Politecnico di Torino

Master thesis of Mechatronic Engineering

Master Course in:

Mechatronic Engineering

Final Master Thesis:

Quality control of GNSS RTK positioning using mobile devices for its application on location apps

Supervisor:
prof. Paolo Dabove

Author:
Pedro Ríos Bravo

June 28, 2021

Abstract

The main purpose of this project is to improve the positioning behaviour of the Android smartphones, using them as station of Global Navigation Satellite System (GNSS). Focusing on the main errors that occurs by different factors and affect on the phone location and applying solutions, to achieve the understanding of the behaviour of the devices under different situations. For the Rinex data collection the app Geo++ Rinex Logger has been the main support, different Android smartphones has been checked during the process and, furthermore, an u-blox device is involved in this project. The collected data, is processed with RTKLib software, which allow to perform a post-processing and a real time study.

Obviously, this project has many applications in an industrialized world, but it is focused on the usage of a better location service of smartphones on apps, to ensure that a non-desired failure cause an undesired performance, ending with unsatisfied clients.

Originality

I ensure that all the contains of the documents are product of my knowledge, being the University to which this document is required, Politècnico di Torino. With this, I comment that I include mentions to the company Repsol, in which I do not use any data that are not public.

With this, I include my sign to affirm all the commented before.

Signed:

Contents

Chapter 1

Introduction

Positioning has been always a human field of study, trying always to achieve and to have the knowledge of where exactly we are. Nowadays, a lot of technologies have been involved in this process, satellites, stations and specific devices have done that the situation improves.

A very interesting example to demonstrate, in a simple way, how the satellites work and perform the positioning calculation, is the lighthouses and ships problem. Imagine that a lighthouse correspond to a satellite and the ship is the rover for which the position wants to be known. The distance between the lighthouse and the ship can be computed using the time that the light wave spends for getting the ship. With this distance, a circumference can be created, and also, if a new lighthouse is involved, another circumference will be created. Then, we could reduce the positioning space of the ship as shown in *Figure 1.1* from the GNSS Data Processing Book [16].

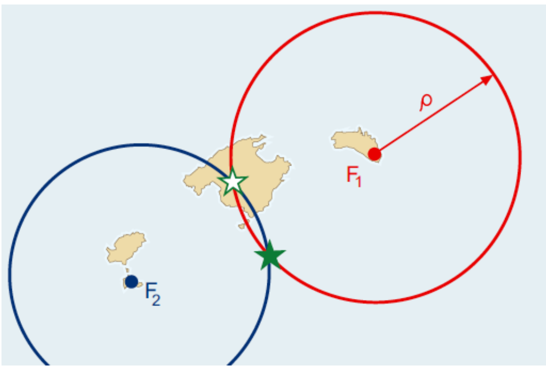


Figure 1.1: Radius for positioning purposes [16]

In 1973 the Defence Department of the United States developed the NAVSTAR/GPS allowing to determine the position using satellites and the adequate receptors. However, this system had some imperfections in the results.

Even today, this kind of errors are present in the modern positioning systems, due to this, some different correction methods appear. These errors can be observed and analysed by some differential techniques, like post-processing (for a not real-time study) and real-time techniques.

The post-processing technique is focused on the result corrected obtained after a time, generally involving a process in which the data are treated previously. This process needs the differential corrections from a base station, that allows to perform the study. This base station could be, from one given by the government to one created by a user, the main requirement of this devices is that it has to be in a known position and to be fixed. Can be observed the behaviour of the system in *Figure 1.2.* [9].

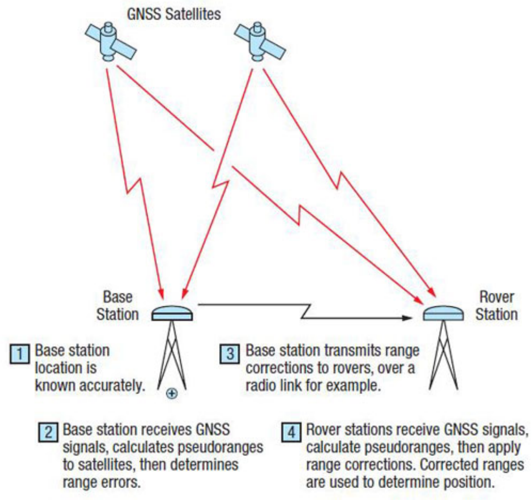


Figure 1.2: Post-processing method scheme [9]

The second method mentioned is the real-time technique or RTK (Real-Time Kinematics), in this case, the phase measurements obtained from the rover given by the different satellites are corrected at the moment with a base station solution. This allows to have a high level accuracy while using the positioning services, probably is the most interesting application for the usage of smartphones and apps, achieving very good precision in them.

1.1 GNSS Architecture

Basically, GNSS involves the satellite navigation systems of GPS, Glonass, Galileo and Beidou which allows the access to continuous positioning information. GNSS can be divided into three parts [11], space segment, basically composed of the satellites; control segment, include all the operations for the correct data treatment; and user segment, that involves all the devices that makes a data receiving. Positioning information is provided thanks to the carrier frequencies which will be different for each GNSS singals.

1.1.1 GNSS Segments

As mentioned previously, GNSS is composed by three segments. The space segment, is referred to the generation and transmission of code and carrier phase signals. The main tool used are high accuracy atomic clocks, which allows to determine the difference between the signal sending and the receiving, these clocks are onboard the satellites.

This segment is composed by all the satellite fleet, ensuring a good number of the to achieve good generation, transmission and receiving of code. Also is important, that the rover, detects a high number of satellite (with a minimum number of 4) to be success in the process.

The control or ground segment is focused on the management of status and configuration of the satellites, in charge of the ephemeris and clock performance, ensure the good GNSS time

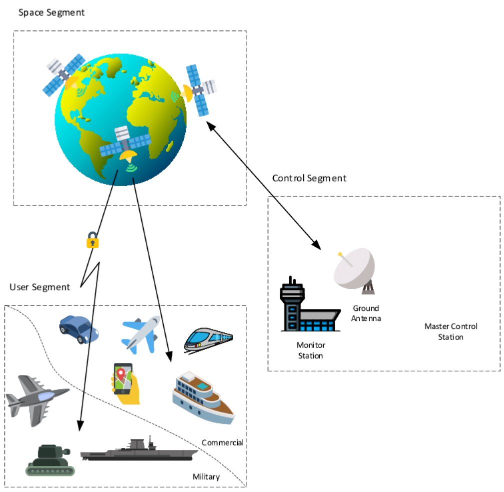


Figure 1.3: GNSS Three segments architecture [11]

scales and keep updated the navigation files that receives the satellites. In thi field is included the base stations for the data corrections.

Finally, the user segment involves all the GNSS receivers or rovers, in which the signals are analysed, the pseudoranges are determined and the navigation equation are solved to obtain the positioning and a good time.

All this fields, involve the different GNSS signals of which will be talked in the following section, analysign how they are integrated in the system.

1.2 GNSS Signals

The satellites transmit information in frequency bands, each band is corresponded to different signals, these are: GPS, GLONASS, Galileo and Beidou. With this signal, the positioning calculation is possible thanks to the travel time that can me computed by the user. The signals [4] are composed by:

- **Carrier:** Depending on the frequency in which the signal is focused, a sinusoidal wave will be given.
- **Ranging Code:** Is the main tool used to determine the time that the signal spends from the satellite to the receiver, is a binary sequence.
- **Navigation Data:** Ephemeris (from the different signals), clock and status of the satellites are provided in these data.

The frequency in which the data are sent, depends on the signal and the capability of the rover/receiver to acquire one of them. In many cases, some devices are only capable to access some signals, obviously, if the receiver is able to reach more satellite signals better for the position accuracy.

1.2.1 GPS Signals

The radio frequency in which the GPS (EEUU) signals are transmitted are named L1, L2 and L5 (L band), whose frequencies are represented in *Table 1.1* [16]. GPS provide two services, Standard Positioning Services (SPS), centred in L1 and Precise Positioning Service (PPS) that uses L1 and L2.

Frequencies (MHz)		
L1	**L2**	**L5**
1 575.42	1 227.60	1 176.45

Table 1.1: Data Transmission GPS Frequencies

Satellites receive information from the antennas on earth, and then, they send information back to the rovers, this information is named navigation message. These data, contains all the necessary requirements that the user needs to compute the position. Clock data, ephemeris, ionospheric model parameters and the almanac are given in the message. The most recent messages are named CNAV, CNAV-2, MNAV and L5-CNAV, being the last one modulated using L5I signal.

1.2.2 Glonass Signals

For this signal (Russian Space Forces) the radio frequencies are also in the L band and have the names G1, G2 and G3 with their respective values in *Table 1.2* [16]. Also, can be differentiated two services, SPS, which transmits in G1 and recently it uses G2, and PPS, that allows two bands, G1 and G2.

Frequencies (MHz)		
G1	**G2**	**G3**
1 589.06	1 242.93	1 201.00

Table 1.2: Data Transmission Glonass Frequencies

Also, the navigation message is present in Glonass, being a bit different with respect to GPS, but containing the necessary information for positioning processing.

1.2.3 Galileo Signals

Galileo (European Union) uses the frequencies E1, E6, E5a and E5b and the services offered are different with respect to the two previous signals, these are: OS (Open Service), PRS (Public Regulated Service), CS (Commercial Service), SAR (Search and Rescue) and SoL (Safety-of-Life). Each frequency support one of the services mentioned and the values of them are shown in *Table 1.3* [16].

Frequencies (MHz)			
E1	**E6**	**E5a**	**E5b**
1 575.42	1 278.75	1 176.45	1 207.14

Table 1.3: Data Transmission Galileo Frequencies

1.2.4 Beidou Signals

Beidou (China) works also in the L-band, using the frequencies named B1, B2 and B3 [16] and using only an Open Service and an Authorized Service.

Frequencies (MHz)		
B1	**B2**	**B3**
1 575.42	1 191.80	1 268.52

Table 1.4: Data Transmission Beidou Frequencies

The values correspond to Beidou Phase III, in which the satellite constellation were increased, and it started to work as an open service.

1.3 GNSS Measurements

1.3.1 Pseudorange

Basically, this kind of GNSS measurement, allow us to know the amount of time that is expended in the reception, being this value computed by the difference among the time registered in the reception, expressed in the frame of the receiver, and the time that is registered in the transmission by the satellite.

The computation of this measurement, for L1 and L2, is performed by the device that carry out the GNSS data acquisition, and the formulation used is represented below [8].

$$\gamma = (f_{L1}/f_{L2})^2 \tag{1.1}$$

Equation 1.1 shows the frequencies L1 and L2, and then, applying the ionospheric corrections and considering both frequenies, the pseudorange computation is performed below.

$$PR = \frac{PR_{L2P(Y)} - \gamma PR_{L1P(Y)}}{1 - \gamma} \tag{1.2}$$

Being the components of *Equation 1.2*:

- **PR**: Here the pseudoreange with the corrections for the ionospheric delays
- **PRi**: Pseudorange for the channel.

1.3.2 Carrier-Phase

This technique, assume that the satellite, which send the information, and the GNSS receiver, and synchronized in time, so the phase of the carrier, will be the same for both. In this case, the time that the signal spend in the travel among both devices is computed by the lag that the signal presents, this is due to the linear variation behavior of the signal with time.

This kind of measurement, is interesting for this project, because will be the one in which the computation and the position generation will be focus. So the smartphone that will be chosen for the performance of the data acquisition will be required to have this measurement among him characteristics.

It is necessary to differentiate between the time expended in the transmission (t1) and the current time (t2), measured by the current phase that arrive to the receiver at the current moment. Is also interesting to know some parameters that take part in the process.

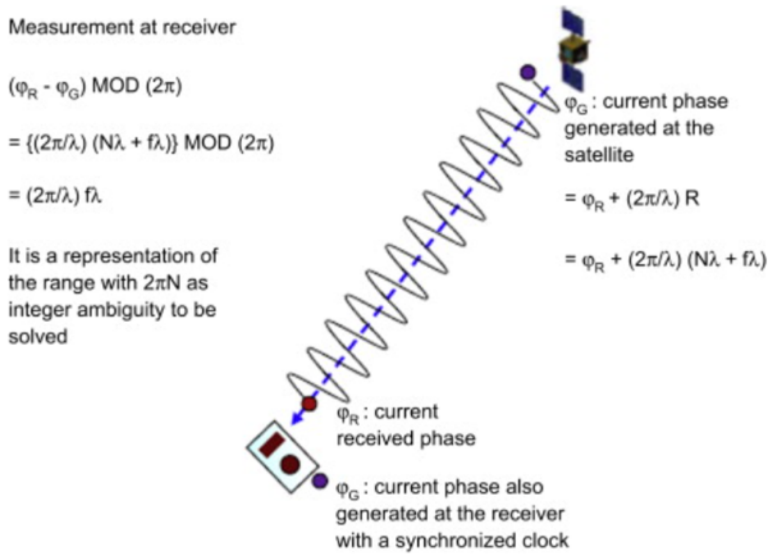


Figure 1.4: Carrier-phase characteristics [7]

Then it is interesting to know how to compute the transmission time and the received time, Tt and Tr respectively *Equation 1.3* and *Equation 1.4*.

$$T_t = k(n_1 2\pi + \alpha_t) \tag{1.3}$$

$$T_r = k(n_2 2\pi + \alpha_r) \tag{1.4}$$

Where:

- $(n_1 2\pi + \alpha_t)$: Transmission carrier phase.
- $(n_2 2\pi + \alpha_r)$: Received carrier phase.
- n_1 and n_2: number of complete 2 pi radians executed by the phases.
- k: factor conversion from phase to time.

With this, the propagation time is easily computed, because is the difference among the time expended by the transmission and the received time.

$$T_r - T_t = k((n_2 - n_1)2\pi + (\alpha_r - \alpha_t)) \tag{1.5}$$

$$\delta T = kN2\pi + k\delta\rho(t) \tag{1.6}$$

1.3.3 Correction methodologies

Above, it has been written about ionosphere corrections, this is due to the effect created by this atmosphere layer in the GNSS signal. For this reason, some computations techniques are performed to achieve better results in the data acquisition. In this section, this techniques are going to be analyzed in a briefly way.

1.3.4 GNSS Receivers

Nowadays, a lot of different devices are able to achieve this kind of measurements, from the most expensive and bulky to the cheapest and smallest. The quality obviously is related with the hardware implemented in the device, and the accuracy reached in the data of some devices is unbelieve.

In this project, two receivers are going to be used, and u-blox, that is a small device that allow the easy transportation of it to get data from different places with no difficulties, and the LEIAR25.R3, which is the receiver located at PoliTO and will be used as a base station also. The last one, presets better characteristic in comparison with the u-blox, but the purposes of them in this thesis are completely different. The market offers many possibilities to choose among different devices, depending on the necessities of the user, even the smartphones, could be considered as a GNSS receiver, but, not in all the cases, the expected results are obtained.

As mentioned above, the smartphone can reach the functions of a GNSS receiver, but, and like is described in this thesis, some of them not present the expected results. The mainly problems found in the smartphones tested are, in a briefly way, that some of them are not able to reach a good number of satellites to obtain the signal and that a lot did not have the possibility of compute a carrier-phase measurement. These complications will be described more precisely in one of the following chapters.

Chapter 2

Objectives

Nowadays, most people have a smartphone and the more applications appears the more useful is the positioning. Food apps, where you can find the nearest restaurant, location apps, that show you the way to get the place that you are looking for and more. The accuracy in this positioning is becoming more and more crucial.

The main purpose of this project, is to show how the smartphones perform their location, and how corrections can be applied to achieve the desired results, ensuring and given the possibilty to introduce this on the apps. Obviously, not all the devices have the same features, and this will be also tested to prove that there is not an unique solution, since we depend on the most recent technology advances. But this is a good turning point, to have in mind that, with the adequate tools, the accuracy can be reached.

Giving a real example of the application of this study, in the company Repsol SA, are developing a new way of payment for the gas station clients.This method allows the user to pay without getting out of the car, only with the app Waylet, which detects the gas station in where the client is located and also the hosepipe that is required, this is a great solution to reduce the time that a person spends on the station.

However, a problem is present in this situation, in many cases, gas stations are located in parallel, one for each side of the road, and a bad positioning of the smartphone while it is using the app, could generate a wrong payment, switching it from the gas station in where the client is to the opposite one.

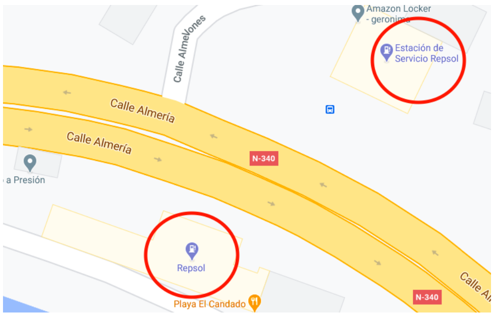


Figure 2.1: Parallel Repsol Gas Stations

Like this, many other apps require a high level usage of positioning services, then the project can be relevant, with the comprehension of the whole process, which will be breaking down below.

Establishing the main points of the project, can be remarkable:

- Setting the devices that will carry on the data collection
- App source where the data will be obtained
- Establishing the errors present on the data without any correction applied
- Creation or selection of a base station to support the rover
- Post-processing methods to apply corrections
- Individual analysis of the smartphones selected
- Analysis of a pure location device
- Comparison among pure location device and smartphones

All this tasks, will be analysed and decomposed in a theoretical and a practical way, to achieve the objectives exposed.

Probably, the most important task to carry out would be the last one, obviously all of them are necessaries to achieve the expected results and to have a whole comprehension of the study. But, with the comparison, the results will show the similarity that presents a normal smartphone with respect to an u-blox, which its purpose is to send raw GNSS data.

If the situation expected is achieved, then it will be possible to affirm that a smartphone could reach really good location performance, applying this to all the apps that requires a service like that.

The actual behavior of the Repsol application is to simple but imprecise, this app assume that the smartphone can have errors and create a circumference with the center in the gas station and a radius that do not interfere with the circumference created for the parallel gas station. With this, if the user is inside of one of the circumferences, the payment will be processed in one or another station.

Figure 2.2: Radius methodology

This is ambiguous, because if a good positioning performance of a smartphone can not be ensure, the actual method could fail. Because of that, this thesis is important, being able to ensure that with an smartphone, a precise positioning can be reached, avoiding this imprecise methods implanted in the apps, with a behavior that do not ensure the correct work of the app.

Chapter 3

State of the Art

Knowing the importance that is currently given to positioning is easy to find a lot of researchs in which this topic is treated. Increasing the accuracy for localization devices, the usage of new processing tools like phyton, a better real-time orbit determination for the satellite and more are becoming increasingly important for our society and are being an remarkable object of study in the present.

3.1 GNSS Positioning techniques

This web [21] introduce the concept of a successful GNSS data treatment, in which first the data is obtained in a code-phase way, and then it is necessary a transformation into X,Y,Z or ENU type.

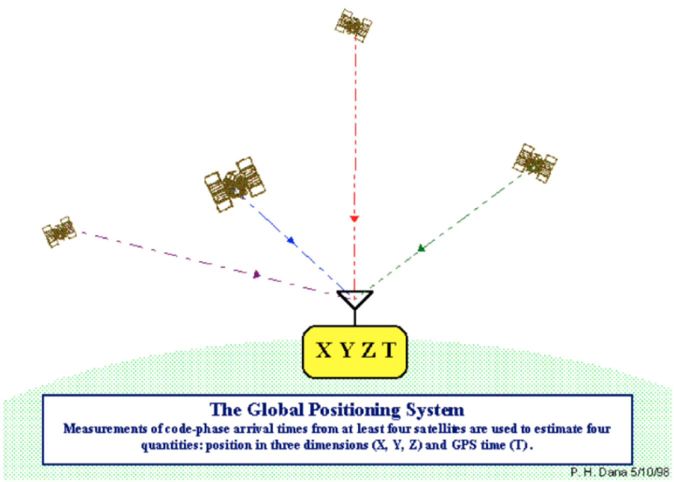


Figure 3.1: GNSS Requirements

Even carrier-phase and code measurement are able to perform a post-processing analysis, which consist on a data collection and after the data are checked a programme is used to obtain positioning results. Here, the negative effects over the signal can be eliminated or reduced by the software of the application.

Another method is the **Real-Time Kinematic Positioning** (RTK) , in which are available different techniques to improve the results, minimizing errors (satellite clock, ionospheric delay...). This method involves a base station, a rover and the satellite constellation, with a minimum of 4. The corrections and the results are obtained instantly.

Precise Point Positioning (PPP) ensure corrections for the satellites broadcast orbit and clock, and even applies solutions for ionospheric and tropospheric errors, estimating their delays. For this method is necessary a high accuracy multi-frequency GNSS receiver.

3.2 GNSS Data Post-Processing

Here [14], the different techniques are explained and compared with each other, so a global image of the available options is obtained.

The initial two methods to compare are the Differential GNSS (DGNSS) and RTK. Both require the existence of a base station receiver. The main difference among them is that in RTK carrier phase measurement are carried out and this gives more accuracy to the method. On the other hand, DGNSS is useful when exits a long distance between the base station and rover.

Then, is interesting to analyse PPP with respect to SBAS method. Both acquire the corrections from the satellite and again, the main difference is that PPP allows carrier phase, and like in the previous case is more accurate than SBAS. But, the advantage of SBAS is that provide the user with free corrections.

If DGNSS is compared with SBAS, the difference appears in the technologies required by each other. SBAS only need a receiver and a GNSS antenna but DGNSS requires, as mentioned before, base station, rover and antenna.

The more precise methodologies, PPP and RTK have a similar precision, but as in the previous comparison, the technologies required are different and the baseline distance. PPP only needs receivier, GNSS antenna and L-Band frequencies, but RTK requires two RTK receivers, base station and rover.

3.3 Real-Time Kinematic using Galileo

GNSS RTK performance improvements using Galileo Satellite Signal [17] focuses on the usage of multiple satellite constelations with Galileo signal measurements (managed by the European Union), applying a real-time correction with the field receivers for numerical analysis using RINEX data.

With at least five satellites, centimeter accuracy is provided, but this is a complex real-time process, obiously a big amount of satellites produce more precission in the measurementes, usign NAVSTAR, Glonass and adding Galileo the satelite availability reach twenty eight.

Is also interesting the effect of Galileo system on the Position Dilution of Precision (PDOP) which has a value near 1, achieving good results and it can be said that the ionosphere impact is small and can be negleted.

As a conclusion, can be said that using Galilleo signal the number of satellites is increased, improving the PDOP and reducing the error limits.

3.4 Floating Car Data (FCD) for mobility applications

Regarding the devices involved in this study, is interesting to know more about the VIASAT devices, these are dedicated to generate positioning information, being this their only application. It is interesting to know how they work and if a samrtphone could equalize their behaviour.

Now a days, practically every car needs GPS information and it is also important that each car could send this information of his position to a place in charge of traffic control.

Floating Car Data (FCD) is the information that includes all the GNSS positioning data required for the real-time processing method to achieve the car location.

This report [18], shows some examples of the FCD obtained at the City of Torino, to prove the feasibility of them achieving some expected points and giving the posibility of a good location service to areas with a lack of sensors. This kind of data is provided by On-Board Unit (OBU) of the vehicles. The data given is too similar to Rinex, it provides Latitude, Longitude, Time... so a real-time method and a post-procesing analysis are admited.

