ALMA MATER STUDIORUM - UNIVERSITA' DI BOLOGNA

SECONDA FACOLTA' DI INGEGNERIA
CON SEDE A CESENA

CORSO DI LAUREA
IN INGEGNERIA AEROSPAZIALE

CLASSE L-9

SEDE DI FORLÌ

ELABORATO FINALE DI LAUREA

IN ELABORAZIONE DATI PER LA NAVIGAZIONE

VEHICLE POSITIONING USING GPS PHASE MEASUREMENTS AND RESOLUTION OF INITIAL AMBIGUITY BY LAMBDA METHOD

CANDIDATO

Seghizzi Alberto Maria

RELATORE

Prof. Matteo Zanzi

Anno Accademico 2011-2012

Sessione III

ABSTRACT

L’oggetto del presente elaborato di tesi è costituito dal progetto preliminare di un sistema GPS per il calcolo del posizionamento relativo fra un utilizzatore mobile(o fisso) e una stazione fissa, di cui sono note le coordinate.

L’accuratezza associata alle misure di fase GPS, sia per il modello relativo alle “singole differenze”, sia per quello relativo alle “doppie differenze”, può essere sfruttata per ottenere una stima della posizione relativa tra i ricevitori, solo dopo aver risolto il problema delle ambiguità iniziali associate alle misure stesse.

Questo aspetto rappresenta l’aspetto più critico delle misure di fase, perché nel caso l’aggancio del segnale di un satellite venga perso, l’ambiguità intera corrispondente, precedentemente individuata, perde di significato, e deve essere determinata nuovamente una volta ripristinato l’aggancio.

Una prima parte della trattazione è quindi incentrata a evidenziare l’aspetto teorico delle equazioni implementate dall’algoritmo, in particolare il modello delle doppie differenze e il metodo LAMBDA, utilizzato per risolvere le ambiguità.

Una seconda parte è invece dedicata ai dati sperimentali ottenuti e al progetto del software in ambiente Matlab.

Summary

List of figures

CHAPTER 1

SCOPE

1.1 Aim

Differential GPS carrier phase measurements have much lower noise and multipath error than that of pseudorange measurements. However, the measurement of the carrier phase has a constant unknown integer ambiguity. Several technical issues are related to solving the integer ambiguity correctly.

In this report , the full procedure for parameter estimation based on the model of double difference GPS observations is reviewed. The LAMBDA method will be used for the integer ambiguity estimation, even if we will not focus our attention on the theoretical and mathematical aspects of the method.

Using the double difference technique, we have develop the relative position between two receivers, called baseline.

Moreover, if the location of one of the receivers, further referred to as the *reference* receiver, is known to some accuracy, the position of the other receiver, the *rover* receiver, can be computed at the same accuracy or at the accuracy of the baseline estimate. This positioning technique is called real-time kinematic positioning (RTK).

The Matlab implementation has been made with two fixed receivers(static mode), but in general there are no difficulties if one is mobile.

1.2 Work organization

This thesis concentrates on precise positioning using dual signal frequency.
The goal is to examine the precision attainable using double-frequency GPS receivers. A method for precise positioning is studied. The method expected to be capable of achieving centimeter-level precision but with its respective restricting conditions. Field experiments using authentic GPS data are conducted to assess if these methods could provide good positioning performance.
The thesis is organized as follows: the history and background of satellite positioning with the fundamental equations and measurement are discussed in **Chapter 2**.
Chapter 3 deals with the LAMBDA method, and a brief paragraph is also dedicated to the least squares and weighted least squares estimation. These chapters are summaries of already existing knowledge that is available in the literature.
Experimental results and test setup are presented in **Chapter 4** and conclusions are summarized and discussed in **Chapter 5**.

CHAPTER 2

GPS SYSTEM AND OBSERVABLES

2.1 GPS

NAVSTAR, the Global Positioning System—commonly known as GPS—was developed for military purposes, by the United States. It is controlled by the U.S. Department of Defense (DOD) and became fully operational in 1995.

