University of Padova

Department of Industrial Engineering

Master's degree in Aerospace Engineering

Advantages of a Multi Constellation GNSS Receiver for Space Navigation and Launchers

Author

Francesco Longhi

Supervisor

Prof. Alessandro Caporali

Company's supervisor

Ing. Samuele Fantinato

Academic Year 2018/2019

Dedicated to my family and my girlfriend Clara.

Abstract

This thesis, elaborated in collaboration with Qascom S.r.l., analyses the applications, the advantages and the integration solutions of a GPS/Galileo receiver for the navigation of launchers and rockets. The research has been conducted with following 4 steps: review of the state of art, trajectories definition, GNSS scenarios analysis and simulation, navigation solution implementation and integration with INS. The launchers taken as reference are SpaceX's Falcon 9 and NASA Sounding Rocket Black Brant XII.

Using a series of Matlab tools, this paper demonstrates that a multi-GNSS receiver is recommended for this kind of applications. Many results of simulations with different combination of system (INS only, GPS only, Galileo only, INS+GPS, INS+Galileo etc..) will be compared to prove this point.

Contents

Chapter 1

State of art of launchers navigation and study approach

1.1 Navigation systems

On board navigation of launchers and sounding rockets is in general based on inertial devices. Measurements from strapdown accelerometers and gyroscopes are integrated from an initial condition yielding a high-rate navigation solution. While self-contained and robust, inertial navigation suffers from a fundamental drawback: the drift. If uncorrected, the inertial propagation errors will boundlessly grow over time. To compensate for this, inertial sensors are made increasingly accurate, raising cost, and often also volume and mass. Still, inertial drift is unavoidable, posing important performance and operational constraints on both launchers and sounding rockets. Trajectory maintenance and injection greatly depend on navigation accuracy (F. Trigo [1]). Not rarely, delivered spacecraft need to perform costly orbital manoeuvres to correct for deficient injection, potentially reducing operation life-time and causing loss of commercial/scientific profit. The drift of inertial sensors poses a strong limitation is also set on launch mission duration and profile. For instance, return phases in missions of reusable vehicles simply cannot cope with the error inertial propagation yields, requiring other measurement sources. In sounding rockets, with lower budget and operational requirements,

when used, inertial sensors are generally of relatively lower grade. The available propagated inertial attitude might potentially limiting scientific payloads. GNSS receivers have long been used to complement INS. In fact, the two measurement systems collaborate: the GNSS receiver bounds inertial drift, while the high-rate inertial measurements bridge the low-rate GNSS outputs.

In a launch environment, however, GNSS technology might be impacted by important vulnerabilities. GNSS is vulnerable to jamming and spoofing (Fantinato [2]) that pose real threat to signal quality. Additionally, signal tracking is not immune to the high-dynamics, vibration and shocks experienced in-flight. Although complete mitigation of all these risks is not possible, improved receiver design/tuning and integrity monitoring can lessen their effects. Moreover, combination with inertial sensors improves availability and robustness.

1.1.1 Architecture

The combination between GNSS and INS can be achieved in three main ways (F. Trigo [1]):

- loosely coupled uses receiver navigation solution to correct the inertial' s one;
- tightly coupled avoids filter cascading by using GNSS raw measurements;
- ultra-tight (or deep for some authors) goes even further by driving the receiver tracking loops with the inertially aided correlators outputs

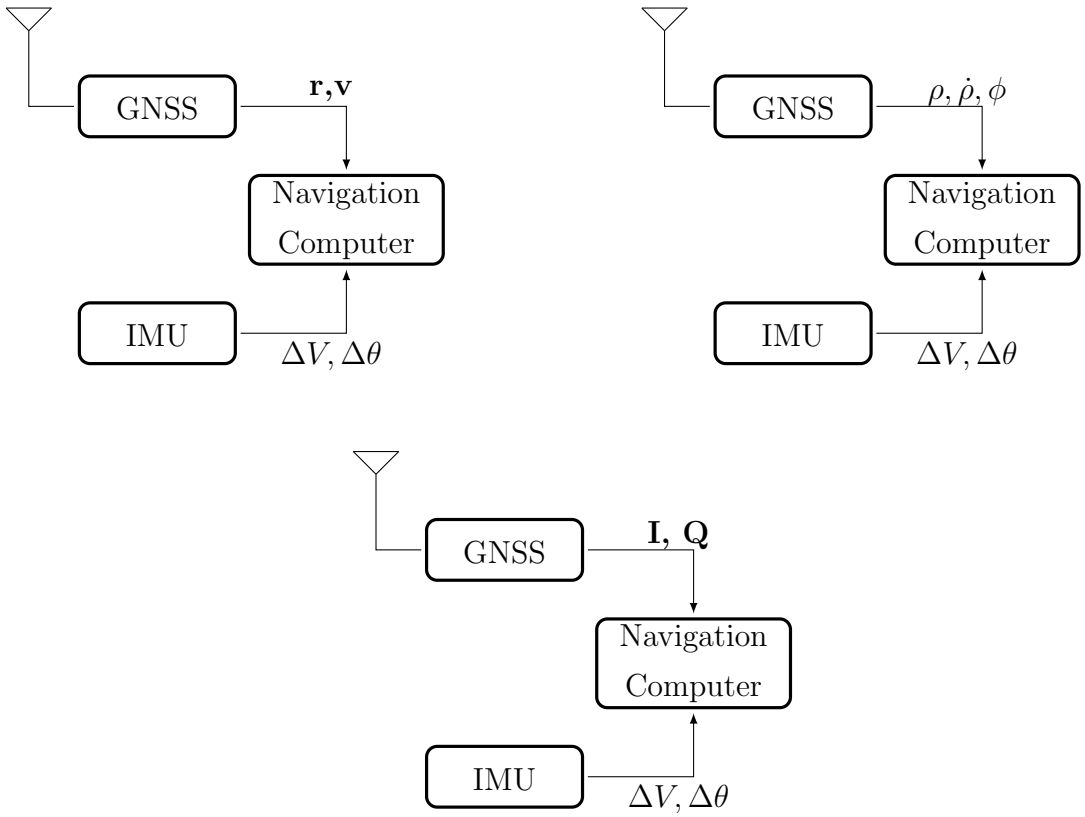


Figure 1.1: Integration GNSS-IMU: loosely coupled, tightly coupled and ultra-tight coupling

Loose and Tight integrations basically differ for the type of information shared between the individual systems: processed GNSS solution is merged with INS solution in the first architecture and raw GNSS measurements are combined with INS-predicted measurements in the second. This yields to a different structure of the two approaches, with two separate filters in the Loose coupling and only a centralized filter in the Tight. The separate filters in the Loose have the advantage to be smaller than the corresponding centralized Tight filter, yielding faster processing times. On the other hand with separate filters in LC (Loose Coupling) (if a Kalman filter is used to perform the GNSS solution), the process noise is added twice, affecting the system performance (Petovello [3]).

1.2 Falcon 9 navigation system

Falcon 9 navigation system is complex and not easily available on-line, but there is a model of IMU (Inertial Measurements Unit) often used inside the rocket, that is the LN-200S by Northrop Grumman.

Figure 1.2: IMU LN-200S

The LN-200S comprises three solid-state fiber-optic gyros and three solid-state silicon Micro Electro-Mechanical Systems accelerometers in a compact package that measures velocity and angle changes in a coordinate system fixed relative to its case.

Performance	
Accelerometer (1σ)	
Bias Repeatability	300 µg, 1σ
Noise	35 µg/√Hz
Scale Factor Accuracy	300 ppm, 1σ
Input Axis Alignment	0.1 mrad
Max Input Accel	40 g
Gyro (1σ)	
Bias Repeatability	1°/hr, 1σ
Scale Factor Stability	100 ppm
Angle Random Walk	<0.07°/√hr
Input Axis Alignment	0.1 mrad
Dynamic Range (max)	1,000°/sec
Bandwidth	200 Hz @ 400 Hz data rate

Figure 1.3: LN-200S Performances, (Northrop Grumman website)

The launcher can carry up to four of these IMU depending on the version. SpaceX also uses a GPS receiver to correct the inertial measures, to improve its navigation in terms of precision and security, multi constellation GNSS receivers might be used in the future and this is the main focus of this thesis.

1.3 Sounding rockets navigation system

Each model of sounding rocket has its own navigation system, in general they all have one or more IMUs and a GPS receiver. Performances depends on quality and missions requirements.
The operational range of sounding rockets is wide and mission requirements are very different, so analyses each one of them is long and unnecessary, for the purpose of this thesis will be considered the following accelerometers and gyro characteristic:

	Bias	σ Random walk
Gyro	10^{-5} [deg/s]	10^{-6} [deg/s]
Accelerometers	10^{-4} [m/s^2]	0.005 [m/s^2]

1.4 Study Approach

Here it is described the approach used for the thesis:
In order to assess the advantages of a Multi constellation navigation for launchers, a simulator representative of navigation systems of launchers (Falcon 9 and Sounding Rockets) has been developed. This is composed by a set of tools already available in Qascom that have been adapted for the study case of this thesis. The simulator is composed by:

- A Trajectory emulator, that integrates a system of differential equations providing the inputs for all the others simulations.

- A GNSS Constellation Simulator, that takes the trajectories and emulates a receiver inside that launcher.

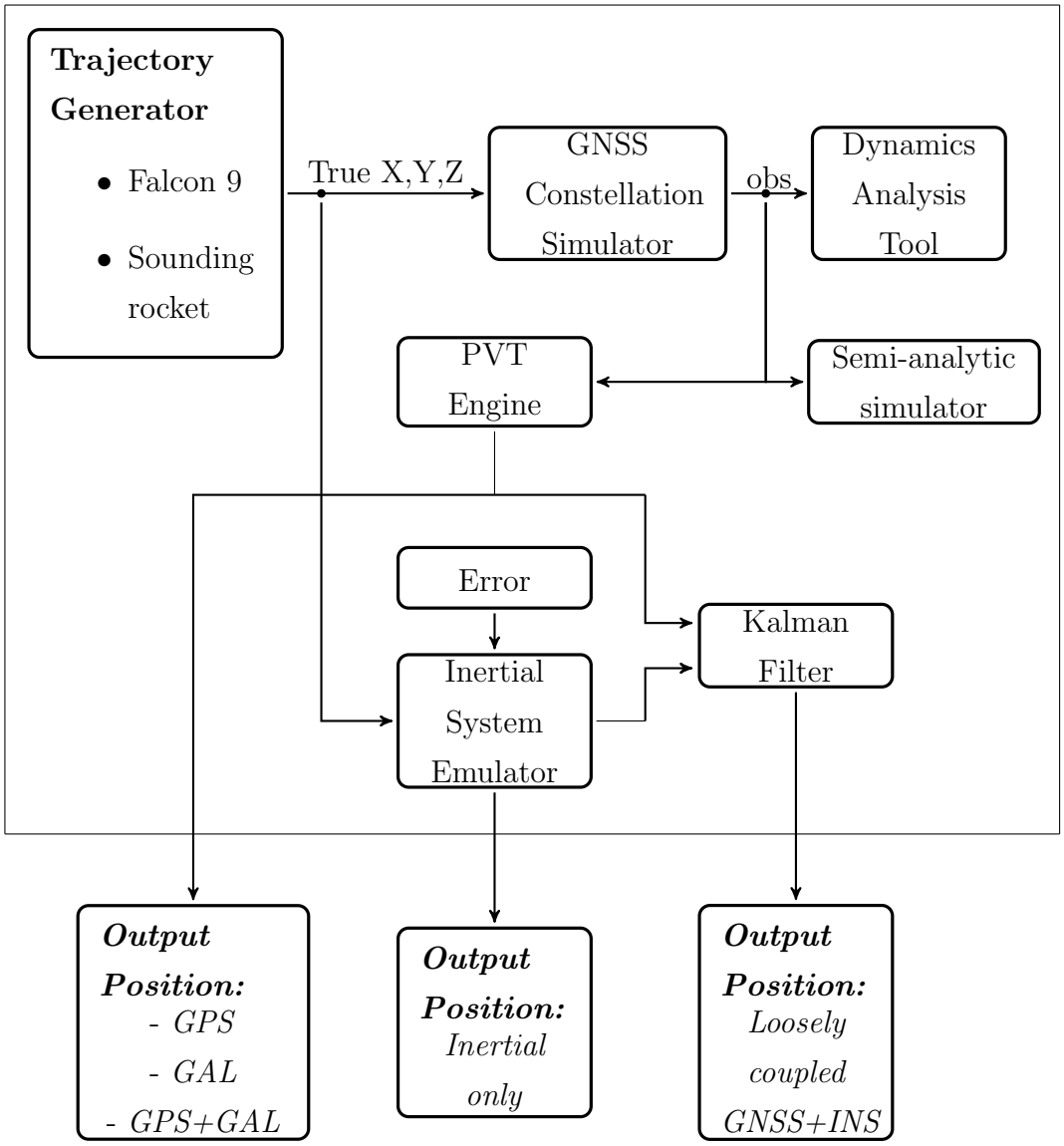
- A tool for the dynamic analysis, that uses the information from previous tools to show the main dynamics.

- A semi-analytic simulator that implements the three tracking loops (DLL, FLL, PLL) to check the capability of the receiver to operate in those challenging scenarios.

- An algorithm for the PVT, that elaborates the observables to obtain the final GNSS output (Position and Velocity).

- A simulator of the inertial system, used also to implement an error model for the IMU measurements.

- Last, a Kalman filter to integrate GNSS and INS data in a loosely coupled architecture.



Except for the trajectories tool, all the others tools have been provided by the company, though some adaptations to make them interfaced and little adaptation for the particular scenario were required. Tools descriptions is in Chapter 4.
The three main outputs of this architecture will be compared to prove the objectives thesis.

Chapter 2

Analysis and simulation of the launchers trajectories

2.1 Main Equations

Trajectories simulation begins with this system of ODE (Turner, 2009 [4] ; Wie 1998 [5]):

$$\begin{Bmatrix} \dot{x} \\ \dot{h} \\ \dot{v} \\ \dot{\gamma} \\ \dot{m} \end{Bmatrix} = \begin{Bmatrix} \frac{Re}{Re+h} v cos(\gamma) \\ v sin(\gamma) \\ \frac{T}{m} - \frac{D}{m} - g sin(\gamma) \\ -\frac{1}{v}(g - \frac{v^2}{Re+h}) cos(\gamma) \\ -\dot{m}_e \end{Bmatrix} \tag{2.1}$$

Where x is the downrange, h is the altitude, v is the speed magnitude, γ is the flight path angle and m is the mass. On the right side: T is thrust, D is drag, g is the gravitational constant.
The Drag force equation is (Curtis, 2010 [6]):

$$D = \frac{1}{2} A \rho v^2 Cd \tag{2.2}$$

Where A is the frontal area, Cd is the drag coefficient and ρ is the air density.

2.2 Reference systems

Each tool utilize his reference system depending on what is his purpose and some of them were not made to interface with others, so different coordinate rotation has been implemented.
The following Reference systems have been used in this thesis:

- ECEF
- Lla
- NED
- Body

ECEF (Earth Centred Earth Fixed) is a reference with origin at the centre of Earth, Z axis pointed in the direction of the North pole, X axis on the equatorial plane to Greenwich and Y always on equatorial plane to complete. This system is not inertial and rotates with Earth.

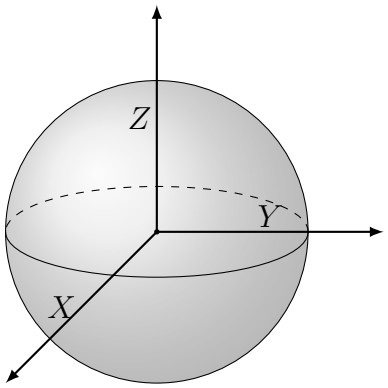


Figure 2.1: ECEF reference system

Lla (Latitude longitude altitude) is the classic geodetic reference, WGS84 is the format for the Earth shape taken as model.

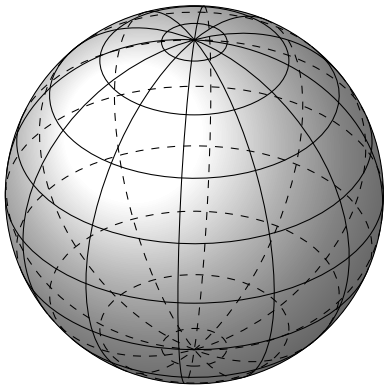


Figure 2.2: Lla reference system

NED (North East Down) is a local reference, so it needs value of lat0, lon0, h0. North axis is pointing to local north, East axis to local east and down to the Earth centre.

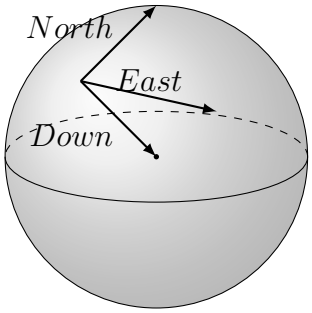


Figure 2.3: NED reference system

Body reference is centred on the rocket, x_b axis pointing towards the nose, y_b orthogonal to x_b, pointing towards the starboard side, z_b to complete.

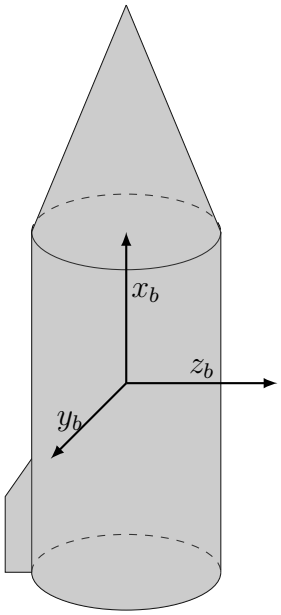


Figure 2.4: Body reference system

2.3 General setting

The following settings are implemented in both simulations:

- Earth radius = 6378000 m
- g constant at sea level = 9.81 m/s^2
- Air density at sea level = 1.225 Kg/m^3
- Temperature at sea level = 20° C
- Pressure at sea level = 101.29 kPa
- Geodetic reference: WGS 84
- Simulation time step = 1 ms

2.3.1 Time step considerations

Setting 1 ms as time step gives a good accuracy in terms of position.
To verify that decrease time step converge to an optimal trajectory a test has been done: the same simulation has been calculated with a first time step of 1 s, then it was decreased following this series:

$$k = \frac{1}{q} \tag{2.3}$$

where q goes from 1 to 100 with unitary steps.

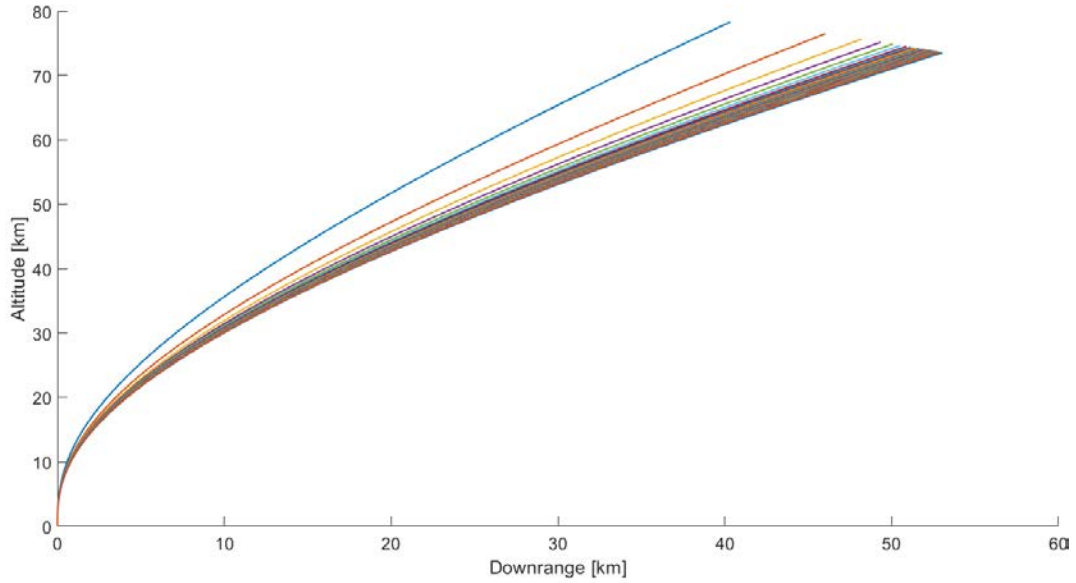


Figure 2.5: converging trajectory with decreasing time step

Thanks to this test, one aspect of the simulation has been validated; if a divergent plot would be obtained, it means that an error during the integration of ODE's system occurs and the dependence on step size was incorrect.
It is intuitive to understand that reducing time step size should converge the solution to the real one.