Here, the data obtained is compared with traffic sensors, which provides information with on-site measurements, this comparison is really interesting because gives knowledge about the accuracy that a VIASAT device can have with respect to a static measurement element, enabling VIASAT with this study as a really good positioning device.

This report shows the travel behaviour, observing the vehicle flow and the mean velocity using this kind of data, performing a comparison between privates cars and fleets. Ensuring with the whole report, the possibility of using FCD as mobility services, analysing also the limitations that presents.

3.5 Common Errors

Sometimes, and because of the phone characteristics, the app is not able to reach a sufficient number os satellites, being the minimum of 4 to reach a good obs file. Is important for the project to have a good smartphone with a good quality in the measurements, however, the study of all kind of devices is also important, to know the deficiencies of the market and how to improve those that are not good enough to achieve a high number of satellites.

This capability of acquire a sufficient number of satellite, was the first problem presented in the smartphones tested, not all of them were able to reach a decent number of satellites which allow to carry out the post-processing studies, due to a good number of data are necessary.

Once this problem was solved, another one came, and was the inexistence of carrier-phase information, some of the mobiles tested, were able to get information from a decent number of satellites, but once the data were analyzed, it is easy to realize that the carrier-phase column was not present. This is a very important fact, because the study was developed focusing on this kind of measurement, which provide a bigger accuracy to the results obtained from the post-processing method.

Finally, a pair of mobile phones that reach the expected situation were found, and the study was developed with these two, ensuring the existence of a sufficient number of satellites and the presence of carrier-phase measurements in the observation file.

3.5.1 Noises

Obviously, all the receivers could present noises in the data acquired, this can be due to the antenna design, the weather factor and more. When this is present in the receivers, some solutions are possible, like the usage of a filter to eliminate the undesired noises [22].

When the biases are produced by instrumental errors and delays, these can be modelled by equations, but the main solution is to get a better receiver, with better hardware characteristics.

3.5.2 Ionosheric and tropospheric delays

Terrestrial atmosphere cointains some layers and depending on the electron density that this layer presents, the GNSS signals will be affected and delayed. This effect can be among a few meters and to a quantity bigger than twenty meters,

Modelling this effect is not an easy task, because the geomagnetic and solar interactions are involved in this behaviour, but the ionospheric effect can be considered as frequency dependent, then this can be measured and also corrected.

On the other hand, the tropospheric effect cannot be considered as frequency dependent, these are the problems originated below the ionosphere. It depends on the zenith angle, being affected mostly by the satellite degree elevation.

3.5.3 Clock errors

As described, the satellite and the receiver can present a bad synchronization with the GNSS time stablish. To correct this kind of differences, is useful to consider both (satellite and receiver) as different devices.

3.6 Post-Processing method

This process involves the data treatment when the positioning information is already collected. Is very important to ensure that in our Rinex file carrier-phase measurements are present, to achieve a more precise result than the one that will be obtained with code measurements.

Carrier phase measurement can be defined by the equation written below.

$$\Phi_{L_f} = \rho + c(dt_{rcv} - dt^{sat}) + T_r - \alpha_f STEC + k_{L_{f,rcv}} - k_{Lf}^{sat} + \lambda_{L_f}(N_{L_f} + \omega) + m_{L_f} + \epsilon_{L_f} \quad (3.1)$$

Describing the equation:

- ρ**:** Geometric range.
- $dt_{rcv} and dt^{sat}$**:** Clock offset from GNSS time scale (receiver and satellite) multiplied by the speed of light (c).

- T_r: Tropospheric delay.
- $\alpha_f STEC$: Frequency-dependent ionospheric delay.
- $k_{L_{f,rcv}} and k_{Lf}^{sat}$: Carrier-phase instrumental delay (frequency-dependent).
- $m_{L_f} + \epsilon_{L_f}$: Multipath and noise.

3.7 Real-time processing method

One known technique is Real-Time Kinematic (RTK), where with two receivers and a distance limited between them, the ambiguity value can be obtained to reach a high (centimeters) accuracy.

Also, it is interesting to consider the usage of geodetic receivers, allowing the performance of differential corrections, this process is named Network Real-Time Kinematic (NRTK). The corrections are applied to ionospheric and tropospheric delays and are broadcasted to the rover.

Using a master station, described by A subscript in equations (3.2) and (3.3), which has a known position , and the smartphone as a rover, the pseudorange corrections (PRC) and carrier-phase corrections (CPC) for real-time methods are described below.

$$PRC^p(t) = \rho_A^p(t) - R_A^p(t) - cdt^p(t) - cdT_A(t) = \alpha I_A^p(t) - T_A^p(t) - E_A^p(t) \tag{3.2}$$

$$CPC(t) = \rho_A^p(t) - \Phi_A^p(t) - \lambda N_A^p(t) - cdt^p(t) - cdT_A(t) = \alpha I_A^p(t) - T_A^p(t) - E_A^p(t) \tag{3.3}$$

The simbols involved by the equations are:

- ρ_A^p: Geometric range.
- R_A^p: Pseudorange measurement
- Φ_A^p: Carrier-phase measurement.
- cdT_A: Biases of the receiver clock multiplied by the speed of light.
- cdt^p: Biases of the satellite clock multiplied by the speed of light.
- αI_A^p: Ionospheric propagation delay.
- αI_A^p: Ionospheric propagation delay.
- T_A^p: Tropospheric propagation delay.
- E_A^p: Ephemeris error.
- λN_A^p: Carrier-phase ambiguity

The rover (B) sees the CPC and PRC values from the master station as the following equations (3.4 and 3.5).

$$R_B^p(t)_{correct} = R_B^p(t) + PRC^p(t) = \rho_B^p(t) - cdT_{AB}(t) + \Delta E_A^p(t) - \Delta I_{AB}^p(t) + T_{AB}^p(t) \quad (3.4)$$

$$\varphi_B^p(t)_{correct} = \rho_B^p(t) + CPC(t) = \rho_B^p(t) - cdT_{AB}(t) - \lambda N_{AB}^p(t) + \Delta I_{AB}^p(t) + \Delta T_{AB}^p(t) + \Delta E_{AB}^p \quad (3.5)$$

The most relevant is the carrier-phase measurement which can reach a centimeter accuracy.

Chapter 4

Positioning Data

Obtaining raw data from different devices can be done with many applications, the one selected for the smartphone usage is Geo++ Rinex Logger app, in which can be obtained the Rinex observable files to procced with the post-processing and real-time studies. Furthermore, another device has been used, u-blox is a low cost location device that can be used as base station and provide raw data, observable files, navigation files and more.

4.1 RINEX

The open source software presents some different version, and both of the were tested in order to compare and check the solutions provided. One of the was the version 2.x, in which the post-processing data obtained were good, but it do not give the possibility of processing Galileo signal. On the other hand, with versions in 3.x, the usage of Galileo signals is a reality, and the data processing obtained with this version allow to get better results. Once both version were tested, obviously the second one (v3.x) was selected for the data treatment.

The pseudorange (code) measurement, equivalent to the difference between the time in which the signal is transmitted by the satellite and when the signal is received by the rover.

The carrier-phase measurement (at one or both carriers) actually being a measurement on the beat frequency between the received carrier of the satellite signal and a receiver-generated reference frequency.

The observation time that is the reading of the receiver clock at the instant of validity of the carrier-phase and/or the code measurements.

And, sometimes the files could include the Doppler measurements.

The name of this Rinex files has the format "aaaabbbc.xxy" and the meaning is described below:

- **aaaa:** Represent the name of the receiver station.
- **bbb:** The day in which the measurement was registered.
- **c:** The time sequence of the day when the data were recorded.

- **xx:** Last two digits of the present year.
- **y:** Kind of file.

Focusing on the letter "y" described previously, different files are presented:

- **o:** Observation files.
- **d:** Observation with Hatanaka compression.
- **n:** Navigation GPS files.
- **m:** Meteorological files.
- **g:** Navigation Glonass files.
- **h:** Navigation message of the geostationary payload GPS and navigation SBAS.
- **f:** Navigation Beidou files.
- **l:** Navigation Galileo files.
- **q:** Navigation QZSS files.
- **p:** Navigation mixed GNSS files.
- **s:** Observation summary.

The header of the Rinex files, contains relative information about the station, antenna or receiver.

4.1.1 Observation files

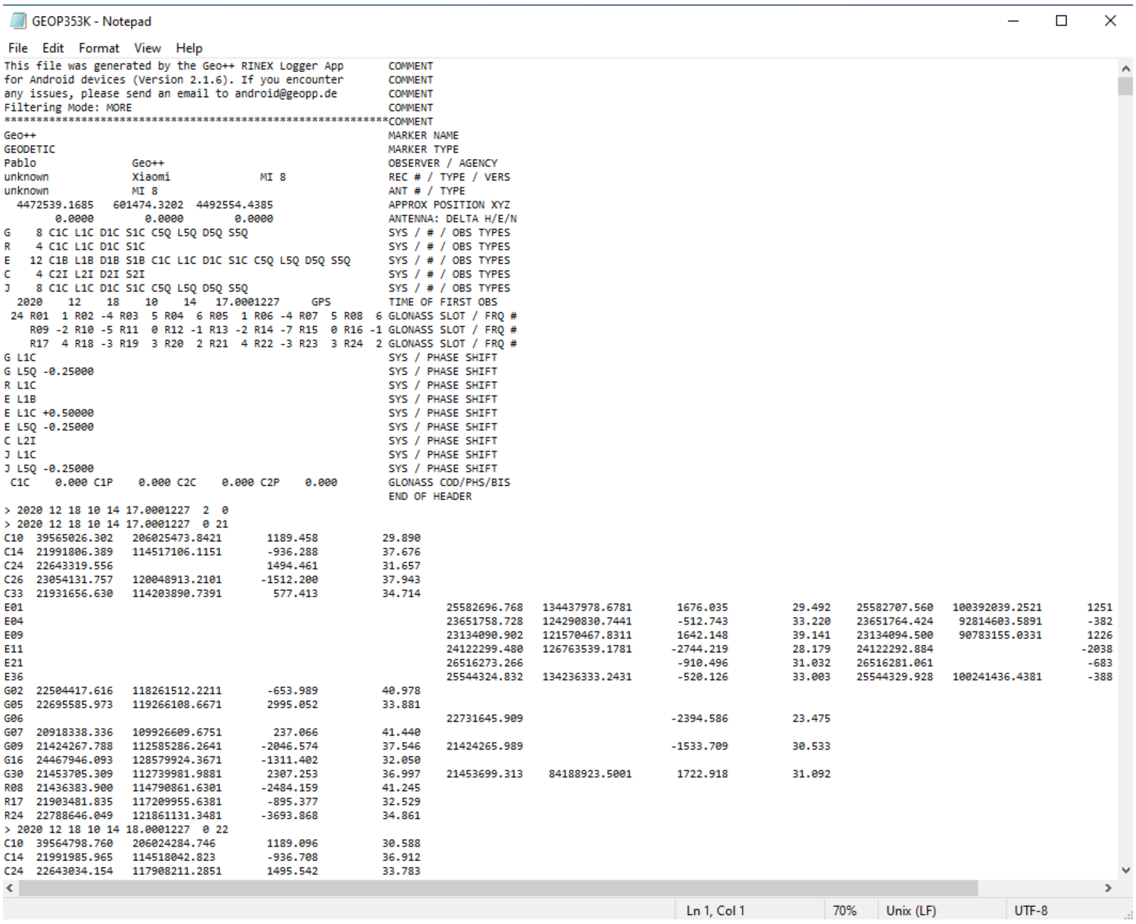
The kind of files that are involved in this field are the base station and the rover files. The first one, comes from a known and static position (permanent GPS station) and it will record 24 hours of data, giving to users the possibility of downloading the desired interval depending on the time of the rover file, being the base station file useful for applying corrections with RTKlib software. The second type, are recorded by the smartphone or the u-blox device, being a measurement of a non-static position (or yes) in which many errors are present and can be fixed as mentioned previously.

The header has a brief description of the collection app (if its used), the device (station name), antenna information, approximate coordinates, the satellites system available, the recording date, number and kind of observations, time interval, first observation time and others.

Can be found in the file: the pseudorange, carrier-phase, SNR and Doppler measurements; described in the first point of this chapter.

The kind of observation in the pseudorange case involves three different measurements, C1 (corresponding to C/A standard code over the frequency L1), P1 (Precise code over L1) and P2 (P code over L2). The Doppler effect gives also two measurements, D1 and D2 in L1 and L2 respectively. For SNR there are also S1 and S2, related again with the two frequencies mentioned.

Depending on the rover, (the different devices described) the information provided could be more precise or not, is very important, for a good develop of the project, to ensure a good device, not forgetting that also the devices with bad conditions have to be analized to have in mind all the possible behaviours.




```
This file was generated by the Geo++ RINEX Logger App      COMMENT
for Android devices (Version 2.1.6). If you encounter      COMMENT
any issues, please send an email to android@geopp.de       COMMENT
Filtering Mode: MORE                                       COMMENT
************************************************************COMMENT
Geo++                                                      MARKER NAME
GEODETIC                                                   MARKER TYPE
Pablo               Geo++                                  OBSERVER / AGENCY
unknown             Xiaomi              MI 8               REC # / TYPE / VERS
unknown             MI 8                                   ANT # / TYPE
  4472539.1685   601474.3202  4492554.4385                 APPROX POSITION XYZ
        0.0000        0.0000        0.0000                 ANTENNA: DELTA H/E/N
G    8 C1C L1C D1C S1C C5Q L5Q D5Q S5Q                     SYS / # / OBS TYPES
R    4 C1C L1C D1C S1C                                     SYS / # / OBS TYPES
E   12 C1B L1B D1B S1B C1C L1C D1C S1C C5Q L5Q D5Q S5Q     SYS / # / OBS TYPES
C    4 C2I L2I D2I S2I                                     SYS / # / OBS TYPES
J    8 C1C L1C D1C S1C C5Q L5Q D5Q S5Q                     SYS / # / OBS TYPES
  2020    12    18    10    14   17.0001227     GPS        TIME OF FIRST OBS
 24 R01  1 R02 -4 R03  5 R04  6 R05  1 R06 -4 R07  5 R08  6 GLONASS SLOT / FRQ #
    R09 -2 R10 -5 R11  0 R12 -1 R13 -2 R14 -7 R15  0 R16 -1 GLONASS SLOT / FRQ #
    R17  4 R18 -3 R19  3 R20  2 R21  4 R22 -3 R23  3 R24  2 GLONASS SLOT / FRQ #
G L1C                                                      SYS / PHASE SHIFT
G L5Q -0.25000                                             SYS / PHASE SHIFT
R L1C                                                      SYS / PHASE SHIFT
E L1B                                                      SYS / PHASE SHIFT
E L1C +0.50000                                             SYS / PHASE SHIFT
E L5Q -0.25000                                             SYS / PHASE SHIFT
C L2I                                                      SYS / PHASE SHIFT
J L1C                                                      SYS / PHASE SHIFT
J L5Q -0.25000                                             SYS / PHASE SHIFT
 C1C    0.000 C1P    0.000 C2C    0.000 C2P    0.000       GLONASS COD/PHS/BIS
                                                           END OF HEADER
> 2020 12 18 10 14 17.0001227  2  0
> 2020 12 18 10 14 17.0001227  0 21
C10  39565026.302   206025473.8421      1189.458         29.890
C14  21991806.389   114517106.1151      -936.288         37.676
C24  22643319.556                       1494.461         31.657
C26  23054131.757   120048913.2101     -1512.200         37.943
C33  21931656.630   114203890.7391       577.413         34.714
E01                                                               25582696.768   134437978.6781      1676.035        29.492   25582707.560   100392039.2521      1251
E04                                                               23651758.728   124290830.7441      -512.743        33.220   23651764.424    92814603.5891      -382
E09                                                               23134090.902   121570467.8311      1642.148        39.141   23134094.500    90783155.0331      1226
E11                                                               24122299.480   126763539.1781     -2744.219        28.179   24122292.884                      -2038
E21                                                               26516273.266                       -910.496        31.032   26516281.061                       -683
E36                                                               25544324.832   134236333.2431      -520.126        33.003   25544329.928   100241436.4381      -388
G02  22504417.616   118261512.2211      -653.989         40.978
G05  22695585.973   119266108.6671      2995.052         33.881
G06                                                               22731645.909                      -2394.586        23.475
G07  20918338.336   109926609.6751       237.066         41.440
G09  21424267.788   112585286.2641     -2046.574         37.546   21424265.989                      -1533.709        30.533
G16  24467946.093   128579924.3671     -1311.402         32.050
G30  21453705.309   112739981.9881      2307.253         36.997   21453699.313    84188923.5001      1722.918        31.092
R08  21436383.900   114790861.6301     -2484.159         41.245
R17  21903481.835   117209955.6381      -895.377         32.529
R24  22788646.049   121861131.3481     -3693.868         34.861
> 2020 12 18 10 14 18.0001227  0 22
C10  39564798.760   206024284.746       1189.096         30.588
C14  21991985.965   114518042.823       -936.708         36.912
C24  22643034.154   117908211.2851      1495.542         33.783
```

Figure 4.1: Observation Rover file

4.1.2 Navigation files

This file contains satellite information, orbits, clock parameters and the accuracy of the pseudorange measurements from the observable satellites. The header provides the date of creation of the file, the name of the satellite constellation and other information. The origin of the file is detailed in the name of it, being described in the point 4.1 of this fourth chapter. Navigation files are provided by different sources, depending on the receptor, we have to choose one that is near to our rover, to achieve a better accuracy in the corrections.

```
tori0460.21n - Notepad
File  Edit  Format  View  Help
     2.10           NAVIGATION DATA                         RINEX VERSION / TYPE
Spider V7.2.2.7822  POLITO              2021 02 16 00:02    PGM / RUN BY / DATE
    8.3819D-09 -7.4506D-09 -5.9605D-08  5.9605D-08          ION ALPHA
    8.8064D+04 -1.6384D+04 -1.9661D+05  6.5536D+04          ION BETA
   -1.862645149231D-09 0.000000000000D+00   405504     2145 DELTA-UTC: A0,A1,T,W
    18                                                      LEAP SECONDS
 ========================================================== COMMENT
   SPIN3 GNSS - SERVIZIO DI POSIZIONAMENTO INTERREGIONALE   COMMENT
              REGIONE PIEMONTE, REGIONE LOMBARDIA           COMMENT
               E REGIONE AUTONOMA VALLE D'AOSTA             COMMENT
                                -                           COMMENT
      WEBSITE: WWW.SPINGNSS.IT - EMAIL: INFO.GNSS@CSI.IT    COMMENT
 ========================================================== COMMENT
                                                            END OF HEADER
 1 21 02 14 23 59 44.0 7.610507309437D-04-7.730704965070D-12 0.000000000000D+00
    1.900000000000D+01 7.334375000000D+01 4.281607068890D-09 5.819234580992D-01
    4.015862941742D-06 1.038461434655D-02 5.606561899185D-06 5.153687667847D+03
    8.638400000000D+04-8.381903171539D-08-1.682963642397D+00 1.452863216400D-07
    9.829542210631D-01 2.815937500000D+02 8.201778440650D-01-8.221770819716D-09
    1.739358101993D-10 1.000000000000D+00 2.145000000000D+03 0.000000000000D+00
    2.000000000000D+00 0.000000000000D+00 4.656612873077D-09 1.900000000000D+01
    7.911000000000D+04 0.000000000000D+00
 2 21 02 14 18 00  0.0-5.768379196525D-04-3.979039320257D-12 0.000000000000D+00
    6.100000000000D+01 8.537500000000D+01 4.401969011525D-09-2.319884020824D+00
    4.518777132034D-06 2.037869498599D-02 5.500391125679D-06 5.153616254807D+03
    6.480000000000D+04 2.812594175339D-07-1.764528288806D+00 9.685754776001D-08
    9.617545280451D-01 2.730625000000D+02-1.545670872630D+00-7.872113627627D-09
    3.653723712471D-10 1.000000000000D+00 2.145000000000D+03 0.000000000000D+00
    2.000000000000D+00 0.000000000000D+00-1.769512891769D-08 6.100000000000D+01
    5.754000000000D+04 0.000000000000D+00
 3 21 02 15 00 00  0.0-8.320948109031D-05-1.011812855722D-11 0.000000000000D+00
    6.800000000000D+01-6.909375000000D+01 4.614478132936D-09-5.571537563532D-01
   -3.660097718239D-06 3.262606565841D-03 4.451721906662D-06 5.153652858734D+03
    8.640000000000D+04 1.117587089539D-08-6.440224302336D-01 5.215406417847D-08
    9.684056177982D-01 2.960625000000D+02 8.510457520087D-01-8.333561396512D-09
   -3.568005890742D-10 1.000000000000D+00 2.145000000000D+03 0.000000000000D+00
    2.000000000000D+00 0.000000000000D+00 1.862645149231D-09 6.800000000000D+01
    7.914000000000D+04 0.000000000000D+00
 4 21 02 14 18 00  0.0-1.802067272365D-04-2.614797267597D-12 0.000000000000D+00
    1.790000000000D+02-4.578125000000D+01 4.246962781451D-09-8.529343807120D-01
   -2.382323145866D-06 1.123445224948D-03 1.195631921291D-05 5.153598321915D+03
    6.480000000000D+04 2.980232238770D-08 4.358979451505D-01-1.490116119385D-08
    9.607090785093D-01 1.458750000000D+02-3.014419640118D+00-7.871756579902D-09
```

Figure 4.2: Navigation file

The header that *Figure 4.2* shows is the same for all the satellite signals, and then the measurements appears [13].

These measurements are divided in two different groups, referred to the same satellite and give information about the orbit. The columns show a first number that indicates the satellite PRN, and after this number the information about the time of the satellite is provided, obtaining finally all the information about position, velocity, orbit... as mentioned previously.

4.2 Geo++ Rinex Logger app

The application used for the data collection using a smartphone, is available in Android 7 or higher. This app allows to access raw positioning data in Rinex format (observation files).

When the app is opened, it can be observed the quantity of satellites available for the smartphone and even which of them are the best option to be observable. Then, with a button on the top of the app, the recording can be played. This interface is shown in *Figure 4.2*.

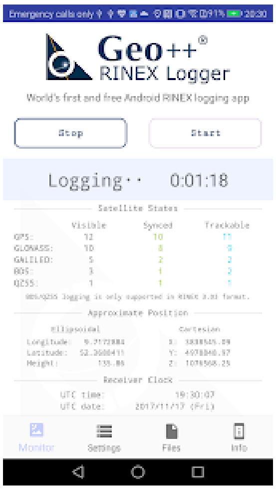


Figure 4.3: Geo++ Rinex Logger Interface

Furthermore, the recorded time is visible all the time, being able to stop the record touching the button "Stop". Then, once the recording is done, automatically the app save the files into a folder in the phone storage.

Also, in the bottom part of the app, the options "Settings", "Files" and "Info" are available for the user.

Chapter 5

Devices

With all the specifications described in the previous chapters, the data processing can be carried out, but first is also important to set which are going to be the devices used, and the functionality of each one.

5.1 Rover

This devices, which will be used as a rover, are included in the user segment, being the element of which the position wants to be known. This rover will provide the observation file relative to the user segment.

