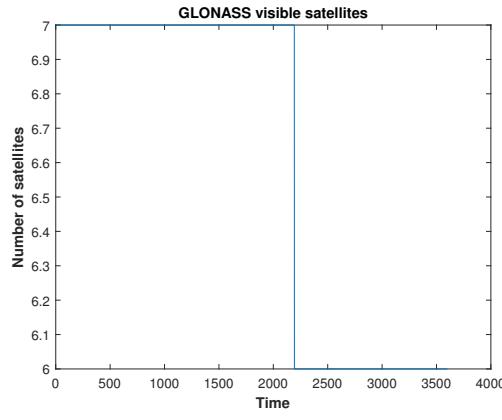


Figure 15. GLONASS constellation visible satellites

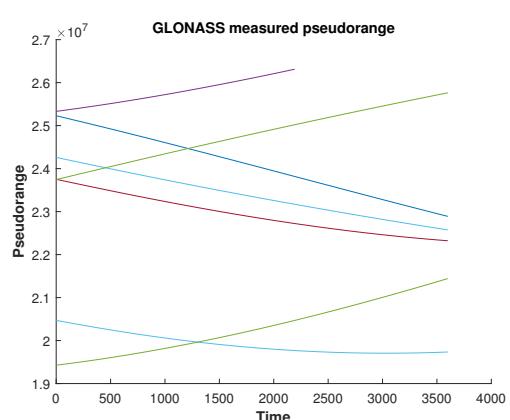


Figure 16. GLONASS pseudoranges measurement

Obtained solution comparison

The aim of this third task is to compare the quality of the obtained solution for different datasets and constellations.

Among all datasets it is possible to see that the average position error is higher for BEIDOU constellation. This could be due to a poorer satellite geometric distribution.

Figures 22, 23, 24, 25 show the comparison between standard deviation of some of the given datasets.

In figures 26, 27 show the behaviour of the error along time. As it can be seen, with the increasing of the satellites number, the medium error decreases.

Performance comparison between LMS and WLMS results

This section provides a comparison bet between LMS and WLMS results, performed on realistic-UERE dataset 1, as shown in figure 28 and 29. For the comparison the GPS constellation has

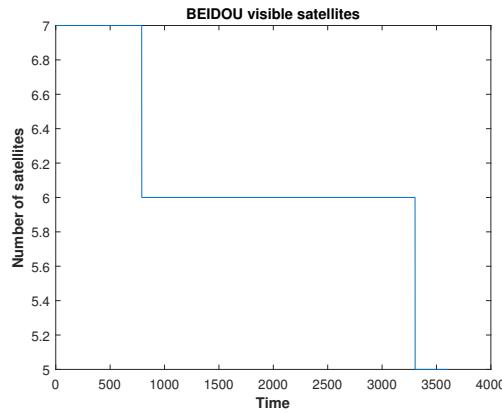


Figure 17. BEIDOU constellation visible satellites

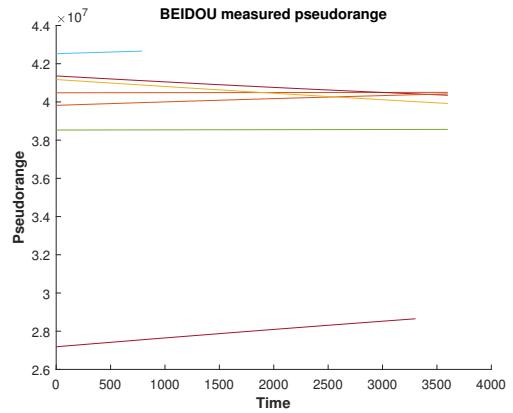


Figure 18. BEIDOU pseudoranges measurement

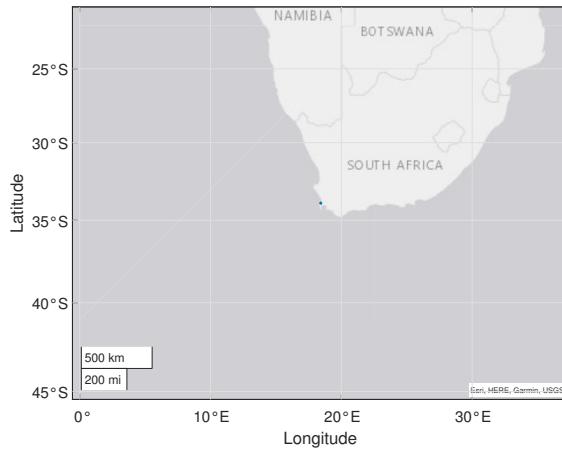


Figure 19. Position coordinate

been chosen.

Tables 1, 2, 3 and 4 show the estimated σ_{UERE} for every available PRN signal of each constellation.

Figure 31 shows the comparison between the error ellipsoids for LMS and WLMS solutions. As it can be seen, the WLMS mean value doesn't change a lot, but the standard deviation is reduced.