2.4 Falcon 9

CRS-5 mission has been taken as reference for the simulation, CRS (Commercial Resupply Service) are missions with resupply for the International Space Station (Sanat K.Biswas [7]).
The mission departed on 10/01/2015 from Cape Canaveral, with an orbit inclination of 51.56°.
The rocket used is the Falcon 9 v1.1, this leads to the following specifications:

- Mean Thrust = 5886000 N
- Burning Time = 180 s
- Mean Mass Flow = 1900 Kg/s
- Starting Mass = 520876 Kg
- Frontal Area = 11 m^2
- First stage mass = 23,100 Kg

2.4.1 Ascending

Integrating the system of ODE (2.1) from t=0 to t=tb with the following initial state vector:

$$\begin{bmatrix} 0m & 0m & 0m & 1.571rad & 520876kg \end{bmatrix}$$

and giving a little pitch control at t=15s to start gravitational turning, the following results have been generated:

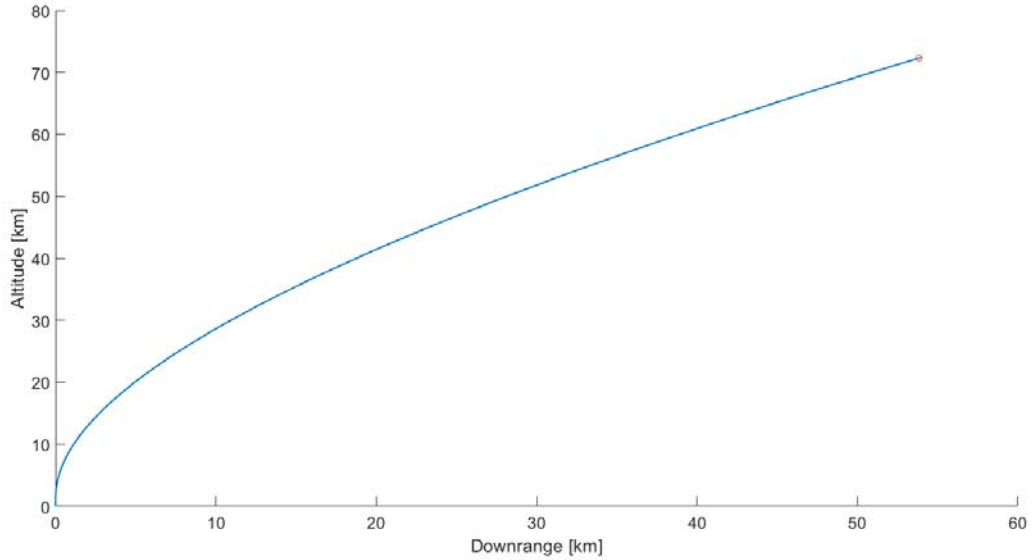


Figure 2.6: Altitude vs downrange for Falcon9 ascending

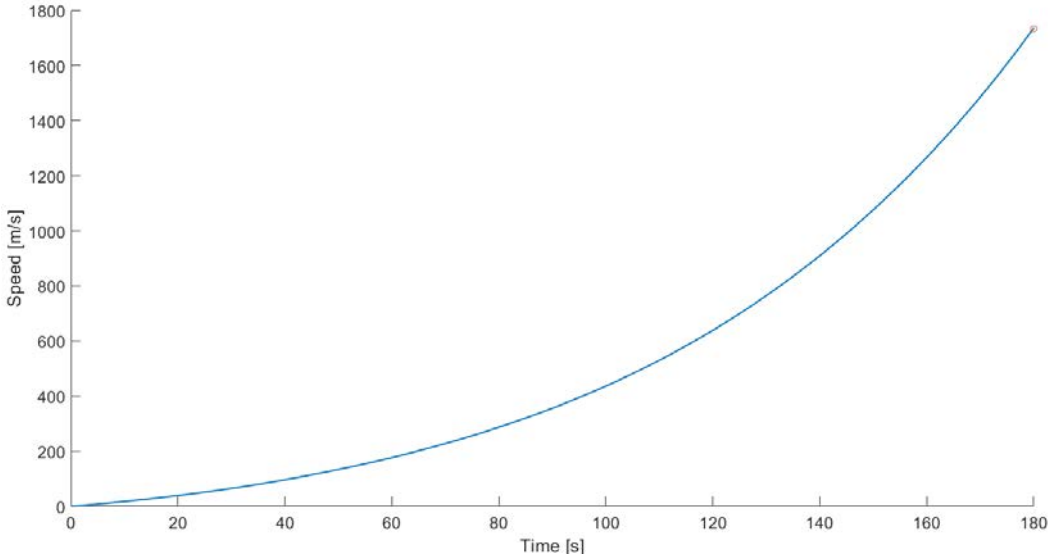


Figure 2.7: Speed vs time for Falcon9 ascending

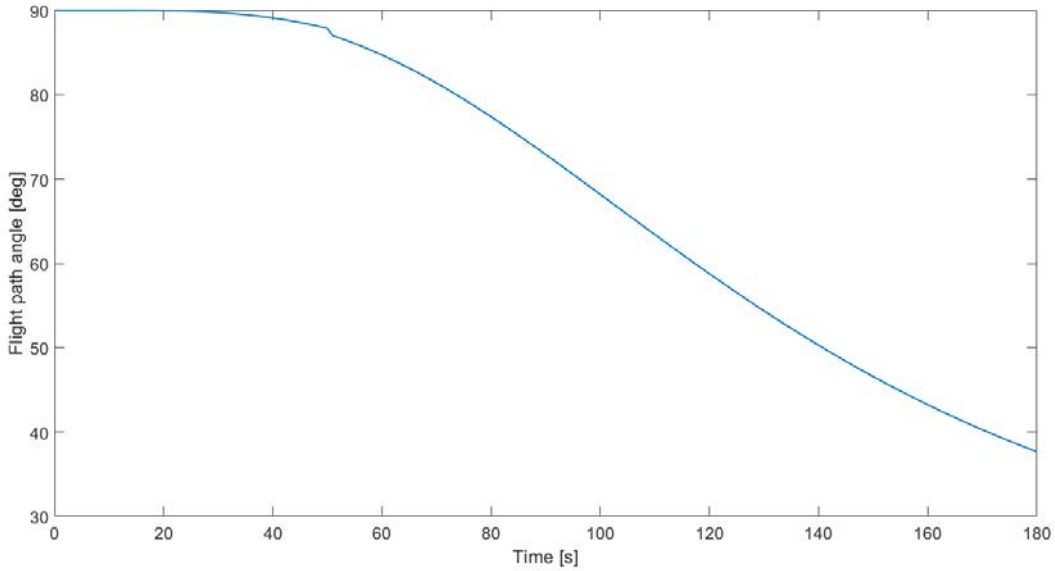


Figure 2.8: Flight path angle vs time for Falcon9 ascending

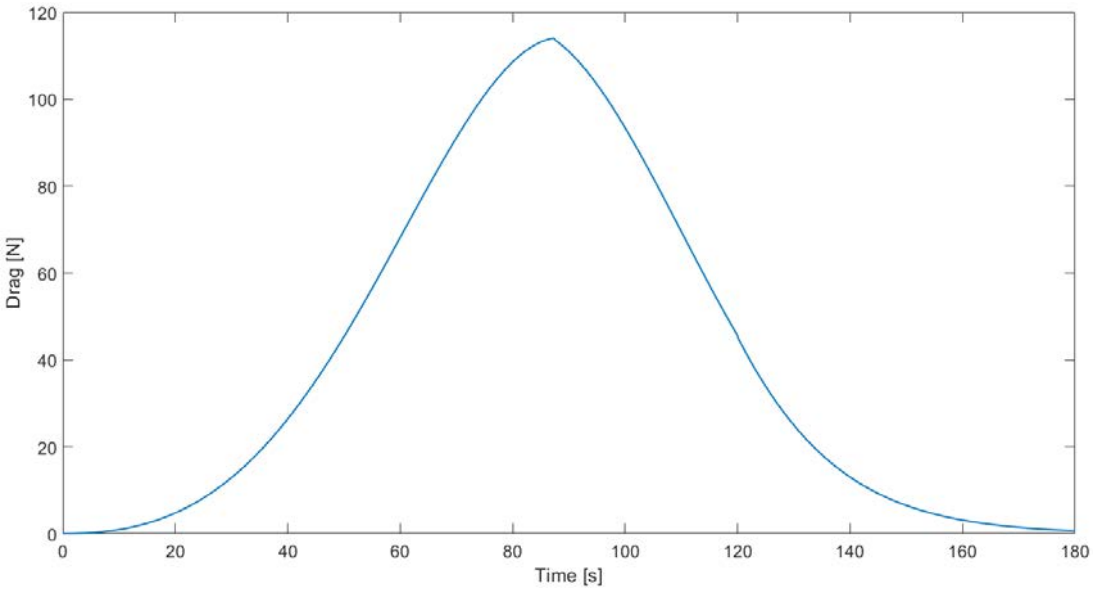


Figure 2.9: Drag vs time for Falcon9 ascending

2.4.2 Maneuvers

As the first stage cut off his engine, a series of events happen:

- Free flight (no thrust) for a few seconds
- Stage separation
- Flip maneuver
- Brake maneuver

Each one of these steps has been implemented with a loop that integrate different equation.
Free flight is simulated with the same system of ODE (2.1) but setting T=0 and $\dot{m}$=0.
Stage separation is just an instant loss of mass.
Flip maneuver consist in a rotation of the pitch angle, so while the centre of mass follow the same ODEs, the attitude here is controlled and pitch reaches value of 180°.
At this point the first stage is slowed down.

2.4.3 Descending

After all the maneuvers, first stage start his descending phase that consist in a controlled fall by using four grid fins that stabilize and slow the rocket.
Inside the code this has been simulated as an increase of the frontal area Aw, Again the same system of ODEs helps compute the trajectory.
Finaly the engine is powered on one last time to slow the stage for landing.

The complete trajectory (ascending, maneuvers and descending) is then plotted on Google Earth thanks to Matlab' s tool:

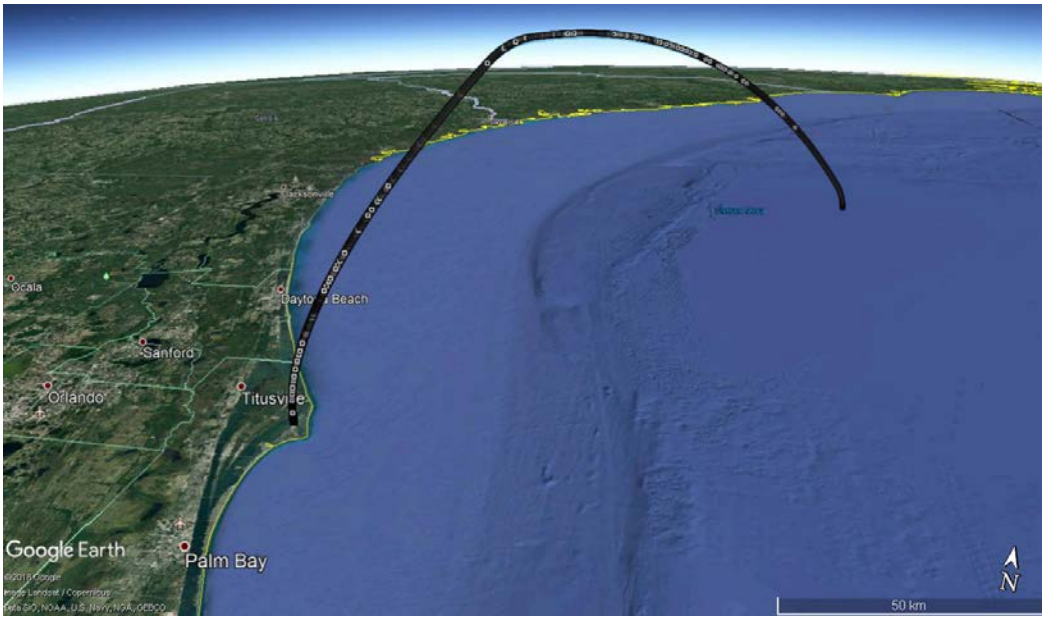


Figure 2.10: Complete Falcon 9 trajectory

The same graphics plotted for ascending are drawn again:

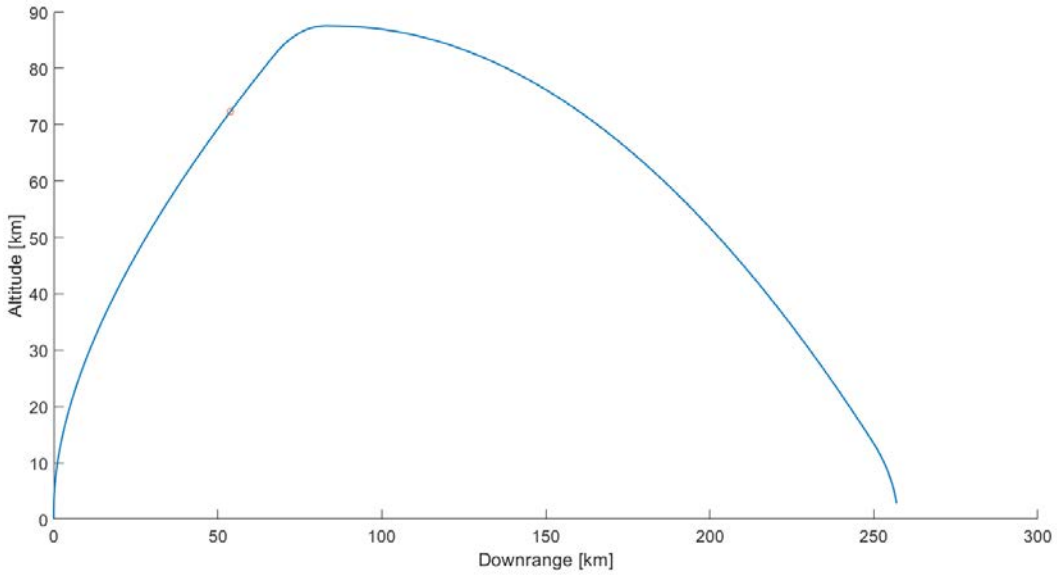


Figure 2.11: Altitude vs Downrange for Falcon9 ascending

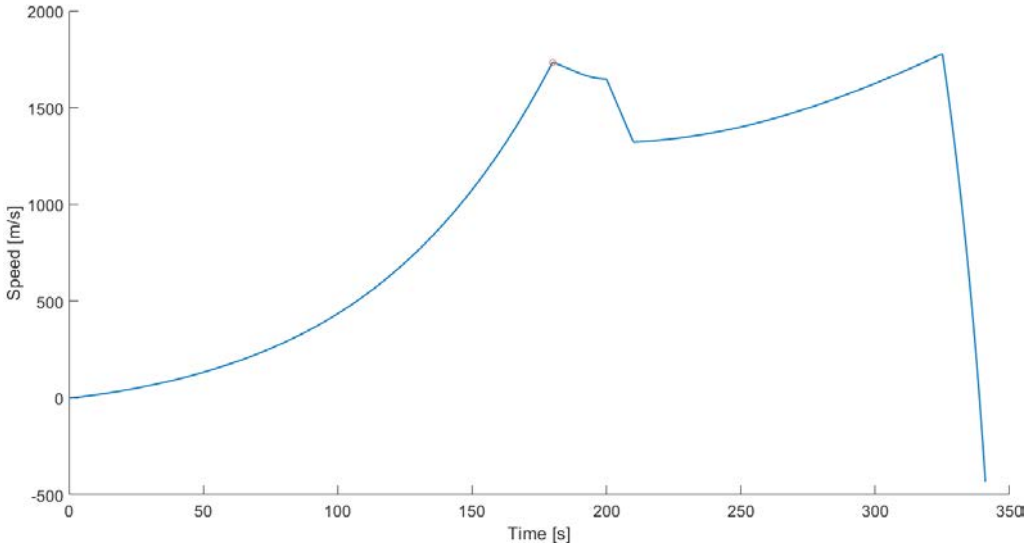


Figure 2.12: Speed vs time for Falcon9 ascending

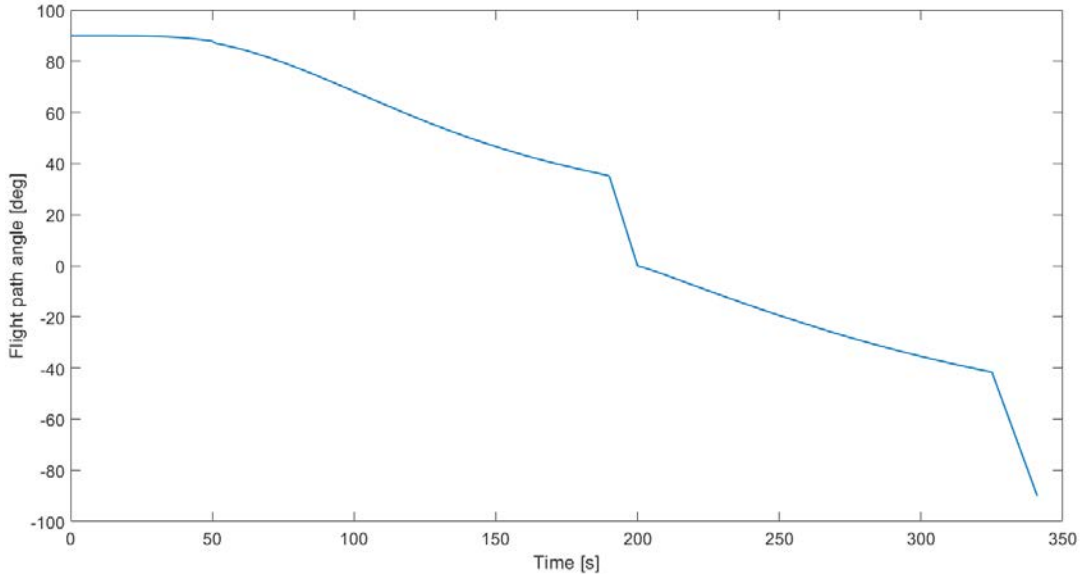


Figure 2.13: Flight path angle vs time for Falcon9 ascending

2.5 Sounding rocket

In this scenario the sounding rocket chosen as reference is the Black Brant XII, it is a four stage rocket capable of bring approximately 110 kg to 1500 km of altitude.