The smartphones that have been used as a rover for this project are: Huawei P8 lite, One Plus 7 and Xiaomi Mi 8. All the devices are able to allow Android 7 and the app can be downloaded. The main characteristcis of the phones are shown below:

Figure 5.1: Huawei P8 Lite [5]

Huawei P8 Lite				
Constellations	**Observations**	**Cost (€)**	**Weight (g)**	**Dimension (mm)**
AGPS, Glonass	L1	160	131	70.6x143

Table 5.1: Huawei P8 Lite Characteristics

Figure 5.2: One Plus 7 [2]

One Plus 7				
Constellations	**Observations**	**Cost (€)**	**Weight (g)**	**Dimension (mm)**
GPS, Glonass	L1+L5 (Dual)	380	131	74.8x157.7

Table 5.2: One Plus 7 Characteristics

Figure 5.3: Xiaomi Mi 8 [6]

Xiaomi Mi 8				
Constellations	**Observations**	**Cost (€)**	**Weight (g)**	**Dimension (mm)**
GPS, Glonass, Galileo, Beidou	Dual Band	319	175	74.8x154.9

Table 5.3: Xiaomi Mi 8 Characteristics

Figure 5.4: Samsung A40 [1]

Samsung A40				
Constellations	**Observations**	**Cost (€)**	**Weight (g)**	**Dimension (mm)**
GPS, Glonass, Galileo, Beidou	Dual Band	159	140	69x144.3

Table 5.4: Samsung A40 Characteristics

Xiaomi Mi 8 introduced in 2018 the dual-band chipset Broadcom BCM47755 GNSS, which allow to use the frequencies L5 and E5a (described in *section 1.2*). One of the most relevant

 Pedro Ríos Bravo

features of a dual-frequency receiver is that it allows performing carrier and code-phase measurements, being really useful to reduce the ionospheric effect, which is the main error in GNSS. With the linear combination of both measurements, the elimination of this effect can be achieved.

Not all the smartphones are able to obtain good measurements, this is due to different factors among those which are the features of them and even is important to consider the low-grade antenna. Furthermore, the u-blox device could be used even as a rover, the characteristics of this location device will be described in section 5.3.

The principle of the rover station is explained in *Figure 1.2*, the data collection can be done by a static or motion rover.

5.2 Base Stations

The control segment is formed by a set of station that controls the space segment, this is thanks to the constant monitoring of the satellites performed by stations located in strategic places. Each one of these stations are located in a place with known coordinates with high accuracy and with GPS dual-band (L1 and L2) receiver.

As a base station can be used the one created by Politècnico di Torino [3], which is located in the building and provide observation and navigation files. Moreover, the u-blox device, if its setted in a specific position can be used also as base station.

Figure 5.5: Antenna from PoliTO

The model used for the project is LEICAR25.R3 and their main parts are shown in *Figure 5.6*.

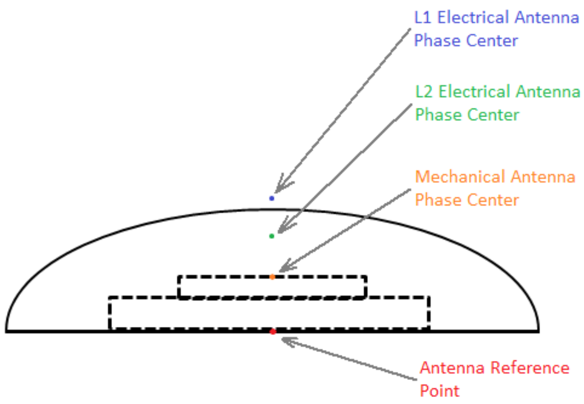


Figure 5.6: Antenna Parts

5.3 u-blox

This module has a ceramic antenna and EEPROM integrated given a higher accuracy. The way to use this device is really simple, it realizes the communication with serial port and allow 4 pins: VCC, RX, TX and GND. The model that was used in this project is u-blox M8T.

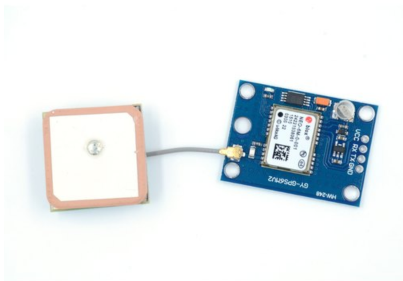


Figure 5.7: u-blox device [15]

Among the most important characteristics stand out:

- Observation files.
- Supply voltage 3.5 - 5 V.
- Capture sensitivity - 148 dBm.
- Tracking sensitivity - 161 dBm.
- Frenquency Band: L1.
- Accuracy of 1 microsecond.

Chapter 6

Data Processing

6.1 Positioning methods using smartphones

The software utilized for the data processing is RTKLib [19], in which the post-processing and the real-time process can be performed.

Due to the main smartphone used for the present work is Xiaomi Mi 8, which includes pseudorange and carrier-phase, is important to describe the main characteristics of these two measurements.

6.2 Geodetic to UTM transformation

For a better comprehension of the data obtained in the position files, which comes into Geodetic coordinates, is good to do a transformation into UTM (East, North and Altitude). Knowing that the data recorded correspond to the UTM Zone 32N and being the hemisphere North and East positioning. The computation is done by the following equations [12].

To do the transformation, the earth is represented by an ellipsoid, and previously of all the computation is important to know the main characteristics of it.

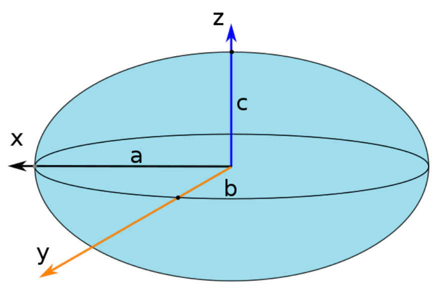


Figure 6.1: Ellipsoid

Where "a" is the semi-major and "b" the semi-minor axis. These two measurements, will be obtained from the WGS84 (World Geodetic System 1984), which is a geodetic coordinate system that allows to compute any point over the earth surface with three given units (x,y,z) [10]. Then, the values for the semi-axis and the curvature polar radius "c" are:

$$a = 6378137m \tag{6.1}$$

$$b = 6356752.314245m \tag{6.2}$$

$$c = 6399593.626m \tag{6.3}$$

First, the UTM zone will be computed, using the geodetic longitude of the point (λ_p) from the position file (Equation 8.4), being also computed the logitude of the central meridian (λ_0) (Equation 8.5) and the longitude variation (Equation 8.6).

$$ZONE_{UTM} = round(\frac{\lambda_p}{6}) + 31 \tag{6.4}$$

$$\lambda_0 = 6 \cdot ZONE_{UTM} - 183 \tag{6.5}$$

$$\Delta\lambda = (\lambda_p - \lambda_0) \cdot \frac{\pi}{180} \tag{6.6}$$

$$\lambda_{pr} = \lambda_p \cdot \frac{\pi}{180} \tag{6.7}$$

With the UTM Zone known, the computation of this parameter should be always 32, which is the one that correspond to Torino. The different parameter that will be involved in the transformation will be described bellow:

- e**:** Eccentricity.
- e'**:** Second ec centricity.
- k_0**:** Scale factor with 0.9996 value.
- E_0**:** Fake East, 500 000 meter.
- N_0**:** Fake North is 0 for North latitudes.

With all this background, the procedure of the transformation can be carried out. Is important that the units of latitude and longitude should be in radians for this calculations.

$$e = \frac{\sqrt{a^2 - b^2}}{a} = 0.081819191 \tag{6.8}$$

$$e' = \frac{\sqrt{a^2 - b^2}}{b} = 0.082094438 \tag{6.9}$$

$$e'^2 = 0.00676817 \tag{6.10}$$

$$A = cos(\lambda_{pr}) \cdot sin(\Delta\lambda) \tag{6.11}$$

$$\xi = \frac{1}{2} \cdot ln(\frac{1 + A}{1 - A}) \tag{6.12}$$

$$\eta = atan\frac{tan(\lambda_{pr})}{cos(\Delta\lambda)} - \lambda pr \tag{6.13}$$

$$\nu = \frac{c}{(1 + e'^2 \cdot cos^2(\lambda_{pr}))^{1/2}} \cdot 0.9996 \tag{6.14}$$

$$\zeta = \frac{e'^2}{2} \cdot \xi^2 \cdot cos^2(\lambda_{pr}) \tag{6.15}$$

$$A_1 = sin(2 \cdot \lambda_{pr}) \tag{6.16}$$

$$A_2 = A1 \cdot cos^2(\lambda_{pr}) \tag{6.17}$$

$$J_2 = \lambda_{pr} \cdot \frac{A_1}{2} \tag{6.18}$$

$$J_4 = \frac{3 \cdot J_2 + A_2}{4} \tag{6.19}$$

$$J_6 = \frac{5 \cdot J_5 + A_2 \cdot cos^2(\lambda_{pr}}{3} \tag{6.20}$$

$$\alpha = \frac{3}{4} \cdot e'^2 \tag{6.21}$$

$$\beta = \frac{5}{3} \cdot \alpha^2 \tag{6.22}$$

$$\gamma = \frac{35}{27} \cdot \alpha^3 \tag{6.23}$$

$$B_\phi = 0.9996 \cdot c \cdot (\lambda_{pr} - \alpha \cdot J_2 + \beta \cdot J_4 - \gamma \cdot J_6) \tag{6.24}$$

With this the computation of X_{East} and Y_{North} is easily calculated.

$$X_{East} = \xi \cdot \nu \cdot (1 + \frac{\zeta}{3}) + 500000 \tag{6.25}$$

$$Y_{North} = \eta \cdot \nu \cdot (1 + \zeta) + B_\phi \tag{6.26}$$

Then, all the equations described will be programmed in MatLab for an automatic processing of the position files, for this reason, the file should be studied to have a better comprehension. The columns with the longitude, latitude and height information are relevant for the computation.

```
% program   : RTKPOST ver.2.4.2
% inp file  : C:\Users\pedro\Desktop\Solutions\2021-01-09_Sol\Xiaomi Mi8\Receiver\2021-01-09,10_39_46_CET\GEOP009K.21o
% inp file  : C:\Users\pedro\Desktop\Solutions\2021-01-09_Sol\Xiaomi Mi8\Base Station\tori0090.21d
% inp file  : C:\Users\pedro\Desktop\Solutions\2021-01-09_Sol\Xiaomi Mi8\Base Station\tori0090.21n
% obs start : 2021/01/09 10:00:00.0 GPST (week2139 554400.0s)
% obs end   : 2021/01/09 10:45:15.0 GPST (week2139 557115.0s)
% pos mode  : static
% freqs     : L1+L2+L5
% solution  : combined
% elev mask : 15.0 deg
% dynamics  : off
% tidecorr  : off
% ionos opt : broadcast
% tropo opt : saastamoinen
% ephemeris : broadcast
% navi sys  : gps glonass
% amb res   : fix and hold
% amb glo   : on
% val thres : 3.0
% antenna1  :                        ( 0.0000  0.0000  0.0000)
% antenna2  :                        ( 0.0000  0.0000  0.0000)
% ref pos   : 45.063364581    7.661277662   310.8187
%
% (lat/lon/height=WGS84/[illegible]5:single,6:ppp,ns=# of satellites)
%  GPST                    latitude(deg) longitude(deg)  height(m)   Q  ns   sdn(m)   sde(m)   sdu(m)  sdne(m)  sdeu(m)  sdun(m) age(s)  ratio
2021/01/09 10:00:00.000   44.891383705    7.050585847  1504.0997   2   4   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   0.00    1.2
2021/01/09 10:00:01.000   44.891383704    7.050585846  1504.0996   2   4   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   1.00    1.2
2021/01/09 10:00:02.000   44.891383704    7.050585846  1504.0996   2   4   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   2.00    1.1
2021/01/09 10:00:03.000   44.891383704    7.050585846  1504.0996   2   4   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   3.00    1.1
2021/01/09 10:00:04.000   44.891383704    7.050585846  1504.0996   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   4.00    1.2
2021/01/09 10:00:05.000   44.891383704    7.050585846  1504.0996   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   5.00    1.2
2021/01/09 10:00:06.000   44.891383704    7.050585846  1504.0996   2   5   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   6.00    1.5
2021/01/09 10:00:07.000   44.891383704    7.050585846  1504.0996   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   7.00    1.2
2021/01/09 10:00:08.000   44.891383704    7.050585846  1504.0996   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   8.00    1.4
2021/01/09 10:00:09.000   44.891383704    7.050585846  1504.0996   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004   9.00    1.3
2021/01/09 10:00:10.000   44.891383704    7.050585846  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  10.00    1.4
2021/01/09 10:00:12.000   44.891383704    7.050585846  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  12.00    1.3
2021/01/09 10:00:13.000   44.891383704    7.050585846  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  13.00    1.1
2021/01/09 10:00:14.000   44.891383704    7.050585846  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  14.00    1.0
2021/01/09 10:00:15.000   44.891383704    7.050585845  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  15.00    1.0
2021/01/09 10:00:16.000   44.891383704    7.050585845  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  16.00    1.3
2021/01/09 10:00:18.000   44.891383704    7.050585845  1504.0995   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  18.00    1.4
2021/01/09 10:00:19.000   44.891383704    7.050585845  1504.0994   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  19.00    1.0
2021/01/09 10:00:20.000   44.891383704    7.050585845  1504.0994   2   5   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  20.00    1.1
2021/01/09 10:00:21.000   44.891383704    7.050585845  1504.0994   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  21.00    1.2
2021/01/09 10:00:22.000   44.891383704    7.050585844  1504.0993   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  22.00    1.2
2021/01/09 10:00:23.000   44.891383704    7.050585844  1504.0993   2   5   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  23.00    1.0
2021/01/09 10:00:24.000   44.891383704    7.050585844  1504.0992   2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  24.00    1.2
2021/01/09 10:00:25.000                                          2   6   0.0004   0.0003   0.0009   0.0001   0.0003  -0.0004  25.00    1.2
```

Figure 6.2: Position file, latitude, longitude and height

Chapter 7

RTKLIB Software

RTKLib [20] is an open source programs package made for the standard and precise positioning with GNSS. RTKLib consist on a library of portable programs and some access points which the library uses.

The standard and precise positioning study is developed with raw GNSS data. It is able to execute real-time and post-processing methods, using a base station for Precise Positioning Point (PPP) for example among others. Furthermore, the software is compatible with the main satellite constellations (GPS, GLONASS, Galileo, Beidou...) descirbed in *section 1.2*.

The software is executed with a graphical interface that can be run either in Windows or in Linux.

This software owns differents applications which can be used for different purposes, this apps will be described in the following sections.

7.1 Features and compatibilities

This software is able to realize different types of GNSS positioning, in real-time and post-processing:

- DGPS/DGNSS
- Simple
- Static (Post-Processing)
- Kinematic (Post-Processing)
- Moving-Baseline
- Fixed
- PPP-Static
- PPP-Kinematic
- PPP-Fixed

In addition, RTKLib owns compatibility with different protocols and standard GNSS formats, allowing the following most relevant GNSS messages:

- RINEX
- CLK
- BINEX
- RTCM
- NTRIP 1.0
- SP3-c
- u-blox LEA-4T/5T/6T
- SkyTraq

These are one of the most important messages, and also is remarkable the compatibility with external communication through Serial, TCP/IP, local registered file and FTP/HTTP.

7.2 Applications

RTKLib includes the following Graphical User Interfaces (GUI AP), which give to the user the possibility of realize different studies and works.

Applications	
Description	**GUI AP**
Initializer	RTKLaunch
Post-Processing	RTKPost
Real-Time	RTKNavi
RINEX Converter	RTKConv
Communication Service	STRSVR
Plotting	RTKPlot
NTRIP Navigator	NTRIP
Downloading of GNSS data and products	RTKGet

Table 7.1: GUI Apps functions

7.2.1 RTKPlot

This application allows to visualize the positioning solutions from RTKPost and RTKNavi. This tool also shows the satellite signals recorded by the receptors, commonly GPS and GLONASS.

The principal window contents all the features shown below:

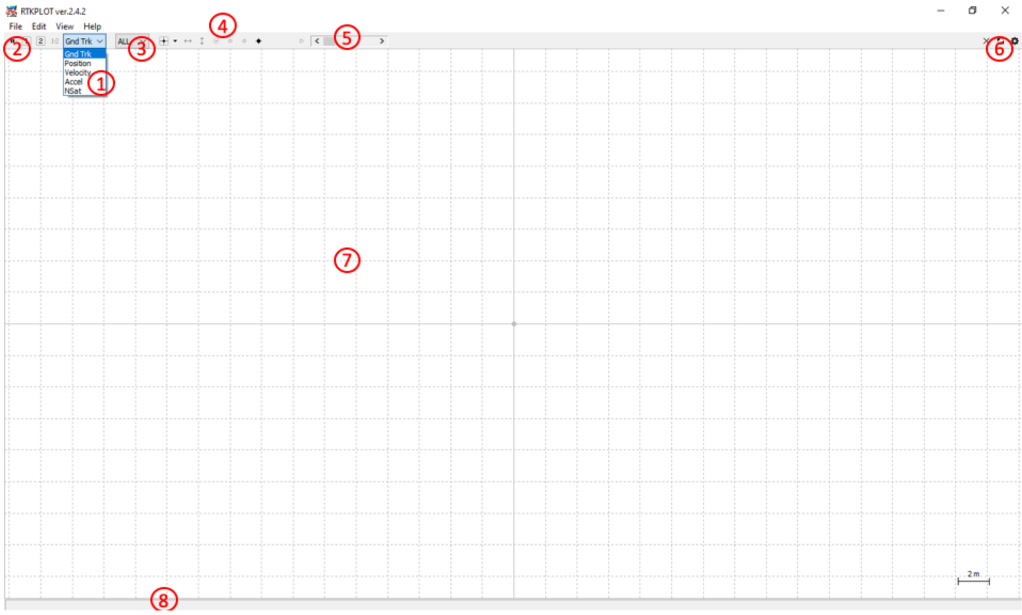


Figure 7.1: RTKPlot characteristics

- **1:** Kind of graphic selection.
- **2:** Toolbar.
- **3:** Signal quality.
- **4:** Toolbottons.
- **5:** Scrollbar.
- **6:** Refresh, configuration and close botton.
- **7:** Graphic area.
- **8:** State bar.

File: This option allows to open nav and obs files, connecting to RTKPost for the analysis.

Edit: Allowing to modify the connexion interval, the selected points and the options to work with RTKPost can be modified.

View: The toolbar, state bar, Google Maps or Earth view, etc, are shown.

Processed points, navigation and observation files

The kind of files that can be plotted by RTKPlot are: processed points (.pos), navigation (.nav) and observation (.obs). We can plot the main characteristics of the files, like the Satellite Visualization, in which the satellites available are shown; the Skyplot, a representation of the position of the satellites that are involved; the Doppler effect, the change in frequency in relation to the observer; and the SNR (Signal-to-Noise-Ratio), the proportion of the signal to noise, being 30 dB the best quality.

Different views are shown in *Figure 7.1*, in number 1, these appear when the post-processing method is performed, these are Gnd Trk, Position, Velocity, Accel and NSat.

Gnd Trk: In this display the error in positioning is shown, being able to see also the Google map view. In this plot the axis are horizontal and vertical, no time axe is available.

Positioning: This graphic shows the position evolution and errors by time, allowing to see the moment in which the signal produces the failures.

Velocity: Is interesting to see this field if the data recording were done in a kinematic way, seeing the velocity reached during the process.

Accel: As in the velocity representation, this plot is interesting in the case of a moving data record.

NSat: This section shows the number of valid satellites, age of differential (s) and the ratio factor for AR validation.

7.2.2 RTKConv

The main function of this app is to transform files in a complex format into a RINEX. This translates the messages from a native receptor.

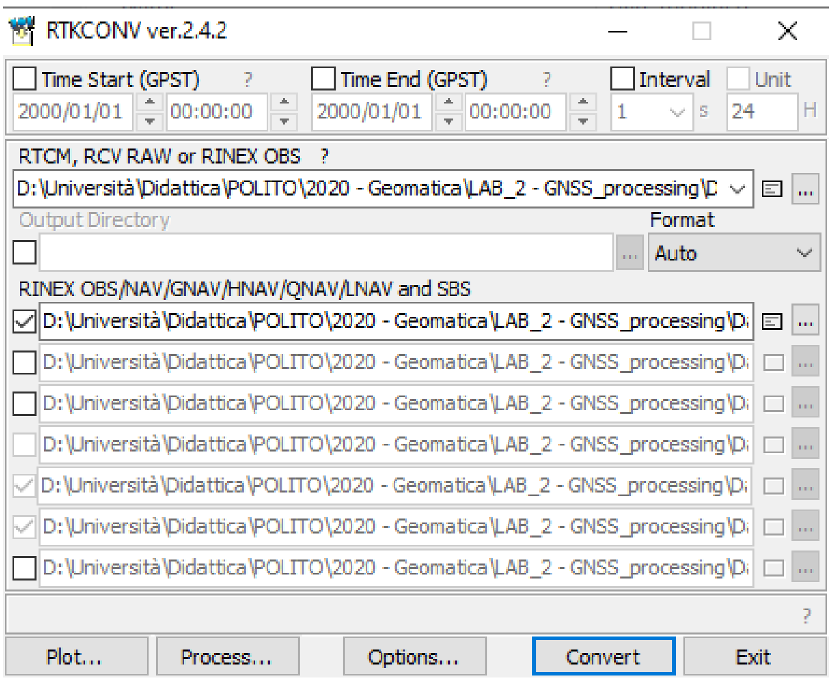


Figure 7.2: RTKConv interface

RTKConv can also extract SBAS messages from receiver raw data. This app is really important as RTKPost uses the Rinex format for the data treatment.

7.2.3 RTKPost

This app is dedicated to post-processing, is prepared to get solutions from data obtained by different measurement methods, using observation and navigation files (from rover and master devices).

The main interface of this app is represented in *Figure 7.3* where can be appreciated, the section which requires an observation file from a rover, the one referred to the base station file (observation) and in the last section a navigation (GPS or GLONASS, clock, SP3... file can be included.

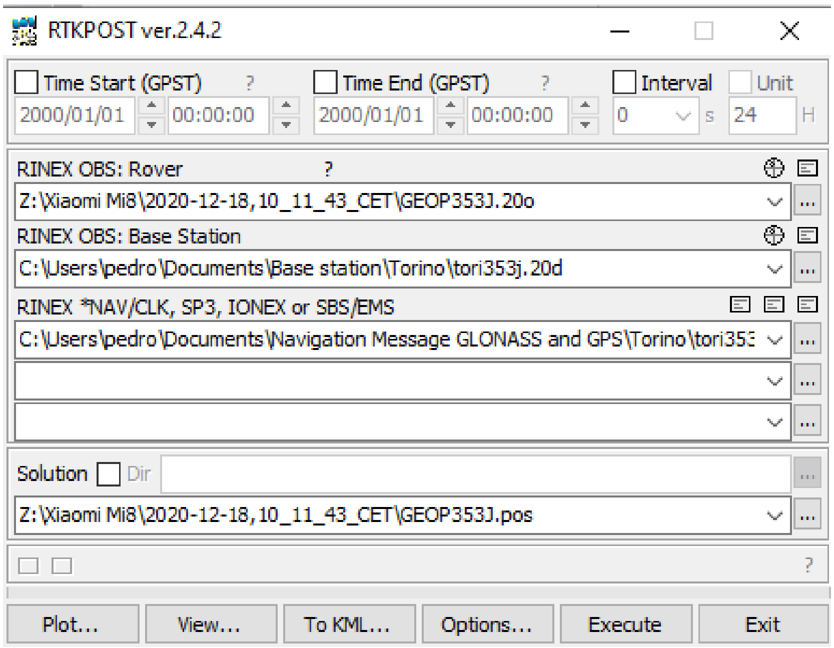


Figure 7.3: RTKPost interface

The bottom options, show the possibility of plotting which able to represent all the graphs explained in *section 7.2.1*; View, allow to see the .pos, generated from the initial files; To KML, is able to represent the positioning into Google Maps or Earth; Execute, is required for running the post-process; and Options will be described more in detail below.