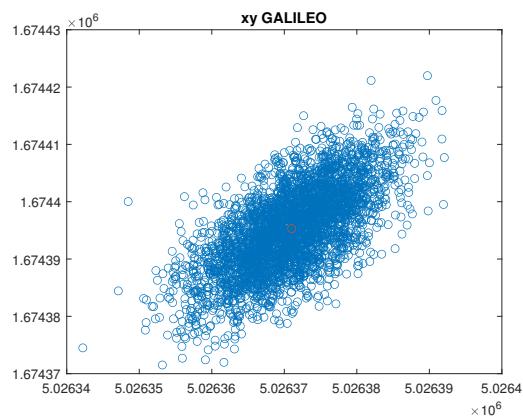


Figure 20. GALILEO 2D estimation

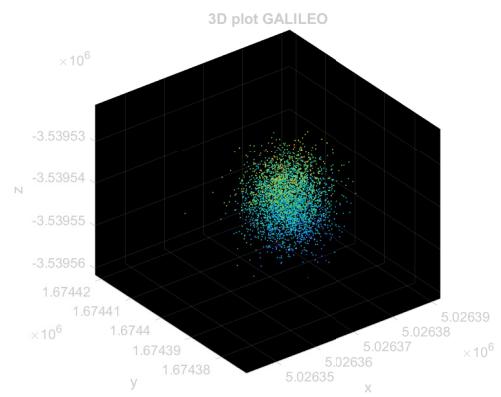


Figure 21. GALILEO 3D estimation

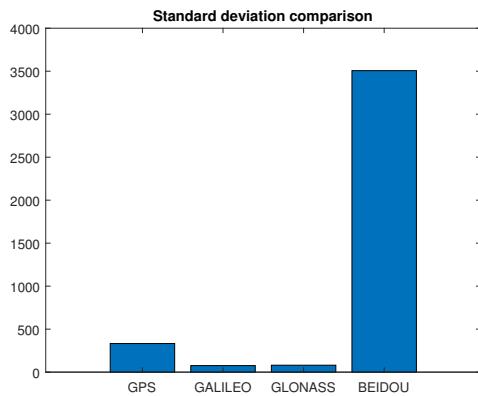


Figure 22. Dataset 1 standard deviation comparison

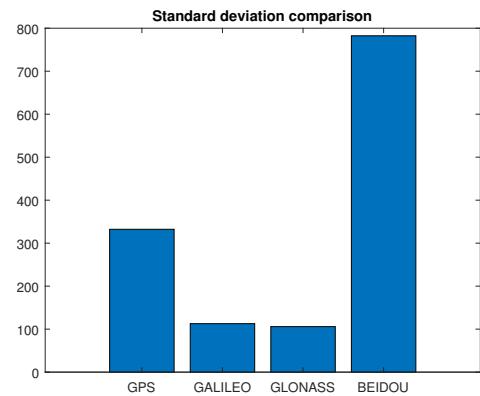


Figure 23. Dataset 1 standard deviation comparison

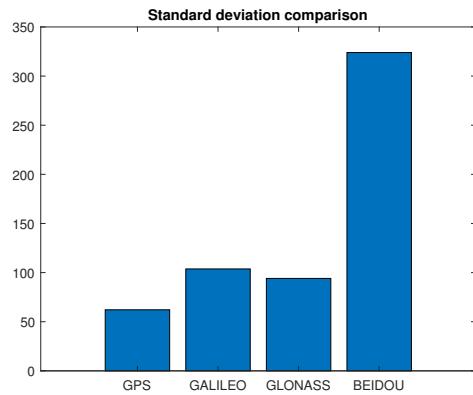


Figure 24. Dataset 4 standard deviation comparison

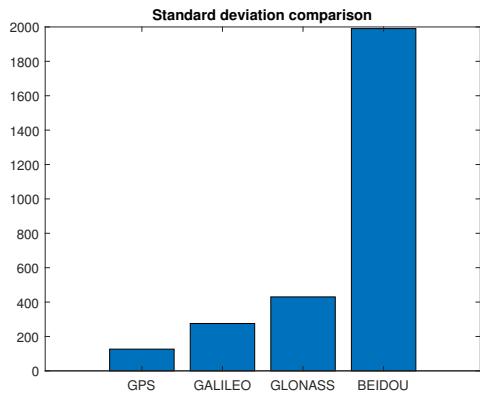


Figure 25. Dataset 6 standard deviation comparison

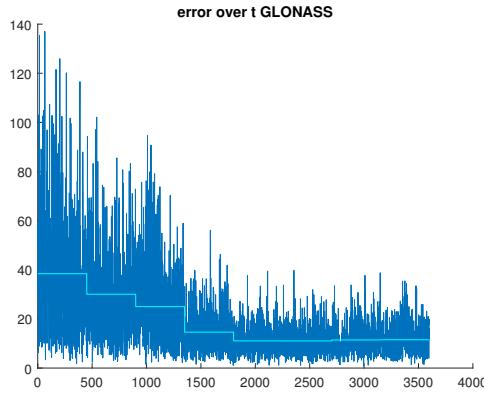


Figure 26. GLONASS constellation error over time (dataset 6)

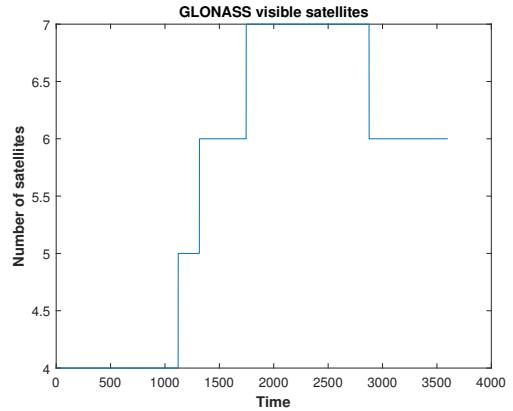


Figure 27. Number of GLONASS visible satellites (dataset 6)

Table 1. σ_{UERE} for GPS signals

PRN	σ_{UERE}
2	6,0247
5	3,3796
10	14,1051
13	2,1577
15	1,6657
20	2,1095
21	2,7813
24	3,9664
29	2,8570
30	8,0198

Table 2. σ_{UERE} for GALILEO signals

PRN	σ_{UERE}
1	14,8763
3	3,7365
5	1,8510
9	2,8234
11	7,6853
18	3,3923
24	7,9007

Table 3. σ_{UERE} for GLONASS signals

PRN	σ_{UERE}
15	7,0426
19	6,3108
20	2,1254
21	4,6442
34	5,3558
37	6,6150
42	2,3685
44	2,7514
47	10,1114
51	6,9193
54	2,3709

Table 4. σ_{UERE} for BEIDOU signals

PRN	σ_{UERE}
2	7,1276
5	3,5678
7	9,9068
9	6,1315
10	7,8890

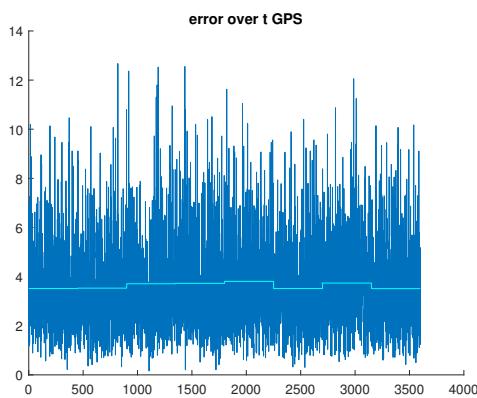


Figure 28. LMS error

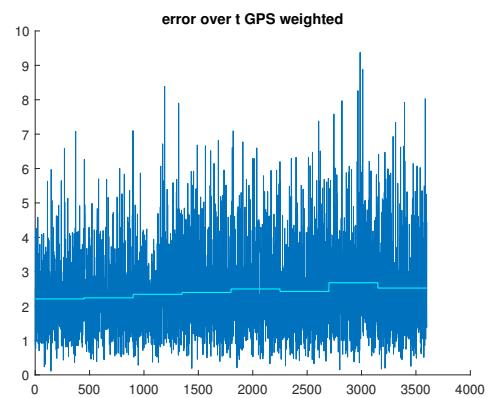


Figure 29. WLMS error

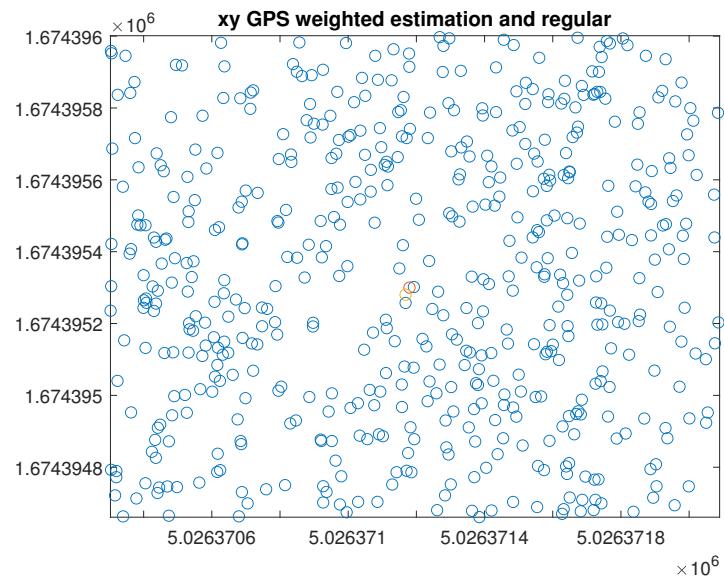


Figure 30. LMS and WLMS estimations

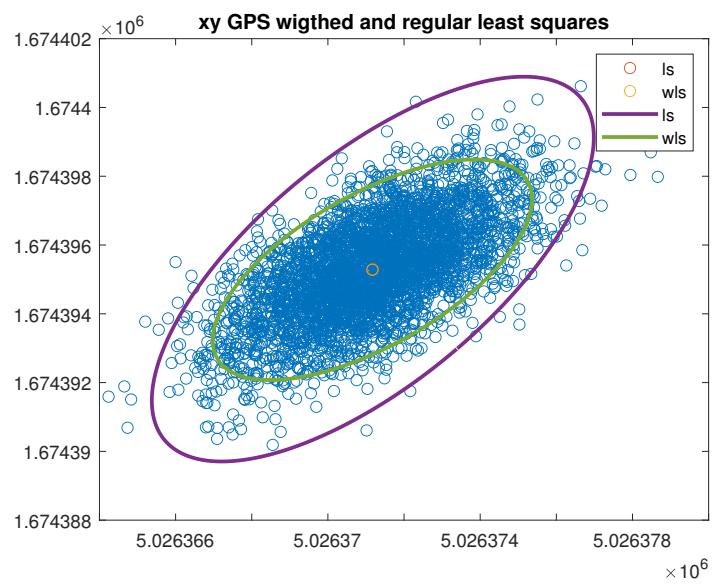


Figure 31. Error ellipsoid

LAB03: Spreading codes

The goal of this lab session is to implement a MATLAB set of functions able to generate the spreading codes and to prove their properties. The term *spreading* is related to the concept of CDM codes and it is useful because the receiver has to discriminate between the satellites.

