Lab 5:

**Multipath Effects in Code Tracking**

Course ENV-542

Advanced satellite positioning

2020

Document reference: Student version

Author(s): Miguel Angel Ribot

Vincenzo Capuano

Cyril Botteron

Teaching assistant: Gabriel Laupré ([gabriel.laupre@epfl.ch](mailto:gabriel.laupre@epfl.ch))

Date of preparation: 30/04/2020

Revision: 8.0

Date of lab : 01-05-2020 – 08-05-2020

Student name : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Goal of the lab

In this lab assignment, we will study the impact that multipath signals can have on the performance of a GNSS receiver. In particular, we will concentrate on its effects on the code tracking loop as it is the most multipath sensible block within the receiver. First, we will derive an analytic model for the direct and multipath signals arriving to the receiver from a particular satellite. Then, we will see how we can derive the multipath error, and how to represent it. Finally, we will investigate the effect of some tracking parameters to mitigate the multipath errors.

***Important: Please write your answers in this electronic document and document well your MATLAB code as you will also need to provide it in a .zip file named as “lab5\_LastName\_FirstName.zip”. The deadline to hand out the .zip file and this document without penalty is May 8th before lunch.***

Exercise 1: analytical model for direct and a single multipath signal

In this Exercise and the following ones, we are going to assume a scenario where our GNSS receiver is receiving two signal components: the line-of-sight (LOS) signal and a single specular multipath component[[1]](#footnote-2), as shown in Fig. 1.

GNSS RX

antenna

LOS

signal

Multipath component

GNSS

satellite

Ground surface

Figure 1 Scenario with a single specular multipath component represented.

For simplicity we will be taking the following assumptions:

1. We can estimate the Doppler of the received GNSS signal very accurately.
2. Our receiver front end has a very large bandwidth ().
3. The receiver is stationary and the Doppler shift experienced by the LOS and the specular multipath is the same.
4. There is no bit transition during the considered observation time.

Given the previous assumptions, and after removing the carrier frequency from the signal, we can model the output of the Prompt correlator while in the tracking stage after an integration time as:

(1)

where:

is the amplitude of the LOS signal component,

is the carrier phase of the LOS signal component referred to the receiver’s local oscillator,

is the autocorrelation function of the code evaluated at ,

is the code delay difference between the true delay of the LOS signal and the local estimate ,

is the multipath-to-signal amplitude ratio (MSAR) with ,

is the multipath component extra delay,

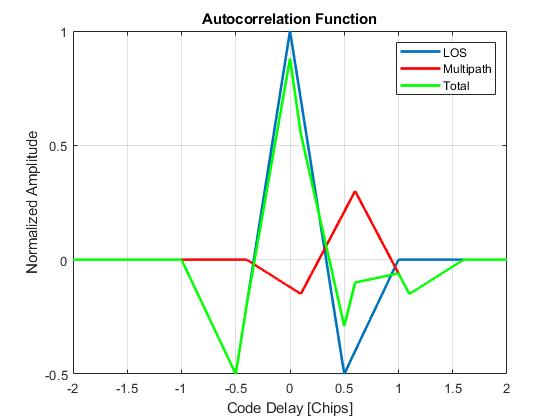
is the multipath component extra phase,

is the additive Gaussian noise component.

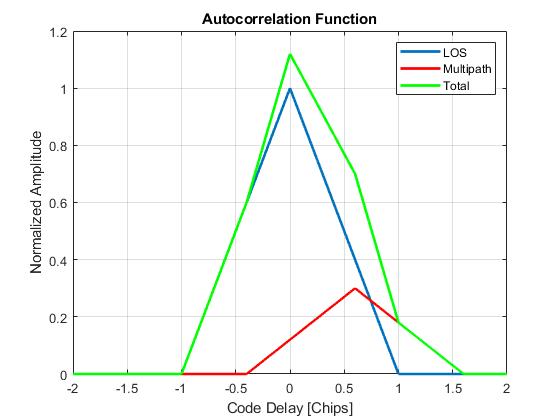
Question 1.1

Modify the function initSettings (and then run settings = initSetings) to initialize a multipath with a MSAR = 0.3, a multipath delay of 0.6 chips and a phase delay of 0º assuming first a GPS L1 BPSK received signal, and then a Galileo E1 BOC(1,1) signal. For each signal, use the plot\_autocorrelation function to plot the autocorrelation of the LOS component, the multipath component, and their sum. Repeat the simulation with a phase delay of 180°.