Figure 2.14: Black Brant XII

2.5.1 Ascending

Ascending phase for Black Brant XII models consists in 4 separate burn with different thrust and burning time. The following table shows the parameters used:

Stage	Thrust [kN]	Burning time [s]
1	516	6
2	45	20
3	111	35
4	467	17

For keeping the attitude, a spin is implemented in many models of sounding rocket, Black Brant XII need a spin too and it has been implemented with a parabolic trend, max spin reached is 3.2 rps:

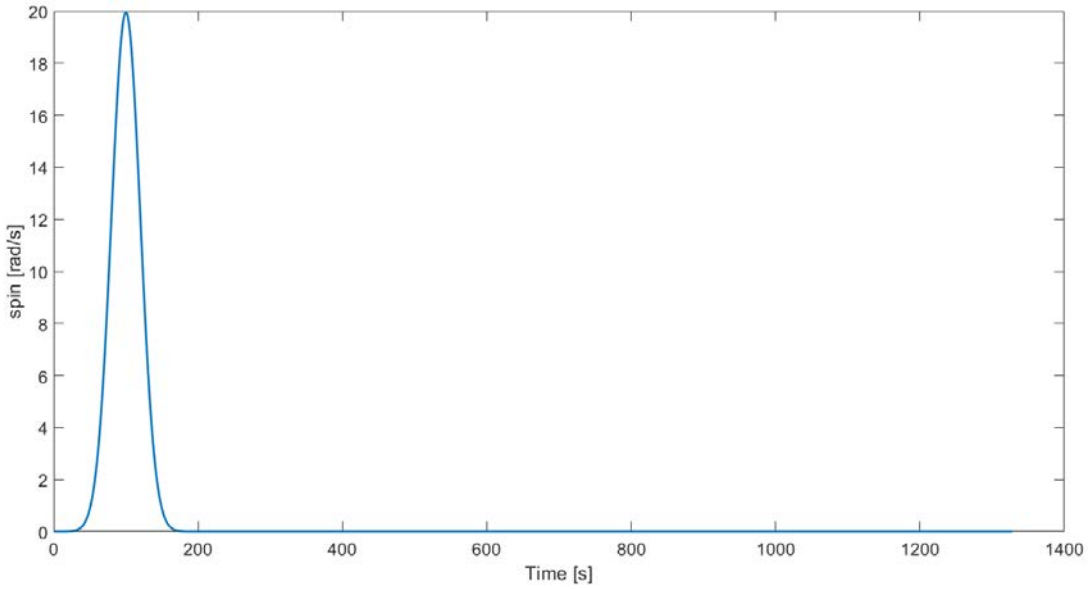


Figure 2.15: Spin trend

2.5.2 Descending

After all stage separations, the payload can now complete his parabolic trajectory, speed and flight path angle decrease and the only forces remaining are gravitational and drag (very low in the highest part of the parabola).
Again integrating the equations the following trajectory is obtained:

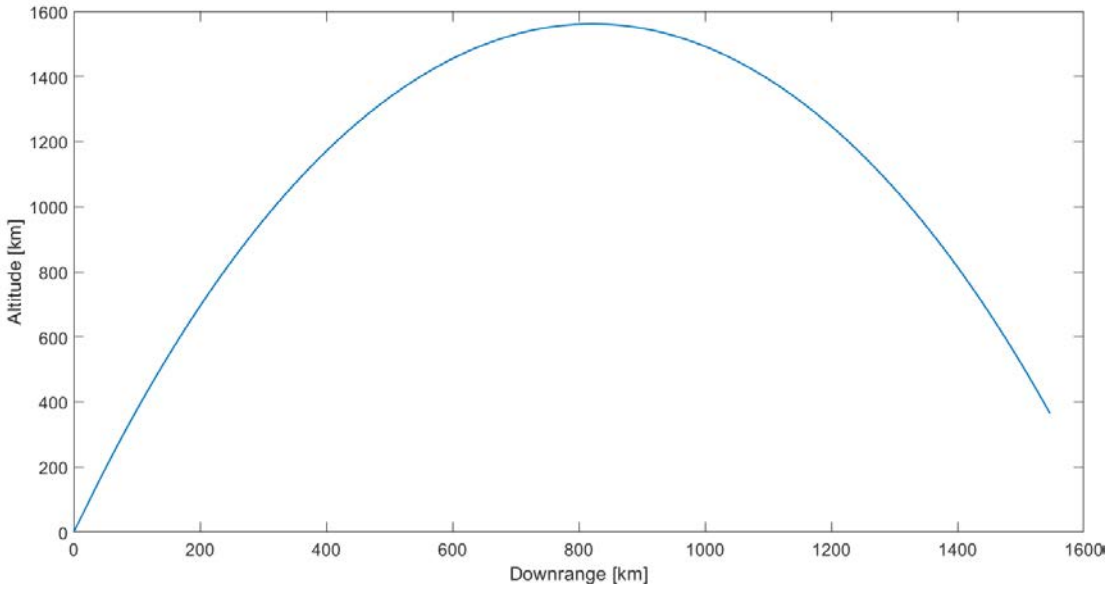


Figure 2.16: Altitude vs downrange for Sounding Rocket

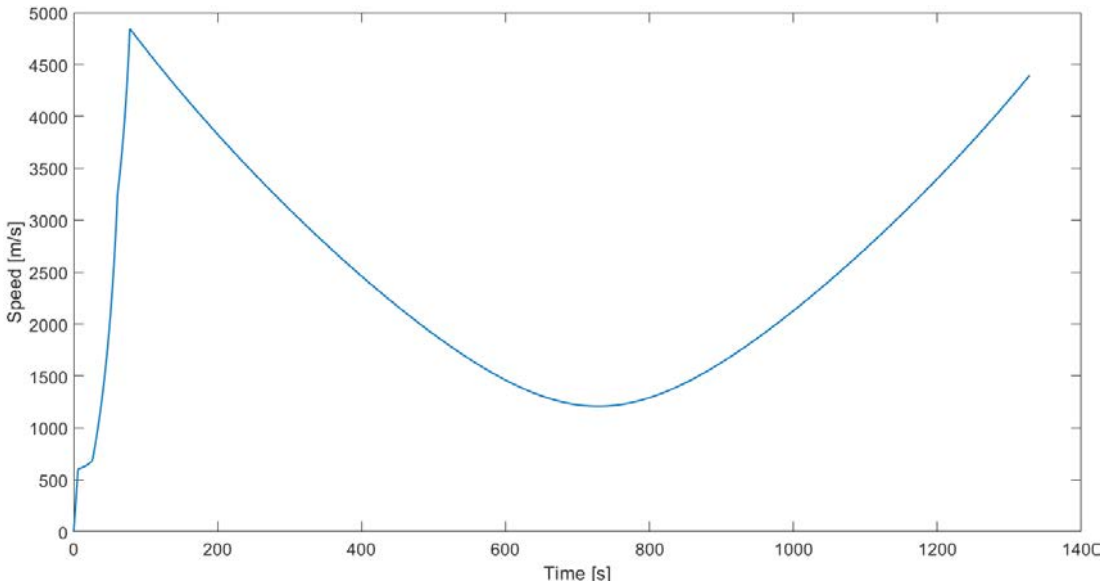


Figure 2.17: Speed magnitude vs time for Sounding Rocket

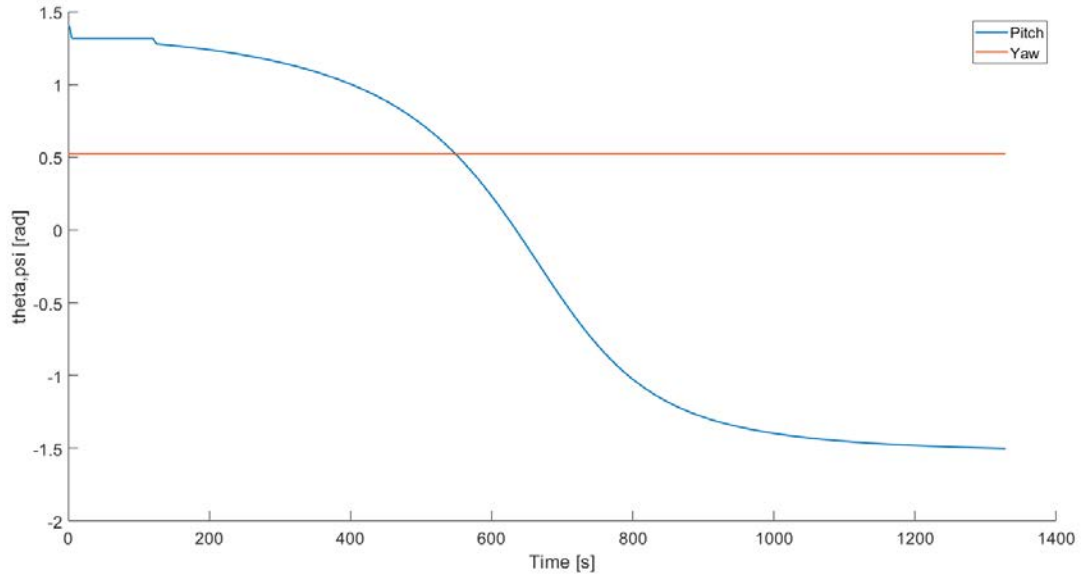


Figure 2.18: Pitch and Yaw vs time for Sounding Rocket

The yaw angle here is always zero because Spirent integrates it using the velocity vector, each step it takes the actual and the previous position, then calculates the yaw angle using the projection of the velocity vector.

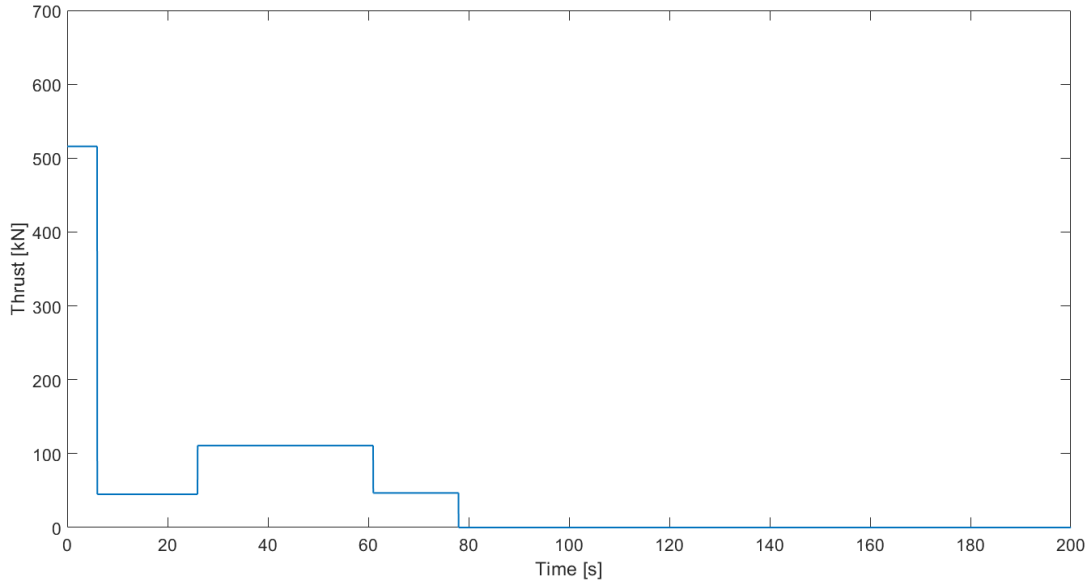


Figure 2.19: Thrust vs time for Sounding Rocket

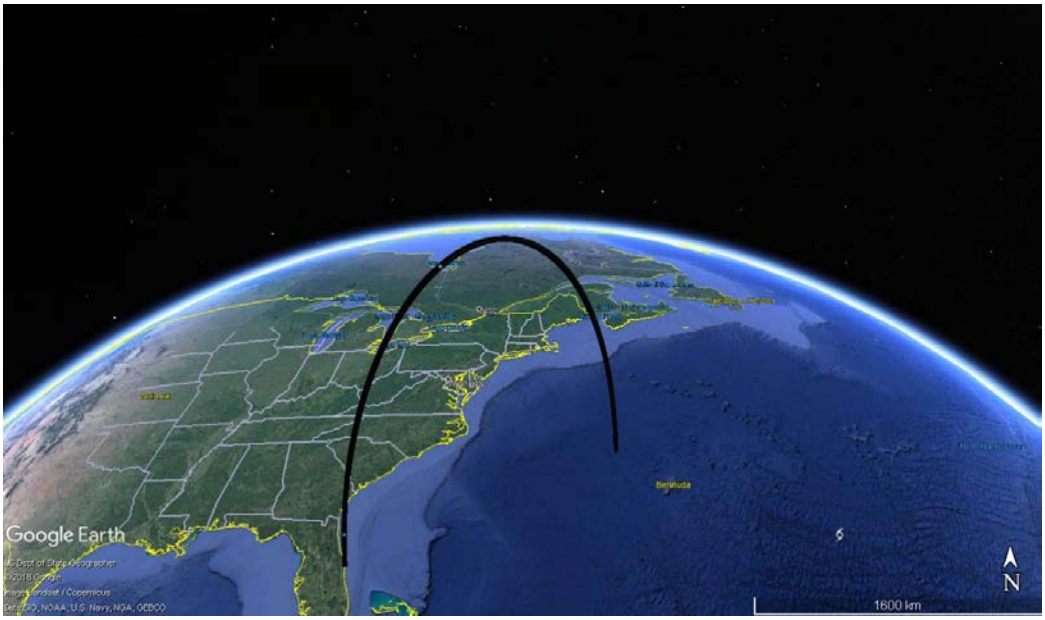


Figure 2.20: Plot on Google Earth for Sounding Rocket

The last part of the simulation, corresponding at the parachute deploy and slow descending, has not been implemented because is not interesting for the purpose of this thesis and many parameters (parachute dimensions, deploy altitude) are unknown.
These trajectory are considered representatives for the scope of the activity even if not all the details are available form NASA and Space X.

Chapter 3

Analysis of the GNSS signal dynamics

3.1 Tool description and outputs

The simulation of GNSS parameters in a complex scenario like the launch of a rocket is made possible thanks to five tools provided by Qascom S.r.l.
In this chapter they will be presented and their output are analysed.

3.1.1 GNSS Constellation Simulator

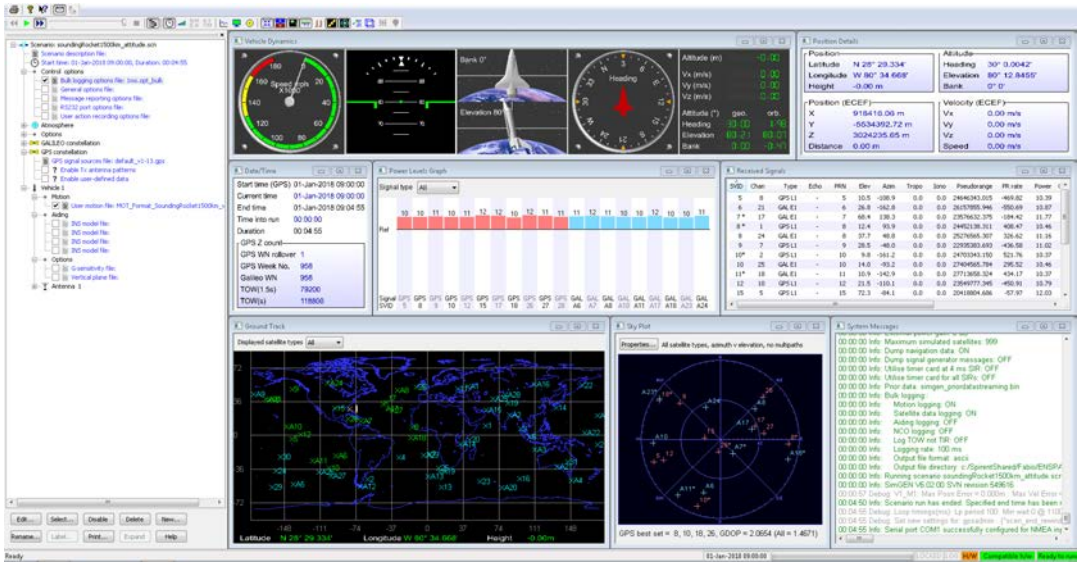


Figure 3.1: GNSS Constellation Simulator by Spirent

As anticipated before, SimGen (by Spirent) is a constellation simulator that simulates the GNSS signals as received in a receiver mounted in different vehicles; it can take in inputs the position, velocity, acceleration, jerk and attitude. Once the scenario is set all it provides all the emulated observables as generated by the receiver (number of satellite, pseudorange, doppler).

Parameters	**Settings for this simulation**
Constellations	GPS, Gal
Frequencies	L1, E1
Date UTC	1/1/2018 (Sounding rocket)
Vehicles	Spacecraft
Iono correction	No
Tropo correction	No
Elevation Mask	10°

One of the strength of this application is the variety of scenarios that it can simulate, in fact all these parameters can be widely different: constellations can also

include any other GNSS system, for example Glonass and Beidu; vehicles include aircraft, trains, cars; elevation mask depends on the antenna position; correction may be implemented.
To calculate all the satellites positions, Spirent uses a series of Keplerian parameters, known at some fixed dates, and propagates the orbits for all the simulation time.
Once set all the receiver information it generates as output all the observables (number of satellite, pseudorange, doppler), it also visualize the sky plot for altitude and azimuth, so the user has a quick view of the visible satellites position in the sky.
Spirent also interpolate the inputs if requested by the users, this is useful in case off different step size between inputs and simulation or in case of specific parameters like heading angle.

3.1.2 GNSS Constellation Simulator outputs

This application gives as output two files: Sat_Data.csv and Motion.csv. The first one contains all the observables and data for each satellite:
Time (ms), Channel, Sat_type, Sat_ID, Sat_PRN, Sat_Pos, Sat_Vel, Azimuth, Elevation, Range, Pseudorange, Doppler.
The second one contain position, velocity, acceleration and asset of the vehicles. Both files are in CSV (comma separated values) format, in this layout they are not easily legible by the user, so others tool need to be used for further elaboration and presentation.

3.1.3 Dynamic Analysis Tool

Analysis Dynamics is a useful tool to process SimgGen's output, it takes as input the SAT_DATA file and re-elaborate it to plot in a more friendly way all the important data like number of satellite vs time, visibility time for each constellation, Doppler amplitude vs frequency, etc. To do so, it need verify a very important value: C/N0.

C/N0 is the Signal-Noise rate normalized with the carrier, to evaluate this number the program requests a series of parameters:

Parameters	**Settings for simulation**
GPS Power	-160 [dBW]
Gal Power	-152 [dBW]
C/N0 Threshold	35 [dBHz]
Receiver Temperature	290 [K]
Sky Temperature	100 [K]
Antenna Loss	0.5 [dB]
Cable Loss	0.3 [dB]

Thanks to this data, and those provided by SimGen, the tool computes the noise value depending on the distance, the front end, electronic and temperature characteristics; than relates it with the signal power and discards all satellites with C/N0 $<$ C/N0 Threshold. Now, with the data of the remaining satellites, the analysis are performed.

3.1.4 Dynamic Analysis Tool outputs

The firsts interesting graphics are the visibility histograms, on the vertical axis there is the time percentage, so in this simulation 100% equals to 1276 s. On the horizontal axis there are the numbers of visible satellites.
Comparing Figures 3.2, 3.3 and 3.4, the first and more obvious advantage of a multi GNSS receiver comes out: a major number of visible satellites, this means more pseudorange measurements that allows to achieve a more accurate position.