Can be observed three main spaces to upload files for the data treatment:

- **RINEX OBS: Rover:** Here the path for observation file is included and it is processed like rover point.
- **RINEX OBS: Base Station:** Here the observation file of the point is added, is processed like Base Station.
- **SBAS Corrections space:** Can be introduced SBAS corrections: NAV/CLK, SP3, IONEX or SBS/EMS.

If the button "Options" is selected, the following interface appears.

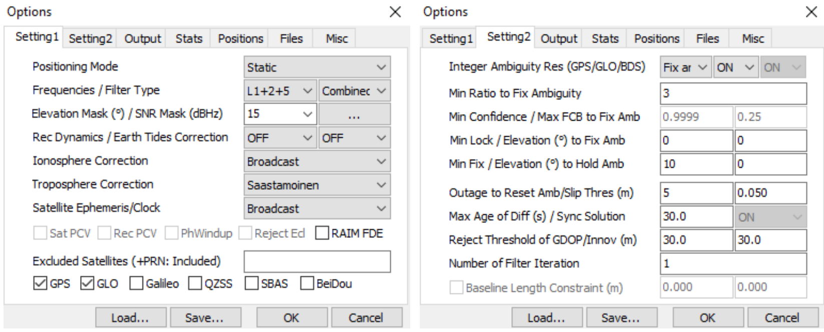


Figure 7.4: Options interface

In this section, some different possibilities can be selected for the post-processing method, like the positioning mode, the frequencies selected, ionospheric and tropospheric corrections, the integer ambiguity resolution, etc.

The options selected for the data treatment can be appreciated in *Figure 7.4*, the most relevant selections are the static positioning mode and the usage of GPS and GLONASS.

7.2.4 RTKNavi

This app of the RTKLib Software is dedicated for positioning in real-time, allowing the data treatment in their different phases, differential, autonomous, by code, carrier-phase... and also using different methods for data flow (serial, NTRIOP...).

With the rover connected to the PC, the app allows to know the coordinates of the receptor, visualize the satellites and all of this in real-time.

The main window of the app, included also the Options, Plot, Start and Exits buttons, can be appreciated in *Figure 7.5*.

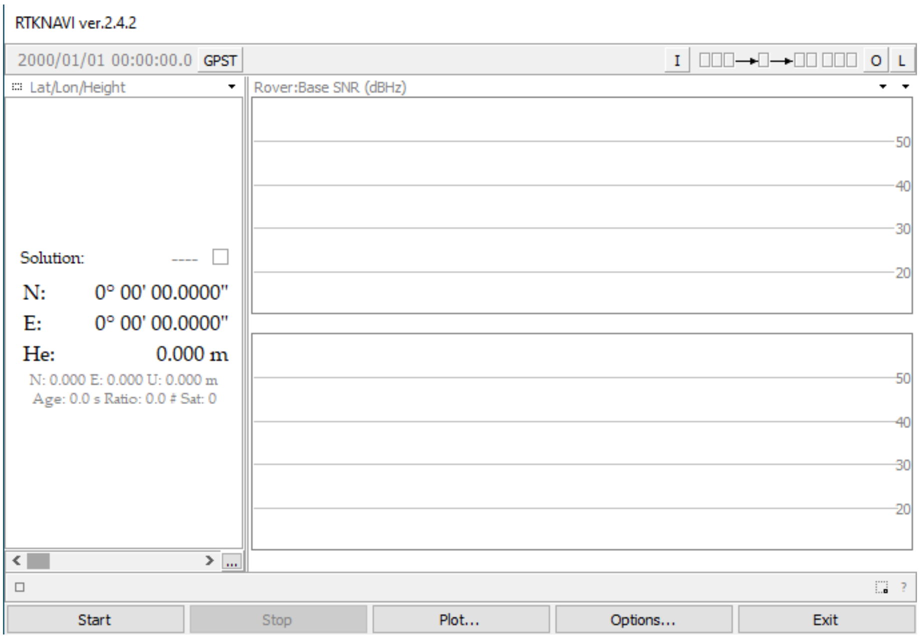


Figure 7.5: RTKNavi interface

In the upper left corner of the app, the GPS time is present, showing the GPS date and hour. Also, is remarkable the section "Solution" in which the coordinates of the receptor are shown, under this space, a small button can be appreciated, this is dedicated for the RTK monitor, giving acces to a real-time report in which shows the software characteristics, the positioning mode, satellite system, different kind of signals and frequencies... Furthermore, the satellite and signal information appears in the right part of the interface, plotting the satellite data in a rosette and the quality of the signals received.

This app, gives also the possibilities to select the positioning mode (static, kinematic, moving-base, PPP...), the option of selecting among one o more frequencies, choose the satellite navigation that is going to be used, change the data outputs, upload coordinates of the receiver, etc. For the data processing is a requirement to stablish the data input and output format.

Chapter 8

Positioning Results

8.1 Data collection and measurement campaign

The measurement campaign have been done using the smartphones Xiaomi and Samsung, u-blox and the base station located in Politècnico di Torino. These devices have been alternated in their use to reach all the combinations present *Table 8.1*.

Deviced Involved			
Xiaomi	**Samsung**	**u-blox**	**BS Polito**
Rover and Base Station	Rover and Base Station	Rover and Base Station	Base Station

Table 8.1: Usage of the deviced involved

The base station of PoliTO is able to reach GPS+GLO+GAL+BDS satellite signals and the precise location of the base is shown in *Figure 8.1*, is on the top of a building.

Figure 8.1: PoliTO Station location

The antenna is a LEIAR25.R3 model, being the receiver a LEICA GR30 loceted in the back office.

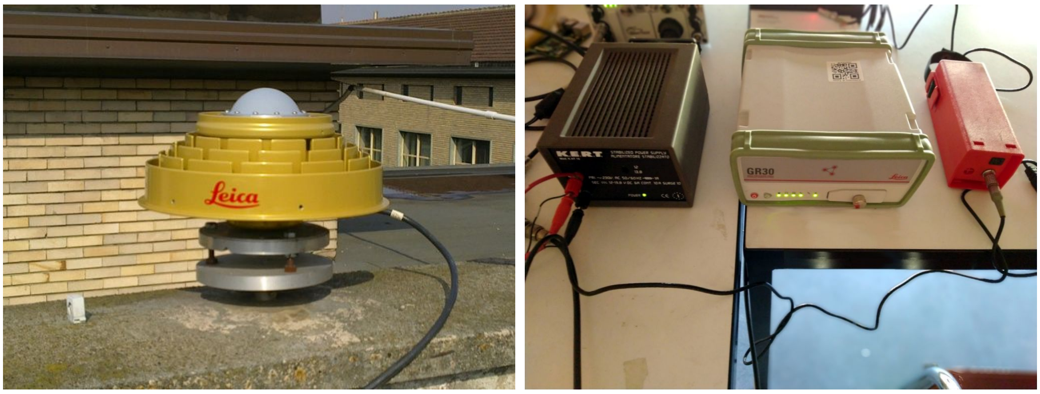


Figure 8.2: Antenna and receiver from PoliTO

With all the elements involved in the campaign defined, the procedure was realized during the day 9th of January and the 14th, 15th and 20th of February.

8.2 Post-Processing results

Once the data were collected, the post-processing method was carried out with the settings established as shown in *Figure 8.2*, selecting Static as positioning mode, the frequencies L1+2+5, GPS and GLO and "Fix and Hold" as Integer Ambiguity Resolution in which the static integer ambiguities are estimated and solved, among other features.

This last mentioned characteristic is really fascinating, because with a carrier-phase integer ambiguity resolution the unknown cycle ambiguities of the carrier-phase dataset are solved as integers obtaining a higher accuracy in the GNSS parameter estimation.

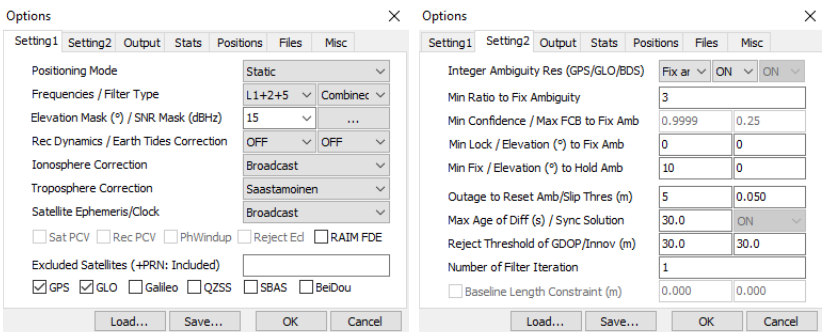


Figure 8.3: Settings for Post-Processing

It is also interesting to know, that when the study have been performed a .pos file is generated, Positioning Solution File, which contains information about the calculated process of the positioning results and what level of accuracy is statically expected.

8.2.1 Xiaomi 21-01-09

The measurement campaign was carried out on 9th of January, and all the combination were studied.

Receiver: Xiaomi, Master: Base Station PoliTO, Nav: Base Station PoliTO

Observing the positioning solution is important to highlight that mostly there are present three different Qs, Q = 1 (green colored) is referred for fixed point position, Q = 2 (yellow colored) is for floating point position and finally Q = 5 (red colored) single position.

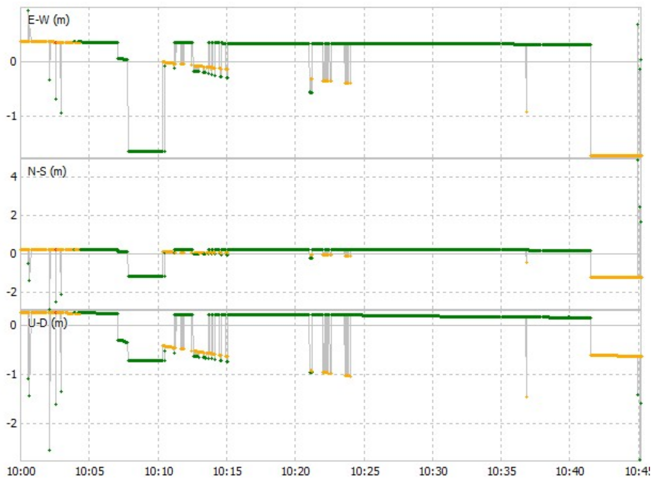


Figure 8.4: Positioning Solution 1 for Xiaomi

Analysing the solution shown, the maximum error values for position and the corresponding Q for the specific time are in *Table 8.2*

Maximum errors		
E-W	**N-S**	**U-D**
Q= 2; d= -1,9 m	Q= 1; d= 2,1 m	Q= 1; d= -2,8 m

Table 8.2: Distance errors and Qs

Mostly, the solution is performed in fixed point position (Q = 1) which means than the accuracy when this occurs is about millimeters.

With the ground tracking, these errors are also visualized, setting in the centre of the plot the original position of the receiver. Being the scale of the squares of 1 meter, *Figure 8.5* shows the maximum dispersion.

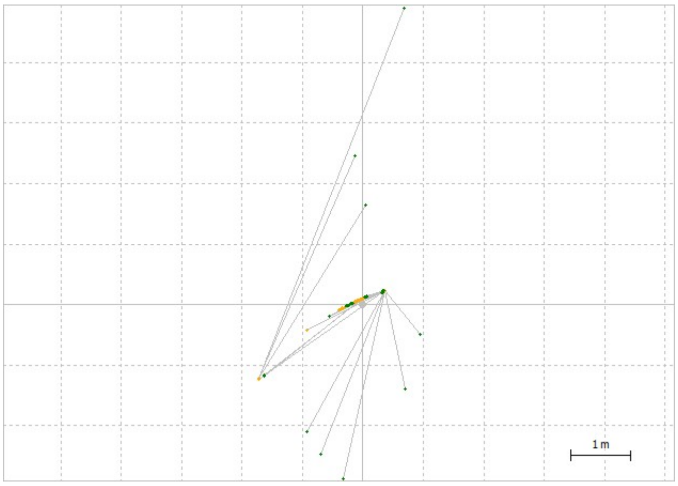


Figure 8.5: Ground Tracking Solution 1 for Xiaomi

The maximum distance represented in *Figure 8.5* is bigger than 5 meters.

8.2.2 Xiaomi and u-blox 21-01-09

Receiver: Xiaomi, Master: u-blox, Nav: u-blox

In this case, the data obtained from u-blox are used as a Master, getting from it the navigation and the base station files for the solution.

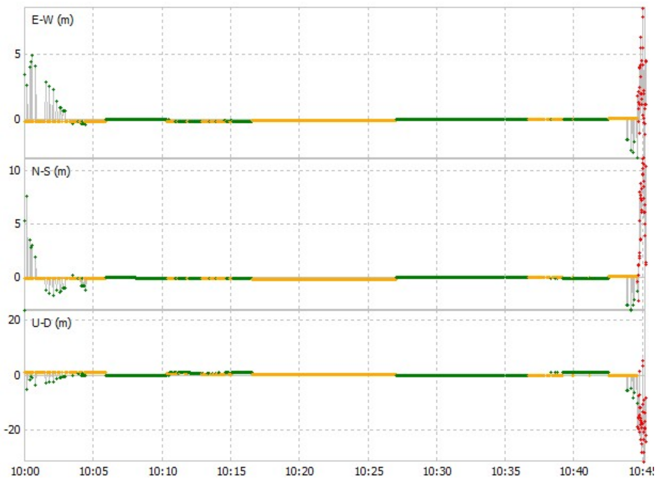


Figure 8.6: Positioning Solution 1 for Xiaomi and u-blox

Here in *Figure 8.6* the solution is really stable, but at the end can be appreciated some dispersion with Q = 5 which indicates that a single positioning its being performed.

Maximum errors		
E-W	**N-S**	**U-D**
Q= 5; d= 5,6 m	Q= 5; d= 10,2 m	Q= 5; d= -24,2 m

Table 8.3: Distance errors and Qs

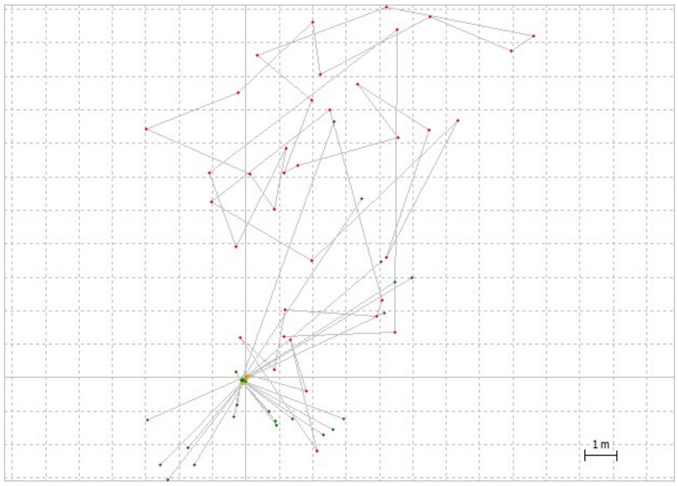


Figure 8.7: Ground Tracking Solution 1 for Xiaomi and u-blox

As can be observable from *Figure 8.7* the dispersion at the final points of positioning solution I also represented, being this distance bigger than 11 meters.

Receiver: Xiaomi, Master: Base Station PoliTO Nav: u-blox

Changing the master for the data recorded by the PoliTO Base Station, the results that can be thought to be better, have a worse response to this, probably originated by using three different devices for the operation, creating errors in the comparison of three files with different origins.

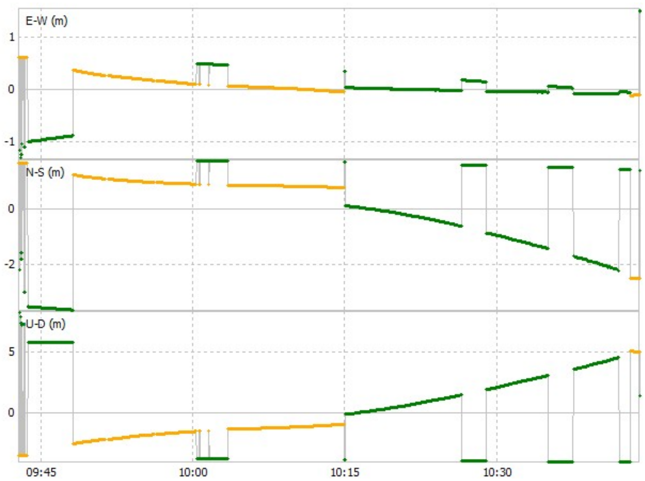


Figure 8.8: Positioning Solution 2 for Xiaomi and u-blox

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= -1,2 m	Q= 1; d= -2,8 m	Q= 1; d= 5,9 m

Table 8.4: Distance errors and Qs

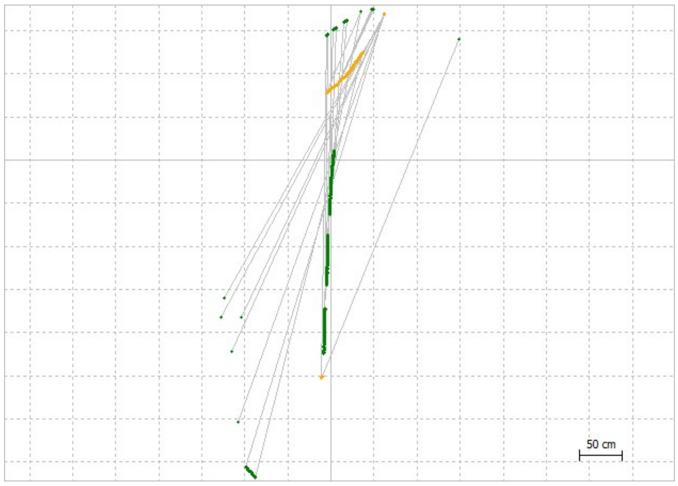


Figure 8.9: Ground Tracking Solution 2 for Xiaomi and u-blox

Receiver: Xiaomi, Master: u-blox Nav: Base Station PoliTO

Observing these results, can be confirmed that one kind of error may occur during the data acquisition, this is thought because of the perfection in the data, being the error of 0 meters.

This option is also a possibility during the performance of the study, not all the data are good and able to be studied, so this is a possibility that is good to keep in mind and to know that the error in the measurement is possible.

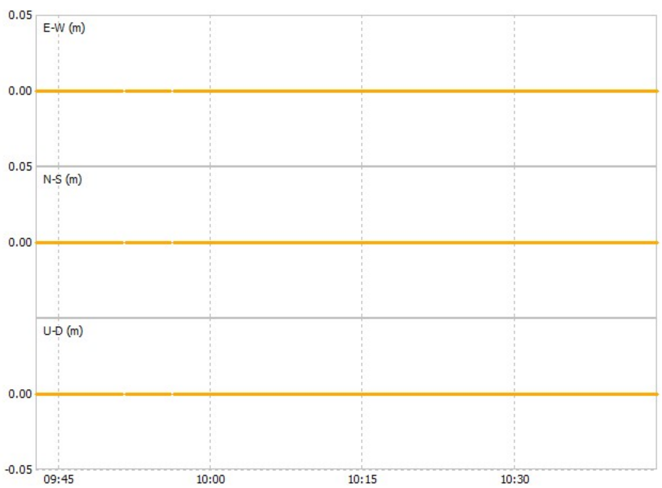


Figure 8.10: Positioning Solution 3 for Xiaomi and u-blox

Maximum errors		
E-W	**N-S**	**U-D**
Q= 2; d= 0 m	Q= 2; d= 0 m	Q= 2; d= 0 m

Table 8.5: Distance errors and Qs

8.2.3 u-blox 21-01-09

Receiver: u-blox, Master: Base Station PoliTO Nav: u-blox

The behaviour of the u-blox, should be more precise than the others presented, in this case, the master and the navigation files are different, one comes from the base station in Politècnico and the other from the own u-blox, which probably will cause a non-regular behaviour. Nevertheless, the result are very similar to the Smartphone ones.

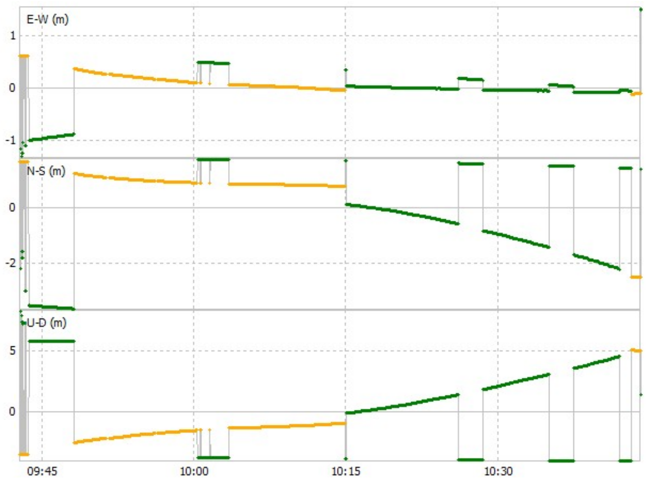


Figure 8.11: Positioning Solution 1 for u-blox

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= -1,2 m	Q= 1; d= -2,7 m	Q= 1; d= 5,5 m

Table 8.6: Distance errors and Qs

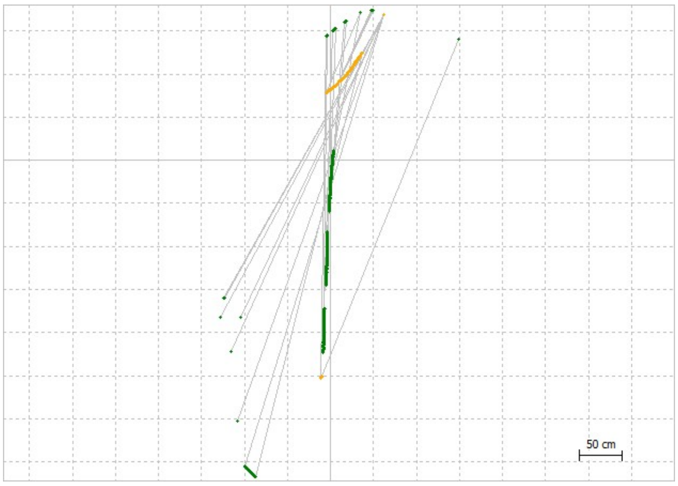


Figure 8.12: Ground Tracking Solution 1 for u-blox

8.2.4 Samsung 21-02-14

Receiver: Samsung, Master: Base Station PoliTO Nav: Base Station PoliTO

This is a really interesting analysis, because it will be the combination used to prove the performance of the Samsung smartphone, comparing it with the u-blox behaviour, obviously in both cases with the same base station and navigation file.

Figure 8.13, shows that in practically the three cases, the error increases when the value of the Q is 1, this is because of the computation capability of the smartphone, and if the calculation method is reduced into a fixed point positioning, the accuracy will be affected. However, when the Q values is 1, the results present an admisible error.