In this lab experience there will be five incremental steps and in this chapter the results of will be shown. Both GPS L1 C/A and Galileo E1b signals will be considered.

GPS codes properties

After the first step, where the generation of GPS codes has been performed and the number of +1 and -1 have been verified thanks to the formula provided, normalized linear and circular auto-correlation and cross-correlation (between two different generated codes), has been performed. It was also verified that the code matched with the GPS ICD code provided by Table 3-IA. For example, PRN 1 corresponds to octal code 1440 (which is equal to 1 1 0 0 1 0 0 0 0 0 in binary code), and the first ten chips are: -1 -1 1 1 -1 1 1 1 1 1.

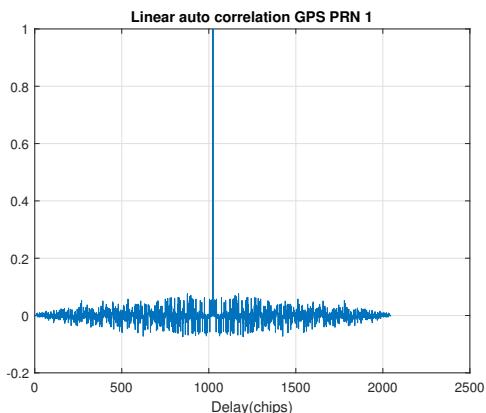


Figure 32. Linear auto-correlation of the PNR code n.1

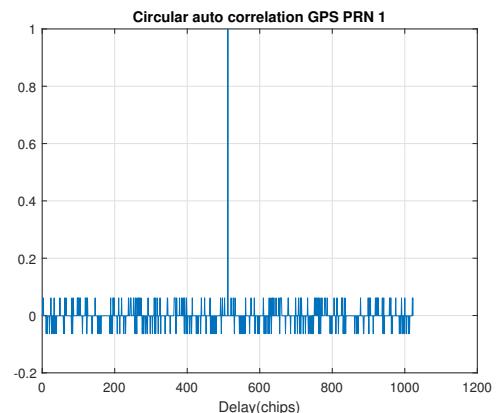


Figure 33. Circular auto-correlation of the PNR code n.1

Figures 32 and 33 show respectively the linear and circular auto-correlation of the PRN code n.1, while Figures 34 and 35 show the linear and circular cross-correlation between GPS PRN codes n.1 and n.2.

The normalized circular correlation function values are shown in Table 5.

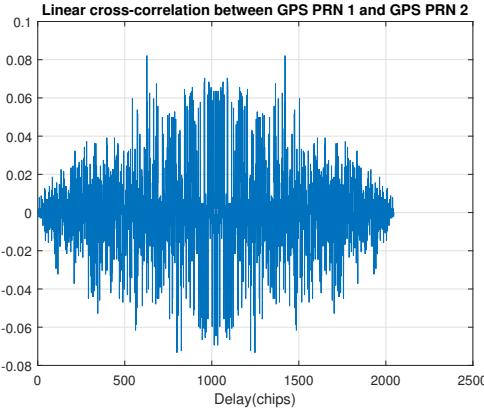


Figure 34. Linear cross-correlation between PNR code n.1 and n.2

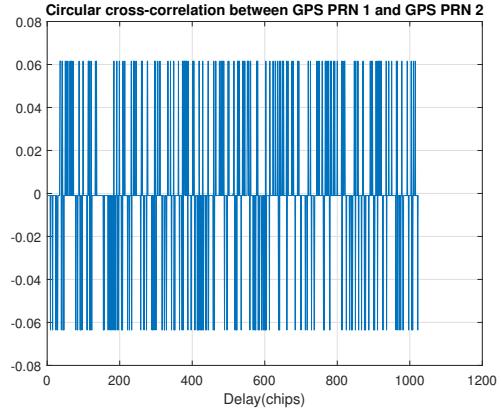


Figure 35. Circular cross-correlation between PNR code n.1 and n.2

Table 5. Circular auto and cross-correlation values

Circular auto-correlation	Circular cross-correlation	
1	1	\
$-\frac{1}{p}$	-0.0635	-0.0635
$-\frac{\beta(m)}{p}$	-0.0009	-0.0009
$\frac{\beta(m)-2}{p}$	0.0615	0.0615

Generation of Galileo codes

In the same way as in the previous step, plots of linear and circular self-correlation and cross-correlation are shown in Figures 36, 38, 37, 39.

The ratios in dB between the maximum amplitude of the cross-correlation auto-correlation function and its highest peak for GPS and GALILEO are respectively 12.1054 and 13.1968. GALILEO signal has a higher peak, which is let the signal to be easily detected with the presence of noise. It is possible to conclude, from the values of the ratios just highlighted, that the sequence of the GALILEO PRN is longer than the GPS one.

Generation of local replica code and codes in time and frequency

For the last two steps of this lab experience a local replica of the code has been generated. Furthermore the generated codes have been plotted in time and frequency domain. The number of samples per chip generated in this code is 16, the code duration in samples is equal to 16368, the code duration in chips is equal to 1023, while the code in seconds is 1ms. The spectral density is as expected, energy of the signal from GPS and Galileo are distributed so that they don't interfere (main lobe splitted in two side lobes for GALILEO).