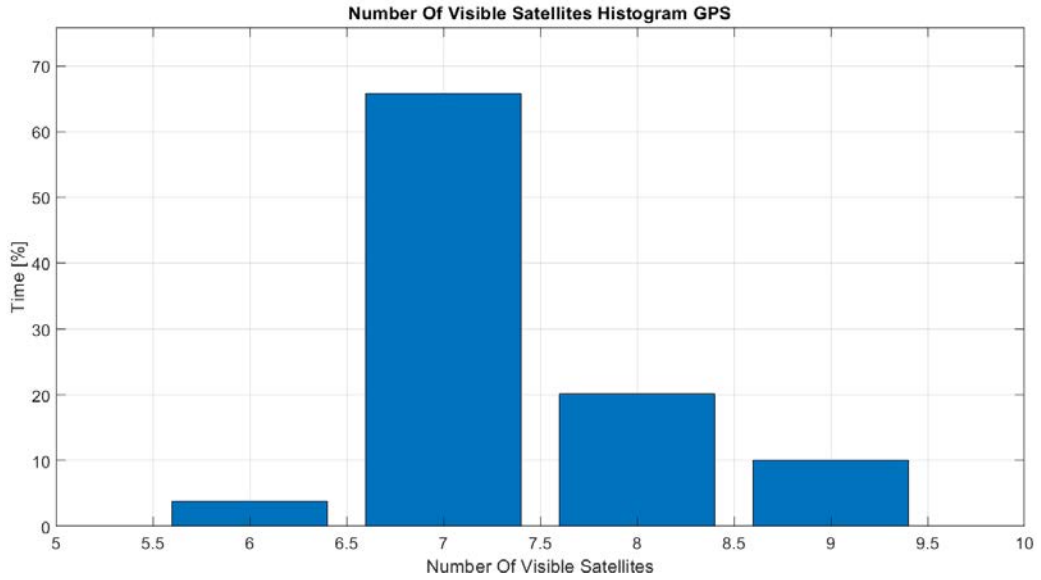


Figure 3.2: GPS visibility time histogram

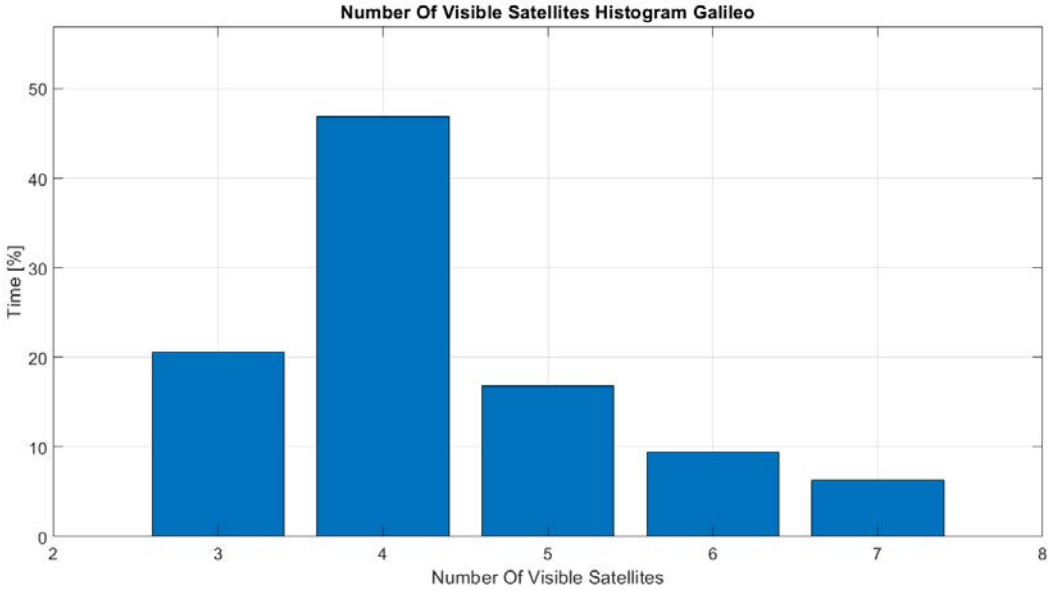


Figure 3.3: GAL visibility time histogram

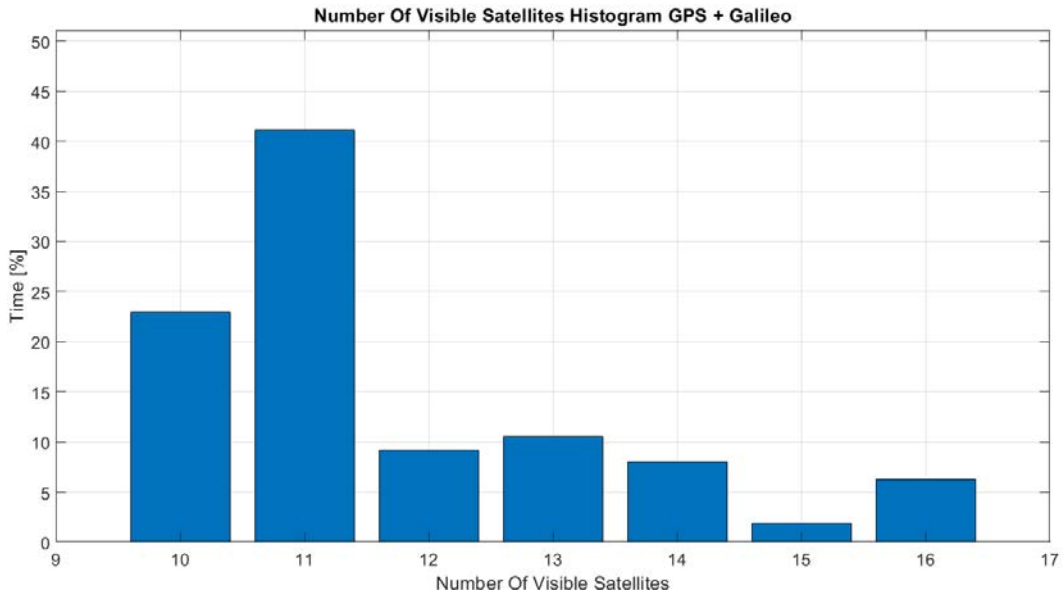


Figure 3.4: Mixed (GPS+Gal) visibility time histogram

Than many others variables are plotted, but here in the following only two more are reported: the Probability density function of the Doppler and Doppler rate values, those values are interesting to assess the range of speed and accelerations of the rocket.

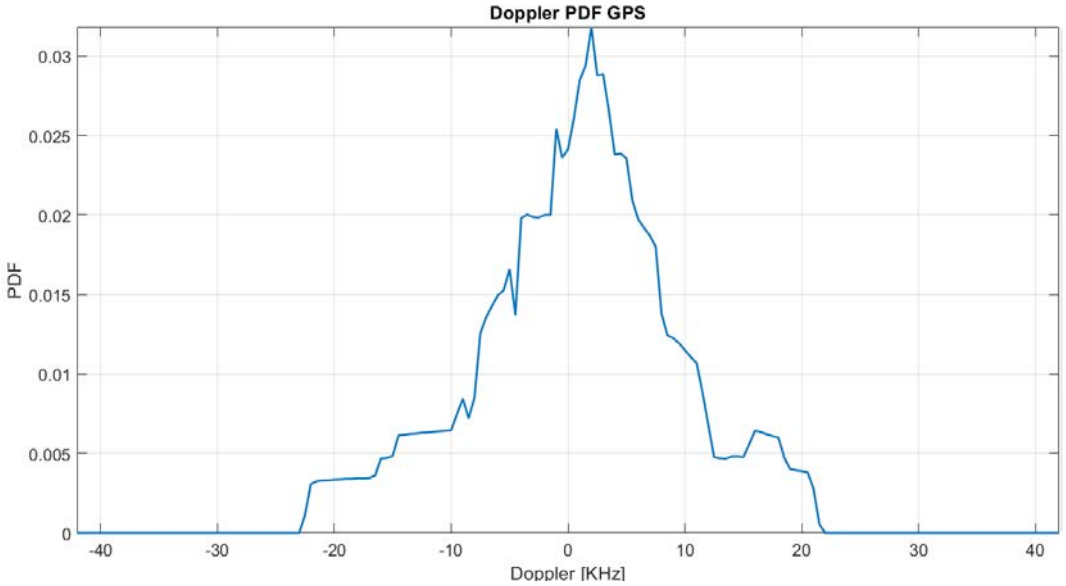


Figure 3.5: GPS Doppler PDF

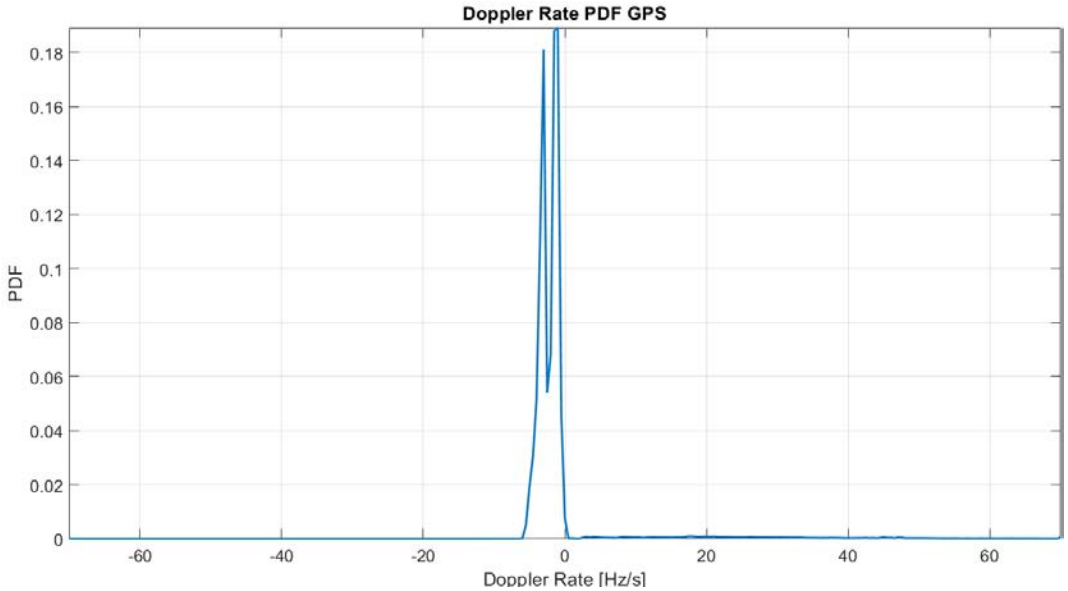


Figure 3.6: GPS Doppler rate PDF

Only GPS data are reported because Galileo data are similar and do not add any more information.

3.1.5 PVT Engine

The PVT algorithm (Position Velocity Time) provides the elaboration of three main input:

- Time.
- Observables.
- Ephemeris.

to gives as output:

- ECEF and LLA position.
- Errors.
- Clock bias.
- DOP.

The process iterates these main steps:
First calculate the a priori travel time, than subtract it at the received time to know the transmission time T_{Trans}, now the satellites position at T_{Trans} can be evaluated through the ephemeris. At this point the least square algorithm is implemented (Caporali [8]):

- Computation of the partial derivative matrix 'H', they are the difference between Satellites positions and the a priori coordinates divided by the a priori ranges.
- Transposition of 'H', gives H^T.
- H^T times H, gives $H^T H$.
- $H^T H$ inversion, gives $H^T H^{-1}$.
- H^T multiplied by receiver bias a priori, gives $H^T Y$.

- and last $H^T H^{-1}$ multiplied by $H^T Y$, gives the four delta values to correct positions and receiver bias.

The process is iterate until convergence, for further explanations see (Caporali [8]).
The algorithm needs at least four satellite to work.
A σ value can be setted to simulate an error on the pseudoranges.

3.1.6 PVT outputs

To compare single-constellation results with multi-constellation, PVT provides four graphics:

- 3D positioning error
- Root mean square error
- DOP
- Error Variance

The first one is the difference between true trajectory and measured by GNSS systems.
Rms (Root mean square) error is the square root of the mean square (the arithmetic mean of the squares of a set of numbers).
DOP (Diluition of precision) is an important value in satellite navigation, it is correlated with the invertibility of the $H^T H$ matrix, the main one is PDOP (Position Diluition Of Precision) and is strictly connected with geometry and number of satellite, value of PDOP > 3 are not acceptable. Practically if the weight unit for the pseudorange measures is 3 metres and PDOP $= 2$, the radius of the circumference at 1σ is six metres. It is possible to obtain others DOP values by representing the inverse $H^T H$ matrix in a local system like NED, HDOP (Horizontal) VDOP (Vertical) TDOP (Time) GDOP (Global) (Caporali [8]).
Variance is the square absolute error minus the average error.

GPS only:

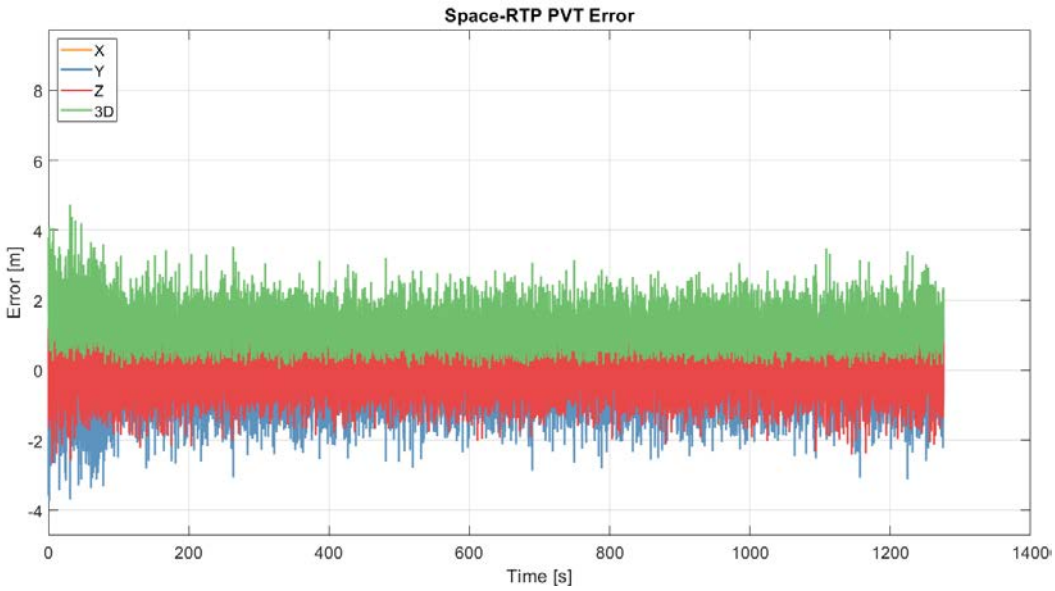


Figure 3.7: Gps only error vs time

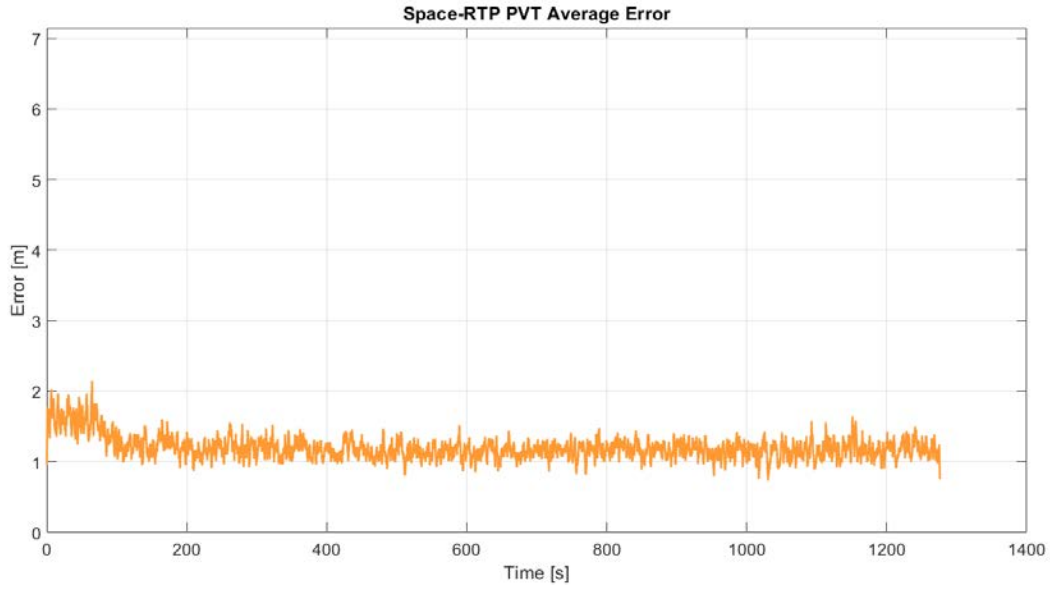


Figure 3.8: Gps only Average Error vs time

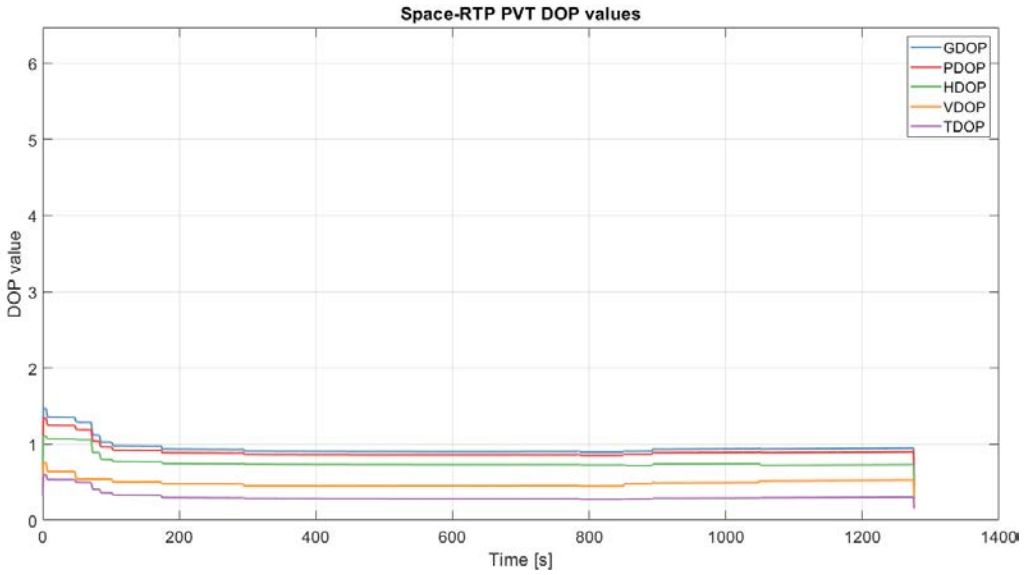


Figure 3.9: Gps only DOP vs time

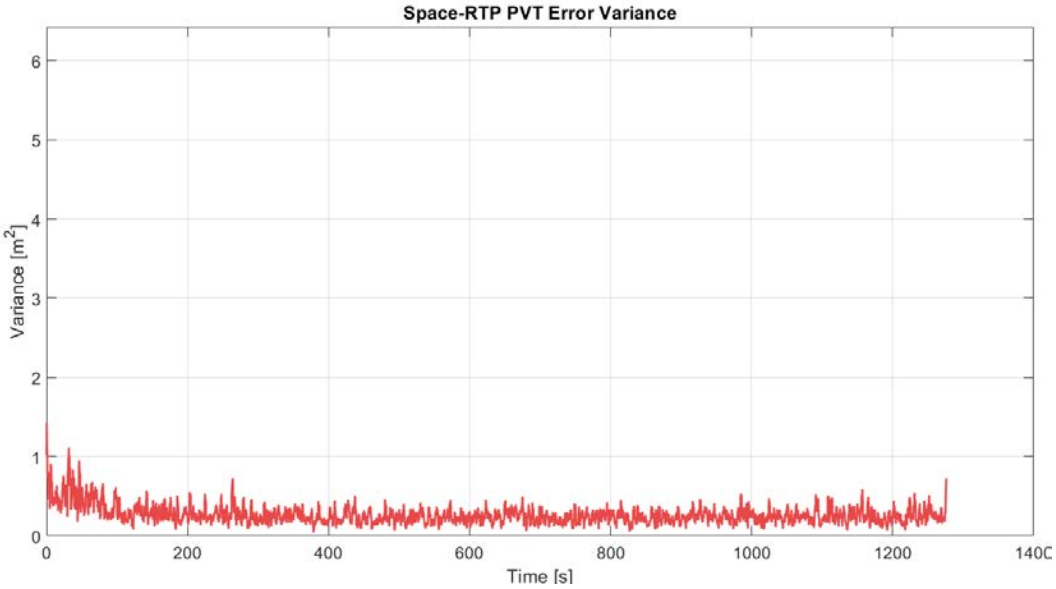


Figure 3.10: Gps only Variance vs time

Galileo only:

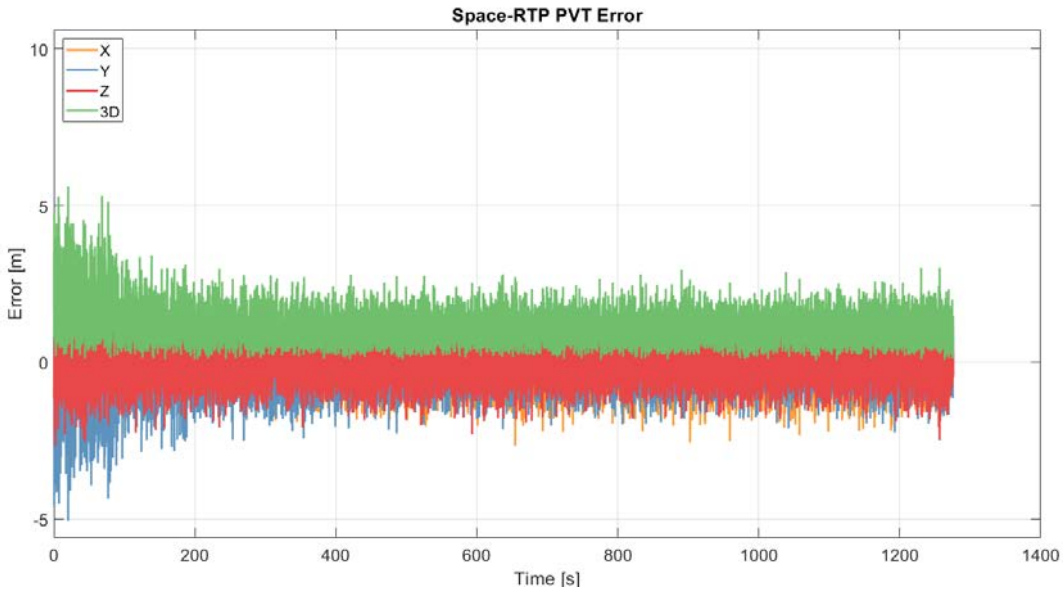


Figure 3.11: Galileo only error vs time

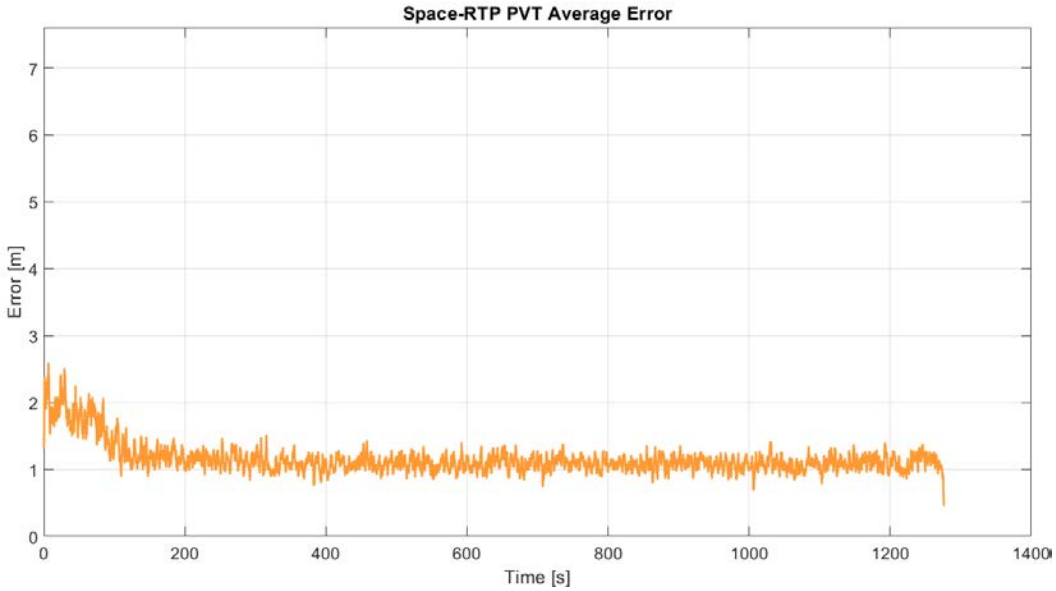


Figure 3.12: Galileo only Average Error vs time

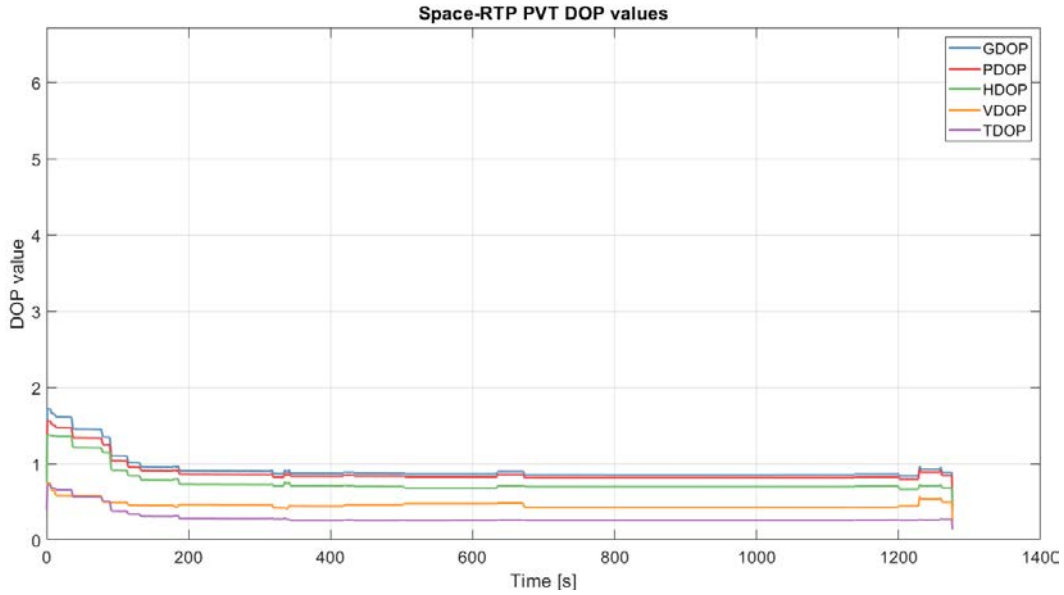


Figure 3.13: Galileo only DOP vs time

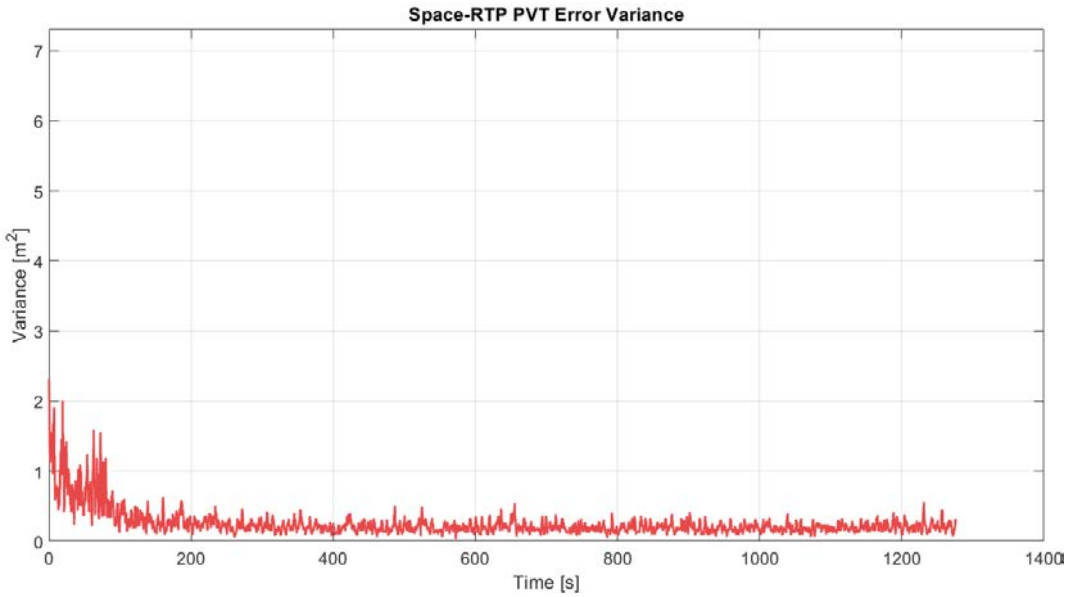


Figure 3.14: Galileo only Variance vs time

GPS + Galileo:

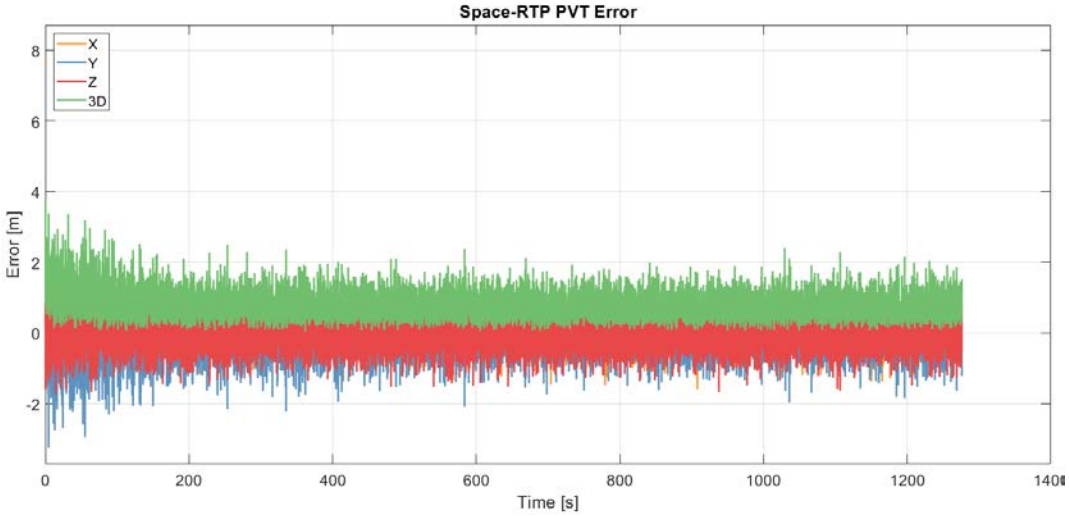


Figure 3.15: Galileo + GPS error vs time

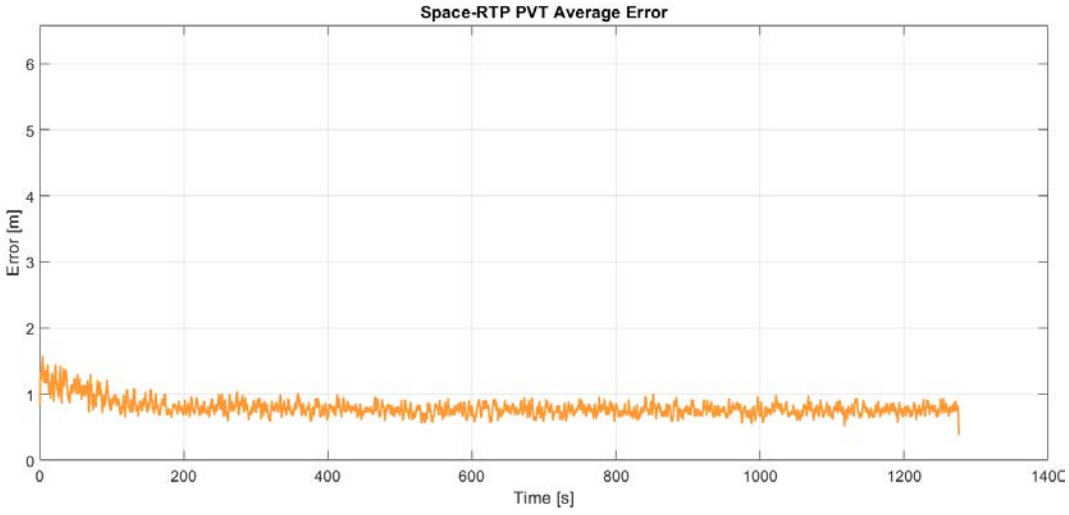


Figure 3.16: Galileo + GPS Average Error vs time

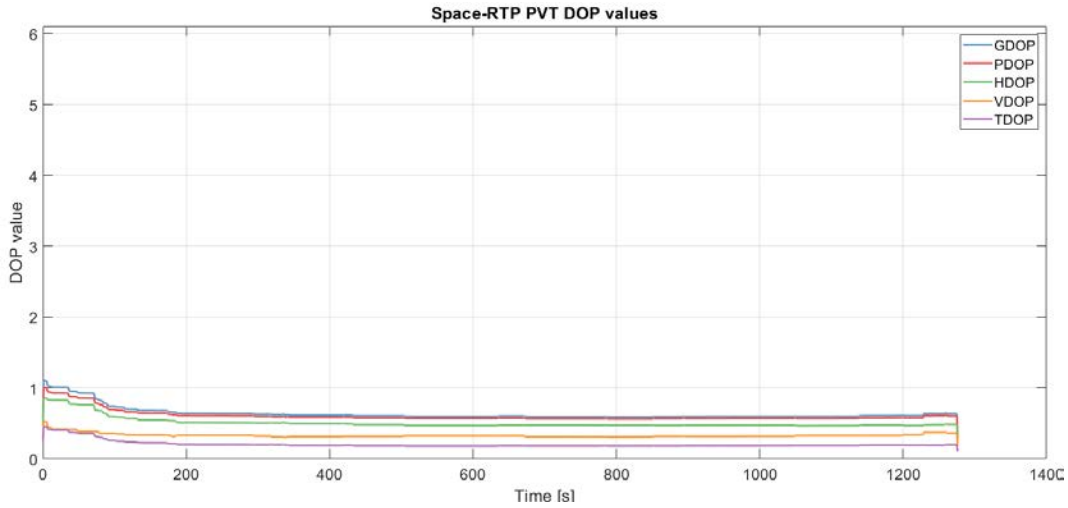


Figure 3.17: Galileo + GPS DOP vs time

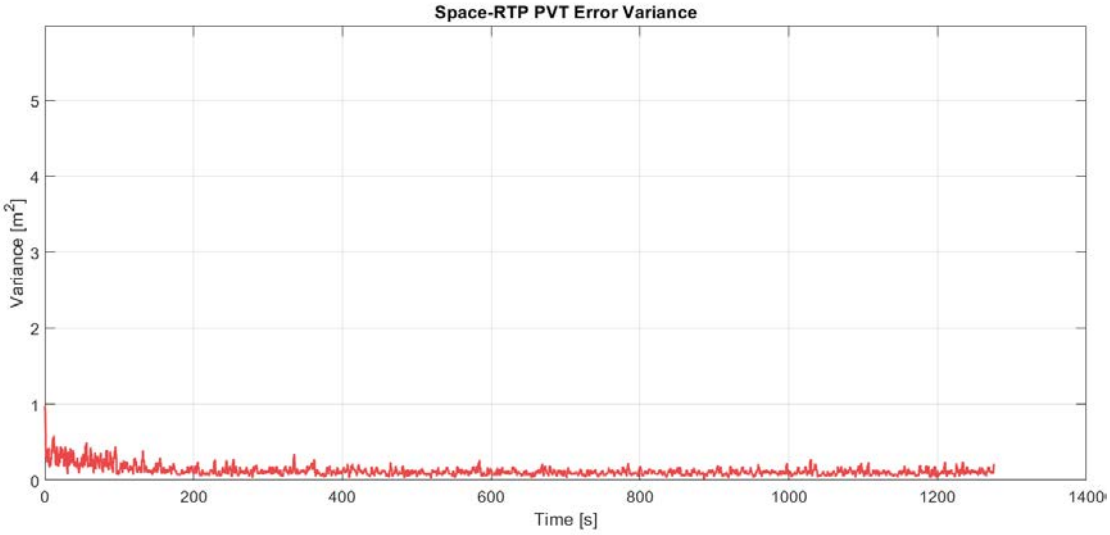


Figure 3.18: Galileo + GPS Variance vs time

3.1.7 Semi-analytic simulator

Simulator Architecture:

The simulator is mainly a multichannel tracking loop and data demodulation simulator, that works after the correlators bank. This tool simulates the receiver

tracking loops; the simulator has been used since the launcher system is operating in a high dynamic environment, it is subjected by high value of doppler, doppler rate and jerk, and it is therefore important to verify the capability of the receiver to keep the signal well tracked.
The alignment between the two PRN codes, received and copy inside the receiver, is represented through a autocorrelation function peak, increasing the matching between the two PRN means increasing this peak.

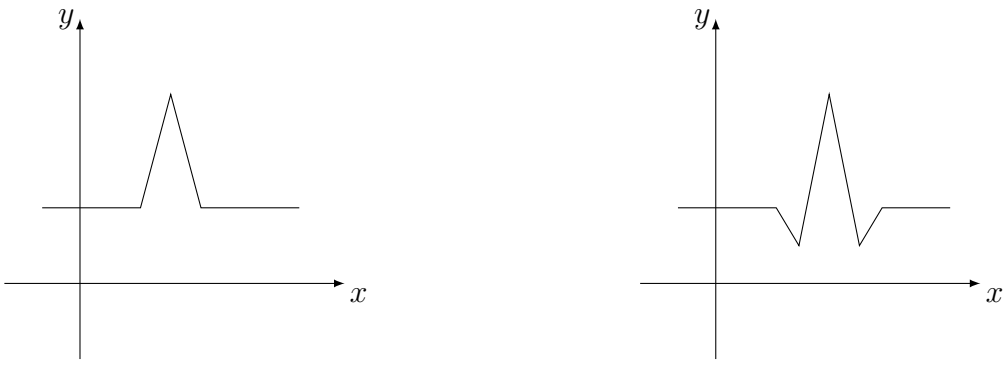


Figure 3.19: GPS (left) and Galileo (right) correlation peak

The three main processing block of a GNSS receiver are: Acquisition, tracking, PVT. The first one compares the received code and the replica inside the receiver to obtain the correlation figure (3.19); once the acquisition is done, tracking loops keep it locked in case of high dynamic environment or interference. At last the PVT algorithm is implemented (Kaplan [9]).
The semianaltyical simulator emulates the main components of a GNSS receiver (Acquisition, Tracking and Data demodulation). It computes as first step the Autocorrelation function of GPS or Galileo signals that is then used to emulate the tracking loop processing by adding at each simulation step the deynamics and power changes of the signal in a very realistic way. :

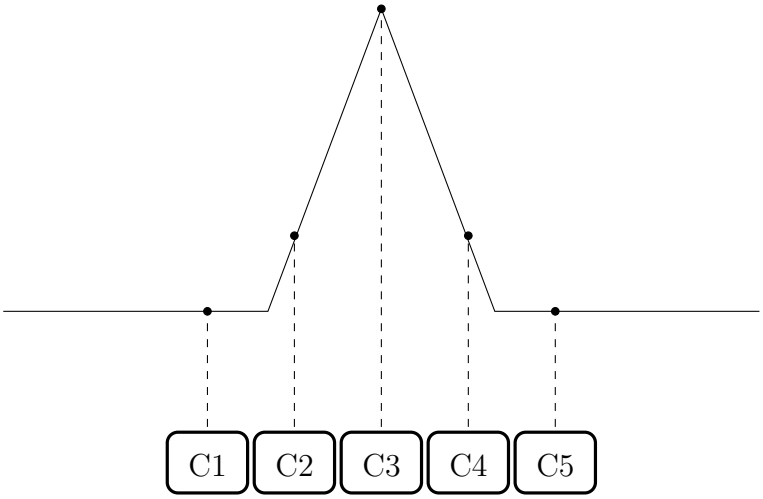


Figure 3.20: Correlator on GPS signal

After the acquisition is done the first time (only one time in this tool), the correlators are setted; C2 is called "early", C3 is "prompt", C4 is "late".
The dynamics shift this signal in time and frequency, if for example the code is shifted in time, one between C2 and C4 correlators is becoming higher in value than the prompt, the DLL observe this and apply a correction. This works also for the others 2 tracking loops but with different checks.

Sizing the receiver's tracking loops means set 4 main parameters:

- Integration time
- DLL bandwidth
- PLL bandwidth
- FLL bandwidth

The first one is the step of the integration, having a high integration time may lead to a higher peak but what happens between the two steps in unknown, there might be negative value that influence the peak amplitude, so it's necessary to have a low integration time that still produces a high enough peak.
DLL (Delay lock loop) aims at tracking the code delay of the incoming GNSS signal. The DLL provides a correction of the current observed delay, and this

correction is applied to the local replica code generators, in order to keep the local replica as "matched" as possible with the incoming signal.