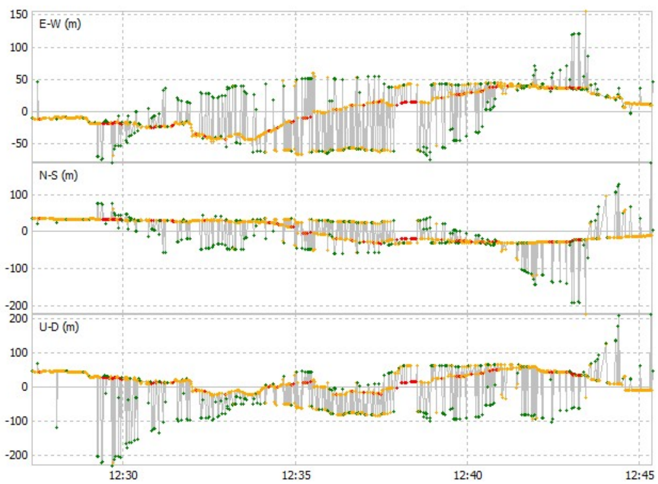


Figure 8.13: Positioning Solution 1 for Samsung

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= 155 m	Q= 2; d= -210 m	Q= 1; d= -211 m

Table 8.7: Distance errors and Qs

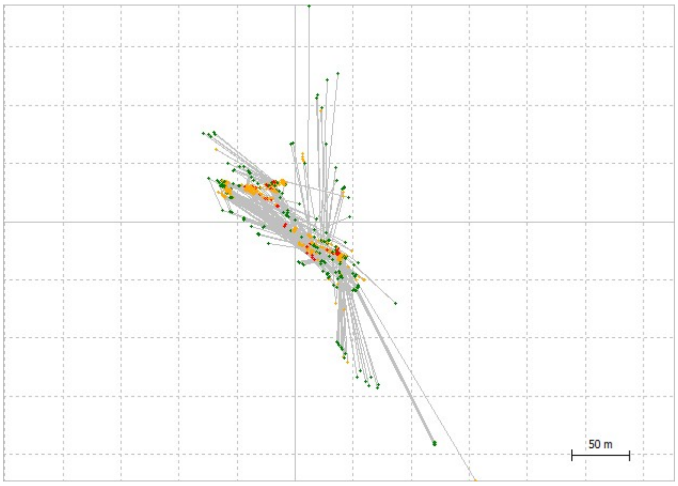


Figure 8.14: Ground Tracking Solution 1 for Samsung

8.2.5 Xiaomi, Samsung and u-blox 21-02-15

Receiver: Samsung, Master: Base Station PoliTO Nav: u-blox

In this case, can be observable that the majority of the data are computed with Q = 5, this indicates that probably the possibility of using the u-blox for the navigation file is not good, it will be checked again in the next measurements.

This kind of combination is being carried out, because it is interesting also to consider the possibility of an u-blox as a base station in places where a good location will be required for the mobile apps, so it is interesting to keep in mind this results.

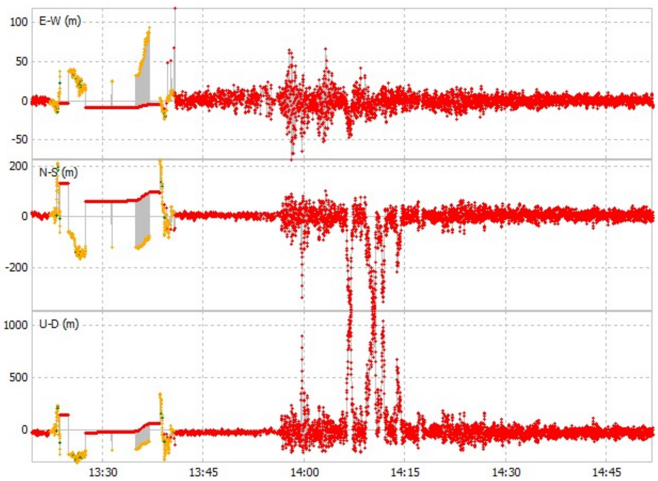


Figure 8.15: Positioning Solution 2 for Samsung

Maximum errors		
E-W	**N-S**	**U-D**
Q= 5; d= 110 m	Q= 5; d= -280 m	Q= 5; d> 1000 m

Table 8.8: Distance errors and Qs

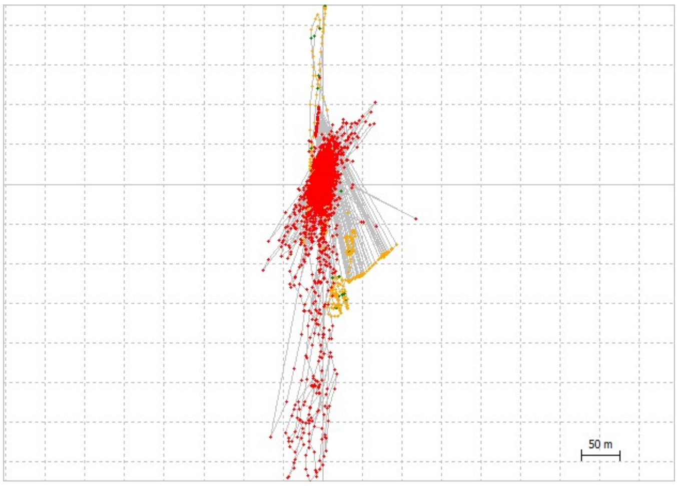


Figure 8.16: Ground Tracking Solution 2 for Samsung

Receiver: Samsung, Master: Base Station PoliTO Nav: Base Station PoliTO

This .pos file analysed will be used in the study, observing the dispersion after the ENU transformation and the comparison with the u-blox .pos file for the same date.

Samsung device, presents more or less a regular positioning of the points, being the worst results obtained when the Q is equal to 1. Probably, this measurement presents some repetitive errors in positioning, but nothing that gets a value that would complicate the smartphone performance. Moreover, the u-blox will be analysed in the same conditions and compared epoch by epoch with the smartphones, and also could occur that u-blox would present the same errors in the same epochs, being in this case the difference among smartphone and u-blox practically null.

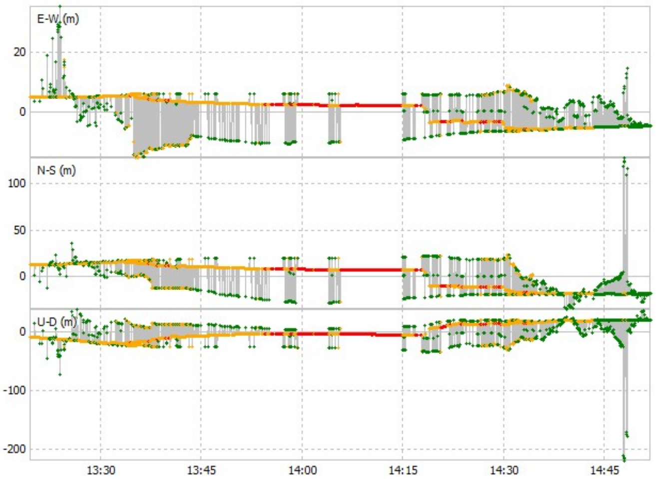


Figure 8.17: Positioning Solution 3 for Samsung

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= 22 m	Q= 1; d= 120 m	Q= 1; d= -208 m

Table 8.9: Distance errors and Qs

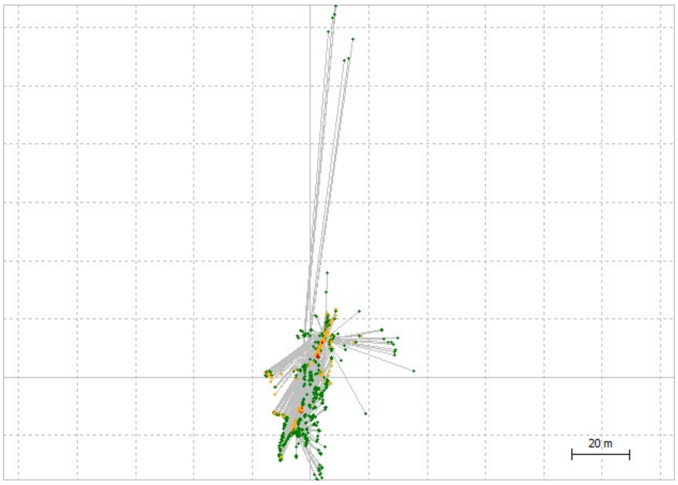


Figure 8.18: Ground Tracking Solution 3 for Samsung

Receiver: Xiaomi, Master: Base Station PoliTO Nav: u-blox

Again, when the u-blox is used to provide the information required for the navigation file, practically all the points are computed with Q = 5, so can be affirmed that is not a good option to consider.

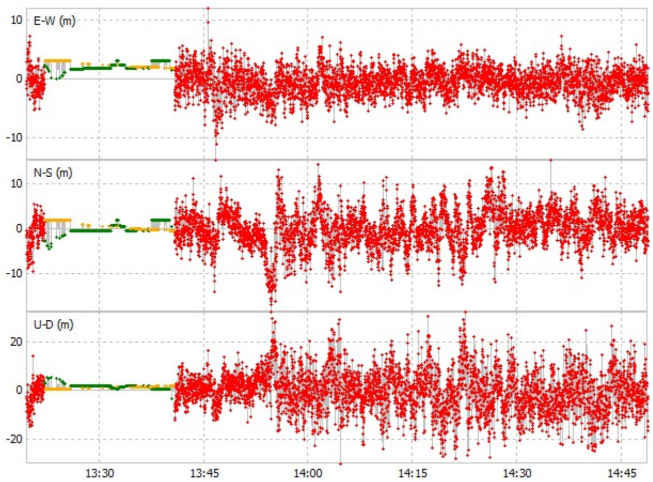


Figure 8.19: Positioning Solution 2 for Xiaomi

Maximum errors		
E-W	**N-S**	**U-D**
Q= 5; d= -13 m	Q= 5; d= -16 m	Q= 5; d= 24 m

Table 8.10: Distance errors and Qs

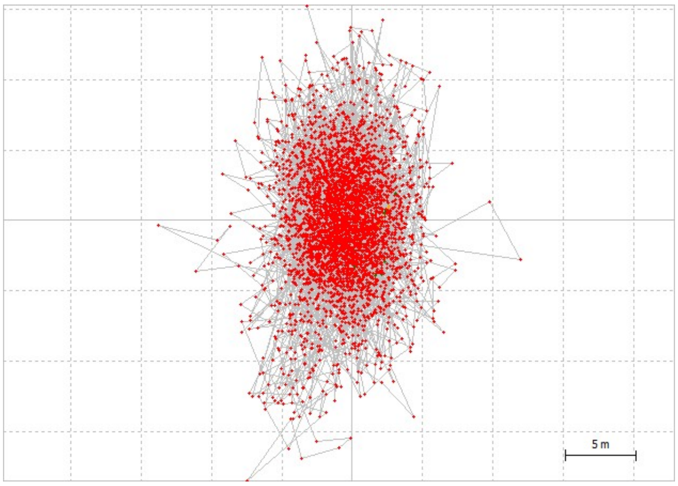


Figure 8.20: Ground Tracking Solution 2 for Xiaomi

Receiver: Xiaomi, Master: u-blox Nav: Base Station PoliTO

In this case, the u-blox is used as a master, since it did not work properly as a navigation file provider.

Here, the results have not got big errors and the response to the combination is good as appreciated in *Figure 8.23*. With this conclusion, the possibility of the usage of an u-blox as a Master y completely right, getting the base station data from one static point provided by an user (there are a lot available for data adquisition).

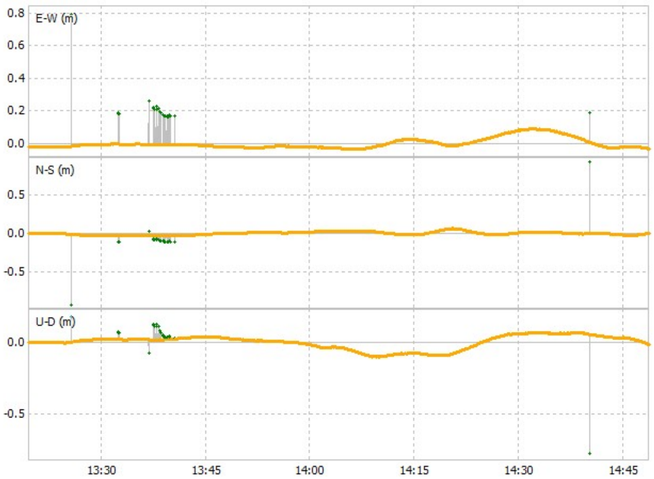


Figure 8.21: Positioning Solution 3 for Xiaomi

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= 0,8 m	Q= 1; d= 0,9 m	Q= 1; d= -0,8 m

Table 8.11: Distance errors and Qs

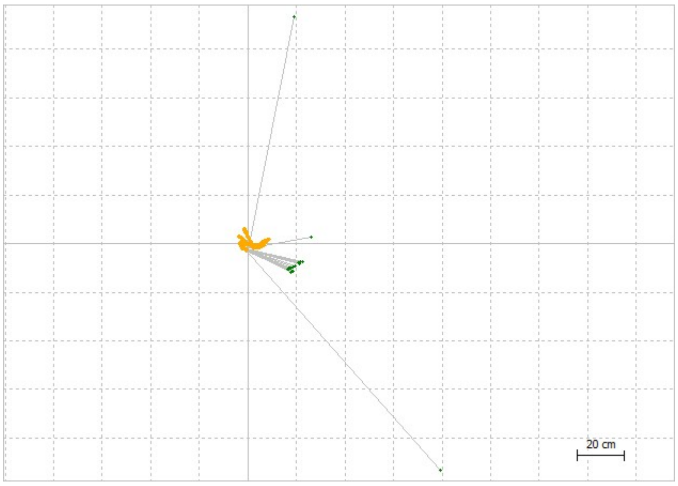


Figure 8.22: Ground Tracking Solution 3 for Xiaomi

Ground tracking, also shows the good response of the combination, only a few points have an insignificant error.

8.2.6 Xiaomi 21-02-15

Receiver: Xiaomi, Master: Base Station PoliTO Nav: Base Station PoliTO

This plots, show the combination used for the Xiaomi study and comparison. The results represented in *Figure 8.23* and *Figure 8.24* indicate that only a few points have errors, but not in a value that could refutate the good response of the smartphone to location requirements.

With this initial analysis, the situation is promising, observing that these erros occurs in some points and do not disable the good behaviour in positioning terms. So the comparison that will be carried out with the u-blox response should have minimal differences.

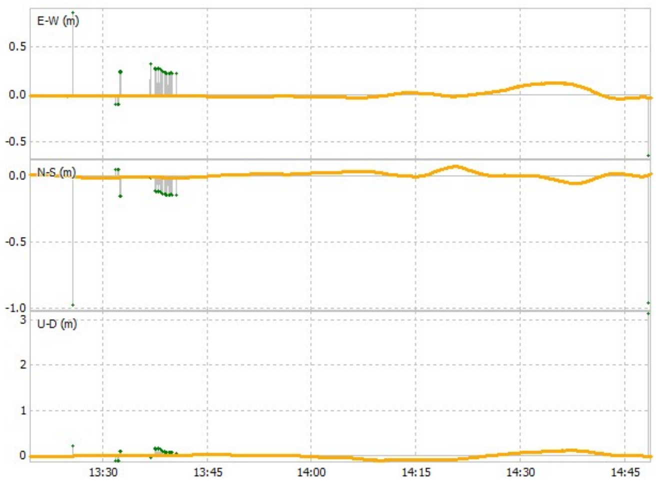


Figure 8.23: Positioning Solution 4 for Xiaomi

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= 0,8 m	Q= 1; d= -1 m	Q= 1; d= 0,1 m

Table 8.12: Distance errors and Qs

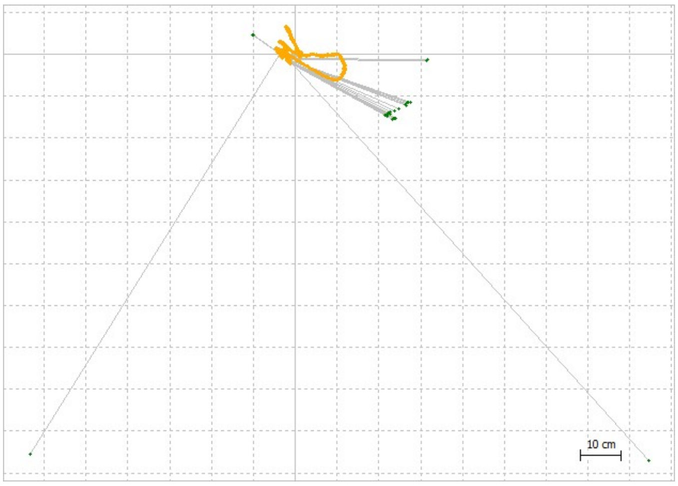


Figure 8.24: Ground Tracking Solution 4 for Xiaomi

8.2.7 u-blox 21-02-15

Receiver: u-blox, Master: Base Station PoliTO Nav: Base Station PoliTO

u-blox is the element selected as a pure location device, so the expected results should be the best in comparison with the smartphones and combinations.

However, the results that are shown in *Figure 8.25* and *Figure 8.26* indicate that they are similar to the Xiaomi ones, but even the smartphone presents better results, which means that the Xiaomi device could be used for location purposes.

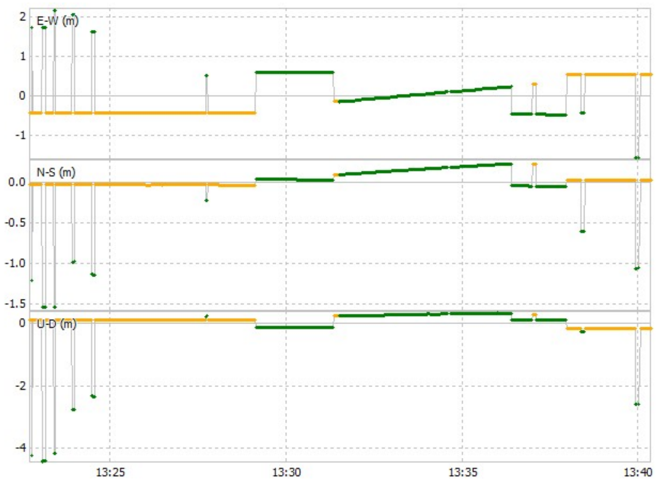


Figure 8.25: Positioning Solution 2 for u-blox

Maximum errors		
E-W	**N-S**	**U-D**
Q= 1; d= 2,1 m	Q= 1; d= -1,5 m	Q= 1; d= -4,1 m

Table 8.13: Distance errors and Qs

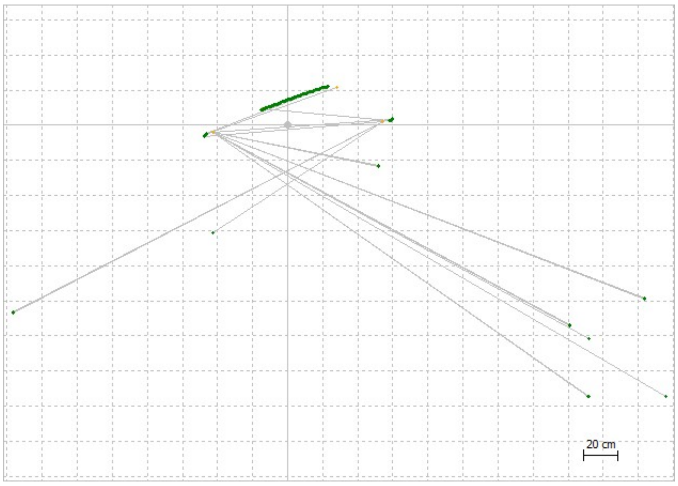


Figure 8.26: Ground Tracking Solution 2 for u-blox

8.2.8 Samsung 21-02-20

Receiver: Samsung, Master: Base Station PoliTO Nav: Base Station PoliTO

Here are shown the results to be compared with U-blox but for a different day, the 20th of February. In this case, Samsung device present results that are practicaly equal to the ones represented before, with big errors in a few points and presenting even Qs equal to five. However, and like has been said before, this kind of data could be also useful for location purposes.

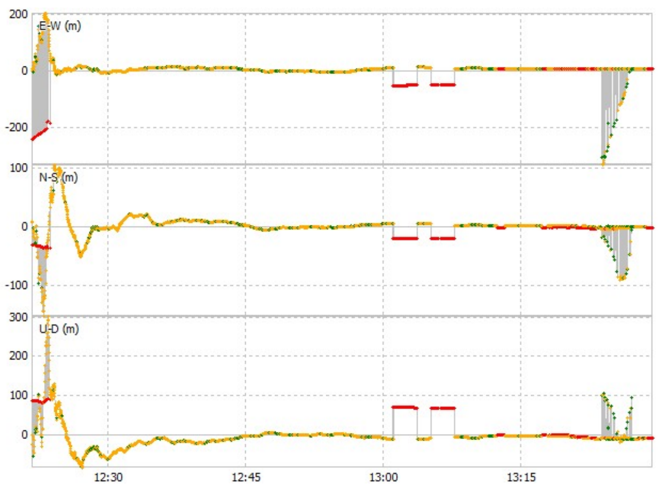


Figure 8.27: Positioning Solution 5 for Samsung

Maximum errors		
E-W	**N-S**	**U-D**
Q= 5; d= -210 m	Q= 2; d= -150 m	Q= 2; d= 302 m

Table 8.14: Distance errors and Qs

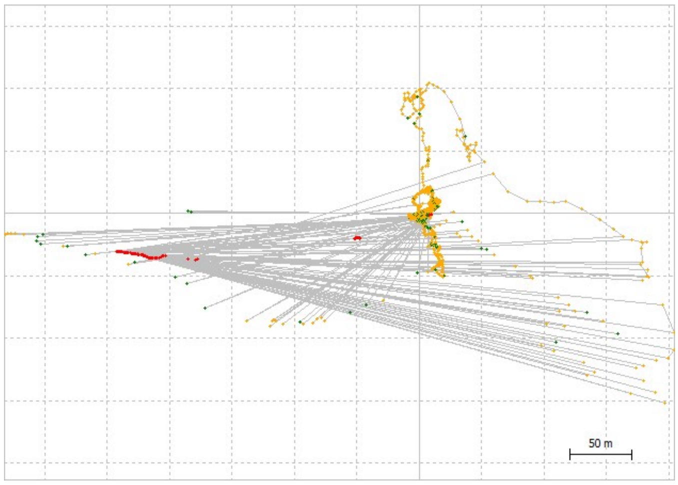


Figure 8.28: Ground Tracking Solution 5 for Samsung

8.2.9 Xiaomi 21-02-20

Receiver: Xiaomi, Master: Base Station PoliTO Nav: Base Station PoliTO

Xiaomi again presents results as good as the ones recorded for the previous day, this means that the actual smartphones are incredible devices with the capability to capture and process GNSS data for location purposes.

Figure 8.29: Positioning Solution 5 for u-blox

Maximum errors		
E-W	**N-S**	**U-D**
Q= ; d= m	Q= ; d= m	Q= ; d= m

Table 8.15: Distance errors and Qs

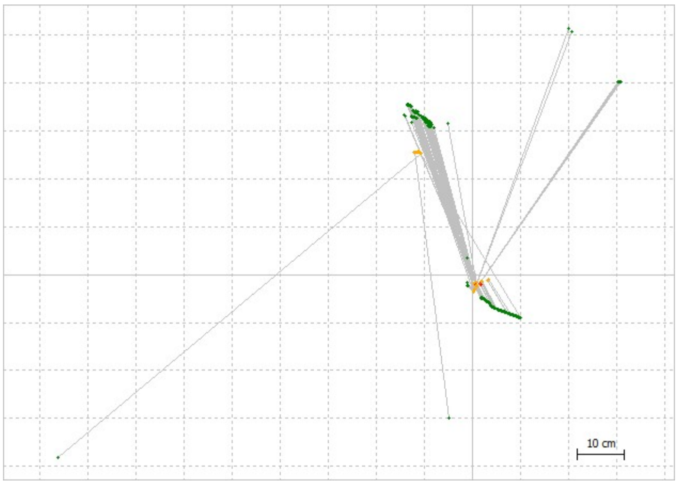


Figure 8.30: Ground Tracking Solution 5 for u-blox

8.2.10 u-blox 21-02-20

Receiver: u-blox, Master: Base Station PoliTO Nav: Base Station PoliTO

Finally, the results from u-blox which will be used for the comparison in this day. Now, the Figures shown a behaviour again worse than the presented by Xiaomi, so the comparison will be successful for it.