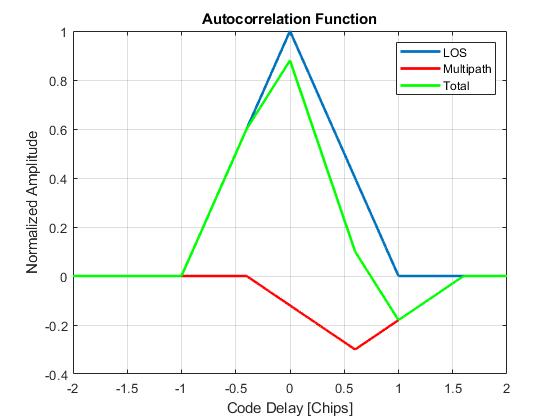
Put the four plots (including clear legends) here:



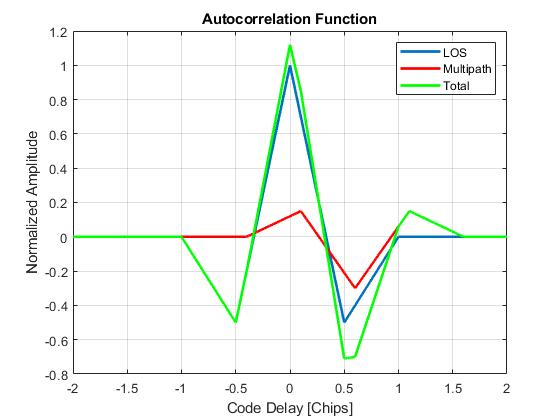
**Galileo E1 BOC(1,1), phase = 0°**



**GPS L1 BPSK, phase = 0°**



**GPS L1 BPSK, phase = 180°**



**Galileo E1 BOC(1,1), phase = 180°**

Question 1.2

Analyze the results obtained.

Does the multipath presence have some effect on the autocorrelation function?

First, we see that the resulting correlation function, taking the 2 components into account, is distorted in every case due to the additional multipath component that comes with a certain delay, phase, and attenuated compared to LOS signal. (So yes, the multipath presence has some effect on the function).

For the GPS signal, we see that multipath interference is constructive (peak of result is higher than without multipath) when the phase delay is 0°, while it is destructive (peak is smaller) with a 180° shift. We observe the opposite behavior with the Galileo signal.

One important thing to notice is that the impact of multipath looks more important on the GPS signal in these settings. The pseudorange error induced could be quite large. This is explained by the geometry of the different correlation functions. With Galileo signal, for the particular delay of 0.6 chip, the multipath component affects only a side peak and not the main peak. If the delay was smaller, for instance 0.2 chip, there the main peak of the BOC function would be affected. In this case the GPS composite signal would have an amplified almost perfect main peak.

Is it possible to analytically treat every multipath component separately?

In practice, probably not. They come with different attenuations, phases and delays and it is probably difficult to identify each of them exactly.