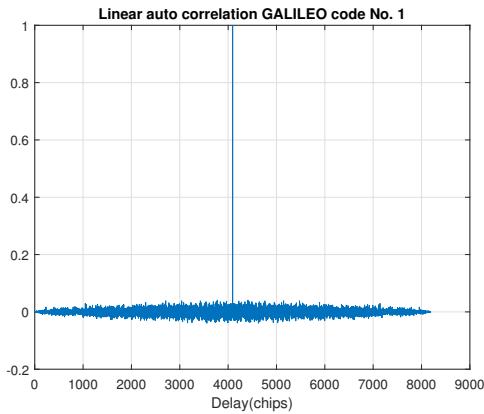


Figure 36. GALILEO linear auto-correlation of the PNR code n.1

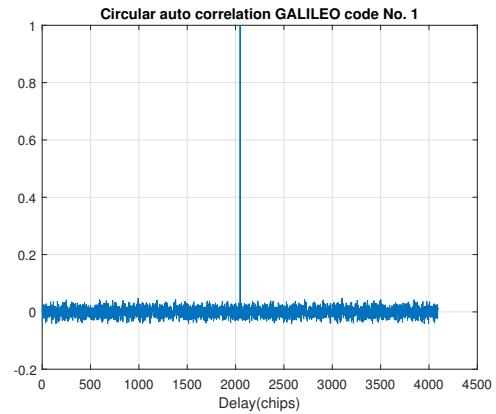


Figure 37. GALILEO Circular auto-correlation of the PNR code n.1

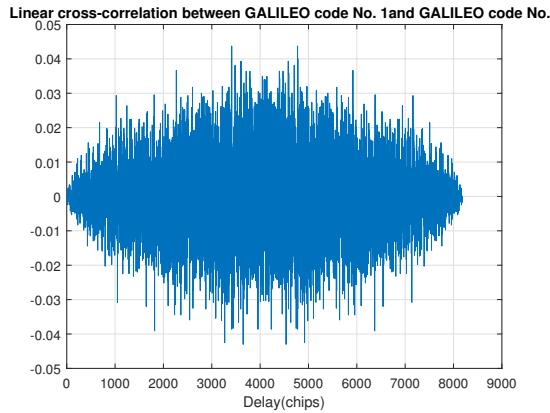


Figure 38. GALILEO linear cross-correlation between PNR code n.1 and n.2

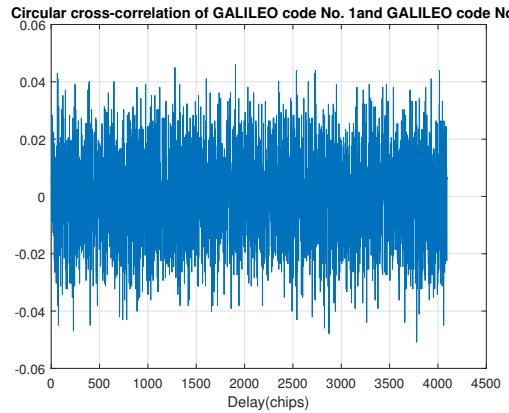


Figure 39. GALILEO circular cross-correlation between PNR code n.1 and n.2

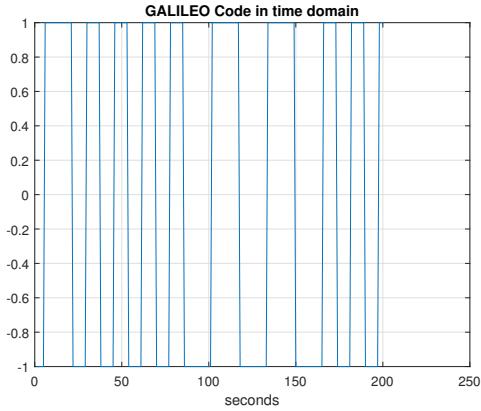


Figure 40. GALILEO code in time domain

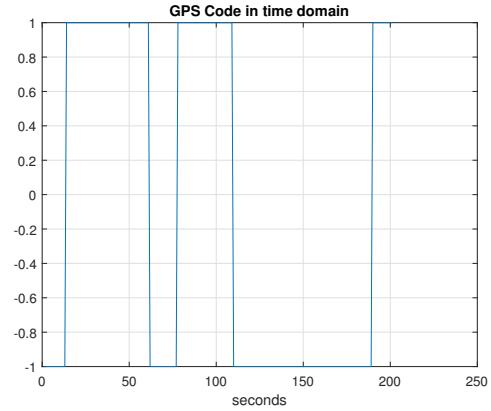


Figure 41. GPS code in time domain

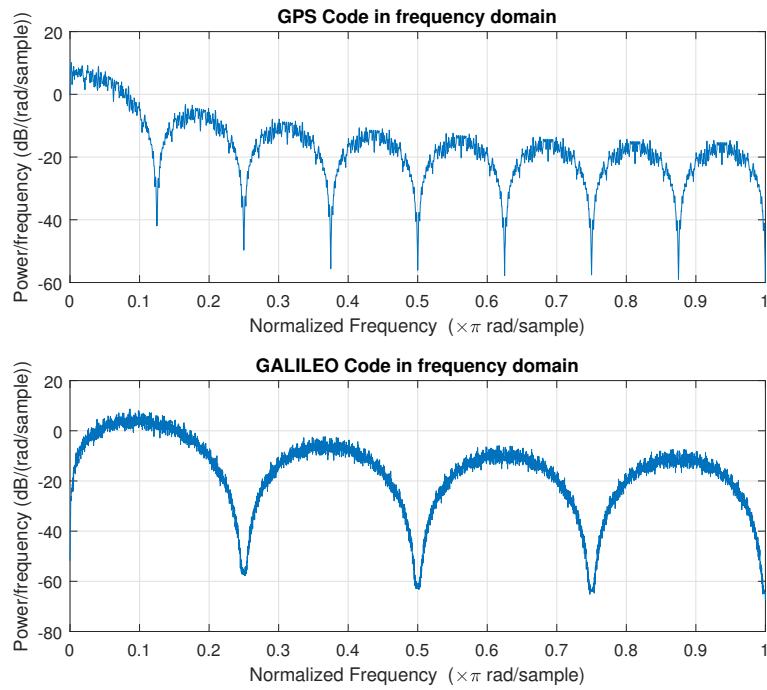


Figure 42. GPS and GALILEO codes in frequency domain

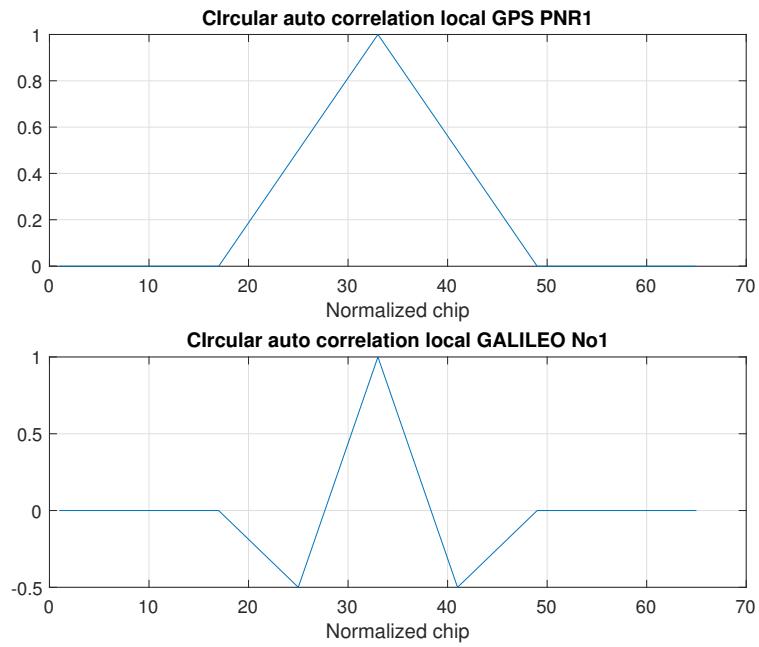


Figure 43. Circular auto-correlation of one code period for GPS and GALILEO BOC(1,1) codes

LAB04: IF Signals Correlations

The objective of this laboratory experience is to implement the acquisition stage of a GNSS receiver by means of the software MATLAB. This was achieved through the sequence of four steps that will be described below and their results will be shown.

Generation of the GNSS carrier and visualization of signals in time and frequency

The first two steps will be described in this one section. First, it has been proceeded to write a MATLAB function capable of generating a carrier used to perform the down-conversion of a GNSS signal from the receiver Intermediate Frequency IF to the baseband using the following parameters:

- . IF carrier frequency: **4.092 MHz**
- . Sampling frequency: **16.368 MHz**
- . Doppler(fixed): **0 Hz**
- . Amplitude: $\sqrt{2}$
- . Signal length(time): **1 ms**

Figures 44 and 45 show the plots of the generated local code, the carrier signal and the product between the two codes both in time (displayed between 100 and 300 seconds) and in frequency domain. As it can be seen a phase shift of the carrier occurs at the change of code's sign.

Correlations

In the third step, the calculation of correlation between the received sequence and a local code has been implemented on the MATLAB code. This step was done by using circular cross-correlation and the Fast Fourier Transform. As shown in the Figures 46 and 47, the results are the same, but the FFT is a much faster procedure than the calculation of correlation by circular cross-correlation when the length of the sequence is large, but this is not the case here. In the received sequence, the PRN code corresponds to the number 3 because as it is shown in Figures 46 and 47 it is the one for which the peak occurs.