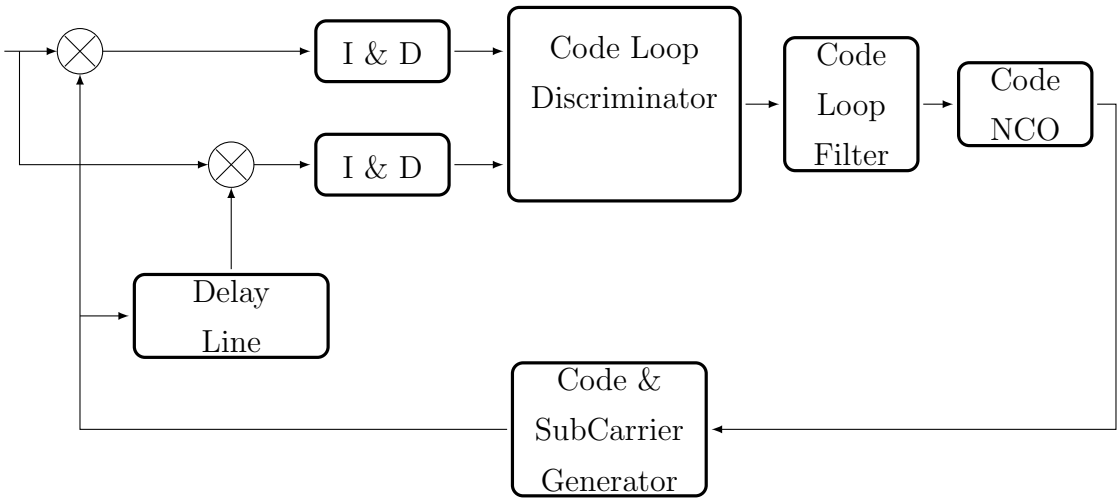


Figure 3.21: DLL high level tracking loop

PLL (Phase lock loop) aims at tracking the phase of the incoming GNSS signal. The PLL provides a correction of the phase in a continuous loop, generating a phase error signal.
FLL (Frequency lock loop) aims at tracking the frequency of the incoming GNSS signal. The FLL provides frequency corrections in a continuous loop, generating a frequency error signal.

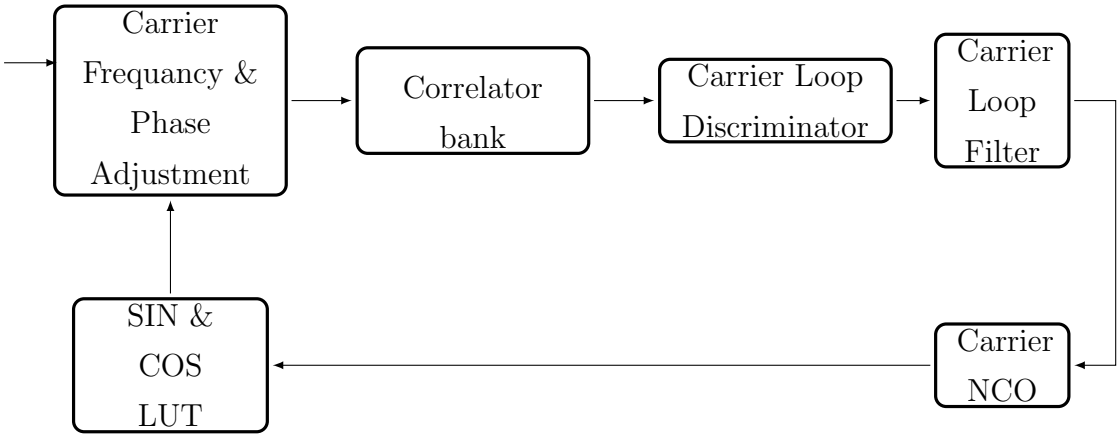


Figure 3.22: PLL and FLL High Level tracking loop

It is not the purpose of this thesis to go deeper inside the explanation of the tracking loops, just a few consideration: all three loops have a filter, the bandwidths setted inside Semi_Analytic are the bandwidths of those filters, sizing them is not an easy task and there is an important paradox to consider, in fact to tolerate dynamic stress, the predetection integration time should be short, the discriminator should be an FLL, and the carrier loop filter should be wide, However, for the carrier measurements to be accurate (have low noise), the predetection integration time should be long, the discriminator should be a PLL, and the carrier loop filter noise bandwidth should be narrow. In practice some compromise must be made to resolve this paradox.

3.1.8 Semi-analytic simulator outputs

Setting the receiver tracking loops with value of a typical space receiver, and choosing to operate in the worst possible scenario by setting the higher Doppler rate and jerk obtained from Analysis Dynamic, $\begin{bmatrix} 24607Hz & 160Hz/s & 0.1Hz/s^2 \end{bmatrix}$ these results are acquired:

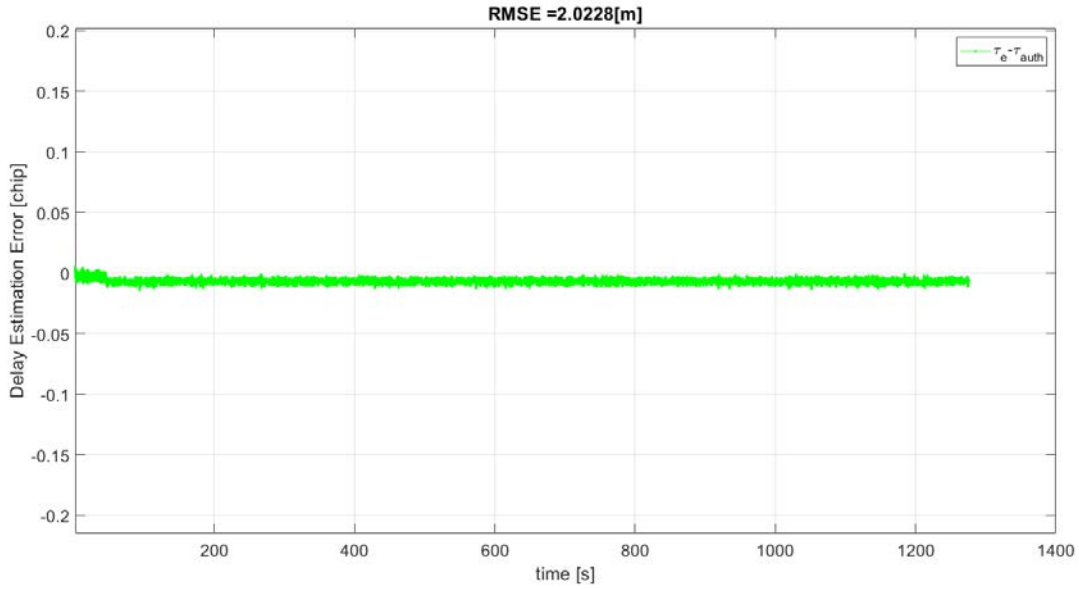


Figure 3.23: Delay error

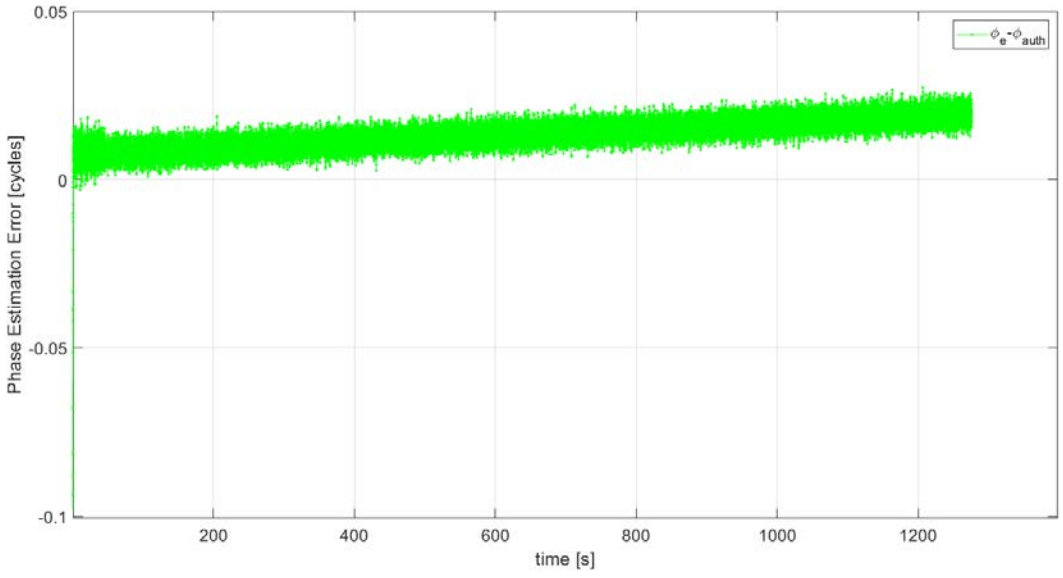


Figure 3.24: Phase error

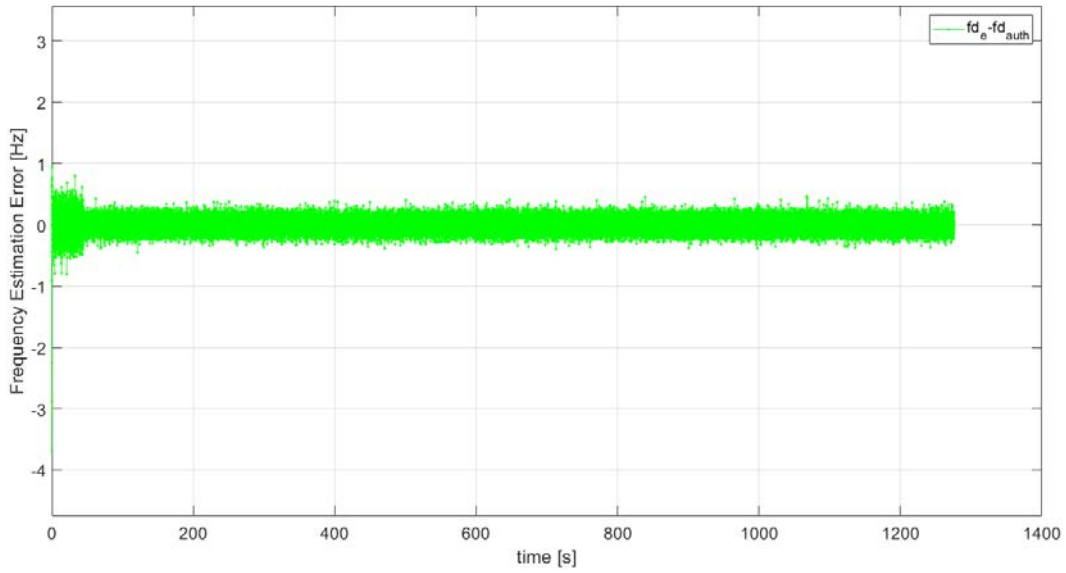


Figure 3.25: Frequency error

These graphics show that the receiver is capable of keeping the signal tracked, errors are all small, even the drift in phase estimation due to third order dynamics.

3.1.9 Inertial System Emulator

GenerateMeas is the tool's name that provides the INS and errors simulation. It takes as input a trajectory file in NED format, than simulate the INS system and apply an error model. To do that it calculates the linear acceleration and attitude in the NED coordinates system, than rotate them in the Body frame since this is the reference system where the INS operates; now an error model is implemented: a standard deviation σ and a bias value are required for both gyro and accelerometers, from starting values of acceleration and attitude, a random walk that contain bias and deviation is applied at the calculated data. [10]
For gyro:

$$\theta(t) = \epsilon \cdot t \tag{3.1}$$

$$\sigma_\theta(t) = \sigma \cdot \sqrt{\delta t \cdot t} \tag{3.2}$$

where ϵ is bias, σ is the white noise standard deviation.
For accelerometers:

$$s(t) = \epsilon \cdot \frac{t^2}{2} \tag{3.3}$$

$$\sigma_\theta(t) = \sigma \cdot t^3/2 \cdot \sqrt{\frac{\delta t}{3}} \tag{3.4}$$

As outputs GenerateMeas gives the true and the INS trajectory, these data, together with PVT data, will be sent into a Kalman filter that will combine them in a loosely coupled integration.

3.1.10 Kalman Filter

The QAVE is a computation engine simulator that exploits a Kalman Filter (KF) technique in order to blend GPS data with IMU data. GPS data consist of time-stamped position estimates, whereas IMU data consist of time-stamped linear accelerations and angular rates estimates.
The main steps of the Kalman filtering are:

1. **Initialization**: based on system model and external information, the state and covariance matrix must be initialized. Assuming that the initialization occurs at time t=0, the following variables must be provided:
   (a) Initial state estimate s_0
   (b) Initial covariance matrix estimate P_0
2. **Kalman Loop**: each iteration is based on two parts:
   (a) **Time update (Prediction)**. Propagate the previous state s_t and covariance matrix P_t, associated to time instant t, up to the current time instant k, k>t. This prediction is based on past history of the system, on the state transition model represented by the matrix $F_{(k,t)}$ (linear

relationship is assumed between state at time instant k and state at time instant t), and on the process noise represented by the covariance matrix $Q_{(k,t)}$ (the process noise is used to model possible parameter uncertainties and un-modelled dynamics), through the equations:
State prediction: $\mathbf{s}_{k|t} = \mathbf{F}_{t,k} \cdot \mathbf{s}_{t|t}$
Covariance prediction: $\mathbf{P}_{k|t} = \mathbf{F}_{t,k} \cdot \mathbf{P}_{t|t} \cdot \mathbf{F}_{t,k}^T + \mathbf{Q}_{t,k}$

(b) **Measurement update (Correction)**. Merge the predicted state with the new observed measurement(s). This update is based on the predicted state and covariance, on the observation model represented by the matrix H_k (linear relationship is assumed between state at time instant k and measurement(s) at time instant k), on the current measurement(s) m_k and on their accuracies represented by the covariance matrix R_k, through the equations:
Innovation or measurement pre-fit residual: $\epsilon_{k|t} = \mathbf{m}_k - \mathbf{H}_k \mathbf{s}_{k|t}$
Innovation covariance or pre-fit residual covariance: $\mathbf{E}_k = \mathbf{R}_k + \mathbf{H}_k \mathbf{P}_{k|t} \mathbf{H}_k^T$
Kalman gain matrix: $\mathbf{K}_k = \mathbf{P}_{k|t} \mathbf{H}_k^T \mathbf{E}_k^{-1}$
State update: $\mathbf{s}_{k|k} = \mathbf{s}_{k|t} + \mathbf{K}_k \epsilon_{k|t}$
Covariance update: $\mathbf{P}_{k|k} = \mathbf{P}_{k|t} - \mathbf{K}_k \mathbf{E}_k \mathbf{K}_k^T = \mathbf{P}_{k|t} - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_{k|t}$
Measurement post-fit residual: $\epsilon_{k|k} = \mathbf{m}_k - \mathbf{H}_k \mathbf{s}_{k|k}$

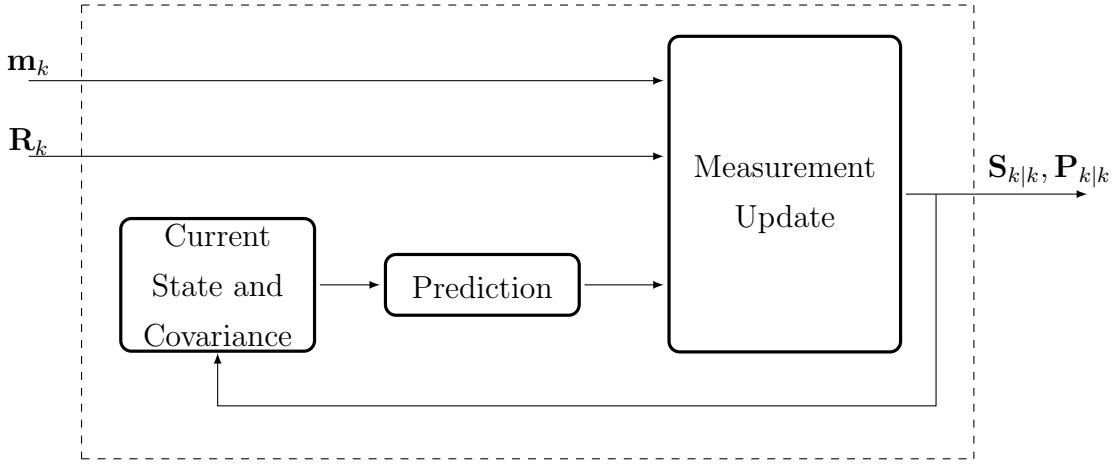


Figure 3.26: Kalman filter representation

3.1.11 Kalman Filter Output

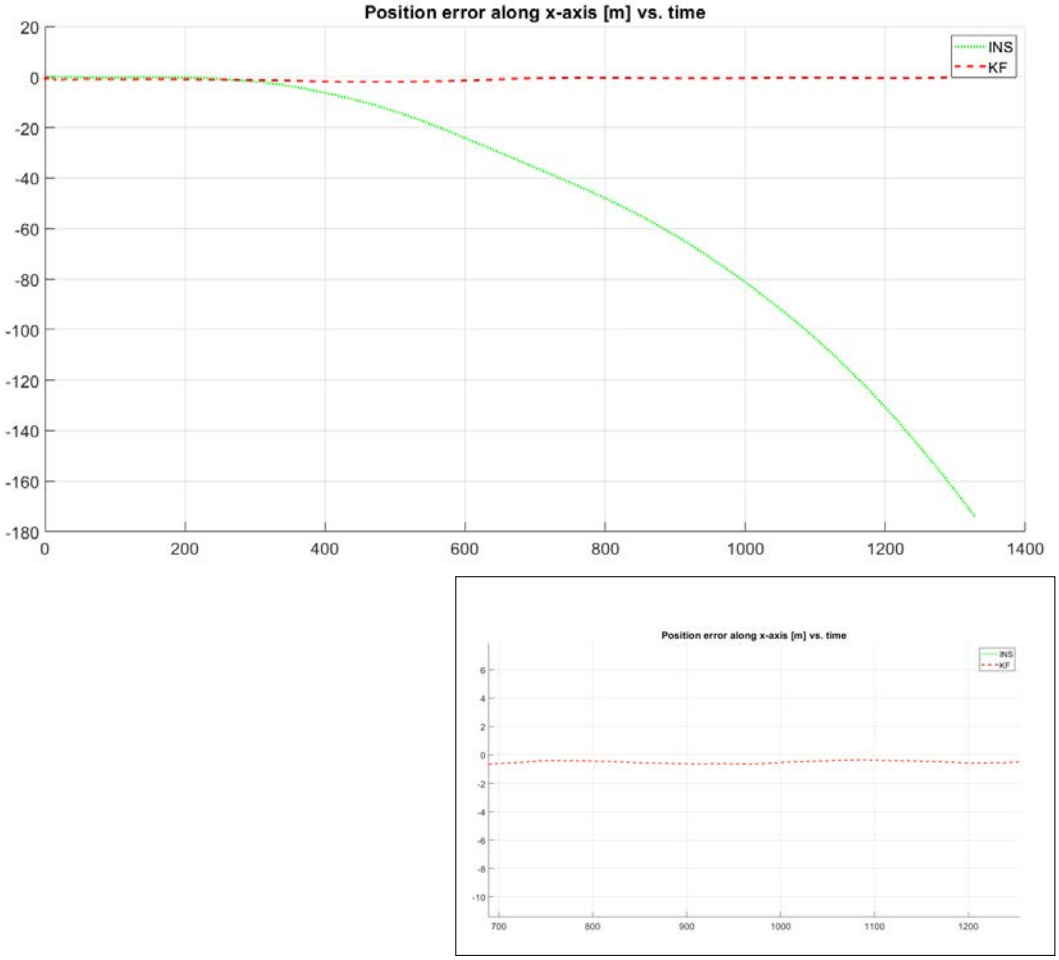


Figure 3.27: INS and Kalman filter x errors vs time, on bottom right zooming Kalman error