Exercise 2: Multipath Error

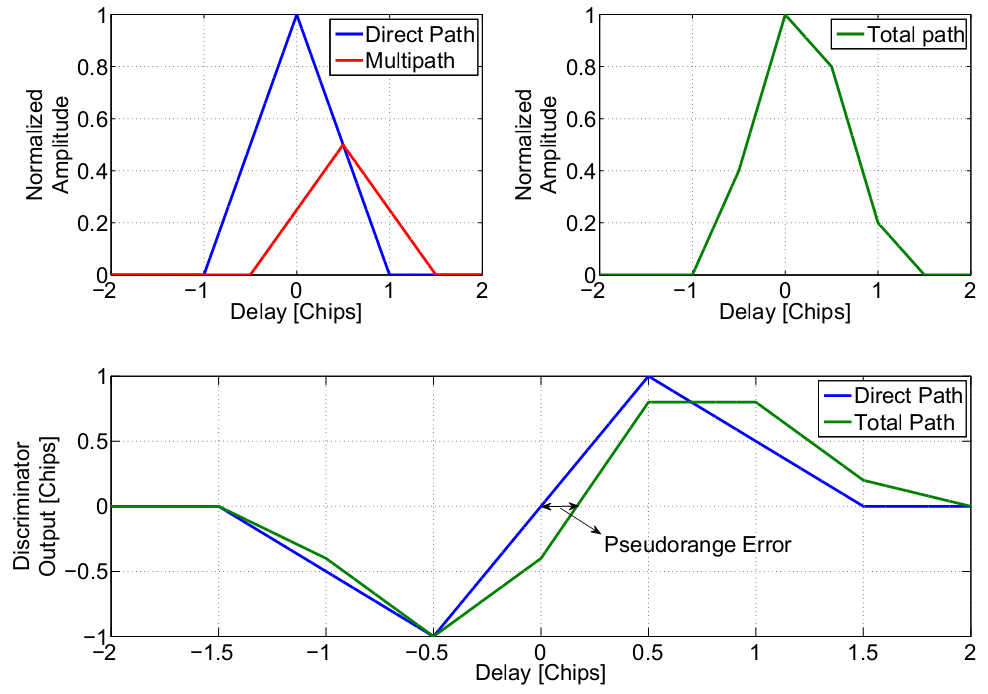
We assume that the code discriminator function used in our GNSS receiver is the **coherent dot product** function, described as:

(2)

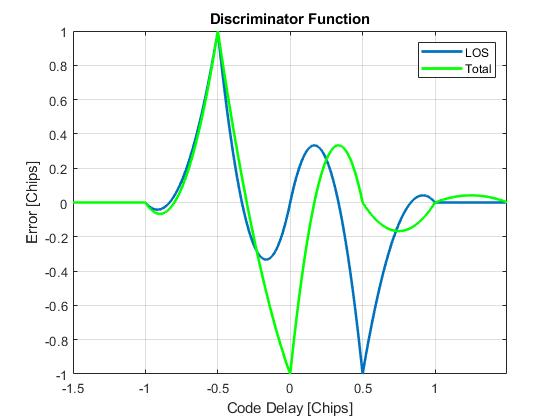
where and are the output of the in-phase component of the prompt, early and late correlators, respectively. During the tracking stage, the PLL keeps all the energy of the received signal on the in-phase component. The correlator spacing is equal to.

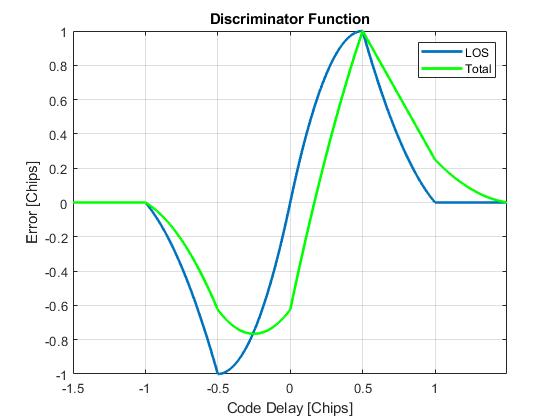
Question 2.1

As seen in the course, due to the presence of a multipath, the correlation function may be distorted and thereby induce a pseudorange or code delay error, as can be seen in the plotted example below:



Use the plot\_discriminator function to plot the discriminator output of the coherent dot product discriminator for of a LOS component, and a LOS + multipath component, assuming first a GPS L1 BPSK received signal, and then a Galileo E1 BOC(1,1) signal. (Multipath Parameters: MSAR=0.5, = 0.5 and = 0).