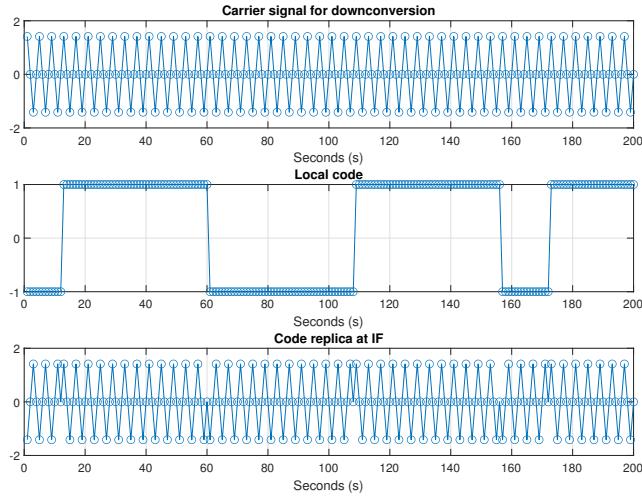


Figure 44. Signals visualization in time domain

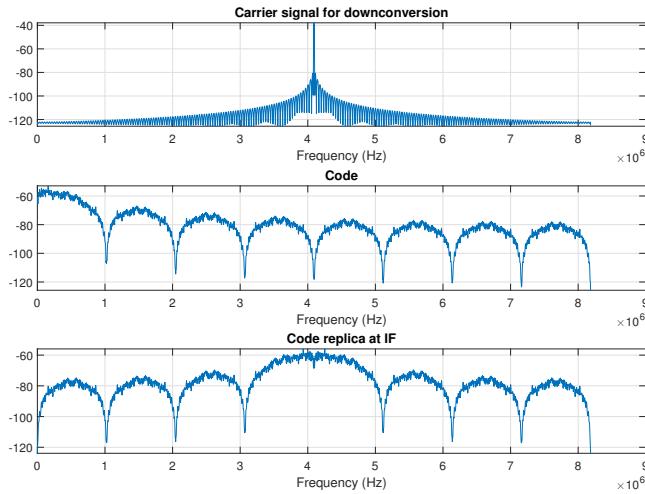


Figure 45. Signals visualization in frequency domain

Serial acquisition

The two-dimensional correlation function in the Delay and Doppler domains is used to compare the incoming signal and a local replica in order to achieve the best estimation of its parameters. The CAF is evaluated over a set of values τ and f_d that define the Search Space. The grid has the dimensions of delay (time) and Doppler shift (frequency).

In this step a MATLAB function capable to acquire an ideal GPS signal using the serial acquisition scheme has been used. The signal parameters needed to perform the acquisition are:

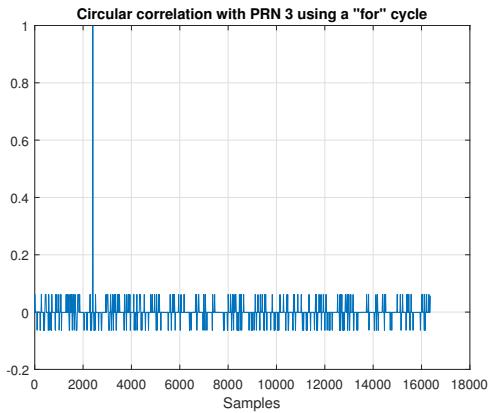


Figure 46. Circular correlation computed by means of a "for" loop

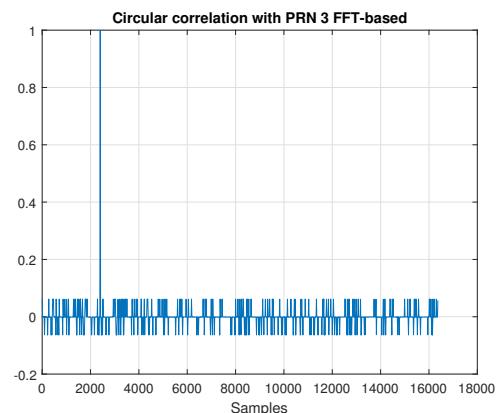


Figure 47. Circular correlation computed by means of FFT

- . IF carrier frequency: **4.092 MHz**
- . Sampling frequency: **16.368 MHz**
- . Signal length: **50 ms**
- . Signal type: **double**
- . Coherent integration time: **1 ms**
- . Constellation and signal: **GPS L1 C/A**
- . PRN: **5**

Figures 48, 49 and 50 show the plotted results, from the acquisition stage the Doppler frequency (f_d) value is between $1666,66Hz$ and $2333,33Hz$ and the Delay is $0.6ms$. The same results are obtained considering each portion of 1ms of the signal, 50ms long.

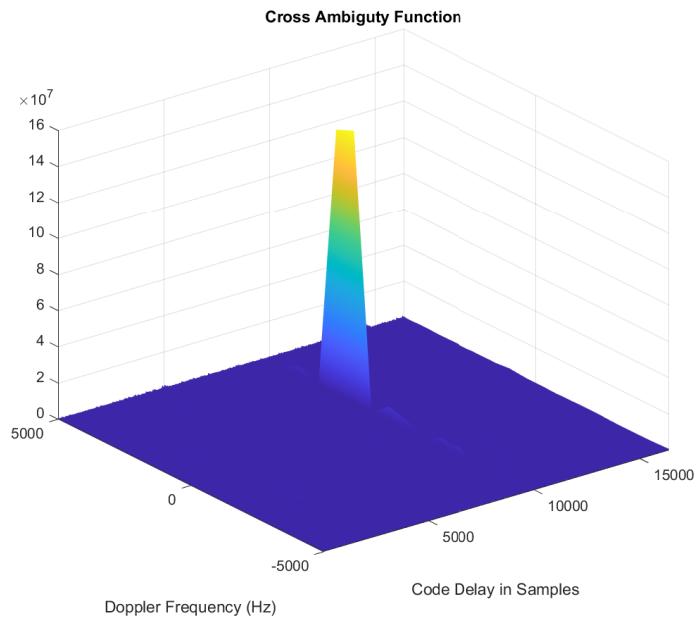


Figure 48. Cross Ambiguity Function in the Search Space

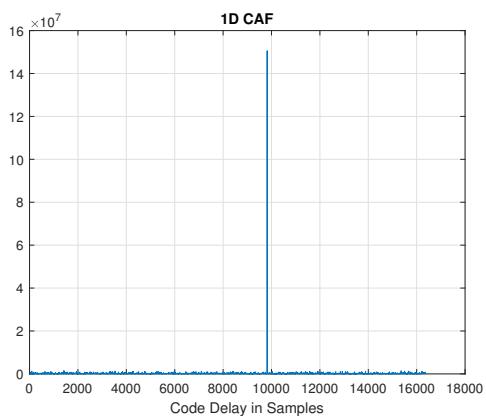


Figure 49. Cross Ambiguity Function depending on Code Delay

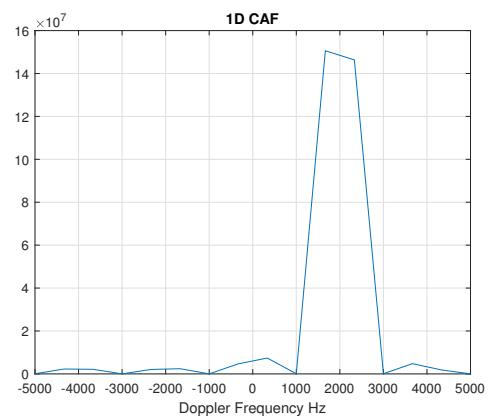


Figure 50. Cross Ambiguity Function depending on Doppler Frequency

LAB05: Acquisition of real signals

The goal of this laboratory experience is to implement a more advanced acquisition stage of a GNSS receiver, capable of acquiring a real gnss signal. The following sections will describe the procedures used and the obtained results.