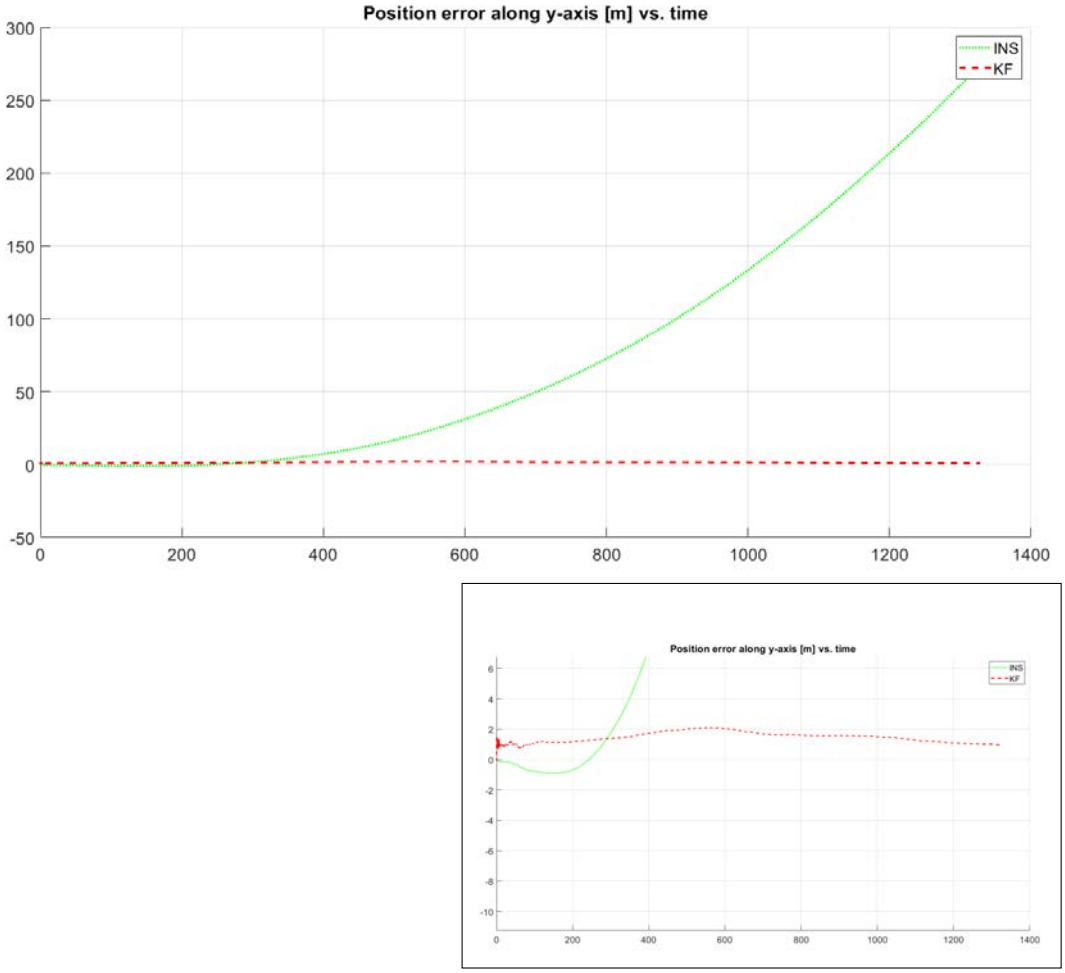


Figure 3.28: INS and Kalman filter y errors vs time, on bottom right zooming Kalman error

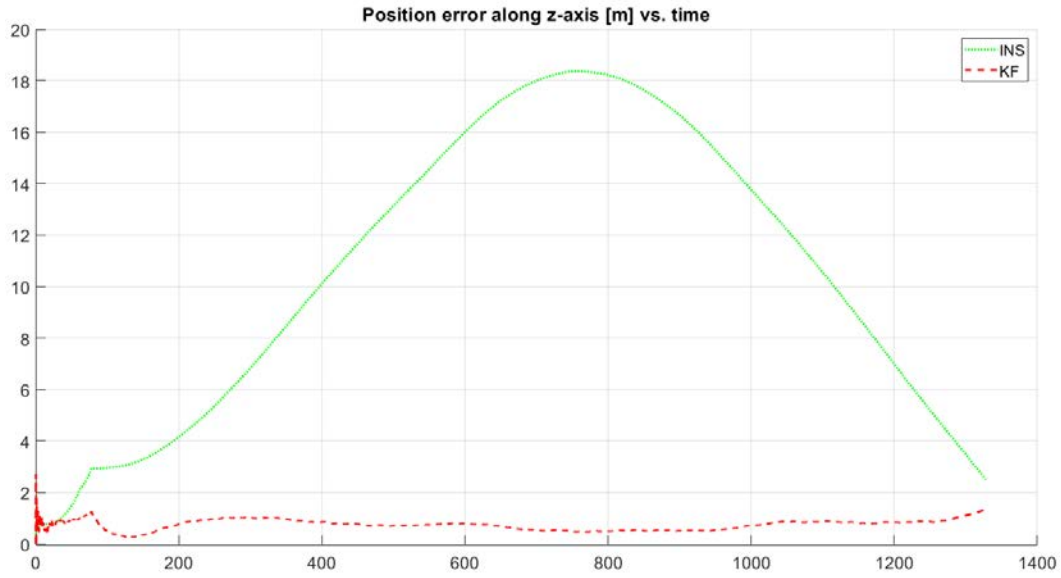


Figure 3.29: INS and Kalman filter z errors vs time

Finally a comparison of the errors between the solutions: GPS+INS, GAL+INS and MultiGNSS+INS:

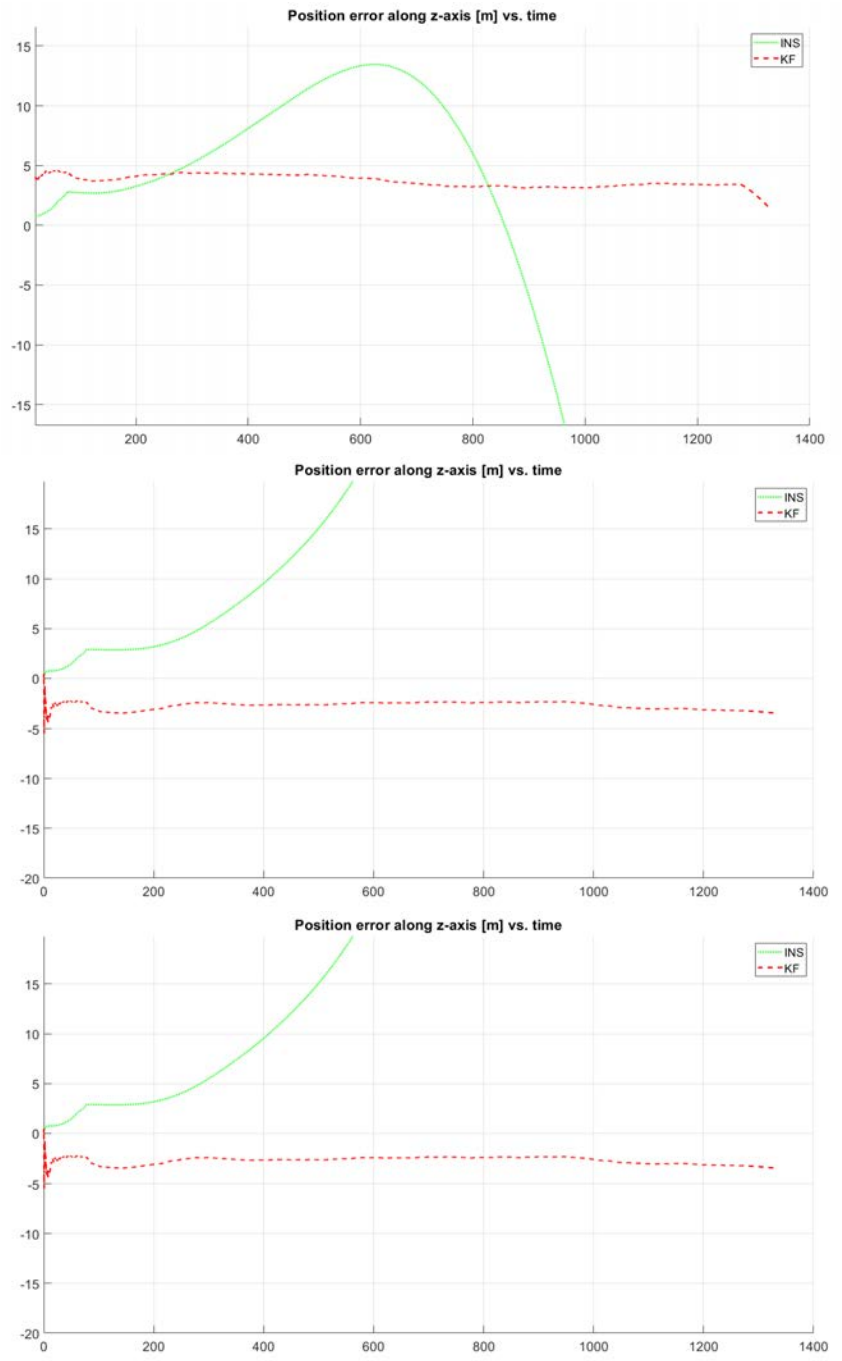


Figure 3.30: Errors on Z axis, GPS, GAL and then GPS+GAL.

3.2 Advantages of Galileo

From the first tool a series of benefits due to the use of Galileo, and multi constellation in general, are shown:

- Increase number of visible satellite.
- Improved DOP, errors and variance.
- Robustness Jamming (next chapter).
- Navigation redundancies

Chapter 4

Receiver performances validation

4.1 Vibration model and consequences

Clocks used for GNSS receiver are influenced by noise, this can have 2 main origins: thermal and mechanical vibration, the second one need to be considered in a hard environment like a rocket. Every launcher gives to customers a PSD (power spectral density) graphic, so the typical amplitude for all frequencies are known.

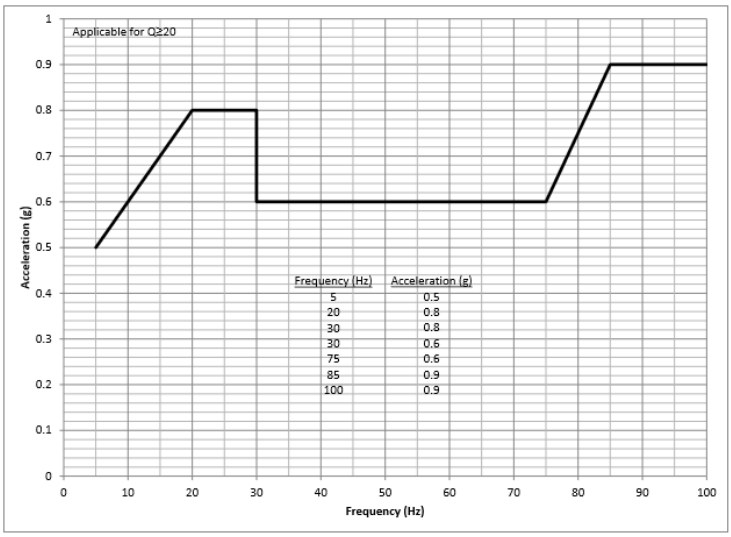


Figure 4.1: Falcon 9 maximum axial equivalent sine environment [12].

The problem is that in case of a receiver placed inside the first stage, and not inside the payload, where this graphic is calculated, acceleration would be significantly different.
For the simulations it has been proposed to operate in a more hazardous environment is necessary, so a different PSD has been created to test the receiver:

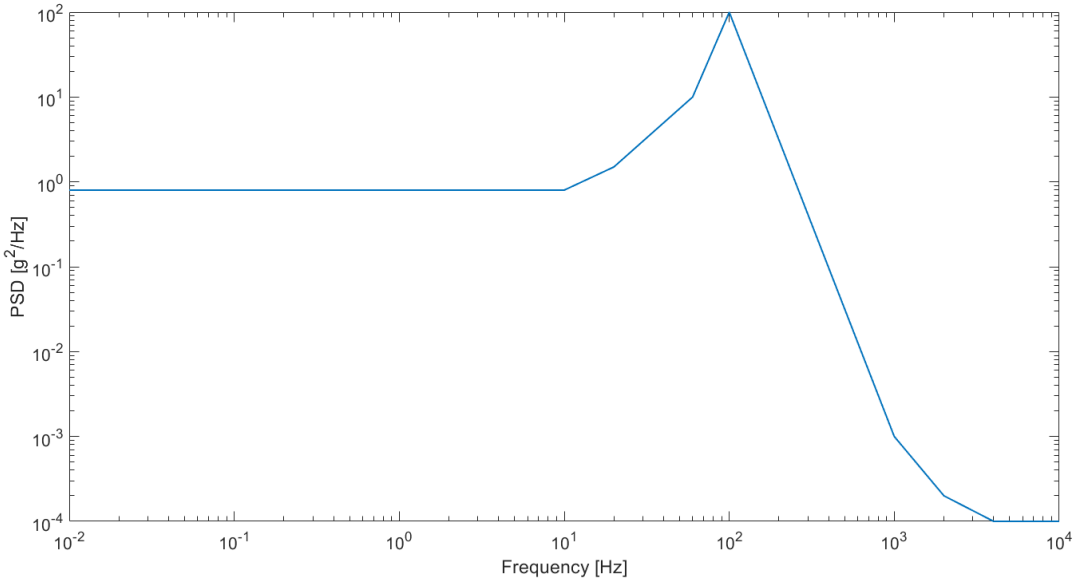


Figure 4.2: Vibration PSD acting on receiver

This PSD is generic and presents a resonance peak at 100 Hz, after that the low pass filter effect of the structures makes the power decrease rapidly. The equation used to calculate the phase noise due by vibrations is [11]:

$$L(f) = 20 \cdot log \frac{acceleration\ sensitivity \cdot acceleration \cdot oscillator\ frequency}{2 \cdot vibration\ frequency} \quad (4.1)$$

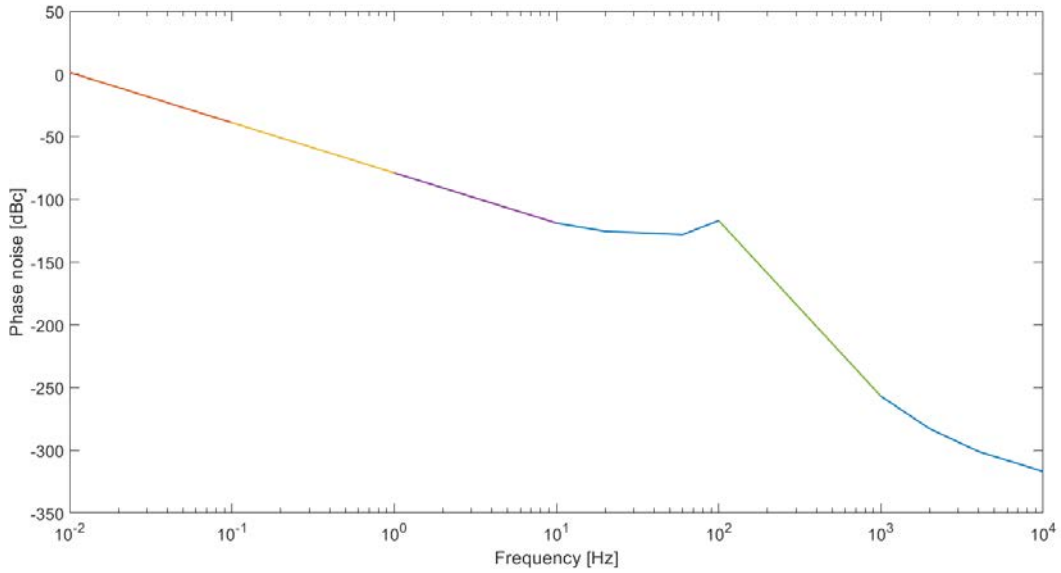


Figure 4.3: Phase noise derived from PSD, different colours due to a check to verify the slope of the curve.

Now for each value of vibration, the phase noise is known.
An other useful parameters is the Allan Variance, it is a measure of frequency stability in clocks, oscillators and amplifiers. From phase noise to Allan variance, some steps need to be done.
The equation used for calculate the variance is [13]:

$$\sigma^2(\tau) = h_{-2}\frac{(2\pi)^2}{6}\tau + h_{-1}2\ln 2 + \frac{h_0}{2\tau} + h_1\frac{1.038 + 3 * \ln(2\pi f_h \tau)}{(2\pi)^2\tau^2} + h_2\frac{3f_h}{(2\pi^2)\tau^2} \quad (4.2)$$

where τ is the time step, h_n are coefficients derived from Phase noise graphic, f_h is the cut frequency for a low pass filter (PSD cut frequency).

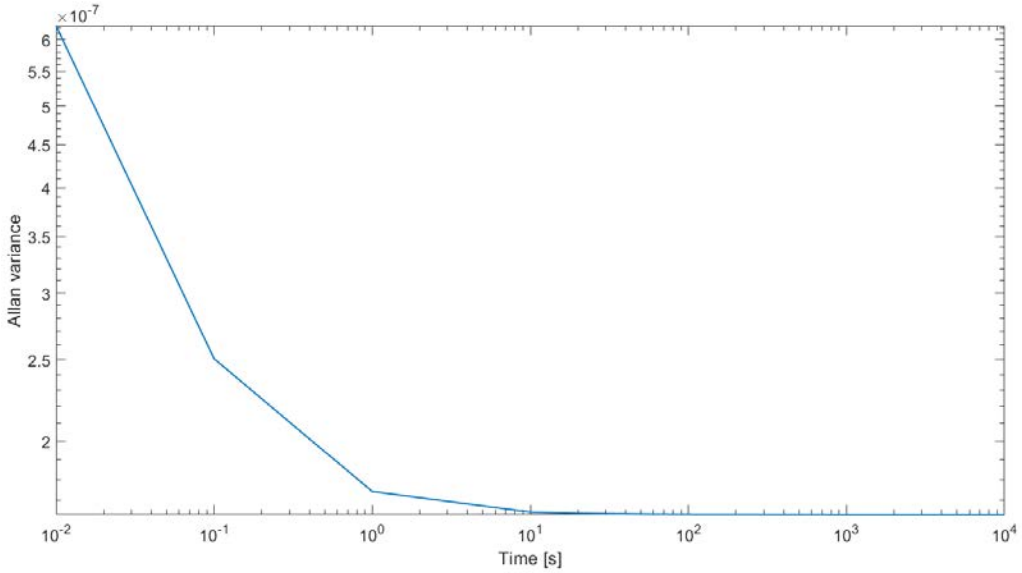


Figure 4.4: Allan Variance derived by phase noise

4.2 Tracking loop bandwidth limits

Through Semi-analytic tool, the ability of tracking the signal by receiver with certain properties has been validated, in this chapter the limits of the loop bandwidth for the PLL, DLL, and FLL will be evaluated.
By making them variables it is possible to see which values can and which can not keep the signal tracked, the dynamic environment is the same used in chapter 3.1.7:

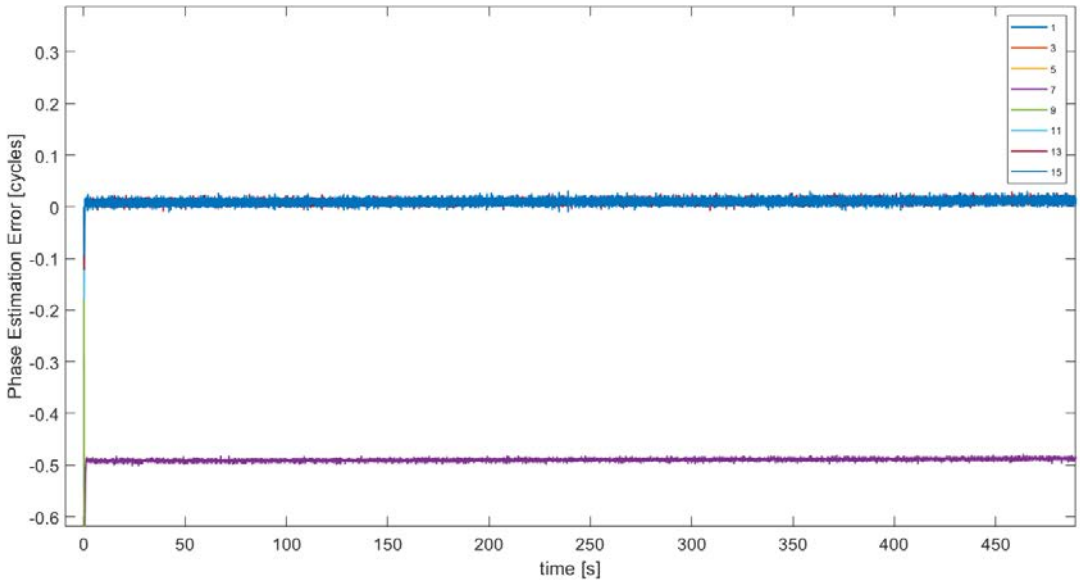


Figure 4.5: Phase estimation error with all bandwidth variables from 1 to 15 Hz

Receiver configuration that have shown a PLL phase estimation error with not around zero are not correctly locked to the received signal.