Put the 2 plots (including clear legends) here:



**GPS E1 BPSK signal**

**Galileo BOC signal**

Question 2.2

Compute visually the code error in meters for BPSK and BOC(1,1) resulting from the existence of the multipath signal in Question 2.1 (when the output of the discriminator function is zero).

|  |  |
| --- | --- |
|  | Multipath Error [m] |
| BPSK | 0.1 chip -> 29.615 |
| BOC(1,1) | 0.1 chip -> 29.6150 |

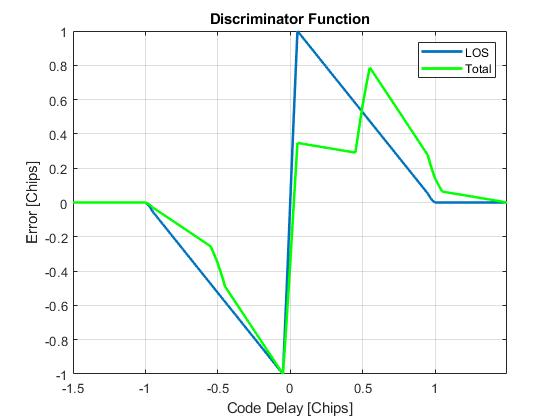
BPSK and BOC(1,1) : 1 chip lasts 1/1023 sec => this represents 296.150 m. So for 0.1 chip, we have an error of 29.615 m.

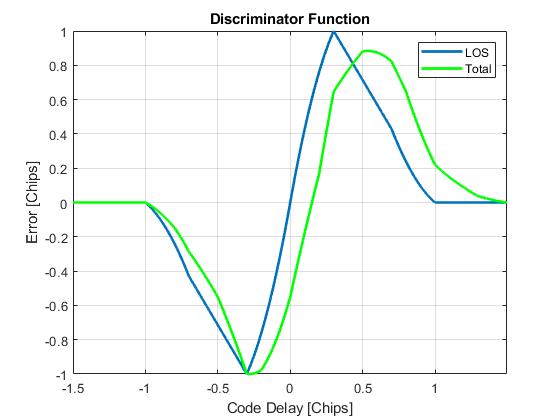
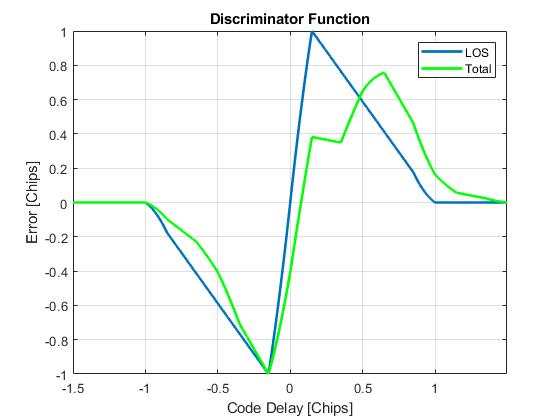
Question 2.3

Repeat Q.2.2 but for =0.6, 0.3, and 0.1. What do you observe? Can you give an explanation?

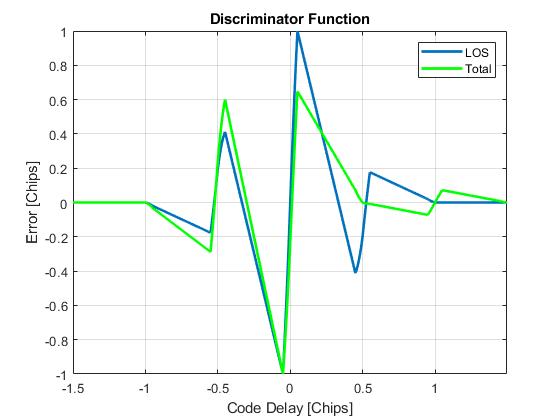
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Multipath Error [m] | | | | |
|  | *d* = 0.6 | *d* = 0.3 | | *d* = 0.1 | |
| BPSK |  |  | |  | |
| BOC(1,1) |  | |  | |  |