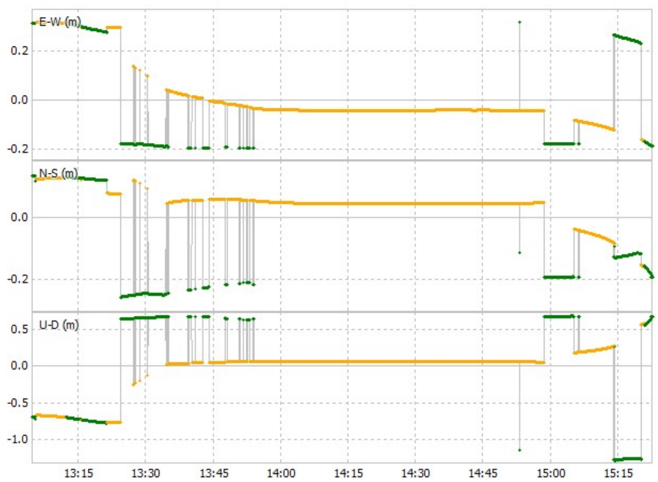


Figure 8.31: Positioning Solution 2 for u-blox

Maximum errors		
E-W	**N-S**	**U-D**
Q= ; d= m	Q= ; d= m	Q= ; d= m

Table 8.16: Distance errors and Qs

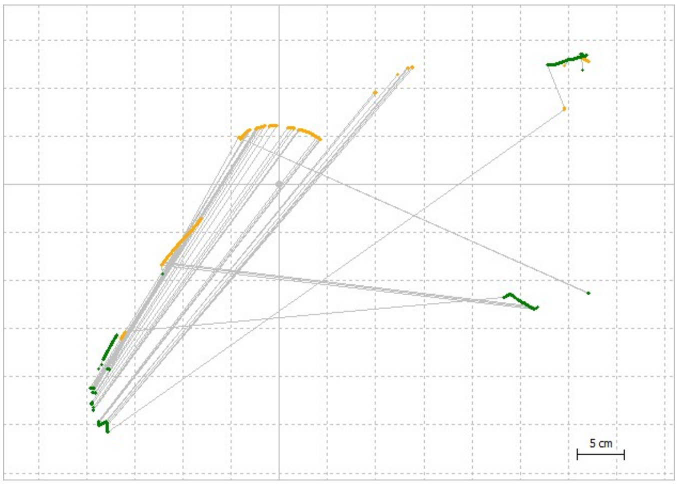


Figure 8.32: Ground Tracking Solution 3 for u-blox

8.2.11 Geodetic to UTM transformation results

Following the equations described in *section 6.3* and the Matlab code shown in the Annex, a matrix with the ENU results will be obtained and used to compare the positioning solutions epoch by epoch and with a Q classification, among different devices. It is interesting to have the comparison between u-blox device and smartphone, to have the values differences among an element that is completely dedicated to this purpose and a device available for everyone, the smartphones.

Once the transformation is done, and the data are saved into a matrix, the comparison mentioned before is performed, giving to the points different colours depending on the Q that is assigned to the epoch.

Talking about the code in the Annex, the first step is to create a function that computes all the equations required for the ENU transformation. After that, the results are called in the next script, creating graphs to see that the points obtained have a dispersion. The procedure is the following, the .xls files are stored into matrices for doing the data treatment, and once this is done, the columns that are interesting for the study are saved, these are the East, North and Height columns. Then, the plotting is performed using for the 'x' axis the East column and for the 'y' axis the North column, being able to observe the dispersion of the points obtained in the data recording.

The plotting which is interesting to see is the one that will be compared below, because they present the same epochs, these are the results that correspond to Xiaomi and u-blox for the 15^{th} of February, with this, a global image of the behaviour of the devices in an individual way will be completely analysed and seeing the comparison, the performance of the smartphone as a location device will be studied.

First, the matrix created in Matlab with the results of the transformation is shown in *Figure 8.27*, appreciating three columns, the first one include the East results, the second one the North and the last one the Height results.

5262x3 double

	1	2	3
1	3.4607e...	4.9727e...	1.5033e...
2	3.4607e...	4.9727e...	1.5033e...
3	3.4607e...	4.9727e...	1.5033e...
4	3.4607e...	4.9727e...	1.5033e...
5	3.4607e...	4.9727e...	1.5033e...
6	3.4607e...	4.9727e...	1.5033e...
7	3.4607e...	4.9727e...	1.5033e...
8	3.4607e...	4.9727e...	1.5033e...
9	3.4607e...	4.9727e...	1.5033e...
10	3.4607e...	4.9727e...	1.5033e...
11	3.4607e...	4.9727e...	1.5033e...
12	3.4607e...	4.9727e...	1.5033e...
13	3.4607e...	4.9727e...	1.5033e...
14	3.4607e...	4.9727e...	1.5033e...
15	3.4607e...	4.9727e...	1.5033e...
16	3.4607e...	4.9727e...	1.5033e...
17	3.4607e...	4.9727e...	1.5033e...
18	3.4607e...	4.9727e...	1.5033e...
19	3.4607e...	4.9727e...	1.5033e...
20	3.4607e...	4.9727e...	1.5033e...
21	3.4607e...	4.9727e...	1.5033e...
22	3.4607e...	4.9727e...	1.5033e...
23	3.4607e...	4.9727e...	1.5033e...
24	3.4607e...	4.9727e...	1.5033e...
25	3.4607e...	4.9727e...	1.5033e...
26	3.4607e...	4.9727e...	1.5033e...
27	3.4607e...	4.9727e...	1.5033e...
28	3.4607e...	4.9727e...	1.5033e...
29	3.4607e...	4.9727e...	1.5033e...
30	3.4607e...	4.9727e...	1.5033e...

Figure 8.33: ENU Data MatLab Matrix

Data from 21-02-15

Then, applying the code described, the representation is done, and the results are represented in *Figure 8.34* for the Xiaomi and *Figure 8.35* for the u-blox.

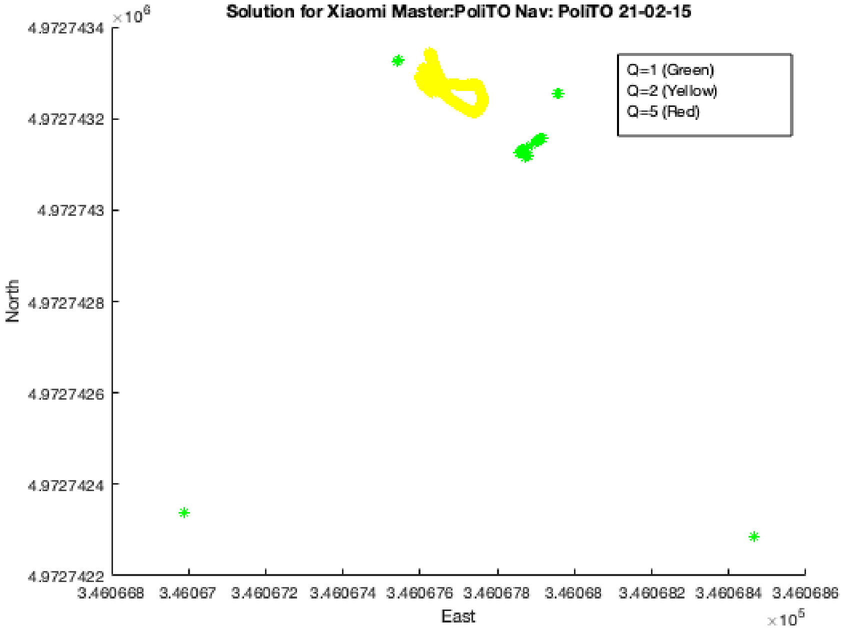


Figure 8.34: Xiaomi ENU Dispersion results

Q percentages		
Q=1	Q=2	Q=5
0,82%	99,18%	0%

Table 8.17: Q Xiaomi percentages

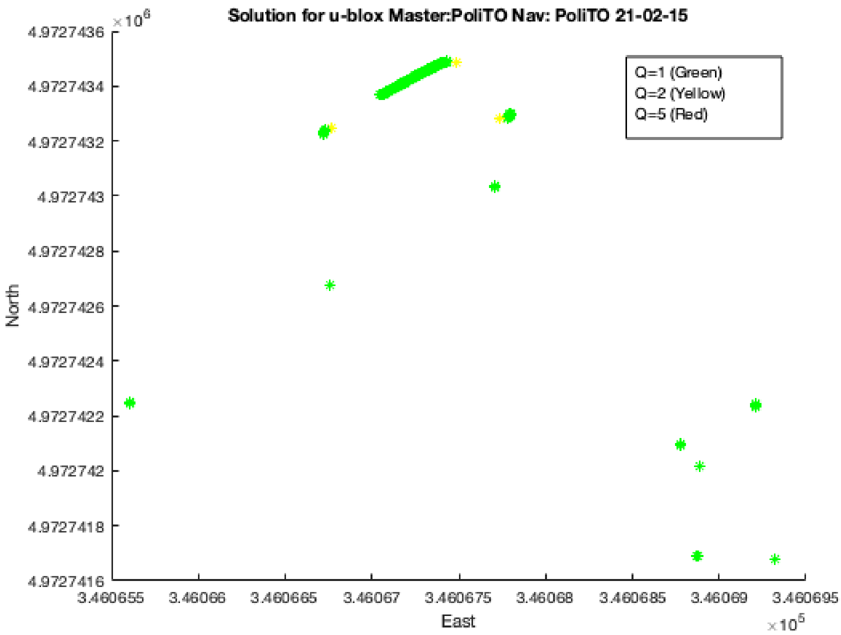


Figure 8.35: u-blox ENU Dispersion results

As can be observed, the dispersion in both devices is more or less the same, even in the smartphone the concentration of points is bigger than in the u-blox measurements, is true that

Q percentages		
Q=1	**Q=2**	**Q=5**
51,86%	48,14%	0%

Table 8.18: Q u-blox percentages

here the Qs differentiation is not performed, but in the next subsection, with the comparison, this will be shown. For this reason, it is feasible to start thinking about the possibility of a Smartphone developing the functions of a pure location device.

Furthermore, the MatLab code includes a computation of the percentage of Qs that present the dataset, and the results are shown in *Table 8.17*.

As a result, can be observed that the majority of the Q are equal to two, which means that the computations mostly are done by floating point, which gives more accuracy to the results.

The other smartphone analysed is the Samsung mentioned in the thesis, so it is interesting first to see the dispersion of the points once the transformation is done. The u-blox file that will be used for the comparison is the same as the used for Xiaomi *Figure 8.35*, due to the date is the same.

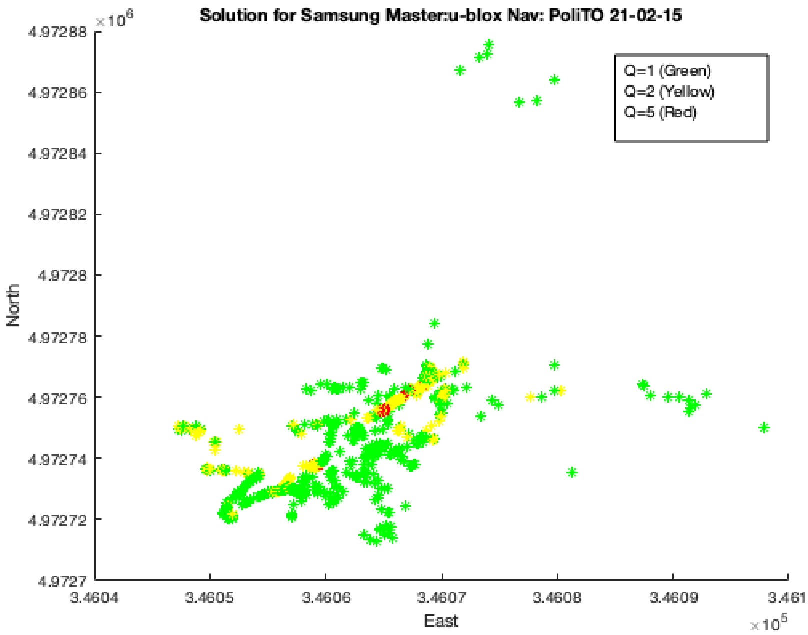


Figure 8.36: Samsung ENU Dispersion results

Q percentages		
Q=1	**Q=2**	**Q=5**
32,56%	47,55%	19,89%

Table 8.19: Q Samsung percentages

These results are not as good as the Xiaomi ones, due to the percentage (*Table 8.19*) of Qs equal to 1 and 5 (fixed and single position respectively), this confirms that the GNSS measurements are getting more importance in the newest smartphones, in which the data obtained for this purpose have bigger accuracy. The dispersion (*Figure 8.36*) shows that the points are not assembled at all.

This effect, will cause differences in the comparison with the u-blox, but is interesting to see that, even with a smartphone that has not got a good location performance, the results obtained could be used for positioning purposes that requires less precision.

Data from 21-02-20

Applying the same procedure over the data obtained for Samsung, Xiaomi and u-blox the 20th of February. the dispersion is shown bellow.

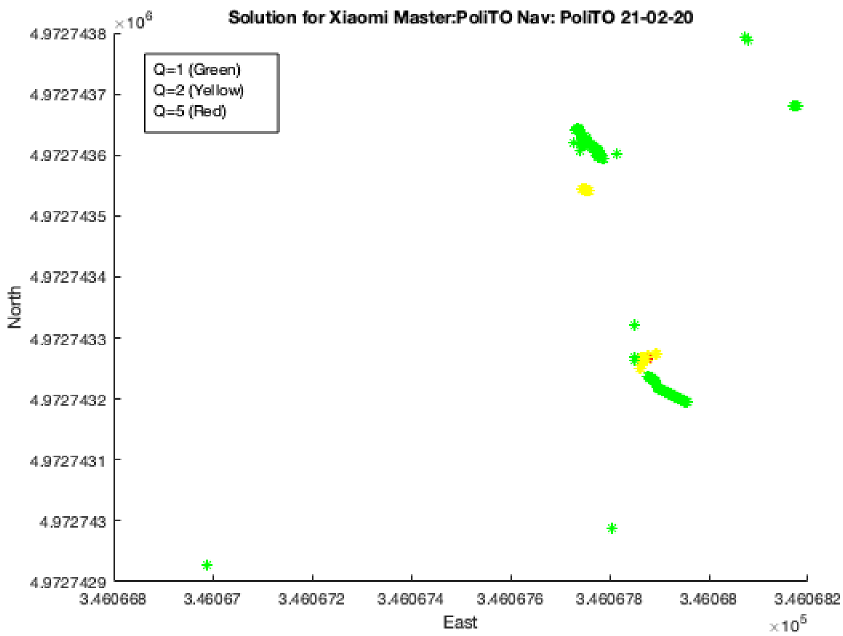


Figure 8.37: Xiaomi ENU Dispersion results

Q percentages		
Q=1	**Q=2**	**Q=5**
8,41%	88,61%	2,98%

Table 8.20: Q Xiaomi percentages

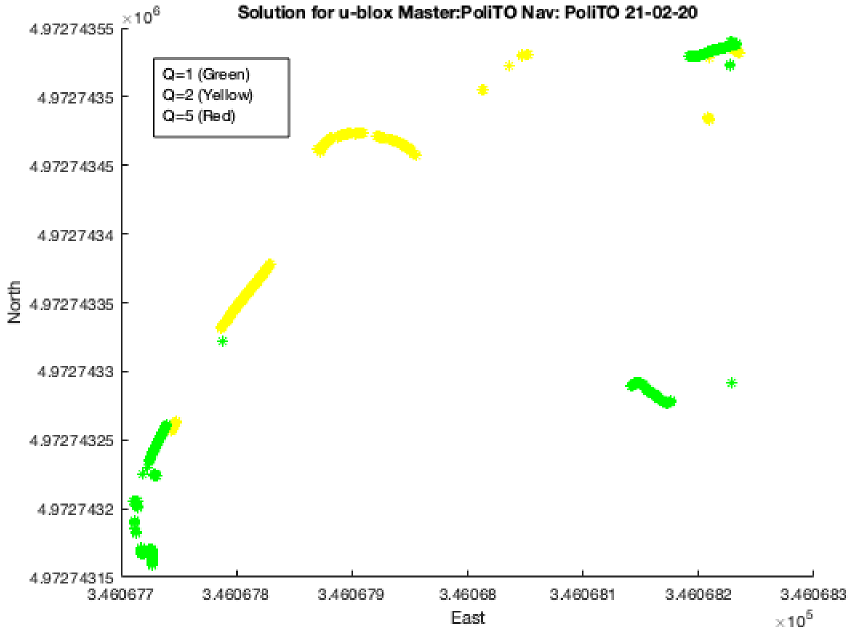


Figure 8.38: u-blox ENU Dispersion results

Q percentages		
Q=1	**Q=2**	**Q=5**
28,09%	71,91%	0%

Table 8.21: Q u-blox percentages

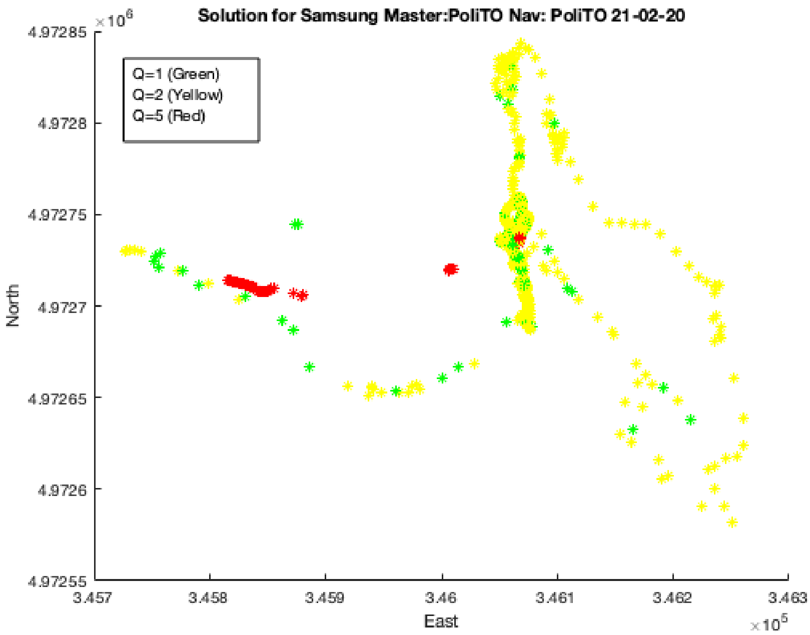


Figure 8.39: Samsung ENU Dispersion results

Observing the dispersion of the data and the Q percentages, the conclusion is approximately the same as in the previous date analysis, (*Table 8.20*) and (*Table 8.21*) show the results from

Q percentages		
Q=1	**Q=2**	**Q=5**
15,96%	68,24%	15,80%

Table 8.22: Q Samsung percentages

Xiaomi and u-blox, being these with a high quality, presenting the Xiaomi data more Q = 2 (floating point) than the u-blox, which means that with Xiaomi a good positioning accuracy will be obtained.

On the other hand, Samsung present a percentage of Q = 5 (single position), which indicates that a worse performance of the positioning computation will be performed, however, the Q = 2 percentage is good, being able to confirm that the comparison with the u-blox device will not be bad at all.

8.2.12 Geodetic to UTM comparison results

Data from 21-02-15

Furthermore, as we talked previously, the comparison epoch by epoch between u-blox and the smartphones is also programmed. This code involves a call to the transformation equations, but previously, the file selection has to be done, in which the data with same date and epoch are compared. Once, the data are transformed, and the matrices for the study are created, the plotting is done selecting the positioning data with the same epoch and also same Q, giving a different colour to the Qs, as mentioned before, green for Q = 1 and yellow for Q = 2, knowing then when the devices are acting in fixed point positioning (green) and in float point positioning (yellow).

Being the files selected for the comparison the ones corresponding to Xiaomi and u-blox data for the 15th of February, being the representation performed in *Figure 8.30* and *Figure 8.31*.

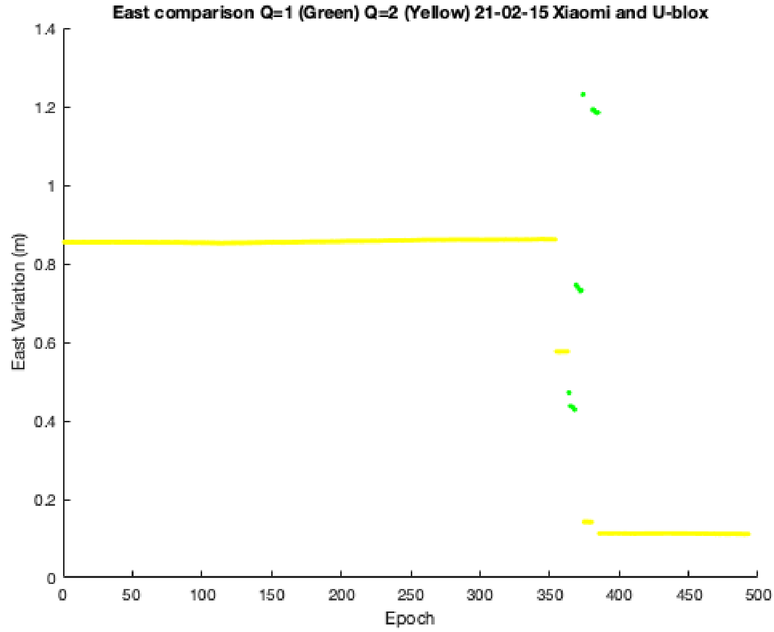


Figure 8.40: East Comparison Xiaomi and u-blox

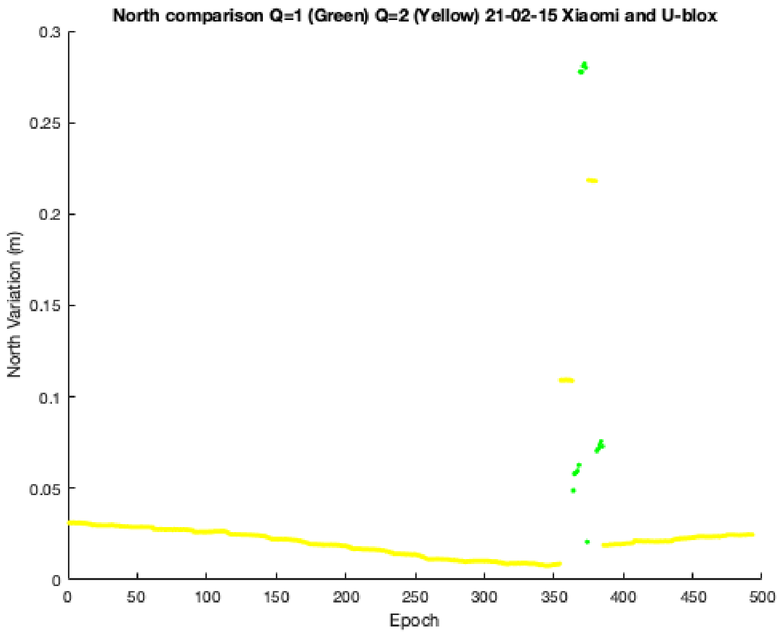


Figure 8.41: North Comparison Xiaomi and u-blox

As can be appreciated in the first graph, the maximum variation among the devices is around 1.2 meters, appearing this values when the Q is equal to 1, fixed point solution, which offers a solution with "less" quality with respect to Q equal to 2. Practically, all the data is obtained by float point positioning, this is a good aspect, because normally a floating point maths offer a bigger range of numbers, increasing the accuracy of the solution, so in this case, is fascinating to have the major part of the data treatment processed with floating point treatment.

