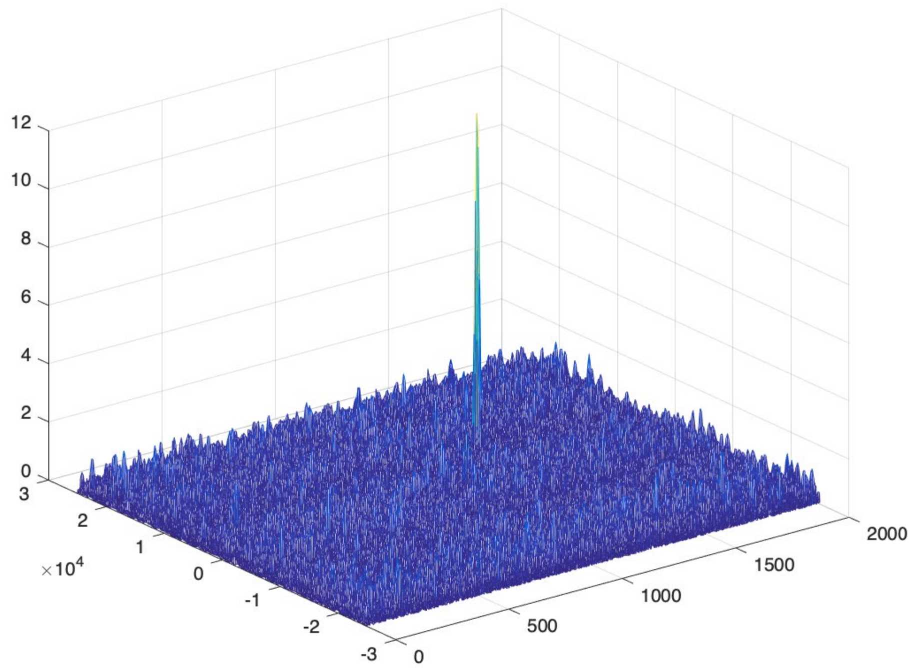
Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

NACC: Navegació Aerea, Cartografía i Cosmografía
GNSS Signal Acquisition laboratory v1.3



v1.1-v1.3
Luis Esteve 2019/2021
luis.esteve@upc.edu

v1.0
Javier Arribas 2018
jarribas@cttc.es

Luis Esteve (luis.esteve@upc.edu), Javier Arribas (jarribas@cttc.es)

Contents

Luis Esteve (luis.esteve@upc.edu), Javier Arribas (jarribas@cttc.es)

Introduction

Welcome to the GNSS signal acquisition laboratory. The main objective of this lesson is to understand, by writing MATLAB scripts and obtaining results, how the GNSS receivers perform the satellite detection and the coarse estimation of the signal synchronization parameters. This is the GNSS receiver process called *acquisition*.

The laboratory ZIP package includes:

- The exercises guide in PDF format (this document)
- A set of custom MATLAB functions in the *libs* subfolder
- A set of recorded GNSS signals in the *signal's* subfolder
- A set of partially completed MATLAB scripts in the *exercise's* subfolder

It is required to have the MATLAB suite or the OCTAVE suite installed in the student computer. The activity is divided in short exercises that the student should complete to solve a problem or to obtain a result.
In some scripts, it is required to add the libs folder to the MATLAB search path to be able to use some custom functions. This can be done in each script by executing:

```
addpath('../../libs');
```

Inside the MATLAB code of each exercise can be found a set of QUESTIONS that the student should solve to produce the desired results. The student should write the answers and copy the figures to a separate document to be delivered to the teacher.

EXERCISE 1: Front-end simulation and signal analysis

The front-end is the first element of the GNSS receiver. In this exercise we will simulate the complete signal chain, from the internal baseband signal generated inside the satellite to the baseband signal recovered by the front-end, including the up-conversion to RF and the down-conversion to baseband.

We will learn how to use the Fast Fourier Transform (FFT) to perform a frequency analysis of the signal in the different points along the signal path.

The exercise is self-explained and it is contained in the MATLAB script *frontend_simulation_exercise.m*, located inside the *exercises/exercise1* subfolder. The student should complete all the questions and write the missing parts of the script to produce a set of figures.