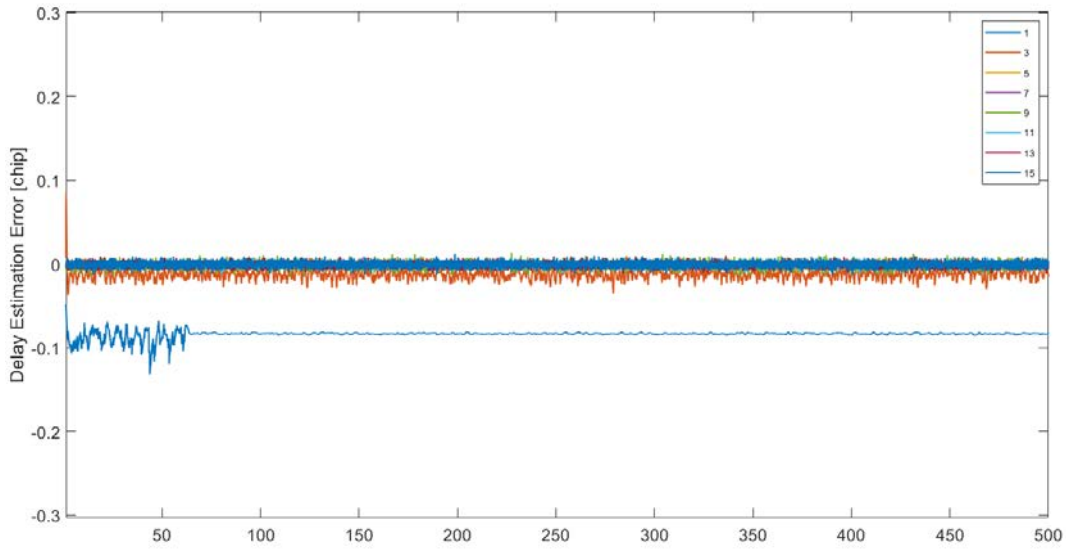


Figure 4.6: Delay estimation error with all bandwidth variables from 1 to 15 Hz

These graphics show how well the different tracking loops keep the signal

tracked. Summing up, many values are taken in consideration to verify if the signal is tracked or not: the two errors just shown but also the power. In our case this table helps to see faster if the receiver is capable of tracking:

bandwidths [Hz]	Loss of lock
1	True
3	True
5	True
7	True
9	True
11	False
13	False
15	False

An other way to better understand the errors due by the bandwidths, is to calculate and plot the standard deviation of the tracking loops error for each attempt:

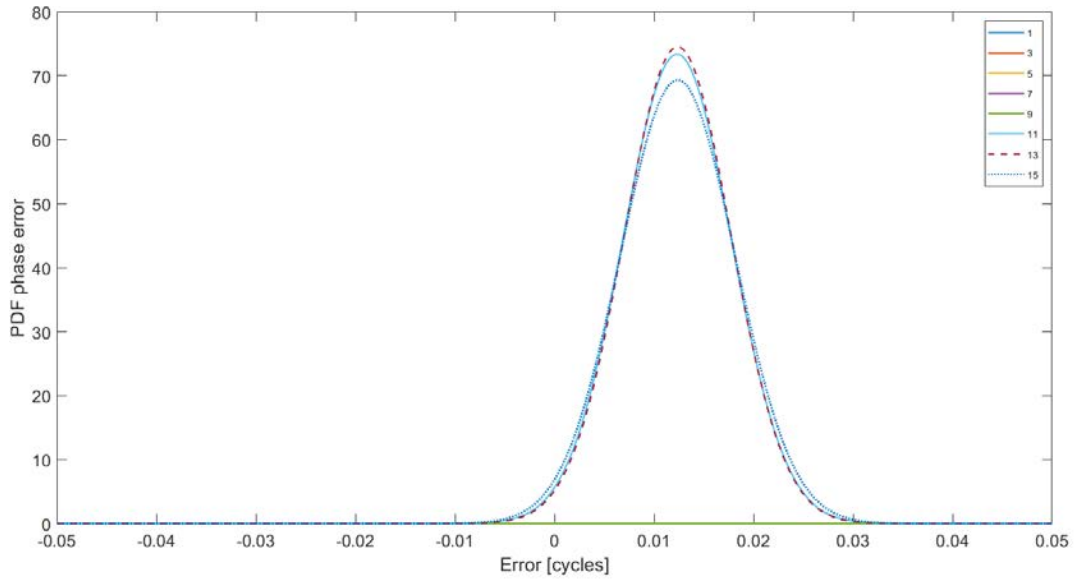


Figure 4.7: Phase PDF (Probability Density Function) with all bandwidth variables from 1 to 15 Hz

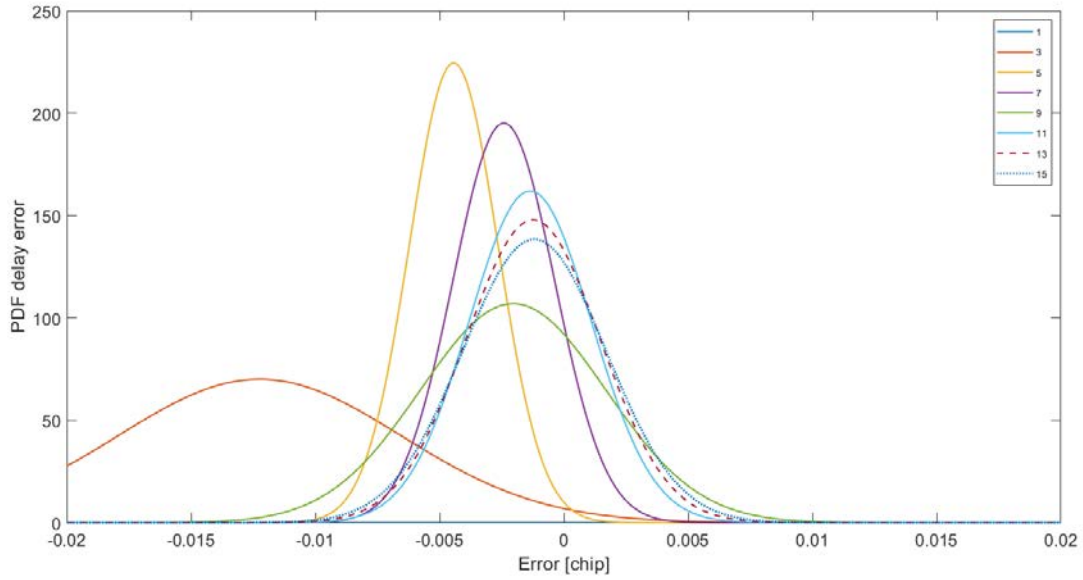


Figure 4.8: Delay PDF (Probability Density Function) with all bandwidth variables from 1 to 15 Hz

What is evident from this analysis is that wide bands are necessary for a receiver under those dynamics, however ,even if not contemplate in these simulations, opening wider the bands means let more noise enter the filter, so values higher than those taken in consideration are not recommended.

4.3 Jamming

RF interference (RFI) has been and will continue to be a significant worry for GNSS users. Because the signals from GNSS satellites are weak when the user equipment receives and processes them, they are especially vulnerable to RFI. Signals that overlap GNSS frequencies are likely to come from transmitters much closer than the satellites and, consequently, can easily overpower GNSS signals and render them unusable.
RFI can be unintentional or intentional; unintentional comprises the set of vulnerabilities introduced by accidental interference that is created by external sources: harmonic emissions from high power transmitters, mobile satellite services, televi-

sion, ultra-wideband radar and personal electronic devices. Intentional interference can be categorized on the base of the objective of the attack (Fantinato, Defending Critical Infrastructures from GNSS Interference [2]):

- Denial of service of position and time (jamming).

- Deception of position and time (spoofing and meaconing).

For what concern jamming, depending on power and modulation of the Jammer, the GNSS receiver is impacted at different levels (Fantinato, Defending Critical Infrastructures from GNSS Interference [2]):

- Antenna-LNA: for power levels above a certain threshold, antenna and LNA may be damaged mainly for electric discharge (electric overstress EOS) or thermal phenomena.

- AGC: if the jammer dynamic are slower than the AGC recovery time (i.e. the time window used to estimate the signal power in the AGC circuitry), a high jammer pulse can trigger the modification of the AGC gain. Therefore, in case of jamming, the AGC levels are lower than the nominal values.

- ADC: ADC isheavily affected by the presence of a jammer. The behavior changes depending on AGC availability. In case offast AGC thatis ableto trigger to the highest power level, the weak GNSS signal and noise are discarded by the effect of the quantization, since typically only 2-4 bits are available.

- Digital Signal Processing: Jamming has a quite important effect also if its power is comparable with the noise floor. In this case jamming affectsboth the acquisition and the tracking stage of a GNSS receiver, causing C/N0 degradation and PVT errors.

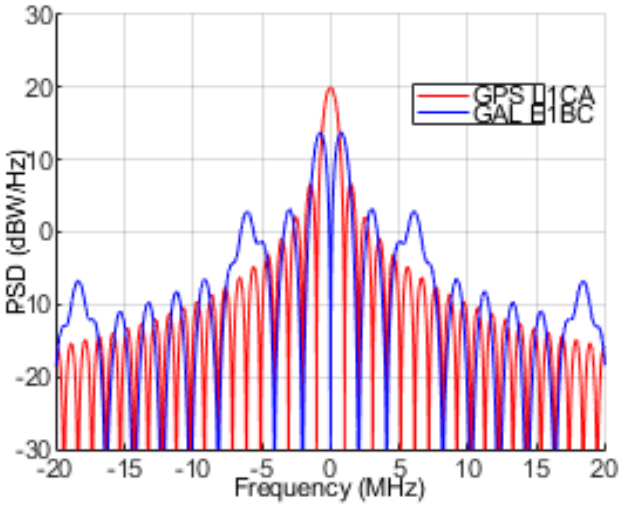


Figure 4.9: GPS and Galileo's signals

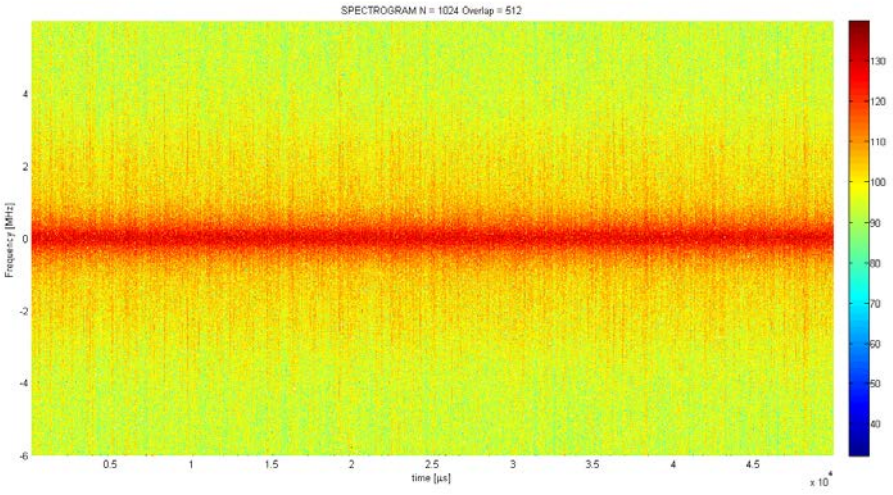


Figure 4.10: Effects of jamming on GPS signal

Two different tests have been made to simulate the behaviour of the receiver under jamming effects: the first one is a obscuration of GPS frequencies for the firsts 50 seconds of launch; the second one is a obscuration of both GNSS signals, so the Kalman filter has to work without GNSS for 50 seconds after launch.

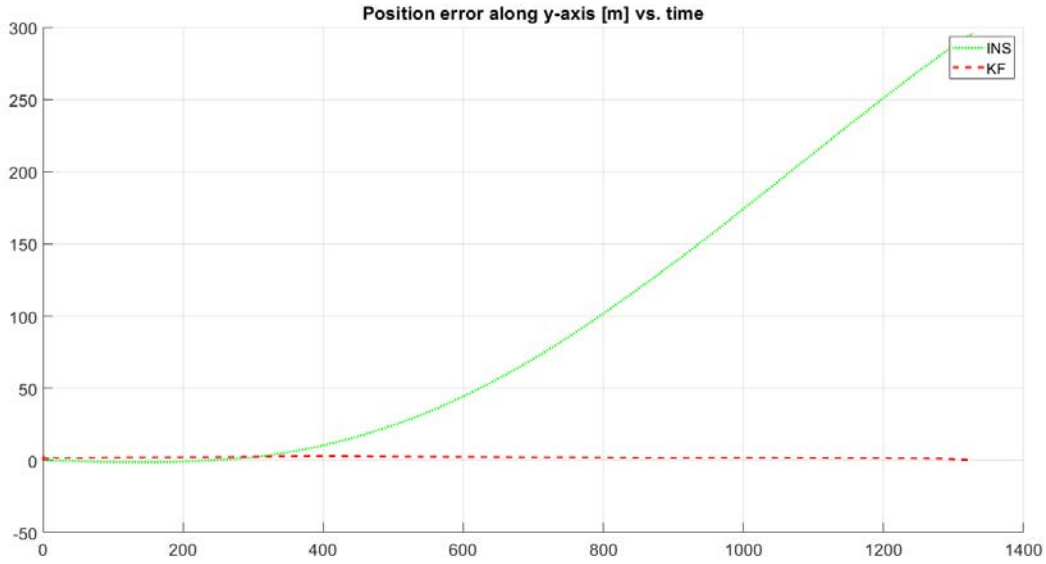


Figure 4.11: Jamming affecting GPS constellation for 50 s

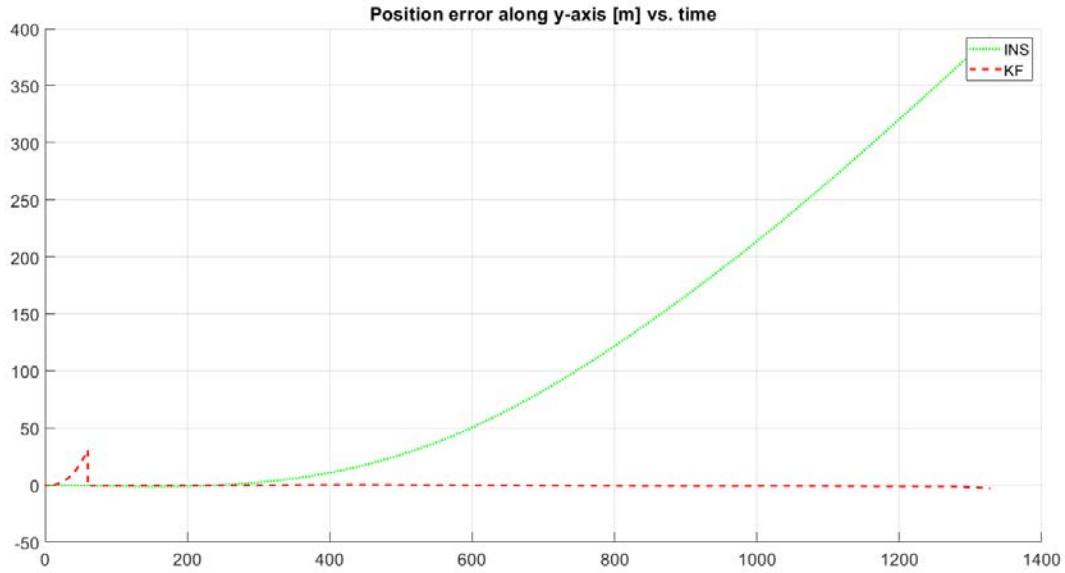


Figure 4.12: Jamming affecting both constellations for 50 s

Observing the two graphics it is possible to understand that if only one signal (GPS) is disturbed, the kalman filter can still work thanks to the other constellation, if both signals are disturbed kalman filter propagate an error that depends on

the asset of the rocket when jamming begins; so others precaution must be taken.

Chapter 5

Conclusions and Future works

The advantages of a multi constellation receiver are well known to space agencies and navigation companies, this paper has the intention to show and prove with data and quantitative analysis that those advantages are relevant also for launcher application.
As anticipated in chapter 3.2, many benefits are generated with a multi GNSS: major umber of visible satellite, better DOP, errors and variance, robustness at Jamming, redundancies; and they have been demonstrated in this thesis. Moreover the study of others elements like the integration, through a Kalman filter, with the inertial system, and the test about vibration effects, poses the basis for future works for this receiver and finally a possible realization and test on a real sounding rocket.

Continuation of this thesis involves analysis of remaining aspects of the Falcon 9 launcher, correction of input values for both Falcon 9 and Sounding rocket when available, integration with anti-spoofing techniques designed in Qascom S.r.l.
If the receiver will flight on a sounding rocket (probably summer 2019), than additional other studies need to be performed before and after the flight, like thermal and structural analysis, antenna positioning, expected results, comparison with actual results.

Bibliography

[1] Guilherme F. Trigo and Stephan Theil, *Improved Hybrid Navigation for Space Transportation* .

[2] Samuele Fantinato, Stefano Montagner, Giovanni Gamba,Andrea Dalla Chiara, Oscar Pozzobon, *Defending Critical Infrastructures from GNSS Interference* .

[3] Mark G. Petovello, *Real-time integration of a tactical-grade IMU and GPS for high-accuracy positioning and navigation*, 2003.

[4] Martin J. L. Turner, *Rocket and Spacecraft Propulsion*, 2009.

[5] Bong Wie, *Space Vehicle Dynamics and Control*, 1998.

[6] Howard D. Curtis, *Orbital Mechanics for Engineering Students*, 2010.

[7] Sanat K. Biswas, Li Qiao, *Simulation of GPS-based Launch Vehicle Trajectory Estimation using UNSW Kea GPS Receiver* , 2016.

[8] Alessandro Caporali, *Posizionamento Satellitare e Determinazione orbitale* .

[9] Elliott Kaplan, Christopher J. Hegarty, *Understanding GPS/GNSS: Principles and Applications* .

[10] Oliver J. Woodman, *An introduction to inertial navigation*, 2007.

[11] Archita Hati, Craig Nelson and David Howe , *Vibration-induced PM Noise in Oscillators and its Suppression.*

[12] SpaceX , *Falcon9 payload user's guide*, 2015.

[13] P. C. Chang, H. M. Peng, and S. Y. Lin , *Allan variance estimated by phase noise measurements.*

Acknowledgement

First I would like to thank the experts who were involved in the validation survey for this research project: Ing. Samuele Fantinato, Ing. Fabio Bernardi, Ing. Luca Canzian, Ing. Carlo Sarto, Dott. Lorenzo dal Corso. Without their passionate participation and input, this thesis could not have been successfully conducted.

I would also like to thank Professor Alessandro Caporali for the help in finding a company interested in my thesis.

Finally, I must express my very profound gratitude to my parents and to my girlfriend for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Author

Francesco Longhi