The second graph, shows a really similar behaviour with respect to the first one, even in this case, the variation is also lower than in the first case, being this value near the 0.3 meters when Q = 1, again the Qs have the same results, being better for Q = 2. If we observe the best results, these are lower than 0.05 meters, which indicates that the behaviour of both devices is really similar.

As a conclusion of this comparison, can be confirmed that the Xiaomi device and u-blox have a similar behaviour, this is really interesting because the Smartphone could achieve a location data treatment practically equal to the one obtained with a device that is completely dedicated to this task.

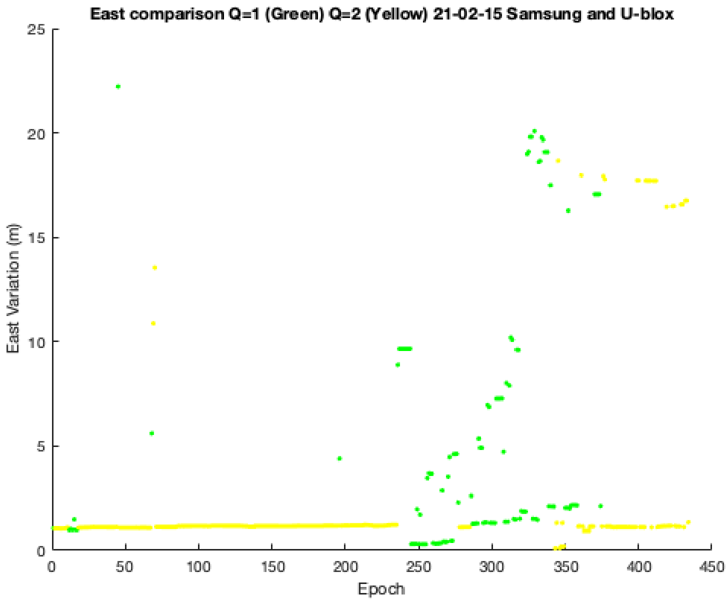


Figure 8.42: East Comparison Samsung and u-blox

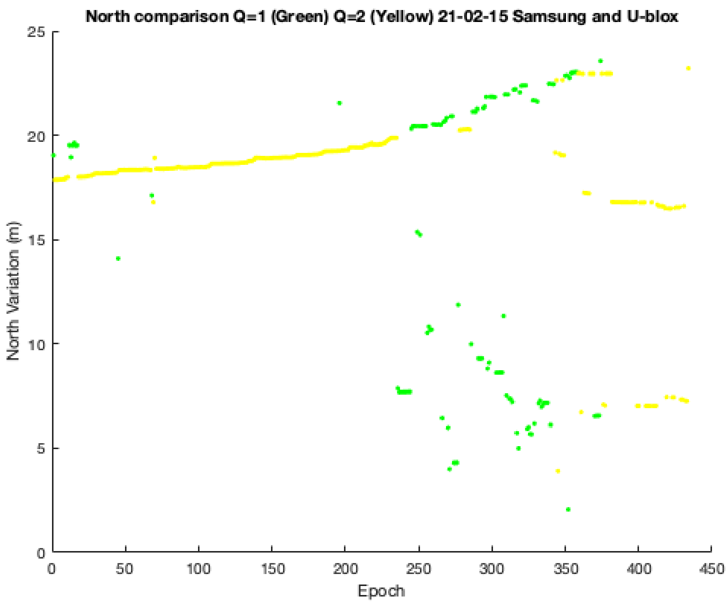


Figure 8.43: North Comparison Samsung and u-blox

As predicted before, the comparison of Samsung and u-blox was not as good as the Xiaomi one, but the results are still interesting. East comparison present really good computation when Q reach the value 2 being the differences with u-blox among the 1,5 meters, however, and when the Q is equal to 1, the differences increases into values that reach the 20 meters. This may be due to the hardware of the smartphone and that is a model older than Xiaomi.

Even the results represented in *Figure 8.43* are worse than the ones shown in *Figure 8.42*, but in this case, the biggest difference is in approximately 25 meters, which could be considered for the app purposes, not with the same precision but could be useful.

Data from 21-02-20

The second dataset, present even better results for Xiaomi thant the one presented before. Here, the differences are around 0.05 meters (for East comparison), which indicates that the behaviour of Xiaomi and u-blox in positioning terms are very similar. Also, the percentage of Q = 2 is higher that the Q = 1 which ensures, as shown in the Figures below, more accurate results.

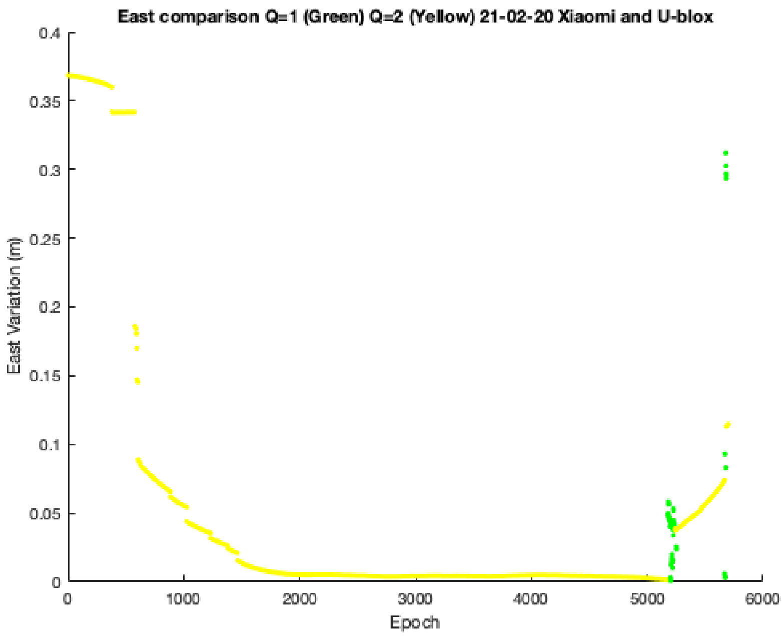


Figure 8.44: East Comparison Xiaomi and u-blox

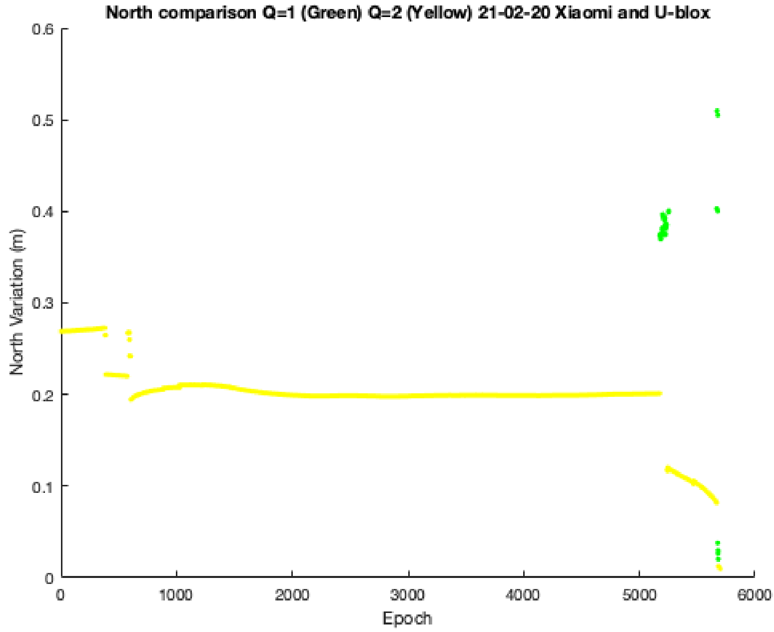


Figure 8.45: North Comparison Samsung and u-blox

Also, in the North comparison, the results obtained ensure that the Xiaomi and u-blox performances are practically equal. With this, the affirmation that the behaviour of a Smartphone as a pure location device is nearest.

Again, Samsung dataset present deviations bigger than the ones presented in Xiaomi. In this case, for some epochs, the difference reach values of 350 meters *Figure 8.46*, this values are only for a few epochs and it is not a problem to be considered a to discard Samsung as a location device.

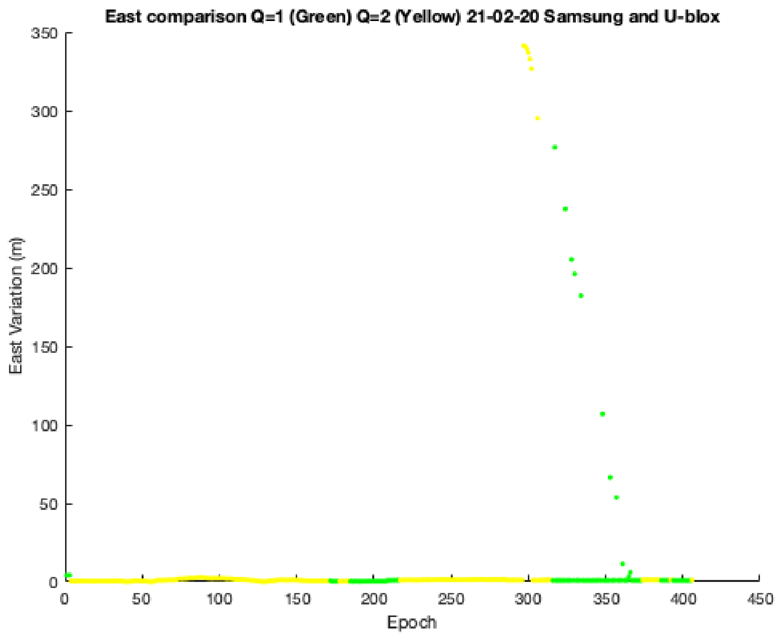


Figure 8.46: East Comparison Samsung and u-blox

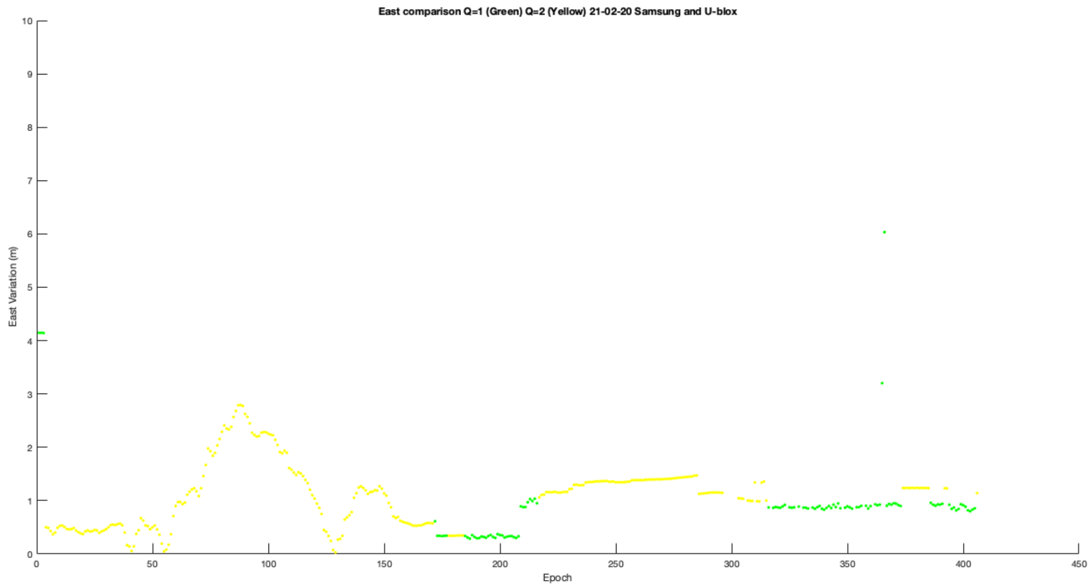


Figure 8.47: East Comparison Samsung and u-blox Zoom

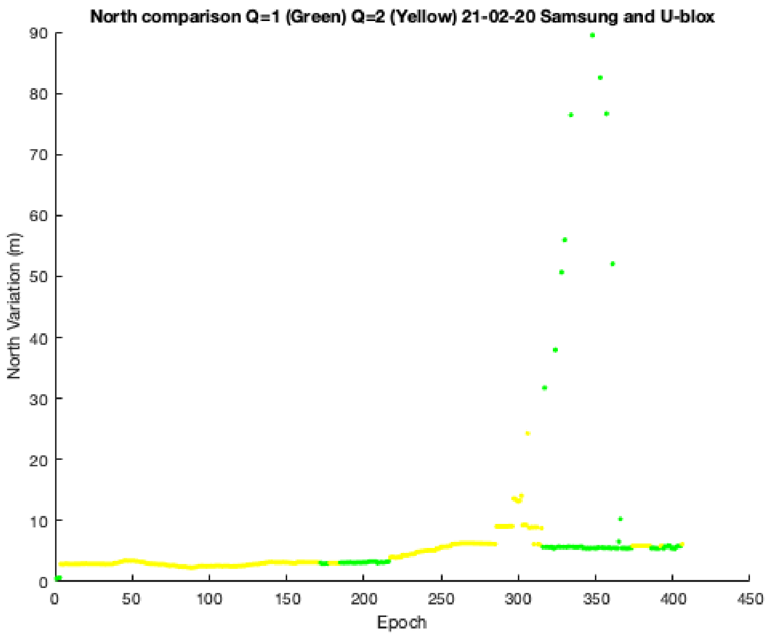


Figure 8.48: North Comparison Samsung and u-blox

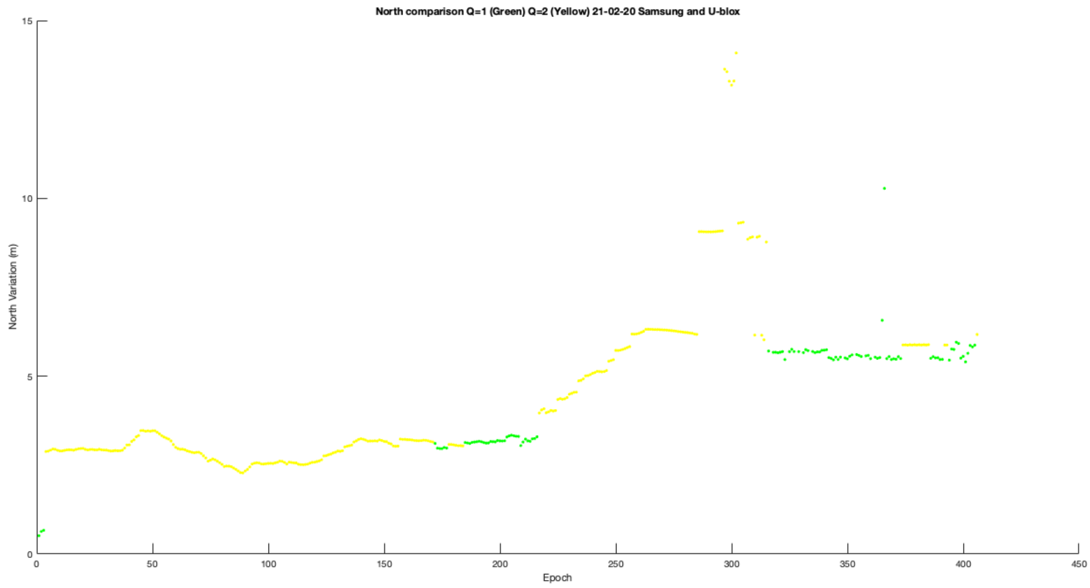


Figure 8.49: North Comparison Samsung and u-blox Zoom

This second comparison of Samsung and u-blox *Figure 8.46*), shows the same performance that the first one, only for a few epochs the values increase to 90 meters of difference, but this is something punctual.

Finally, with the whole comparisons performed, for Xiaomi and Samsung, in two different days, the possibility of the usage of the smartphones as a positioning devices could be confirmed. With this, an app could use these functionalities for the purposes required.

Chapter 9

Conclusions

Basically, the main goal of the thesis is to study the capability of the smartphones to reach GNSS positioning levels similar to pure location devices. If this is proved, then could be demonstrated that the applications installed in the smartphones could take advantage of this feature, providing to the user, a good accuracy and ensuring the correct behaviour of the app, avoiding the errors that could cause a bad positioning of the phone.

It is interesting to exploit this characteristic of the smartphones, because sometimes and as mentioned in the thesis, some apps use location to perform payment processes, so it is important to use the location correctly.

Regarding all the steps followed in the development of the thesis, it has been performed, initially some attempts for the data collection, getting results from different devices, with these data, a classification of them was made, discarding the datasets that not achieve the minimum required features. This bad quality in the data could be originated by the hardware of the smartphone, the capability to accept the version of the app Geo++ or different conditions. After this analysis, two smartphones were selected, Samsung and Xiaomi, which achieve the expected results in the dataset.

The software used for the data processing was RTKLib, which allows developing a post-processing study to check the good quality of the data. This program is analyzed also in the document, to be able to understand how it works.

Once the data are selected and the software is understood, the analysis was carried out, obtaining the .pos files, which are more or less like a text document in which the whole location information of the device is registered. This kind of file present a latitude, longitude and height format, and for that reason, it is required a transformation into ENU (East, North and Up) format, to be able to compare and to have a better comprehension of the data.

Then, the main purpose of the thesis can be performed, and this is, as mentioned in the whole thesis, to be able to confirm that an smartphone can carry out the functions of a location device (as the u-blox) applying this capability into the apps that requires this kind of services. It was observed that the individual behavior of each device involved in the study (Samsung, Xiaomi and u-blox) present good features and a good quality of positioning data, after of the post-processing data treatment. With this, it is not very difficult to have a first approximation and to confirm that they are able to carry out location performances. It is true, that Samsung presents results that are not as good as Xiaomi ones, but even with this kind of data, a good positioning performance is done by the device.

After proving and seeing the behavior of the devices individually, it was interesting to see a comparison between a pure location device, u-blox. This study were carried out comparing the devices performances in the same epoch, obtaining the differences among them, so if the results would have present big differences during the whole comparison, the final conclusion of the thesis would have been oriented into other way. The case was that the values obtained in the study were interesting, putting the focus over the Xiaomi, that is the one that presents the best results, the performance of it as a location device is perfect, presenting insignificant differences with the u-blox for the same day and time. As described during the thesis, exist some devices, as u-blox, that specifically are created with location purposes and they could be assembled in cars to achieve better location performance, but with a devices with the Xiaomi features, this kind of devices will not be necessary.

Knowing this, the cost of introduce a positioning element in the car, will be eliminated, being able to achieve, as mentioned in the Objectives, the possibility of paying into gas stations without getting out of the car, because the positioning problems will not appear. This is only one of the multiple applications that could be performed with a good location behavior.

The Samsung case could cause confusion, but the performance presented is completely normal, this is due to the hardware of the smartphone and because it is older than Xiaomi. However, the results are good enough to ensure a good location performance of this device. Probably it will not have the same accuracy as Xiaomi, but it will provide to the users an admissible precision in the apps when location.

Summarizing, this study proves that there are devices that are not able to carry out this kind of study, due to different facts (GPS available, compatibility with Geo++ app...), also ensure, that the most recent smartphones are able to perform good positioning functions, because the comparison with a pure location device demonstrate that, the differences are minimal.

Chapter 10

Observations

It would be interesting to indicate, that at the beginning of the thesis study, many smartphones ware tested to see if they could provide the raw data, in order to extract them for post-processing, however, a lot of them could not afford this. By this fact, we could obtain wrong conclusion and to think that not all the device are able to have a good behavior in location terms, which obviously is true, but the impossibility to obtain the raw data for the study from an smartphone, do not ensure that the device could not act for positioning purposes.

Furthermore, would be interesting to talk about the direct application into the Repsol app, Waylet. Waylet is an app that allow the users to pay with the own application, and one of the most innovative developments inside the app is that this allows the user to pay into the gas station without getting out of the car. As explained in the objectives, sometimes an error in the smartphone positioning could cause a wrong behavior in the payment, confusing the app and creating the possibility of a payment in a parallel station. This is avoid by the smartphones that present the behavior of Xiaomi and Samsung, obtained the expected behavior of the app. Obviously, this is also comparably to other applications that require the same location performances.

Bibliography

[1] -. *BackMarket*. 2021. URL: https://www.backmarket.es/samsung-galaxy-a40-64-gb-negro-libre-segunda-mano/276027.html?shopping=gmc (visited on 02/09/2021).

[2] -. *bludiode*. 2021. URL: https://bludiode.com/es/oneplus/one-plus-7-pro-gm1910-8256gb-blue-8426/ (visited on 02/09/2021).

[3] -. *EUREF Permanent GNSS Network*. 2021. URL: http://www.epncb.oma.be/_networkdata/siteinfo4onestation.php?station=TORI00ITA (visited on 06/07/2021).

[4] -. *GNSS Signal*. 2021. URL: https://gssc.esa.int/navipedia/index.php/GNSS_signal (visited on 01/09/2021).

[5] -. *Tienda sigloXXI*. 2021. URL: https://tiendasigloxxi.es/es/moviles-libres/63175-huawei-p8-lite-2017-oro-single-sim-6901443158041.html (visited on 02/09/2021).

[6] -. *Xiaomi*. 2021. URL: https://www.mi.com/es/mi8/specs/ (visited on 02/09/2021).

[7] Rajat Acharya. *Carrier Phase*. 2014. URL: https://www.sciencedirect.com/topics/engineering/carrier-phase (visited on 06/02/2021).

[8] LLC ARINC Engineering Services. "Navstar GPS Space Segment/Navigation User Interfaces". In: *NAVSTAR GLOBAL POSITIONING SYSTEM* -.1 (2004), p. 207.

[9] BLACKROC. *Differential GPS*. 2020. URL: http://www.blackroc.com/differential-gps/?doing_wp_cron=1609234731.7664461135864257812500 (visited on 12/29/2020).

[10] Ignacio Alonso Fernández-Coppel. *Localizaciones Geográficas, EI DATUM*. 2001. URL: https://www.aristasur.com/sites/as/users/3/arch/datum-cartografia.pdf (visited on 03/27/2021).

[11] Research Gate. *GNSS three-segment architecture*. 2020. URL: https://www.researchgate.net/figure/GNSS-three-segment-architecture_fig2_336781393 (visited on 12/29/2020).

[12] Dirección General de Geografía y Medio Ambiente. *Geodetic coordinates to UTM*. 2014. URL: https://slideplayer.es/slide/3121137/ (visited on 03/20/2021).

[13] Werner Gurtner. *International GNSS Service (IGS), RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104)*. 2018. URL: http://acc.igs.org/misc/rinex304.pdf (visited on 06/07/2021).