Parallel acquisition

The first step in this lab is to write a MATLAB function able to acquire an ideal GPS signal using the parallel acquisition in time domain scheme. Through this function it was possible to acquire two ideal signals provided in the training material:

- . **SignalRX_1.bin** containing a GPS signal;
- . **SignalRX_2.bin** containing a GALILEO signal.

Figures 51 and 52 show the obtained results. The gain in processing time was also evaluated; in fact, for the same signal, the time required to acquire via serial scheme is **212** seconds, while parallel acquisition took only **0.144** seconds.

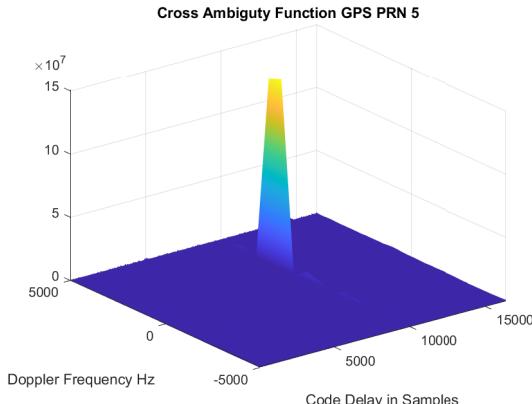


Figure 51. CAF of a GPS signal using parallel acquisition

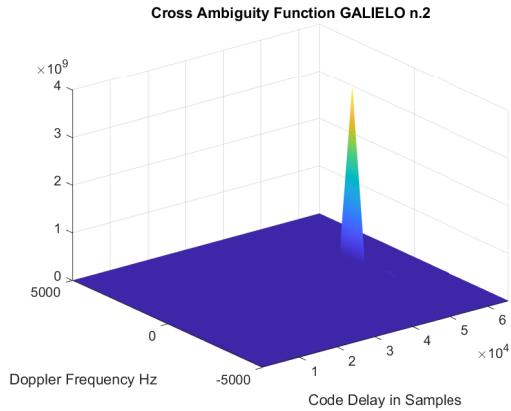


Figure 52. CAF of a GALILEO signal using parallel acquisition

The effect of noise

In this stage the same analysis will be carried out using an ideal but noise-containing signal. In the CAF evaluation the noise samples have an effect on each cell altering the result, the following

techniques can be performed to solve the problem:

- . Non-coherent integration time extension;
- . Coherent integration time extension;
- . Combination of the two

To implement a MATLAB function capable of performing non-coherent integration time extension the following parameters were initialized:

- . Sampling frequency: $16,36MHz$
- . IF : $4,32MHz$
- . Chip rate : $1,023MHz$
- . T_{coh} : 0.001
- . $L = 16368Hz$
- . f_{max} : $5000Hz$
- . f_{min} : $-5000Hz$
- . Δf : $666,667Hz$

Using the coherent integration time, the CAF has been obtained combining $N_c = 3$ correlation functions.

Figures 53,54 and 55 show the obtained results from the acquisition of the provided signals. It can be seen that PRN signal n. 18 is more noisy, while in the PRN signal n. 21 the peak cannot be detected due to the low C/N_0 .

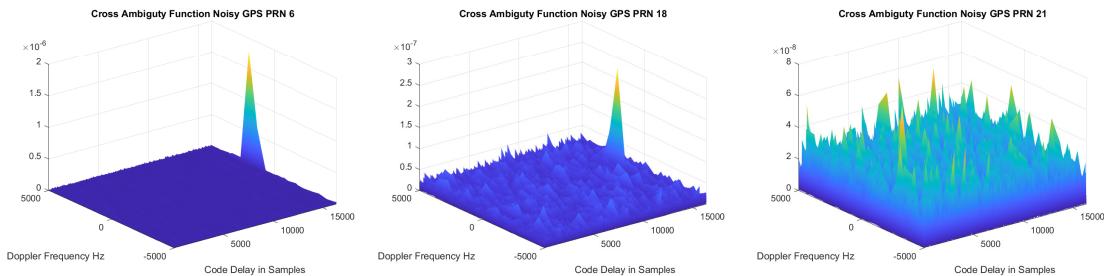


Figure 53. CAF of PRN signal n.6

Figure 54. CAF of PRN signal n.18

Figure 55. CAF of PRN signal n.21

Provided that the most noisy signal is the 21st, the coherent and non-coherent time extension has been performed on that one. Figures 56 and 57 show the improved results.

Figure 58 shows the result of the mixed technique, this allows to lower the coherent time extension (lowering then the probability of bit transition, in a real scenario).

Cross Ambiguity Function Noisy GPS PRN 21 with coherent time extension Nc = 7

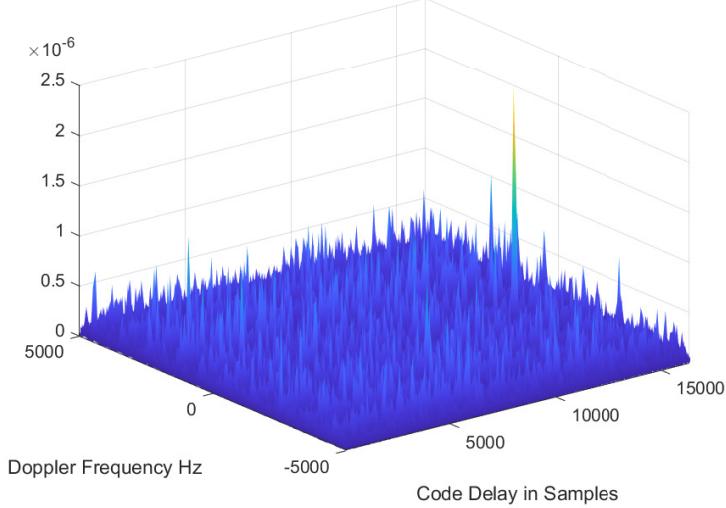


Figure 56. CAF of PRN signal n.21 with coherent time extension

Cross Ambiguity Function Noisy GPS PRN = 21 with non-coherent time extension Nc = 10

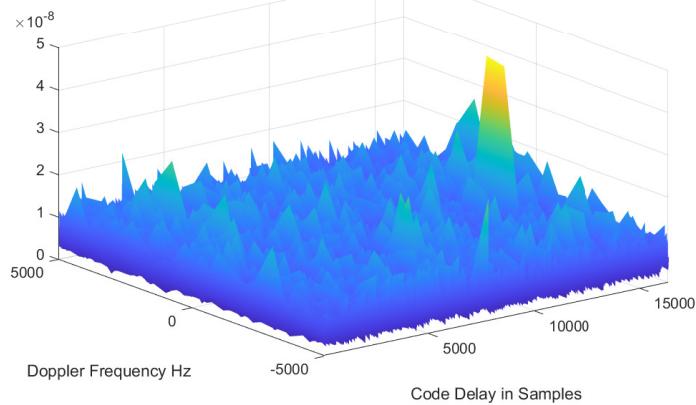


Figure 57. CAF of PRN signal n.21 with non-coherent time extension

Real GNSS Signals

The last task of this lab consists in performing the acquisition of real GPS L1 and GALILEO E1b signals.

The data were acquired on 5/12/22 around 5 PM in three typical conditions of the city environment: open sky (terrace of the Polytechnic of Turin), mild urban (in front of the Mixto bar) and urban canyon (in the courtyard of the classrooms I of the Polytechnic of Turin).

Figure 59 shows the skyplot of that day.

Table 6 shows the real PRNs acquired during the lab experience for both GPS and GALILEO satellites.