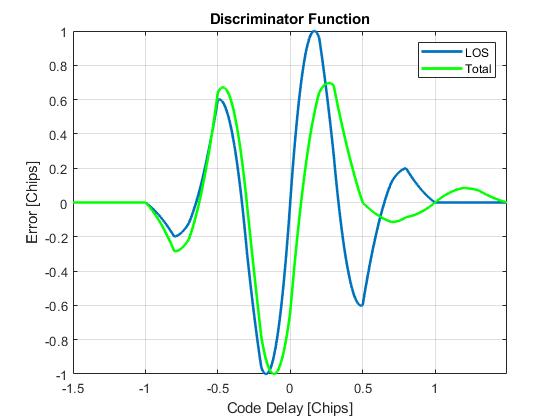
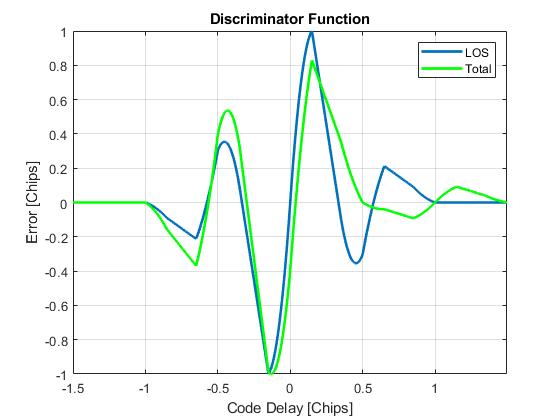
Explain the results you have obtained.

Discriminator outputs for GPS signal:



**d = 0.6 d = 0.3 d = 0.1**

Discriminator outputs for Galileo signals:



**d = 0.6 d = 0.3 d = 0.1**

Note pour moi quand je reprends, matlab planté:

Pour BPSK, d = 0.6, on a discr\_LOS = 0 pour l'indice 2001, on a discr\_total = 0 pour l'indice 2151, => regarder dtau(2001) – dtau(2151) pour avoir la chip diff.

* dtau(find(~discr\_LOS)) - dtau(find(~discr\_total))

Question 2.4

Instead of estimating visually the code delay error from the total discriminator function, we can write a small Matlab function to compute the code multipath error from a given discriminator function. An example of Matlab code is provided below that you can use to complement the “compute\_multipath\_error (settings,discr\_function)”:

function multipath\_error = compute\_multipath\_error(settings,discr\_total)

% Locate the closest zero-crossing to the center in the

% Discriminator function

zerocross = diff(sign(discr\_total));

zerocross\_idx = find(zerocross~=0);

N\_samples = length(settings.dtau)-1;

[~, min\_pos] = min(abs(zerocross\_idx - N\_samples/2));

delay\_error = (zerocross\_idx(min\_pos) - N\_samples/2)\*settings.delay\_step;

Chip\_distance = 3e8/1.023e6;

multipath\_error = delay\_error\*Chip\_distance;

To test if it works correctly, use this function to refine your answers to Q2.2 and Q2.3 above.

Exercise 3: Multipath Error Envelope

The maximum multipath code error occurs when the multipath is in-phase (0 degree), i.e. constructive multipath, or out-of-phase (180 degrees), i.e. destructive multipath, with the LOS component. Therefore, for these two phases’ values, and taking in consideration different delays, a multipath error envelope (MEE) can be derived where the multipath code error for different phases lies within this envelope.

Question 3.1

Plot the MEE using the provided function mee for BPSK and BOC(1,1) for MSAR=0.5, 0 <<2, and *d* = 0.5. What is the maximum code error for both signals and for which delay?

Put the 2 plots (including clear legends) here:

[Include your plots here]

Replace this box

Question 3.2

Repeat the previous question for *d* = 0.6, 0.3, and 0.1.

Put the 2 plots (including clear legends) here:

[Include your plots here]

Replace this box

Question 3.3

Compare the results obtained taking into consideration the chip spacing and the signal modulation.

What do you observe?

What conclusions can be made?

Exercise 4: Multipath Error with HRC Discriminator

Question 4.1

Write a Matlab function called

hrc\_discriminator(settings,multipath\_delay,multipath\_phase)

to implement the HRC discriminator as:



where *VE* and *VL* are two additional correlators with a chip spacing of 2*d*.

**Hint:** *You can get inspiration from the discriminator(settings,multipath\_delay,multipath\_phase) function.*

Question 4.2

Call the function mee\_hrc and compute the multipath error envelope for the HRC discriminator for ***d* = 0.6, 0.4, and 0.2**.

Plot the results for both the BPSK(1) and BOC(1,1) modulations

Put the 2 plots (including clear legends) here:

[Include your plots here]

Replace this box

Question 4.3

Analyze the obtained results and compare them to the results obtained with the coherent dot product discriminator.

1. Sometimes might be also referred as non line-of-sight (NLOS) signal. [↑](#footnote-ref-2)