[14] NovAtel Inc. "An introduction to GNSS". In: *NovAtel* 2.1 (2020), p. 125.

[15] IndiaMart. *U-Blox device*. 2020. URL: https://www.indiamart.com/proddetail/ublox-neo-6m-gps-module-19990293130.html (visited on 01/10/2021).

[16] J.M. Juan Zornoza J. Sanz Subirana and M. Hernández-Pajares. "GNSS DATA PROCESSING". In: *Eropean Space Agency* 1.1 (2013), p. 238.

[17] A. Celms J. Zvirgzds. "GNSS RTK PERFORMANCE IMPROVEMENTS USING GALILEO SATELLITE SIGNAL". In: *LATVIAN JOURNAL OF PHYSICS AND TECHNICAL SCIENCES* 1.1-2 (2020), p. 7.

[18] A. Ajmar; E. Arco; P. Boccardo; F. Perez. "Floating car data (fcd) for mobility applications". In: *International Society for Photogrammetry and Remote Sensing* 1.1 (2020), p. 8.

[19] Open Source. *RTKLIB: An Open Source Program Package for GNSS Positioning*. 2020. URL: `http://www.rtklib.com` (visited on 06/11/2020).

[20] T. Takasu. "RTKLIB ver. 2.4.1 Manual". In: *http://www.rtklib.com* -.1-2 (2020), p. 88.

[21] TALLYSMAN. *GNSS Positioning Techniques*. 2021. URL: `https://www.tallysman.com/gnss-positioning-techniques/` (visited on 06/07/2021).

[22] Guochang Xu. "GPS. Theory, Algorithms and Applications". In: *Springer* -.2 (2007), p. 354.

Appendix A

Annex

A.1 Programming Code

A.1.1 Transformation into ENU

```
function [East,North,Alt,Zone]=lla2enu(lat,lon,alt)
% Transformation from Lat Long Alt to ENU

latr=deg2rad(lat);
lonr=deg2rad(lon);

%Definition of the elipsoid data (WGS84).
%First, the semi-major and semi-minor axis

a=6378137; %semi-major
b=6356752.314245; %semi-minor

%Its also important to know that the hemisphere is the North and the
%Direction is East
%The excentricity is:

exc=0.081819191;
exc1=0.082094438;
exc1_2=exc1^2;

%The curvature polar radius

c=6399593.626;

%The computation of the zone

zn=round((lon/6)+31);

%the meridian zone
```

```
znm=6*zn-183;

%delta lambda

dlam=lonr-((znm*pi)/180);

%Being Xi the vertical desviation and Eta the meridian desviation

A=cos(latr)*sin(dlam);
Xi=(1/2)*log((1+A)/(1-A));
Eta=atan((tan(latr))/cos(dlam))-latr;

%Being the scale factor at the central meridian 0.9996

Ni=(c/(1+exc1_2*(cos(latr))^2)^(1/2))*0.9996;
Zeta=(exc1_2/2)*Xi^2*(cos(latr))^2;

A1=sin(2*latr);
A2=A1*(cos(latr))^2;
J2=latr+(A1/2);
J4=((3*J2)+A2)/4;
J6=(5*J4+A2*(cos(latr))^2)/3;
alpha=(3/4)*exc1_2;
beta=(5/3)*alpha^2;
gamma=(35/27)*alpha^3;
B_phi=0.9996*c*(latr-(alpha*J2)+(beta*J4)-(gamma*J6));

%And finally the East North coordinates

East= Xi*Ni*(1+Zeta/3)+500000;
North=Eta*Ni*(1+Zeta)+B_phi;
Alt=alt;
Zone=zn;
return
```

A.1.2 Analysis of the data and dispersion representation

```
%PEDRO RIOS BRAVO
%% 21-01-09
%Solution for Xiaomi Master:PoliTO Nav:PoliTO

clear East North Alt Zone Q1 Q2 Q5
sol1=xlsread('21-01-09-Xiaomi.xls');
[m,n]=size(sol1);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol1(i,3),sol1(i,4),sol1(i,5));

end
enu1=[East,North,Alt];

%Comparison of the solution by epoch

figure
plot(East,North,'*')
title('Solution for Xiaomi Master:PoliTO Nav:PoliTO 21-01-09')
xlabel('East')
ylabel('North')


%Solution for Xiaomi+ublox Master:U-blox Nav: U-blox

clear East North Alt Zone
sol2=xlsread('21-01-09-XiaomiUblox1.xls');
[m,n]=size(sol2);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol2(i,3),sol2(i,4),sol2(i,5));

end
enu2=[East,North,Alt];


figure
plot(East,North,'*')
title('Solution for Xiaomi+ublox Master:U-blox Nav: U-blox 21-01-09')
xlabel('East')
ylabel('North')

%Solution for Xiaomi+ublox Master:U-blox Nav: PoliTO

clear East North Alt Zone
sol3=xlsread('21-01-09-XiaomiUblox3.xls');
[m,n]=size(sol3);

for i=1:m
```

```
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol3(i,3),sol3(i,4),sol3(i,5));

end
enu3=[East,North,Alt];



figure
plot(East,North,'*')
title('Solution for Xiaomi+ublox Master:U-blox Nav: PoliTO 21-01-09')
xlabel('East')
ylabel('North')

%Solution for Ublox Master:PoliTO Nav: Ublox

clear East North Alt Zone
sol4=xlsread('21-01-09-Ublox.xls');
[m,n]=size(sol4);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol4(i,3),sol4(i,4),sol4(i,5));

end
enu4=[East,North,Alt];


figure
plot(East,North,'*')
title('Solution for Ublox Master:PoliTO Nav: Ublox 21-01-09')
xlabel('East')
ylabel('North')

%% 21-02-14

%Solution for Samsung Master:PoliTO Nav: PoliTO

clear East North Alt Zone
sol5=xlsread('21-02-14-Samsung.xls');
[m,n]=size(sol5);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol5(i,3),sol5(i,4),sol5(i,5));

end
enu5=[East,North,Alt];


figure
plot(East,North,'*')
title('Solution for Samsung Master:PoliTO Nav: PoliTO 21-02-14')
xlabel('East')
```

```
ylabel('North')

%% 21-02-15

%Solution for Samsung Master:PoliTO Nav: PoliTO
clear East North Alt Zone
sol6=xlsread('21-02-15-Samsung.xls');
[m,n]=size(sol6);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol6(i,3),sol6(i,4),sol6(i,5));

end
enu6=[East,North,Alt];

figure
hold on
for i=1:m
if sol6(i,6)==1
plot(East(i),North(i),'*g')
end
if sol6(i,6)==2
plot(East(i),North(i),'*y')
end
if sol6(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Samsung Master:PoliTO Nav: PoliTO 21-02-15')
xlabel('East')
ylabel('North')

%Computation of the Q percentages for Samsung

syms Q1 Q2 Q5
Q1=0;
Q2=0;
Q5=0;

for i=1:m
if sol6(i,6)==1
Q1=Q1+1;
end
if sol6(i,6)==2
Q2=Q2+1;
end
if sol6(i,6)==5
Q5=Q5+1;
end
end
```

```
Q1Samp=(Q1/m)*100
Q2Samp=(Q2/m)*100
Q5Samp=(Q5/m)*100

%Solution for Xiaomi Master:PoliTO Nav: PoliTO

clear East North Alt Zone
sol7=xlsread('21-02-15-Xiaomi.xls');
[m,n]=size(sol7);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol7(i,3),sol7(i,4),sol7(i,5));

end
enu7=[East,North,Alt];

figure
hold on
for i=1:m
if sol7(i,6)==1
plot(East(i),North(i),'*g')
end
if sol7(i,6)==2
plot(East(i),North(i),'*y')
end
if sol7(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Xiaomi Master:PoliTO Nav: PoliTO 21-02-15')
xlabel('East')
ylabel('North')

%Computation of the Q percentages for Xiaomi

syms Q1 Q2 Q5
Q1=0;
Q2=0;
Q5=0;

for i=1:m
if sol7(i,6)==1
Q1=Q1+1;
end
if sol7(i,6)==2
Q2=Q2+1;
end
if sol7(i,6)==5
Q5=Q5+1;
end
end
```

```
Q1Xiap=(Q1/m)*100
Q2Xiap=(Q2/m)*100
Q5Xiap=(Q5/m)*100

%Solution for Ublox Master:PoliTO Nav: PoliTO

clear East North Alt Zone
sol8=xlsread('21-02-15-Ublox.xls');
[m,n]=size(sol8);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol8(i,3),sol8(i,4),sol8(i,5));

end
enu8=[East,North,Alt];

%Computation of the Q percentages for Xiaomi

syms Q1 Q2 Q5
Q1=0;
Q2=0;
Q5=0;

for i=1:m
if sol8(i,6)==1
Q1=Q1+1;
end
if sol8(i,6)==2
Q2=Q2+1;
end
if sol8(i,6)==5
Q5=Q5+1;
end
end

Q1Ubxp=(Q1/m)*100
Q2Ubxp=(Q2/m)*100
Q5Ubxp=(Q5/m)*100


figure
hold on
for i=1:m
if sol8(i,6)==1
plot(East(i),North(i),'*g')
end
if sol8(i,6)==2
plot(East(i),North(i),'*y')
end
if sol8(i,6)==5
```

```
plot(East(i),North(i),'*r')
end
end

title('Solution for u-blox Master:PoliTO Nav: PoliTO 21-02-15')
xlabel('East')
ylabel('North')


%Solution for Samsung Master:PoliTO Nav: Ublox

clear East North Alt Zone
sol9=xlsread('21-02-15-SXU1.xls');
[m,n]=size(sol9);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol9(i,3),sol9(i,4),sol9(i,5));

end
enu9=[East,North,Alt];


figure
hold on
for i=1:m
if sol9(i,6)==1
plot(East(i),North(i),'*g')
end
if sol9(i,6)==2
plot(East(i),North(i),'*y')
end
if sol9(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Samsung Master:PoliTO Nav: u-blox 21-02-15')
xlabel('East')
ylabel('North')


%Solution for Samsung Master:Ublox Nav: PoliTO

clear East North Alt Zone
sol10=xlsread('21-02-15-SXU2.xls');
[m,n]=size(sol10);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol10(i,3),sol10(i,4),sol10(i,5));

end
enu10=[East,North,Alt];
```

```
figure
hold on
for i=1:m
if sol10(i,6)==1
plot(East(i),North(i),'*g')
end
if sol10(i,6)==2
plot(East(i),North(i),'*y')
end
if sol10(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Samsung Master:u-blox Nav: PoliTO 21-02-15')
xlabel('East')
ylabel('North')

%Solution for Xiaomi Master:PoliTO Nav: Ublox

clear East North Alt Zone
sol11=xlsread('21-02-15-SXU3.xls');
[m,n]=size(sol11);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol11(i,3),sol11(i,4),sol11(i,5));

end
enu11=[East,North,Alt];


figure
hold on
for i=1:m
if sol11(i,6)==1
plot(East(i),North(i),'*g')
end
if sol11(i,6)==2
plot(East(i),North(i),'*y')
end
if sol11(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Xiaomi Master:PoliTO Nav: u-blox 21-02-15')
xlabel('East')
ylabel('North')


%Solution for Xiaomi Master:Ublox Nav: PoliTO
```

```
clear East North Alt Zone
sol12=xlsread('21-02-15-SXU4.xls');
[m,n]=size(sol12);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol12(i,3),sol12(i,4),sol12(i,5));
end
enu12=[East,North,Alt];


figure
hold on
for i=1:m
if sol12(i,6)==1
plot(East(i),North(i),'*g')
end
if sol12(i,6)==2
plot(East(i),North(i),'*y')
end
if sol12(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Xiaomi Master:u-blox Nav: PoliTO 21-02-15')
xlabel('East')
ylabel('North')


%% 21-02-20

%Solution for Samsung Master:PoliTO Nav: PoliTO

clear East North Alt Zone
sol13=xlsread('21-02-20-Samsung.xls');
[m,n]=size(sol13);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol13(i,3),sol13(i,4),sol13(i,5));
end
enu13=[East,North,Alt];

%Computation of the Q percentages for Samsung

syms Q1 Q2 Q5
Q1=0;
Q2=0;
Q5=0;

for i=1:m
if sol13(i,6)==1
```

```
Q1=Q1+1;
end
if sol13(i,6)==2
Q2=Q2+1;
end
if sol13(i,6)==5
Q5=Q5+1;
end
end

Q1Samp=(Q1/m)*100
Q2Samp=(Q2/m)*100
Q5Samp=(Q5/m)*100


figure
hold on
for i=1:m
if sol13(i,6)==1
plot(East(i),North(i),'*g')
end
if sol13(i,6)==2
plot(East(i),North(i),'*y')
end
if sol13(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Samsung Master:PoliTO Nav: PoliTO 21-02-20')
xlabel('East')
ylabel('North')

%Solution for Xiaomi Master:PoliTO Nav: PoliTO

clear East North Alt Zone
sol14=xlsread('21-02-20-Xiaomi.xls');
[m,n]=size(sol14);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol14(i,3),sol14(i,4),sol14(i,5));
end
enu14=[East,North,Alt];

%Computation of the Q percentages for Xiaomi

syms Q1 Q2 Q5
Q1=0;
Q2=0;
Q5=0;

for i=1:m
```

```
if sol14(i,6)==1
Q1=Q1+1;
end
if sol14(i,6)==2
Q2=Q2+1;
end
if sol14(i,6)==5
Q5=Q5+1;
end
end

Q1Xiap=(Q1/m)*100
Q2Xiap=(Q2/m)*100
Q5Xiap=(Q5/m)*100

figure
hold on
for i=1:m
if sol14(i,6)==1
plot(East(i),North(i),'*g')
end
if sol14(i,6)==2
plot(East(i),North(i),'*y')
end
if sol14(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for Xiaomi Master:PoliTO Nav: PoliTO 21-02-20')
xlabel('East')
ylabel('North')

%Solution for Xiaomi Master:PoliTO Nav: PoliTO

clear East North Alt Zone
sol15=xlsread('21-02-20-Ublox.xls');
[m,n]=size(sol15);

for i=1:m
[East(i,1),North(i,1),Alt(i,1),Zone]=lla2enu(sol15(i,3),sol15(i,4),sol15(i,5));
end
enu15=[East,North,Alt];


%Computation of the Q percentages for Ublox

syms Q1 Q2 Q5
Q1=0;
Q2=0;
Q5=0;
```

```
for i=1:m
if sol15(i,6)==1
Q1=Q1+1;
end
if sol15(i,6)==2
Q2=Q2+1;
end
if sol15(i,6)==5
Q5=Q5+1;
end
end

Q1Ubxp=(Q1/m)*100
Q2Ubxp=(Q2/m)*100
Q5Ubxp=(Q5/m)*100


figure
hold on
for i=1:m
if sol15(i,6)==1
plot(East(i),North(i),'*g')
end
if sol15(i,6)==2
plot(East(i),North(i),'*y')
end
if sol15(i,6)==5
plot(East(i),North(i),'*r')
end
end
title('Solution for u-blox Master:PoliTO Nav: PoliTO 21-02-20')
xlabel('East')
ylabel('North')
```

A.1.3 Comparison among the devices

```
%PEDRO RIOS BRAVO
% COMPARISON BY EPOCH AND Q, PEDRO RIOS BRAVO

%% Xiaomi and u-blox 21-02-15

clear
clc

%reading the solutions that we are going to compare

sol1=xlsread('21-02-15-Xiaomi.xls');
[m,n]=size(sol1);
sol2=xlsread('21-02-15-Ublox.xls');
[h,j]=size(sol2);

%Then the solution is transformed into ENU
%Xiaomi

for i=1:m
[East1(i,1),North1(i,1),Alt1(i,1),Zone]=lla2enu(sol1(i,3),sol1(i,4),sol1(i,5));
end
enu1=[sol1(:,2),East1,North1,Alt1,sol1(:,6)];

%Ublox

for i=1:h
[East2(i,1),North2(i,1),Alt2(i,1),Zone]=lla2enu(sol2(i,3),sol2(i,4),sol2(i,5));
end
enu2=[sol2(:,2),East2,North2,Alt2,sol2(:,6)];


%the the comparison have to be made between the same epochs and also the
%same Q, so we are going to check the values of each table.

k=1;
for i=1:m
for j=1:h
if enu1(i,1)==enu2(j,1) && enu1(i,5)==enu2(j,5)
comp(k,:)=[abs(enu1(i,2)-enu2(j,2)),abs(enu1(i,3)-enu2(j,3)),abs(enu1(i,4)-enu2(j,4)),
enu1(i,5)];
k=k+1;
else
end
end
end

%East representation

[r,s]=size(comp);
```

```
figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,1),'.g')
end
if comp(i,4)==2
plot(i,comp(i,1),'.y')
end
end
title('East comparison Q=1 (Green) Q=2 (Yellow) 21-02-15 Xiaomi and U-blox')
xlabel('Epoch')
ylabel('East Variation (m)')

hold off

%North representation

figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,2),'.g')
end
if comp(i,4)==2
plot(i,comp(i,2),'.y')
end
end
title('North comparison Q=1 (Green) Q=2 (Yellow) 21-02-15 Xiaomi and U-blox')
xlabel('Epoch')
ylabel('North Variation (m)')

hold off

%% Samsung and u-blox 21-02-15

clear
clc
sol1=xlsread('21-02-15-Samsung.xls');
[m,n]=size(sol1);
sol2=xlsread('21-02-15-Ublox.xls');
[h,j]=size(sol2);

%Then the solution is transformed into ENU
%Xiaomi

for i=1:m
[East1(i,1),North1(i,1),Alt1(i,1),Zone]=lla2enu(sol1(i,3),sol1(i,4),sol1(i,5));
end
enu1=[sol1(:,2),East1,North1,Alt1,sol1(:,6)];
```

```
%Ublox

for i=1:h
[East2(i,1),North2(i,1),Alt2(i,1),Zone]=lla2enu(sol2(i,3),sol2(i,4),sol2(i,5));
end
enu2=[sol2(:,2),East2,North2,Alt2,sol2(:,6)];


%the the comparison have to be made between the same epochs and also the
%same Q, so we are going to check the values of each table.

k=1;
for i=1:m
for j=1:h
if enu1(i,1)==enu2(j,1) && enu1(i,5)==enu2(j,5)
comp(k,:)=[abs(enu1(i,2)-enu2(j,2)),abs(enu1(i,3)-enu2(j,3)),abs(enu1(i,4)-enu2(j,4)),
enu1(i,5)];
k=k+1;
else
end
end
end

%East representation

[r,s]=size(comp);

figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,1),'.g')
end
if comp(i,4)==2
plot(i,comp(i,1),'.y')
end
if comp(i,4)==5
plot(i,comp(i,1),'.r')
end
end
title('East comparison Q=1 (Green) Q=2 (Yellow) 21-02-15 Samsung and U-blox')
xlabel('Epoch')
ylabel('East Variation (m)')

hold off

%North representation

figure
hold on
```

```
for i=1:r
if comp(i,4)==1
plot(i,comp(i,2),'.g')
end
if comp(i,4)==2
plot(i,comp(i,2),'.y')
end
if comp(i,4)==5
plot(i,comp(i,1),'.r')
end
end
title('North comparison Q=1 (Green) Q=2 (Yellow) 21-02-15 Samsung and U-blox')
xlabel('Epoch')
ylabel('North Variation (m)')

hold off


%% Xiaomi and u-blox 21-02-20

clear
clc

%reading the solutions that we are going to compare

sol1=xlsread('21-02-20-Xiaomi.xls');
[m,n]=size(sol1);
sol2=xlsread('21-02-20-Ublox.xls');
[h,j]=size(sol2);

%Then the solution is transformed into ENU
%Xiaomi

for i=1:m
[East1(i,1),North1(i,1),Alt1(i,1),Zone]=lla2enu(sol1(i,3),sol1(i,4),sol1(i,5));
end
enu1=[sol1(:,2),East1,North1,Alt1,sol1(:,6)];

%Ublox

for i=1:h
[East2(i,1),North2(i,1),Alt2(i,1),Zone]=lla2enu(sol2(i,3),sol2(i,4),sol2(i,5));
end
enu2=[sol2(:,2),East2,North2,Alt2,sol2(:,6)];


%the the comparison have to be made between the same epochs and also the
%same Q, so we are going to check the values of each table.

k=1;
for i=1:m
```

```
for j=1:h
if enu1(i,1)==enu2(j,1) && enu1(i,5)==enu2(j,5)
comp(k,:)=[abs(enu1(i,2)-enu2(j,2)),abs(enu1(i,3)-enu2(j,3)),abs(enu1(i,4)-enu2(j,4)),
enu1(i,5)];
k=k+1;
else
end
end
end

%East representation

[r,s]=size(comp);

figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,1),'.g')
end
if comp(i,4)==2
plot(i,comp(i,1),'.y')
end
end
title('East comparison Q=1 (Green) Q=2 (Yellow) 21-02-20 Xiaomi and U-blox')
xlabel('Epoch')
ylabel('East Variation (m)')

hold off

%North representation

figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,2),'.g')
end
if comp(i,4)==2
plot(i,comp(i,2),'.y')
end
end
title('North comparison Q=1 (Green) Q=2 (Yellow) 21-02-20 Xiaomi and U-blox')
xlabel('Epoch')
ylabel('North Variation (m)')

hold off

%% Samsung and u-blox 21-02-20

clear
```

```
clc
sol1=xlsread('21-02-20-Samsung.xls');
[m,n]=size(sol1);
sol2=xlsread('21-02-20-Ublox.xls');
[h,j]=size(sol2);

%Then the solution is transformed into ENU
%Samsung

for i=1:m
[East1(i,1),North1(i,1),Alt1(i,1),Zone]=lla2enu(sol1(i,3),sol1(i,4),sol1(i,5));
end
enu1=[sol1(:,2),East1,North1,Alt1,sol1(:,6)];

%Ublox

for i=1:h
[East2(i,1),North2(i,1),Alt2(i,1),Zone]=lla2enu(sol2(i,3),sol2(i,4),sol2(i,5));
end
enu2=[sol2(:,2),East2,North2,Alt2,sol2(:,6)];


%the the comparison have to be made between the same epochs and also the
%same Q, so we are going to check the values of each table.

k=1;
for i=1:m
for j=1:h
if enu1(i,1)==enu2(j,1) && enu1(i,5)==enu2(j,5)
comp(k,:)=[abs(enu1(i,2)-enu2(j,2)),abs(enu1(i,3)-enu2(j,3)),abs(enu1(i,4)-enu2(j,4)),
enu1(i,5)];
k=k+1;
else
end
end
end

%East representation

[r,s]=size(comp);

figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,1),'.g')
end
if comp(i,4)==2
plot(i,comp(i,1),'.y')
end
if comp(i,4)==5
```

```
plot(i,comp(i,1),'.r')
end
end
title('East comparison Q=1 (Green) Q=2 (Yellow) 21-02-20 Samsung and U-blox')
xlabel('Epoch')
ylabel('East Variation (m)')

hold off

%North representation

figure
hold on
for i=1:r
if comp(i,4)==1
plot(i,comp(i,2),'.g')
end
if comp(i,4)==2
plot(i,comp(i,2),'.y')
end
if comp(i,4)==5
plot(i,comp(i,1),'.r')
end
end
title('North comparison Q=1 (Green) Q=2 (Yellow) 21-02-20 Samsung and U-blox')
xlabel('Epoch')
ylabel('North Variation (m)')

hold off
```