Cross Ambiguity Function Noisy GPS PRN 21 with coherent time extension Nc =

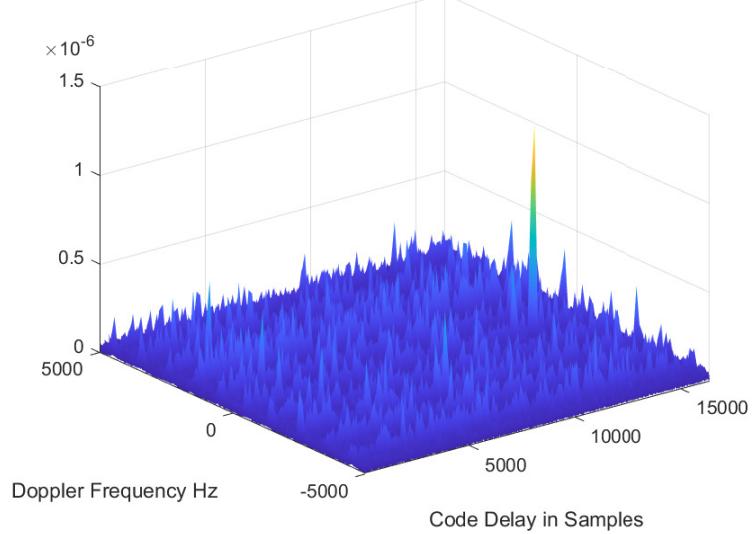


Figure 58. CAF of PRN signal n.21 with non-coherent and coherent time extension

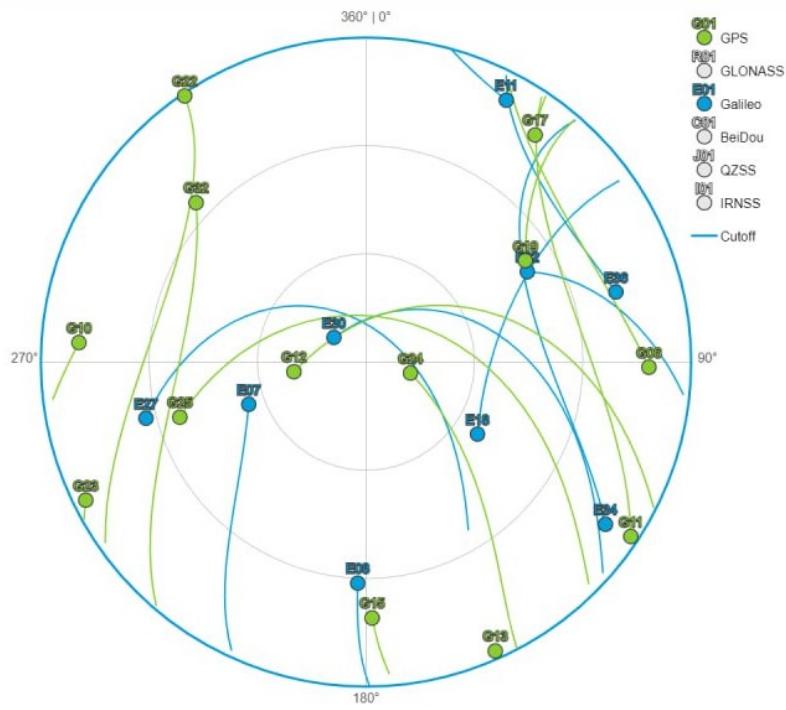


Figure 59. 5/12/22 5 PM skyplot

Comparing the results with the skyplot it is evident that, considering for example the GPS constellation, in an open sky condition, PRN 6 and PRN 10 have been acquired because no

Table 6. PRNs acquisitions in different ambient conditions ($N_c = 5, N_{nc} = 5$)

	ROOFTOP	MIXTO	ROOM I
GPS PRNs acquisitions	6,10,17,19,24,25,32	17,19,24,25,32	17,19,24,32
GALILEO PRNs acquisitions	2,7,18,27,30	2,7,11	2,7,8,18,30

obstacle were found between receiver and SV. In fact, in other conditions those satellites were not visible. Some other signals were not acquired, although the respective satellites were visible, probably due to noise or multipath.

LAB06: Tracking

The goal of this sixth lab is to focus on the code tracking loop.

A first step will be to plot the correlation function of the prompt and the output of the Early and Late correlators, in the absence of noise. After that the open loop discrimination function in so that the S-curves can be plotted. Ultimately the influence of multipath will be analyzed.

Correlators and discriminator

A MATLAB script able to perform the difference between the early and the late correlation of the prompt code has been implemented. This allows the receiver to estimate the actual code delay.

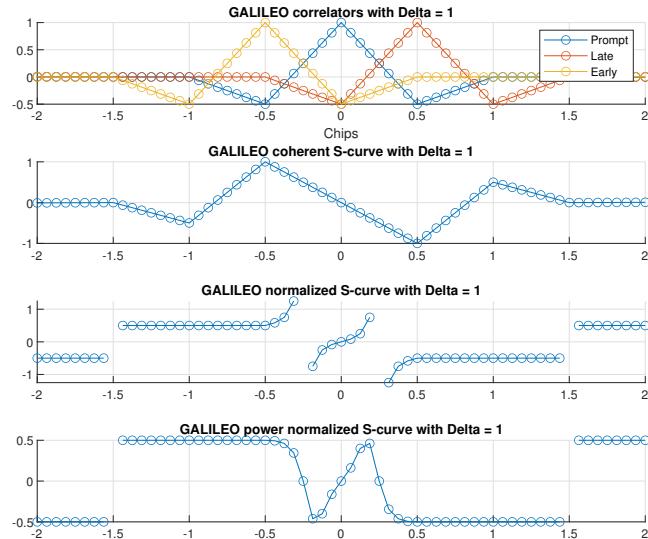


Figure 60. GALILEO correlators with $\Delta = 1$

Figures 60 and 61 show the Early and Late correlators with $\Delta = 1$ for GALILEO E1, BOC(1,1) and GPS L1 signals. Figures 62, 63, 64 and 65 show the the same plots with $\Delta = 0.50$ and $\Delta = 0.25$ respectively for a GPS and a GALILEO signal.

In this step also the discrimination function of the Early and Late correlators has been computed. As can be seen the S-curves are also shown in the figures.

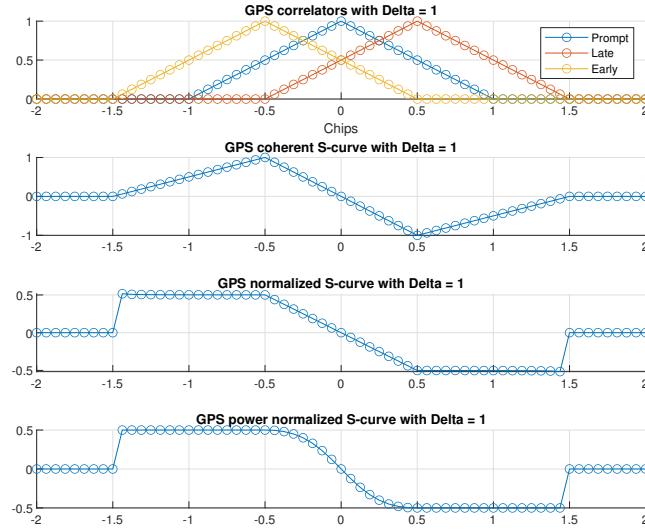


Figure 61. GPS correlators with $\Delta = 1$

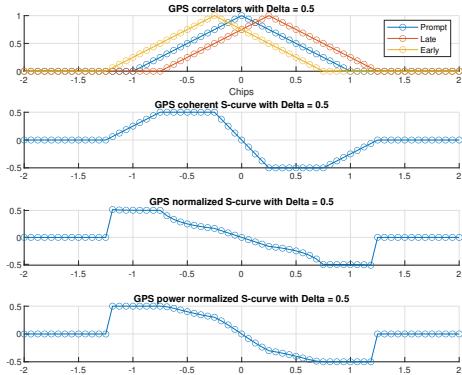


Figure 62. GPS correlators with $\Delta = 0.5$

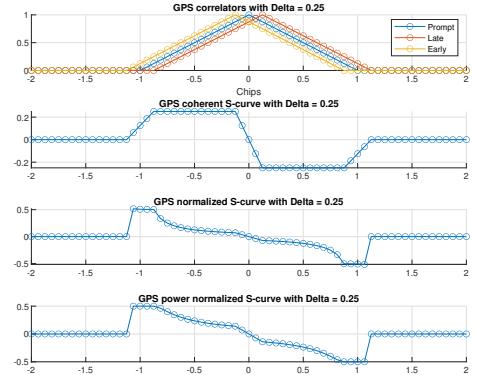


Figure 63. GPS correlators with $\Delta = 0.25$

They have been plotted considering the following discrimination functions:

- . Early minus Late $s = S_E - S_L$
- . Early minus Late normalized $s = \frac{1}{2} \frac{S_E - S_L}{S_E + S_L}$
- . Early minus Late power normalized $s = \frac{1}{2} \frac{(S_E)^2 - (S_L)^2}{(S_E)^2 + (S_L)^2}$

A further step is to introduce a simple front-end filter model, to be applied at the received code. In this case, the choice fell on a Butterworth filter of order 4. Figure 66 shows the results for GPS signal.

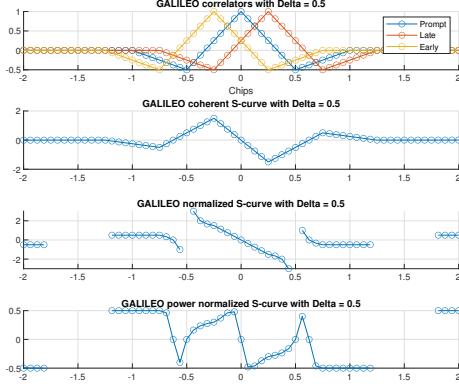


Figure 64. GALILEO correlators with $\Delta = 0.5$

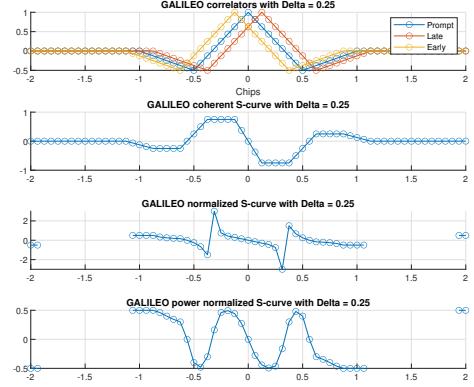


Figure 65. GALILEO correlators with $\Delta = 0.25$

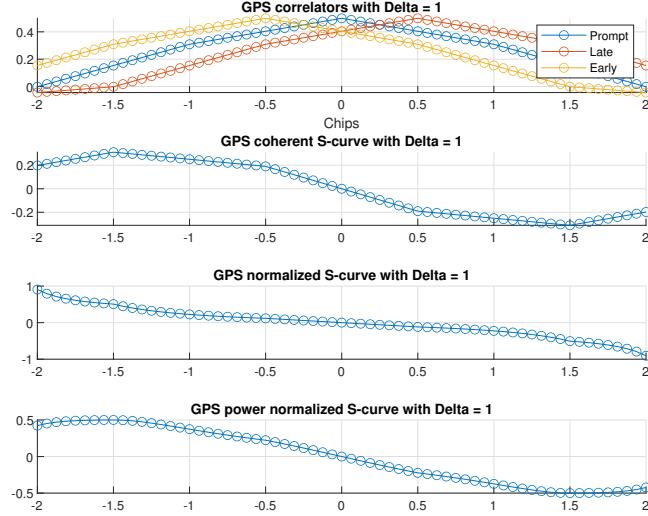


Figure 66. GPS correlators with $\Delta = 1$, filtered signal

Multipath

The last step is to calculate the correlation function in the presence of multipath. As it can be seen in Figure 67 the S-curves are distorted. The presence of Butterworth filter (Figure 68) smooths the S-curves.

Tables 7 and 8 show slope values of the linear part of the S-curve of the discriminator for GPS and GALILEO respectively. Tables 9 and 10 show bias values of the S-curve of the discriminator for GPS and GALILEO respectively. Biases and slopes are computed with respect of the unity of one sample.

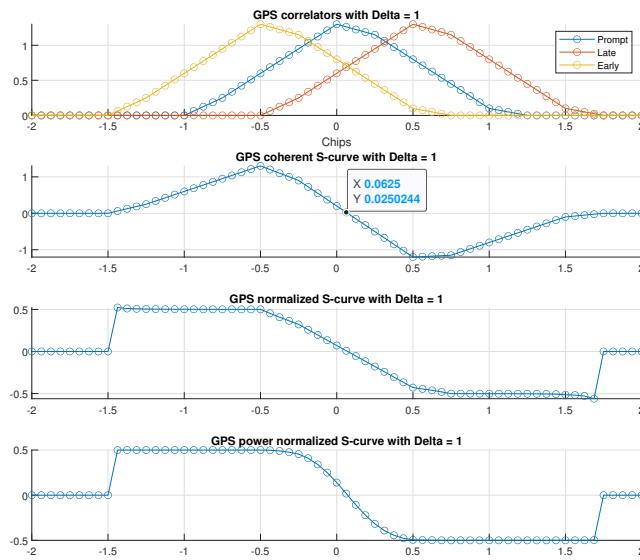


Figure 67. GPS correlators with $\Delta = 1$, in the presence of multipath

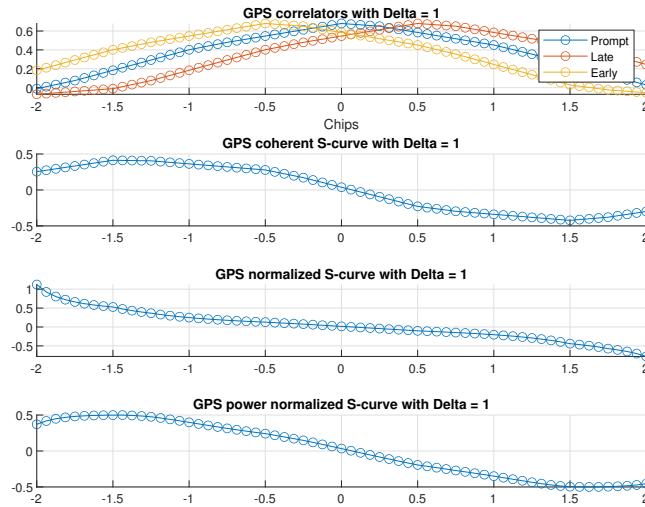


Figure 68. GPS correlators with $\Delta = 1$, filtered signal in the presence of multipath

Table 7. GPS slope

	COHERENT	NORMALIZED	POWER
$\Delta = 1$	-0,125	-0,063	-0,123
$\Delta = 0.5$	-0,125	-0,042	-0,083
$\Delta = 0.25$	-0,125	-0,036	-0,071

Table 8. GALILEO slope

	COHERENT	NORMALIZED	POWER
$\Delta = 1$	-0,125	0,063	0,162
$\Delta = 0.5$	-0,125	-0,375	-0,48
$\Delta = 0.25$	-0,125	-0,15	-0,275

Table 9. GPS bias

	COHERENT	NORMALIZED	POWER
$\Delta = 1$	1,14	1,14	1,14
$\Delta = 0.5$	1,14	1,14	1,14
$\Delta = 0.25$	0,8	0,8	0,761

Table 10. GALILEO bias

	COHERENT	NORMALIZED	POWER
$\Delta = 1$	1,14	1,14	1,14
$\Delta = 0.5$	1,14	1,14	1,14
$\Delta = 0.25$	0,8	0,8	0,761