EXERCISE 2: GNSS signal spectrum analysis

The goal of this exercise is to integrate a GNSS signal generator into the front-end simulator written in the EXERCISE 1. The student will be able to explore the signal spectrum of a realistic GPS L1 CA signal and identify important parameters such as the chip rate, sampling frequency, SNR, CN0 and the receiver bandwidth among others.

The exercise is self-explained and it is contained in the MATLAB script *gps_l1_ca_analysis_exercise.m*, located inside the *exercises/exercise2* subfolder. The student should complete all the questions and write the missing parts of the script to produce a set of figures.

EXERCISE 3: GNSS signal autocorrelation analysis

This exercise is devoted to understand the autocorrelation and cross-correlation properties of the GNSS CDMA signal, which are the basis of the GNSS signal acquisition. We will use the GPS L1 CA generator to produce a set of satellite signals and then compute some correlation products.

The exercise is self-explained and it is contained in the MATLAB script *correlation_properties_exercise.m*, located inside the *exercises/exercise3* subfolder. The student should complete all the questions and write the missing parts of the script to produce a set of figures.

EXERCISE 4: Acquisition engine

Finally, in this exercise we will learn how to implement a GNSS satellite acquisition engine. The chosen implementation strategy is the so-called Parallel Code Phase Search (PCPS) that make use of the FFT circular correlation properties to parallelize the Code Phase (PRN delay) search. In order to alleviate the student workload, the PCPS algorithm is already implemented in a MATLAB function called *pcps_acquisition* contained in the MATLAB file *exercises/exercise4/pcps_acquisition.m*.

The student should write a new MATLAB script called "*acquisition_engine.m*" that uses the *pcps_acquisition* function and implements the following functionalities:

1. Loads a captured signal file containing samples,
2. Defines all the required acquisition parameters: Fs, Doppler step, Doppler max and Doppler min, and sets the threshold, and
3. Performs a loop that tries to acquire all the satellite's PRN's available in GPS L1 constellation (1 to 32)

The captured signal is stored in a MATLAB file located in *signals/gps_l1_ca_2msps.mat*. The sampling frequency was set to 2 MSPS. Notice that the front-end used to capture the signal (USRP 1.0) has a very inaccurate LO, thus, it is present a parasitic IF in Doppler shifts. Explore with an acquisition window of +25 kHz to -25 kHz.

Example of execution of the PCPS acquisition:

```
load('../../signals/gps_l1_ca_2msps.mat')
pcps_acquisition(rawSignal, 2e6, -10e3, 10e3, 1000, 2.5, 1)
```

Luis Esteve (luis.esteve@upc.edu), Javier Arribas (jarribas@cttc.es)

Question 1:

The *pcps_acquisition* implements the PCPS acquisition using the circular autocorrelation properties of the FFT. Read carefully the source code and **explain line-by-line with your own words** what is implemented in the lines comprised between the frequency sweep loop start (`for frqBinIndex=1:1:numberOfFrqBins`) and the loop `end`.

Question 2:

This version of the PCPS acquisition implements a variation in the test statistic function T(x). Inspect the *pcps_acquisition* code and:

- Explain, **with your own words**, how T(x) is computed.
- Adjust the threshold to acquire satellites and avoid false alarms. HINT: thresholds values for the implemented T(x) are in the range of \gamma \in [0,10].
- Which satellites are visible in the gps_l1_ca_2msps.mat signal capture?
- Capture a figure showing the acquisition grid of the most powerful satellite detected.
- Capture a figure showing the acquisition grid of the less powerful satellite detected.

Question 3:

- Compute a table with all the detected satellites, including PRN number, Code Delay in samples and Doppler frequency estimations.

- Assuming that the mean Doppler frequency shift for the visible satellites for a static receiver on the Earth surface should be 0 Hz, compute the parasitic IF frequency present on the front-end using the obtained satellite Doppler estimations.

- How affect's the acquisition Doppler grid granularity to the obtained results? HINT: increase and decrease the value of *doppler_step_hz* variable and compute the table with all the detected satellites. What is the theoretical limit of this variable to be able to correctly detect the satellites?