GPS positioning is based on trilateration, i.e. computing the receiver position when distances to some reference points (here, the satellites) are known. The atomic clocks of all GPS satellites are mutually synchronized[1] and thus the satellites (also referred to as *space vehicles*, SV) are able to transmit time synchronized ranging signals.

Currently, GPS signals are transmitted in the L-band at two frequencies: the Link 1 (L1) signal at 1575.42 MHz and Link 2 (L2) at 1227.60 MHz. However, the L2 signal is encrypted and only available for those authorized by DOD. Some highend receivers are able to measure some parts of the L2 signal, but unfortunately those receivers are too expensive for mass-market. Initially, the civil-use L1 signal was intentionally distorted by dithering the satellite clocks. This interference, called *Selective Availability* (SA), was switched off by a presidential order in May 2000.

The GPS satellite constellation consists of about 32 satellites with orbits of radius 26000 kilometers. The period for this radius is a few minutes less than 12

[1] In the ideal case. The clocks are however subject to drifts due to e.g relativistic effects

hours. During one pass, a satellite is visible to a stationary receiver for a few hours . A GPS user is usually able to get a position fix in about one minute.

2.2 Coordinate Frames

In order to express locations, a coordinate frame must be defined. For terrestrial positioning, a natural choice is a system where the origin lies at the center of the Earth. There are two approaches to this:

Earth-Centered Earth-Fixed (ECEF) : The axes are fixed to some points on the Earth and the frame rotates.

Earth-Centered Inertial (ECI) The frame retains its directions with respect to some fixed celestial bodies despite the rotation of the Earth.

For personal positioning, ECEF is the obvious choice because it retains the coordinates of objects stationary with respect to the Earth time-invariant. However, ECI is useful in some position computations, especially in inertial navigation systems because Newton's laws of motion do not apply in ECEF frames due to the rotation.
From this point on, ECEF frames are considered unless otherwise mentioned.

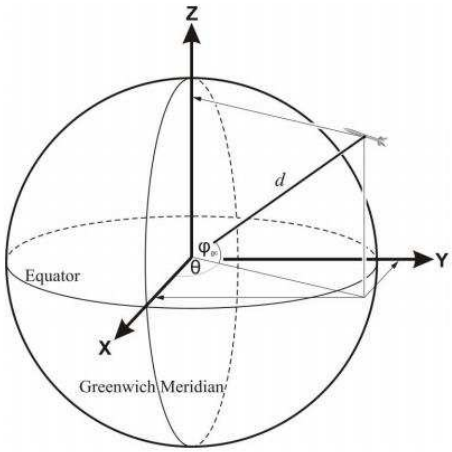

Figure 1: ECEF coordinate system

The coordinate frame needs to be fixed on the surface of the Earth. The conventional way of doing this is to fix the *z*-axis along the axis of rotation of the Earth, the *x*-axis along the equatorial plane to some reference meridian and to define the *y*-axis such that it completes the frame to a right-handed Cartesian coordinate system.

This principle is depicted in Fig 1. Actually, the axis of rotation of the Earth varies within time. Therefore, the z-axis can be chosen to pass through the *Conventional Terrestrial Pole* which is an average estimate of the North Pole between 1900 and 1905. A frame defined this way is called a *Conventional Terrestrial Reference Frame (CTRF)* .

Cartesian coordinates are inconvenient for expressing locations on the Earth. Therefore, the positions are usually expressed to the user in the *latitude-longitude-altitude* (LLA) coordinate system. However, to express altitude, the surface of the Earth must be modeled. Conventionally, this is done by modeling the Earth as an oblate ellipsoid. For example, the WGS84 reference ellipsoid is defined to have semi-major axis 6356.7523142 km and squared eccentricity 0.00669437999014. Zero eccentricity would correspond to a sphere.

In relative positioning, where the goal is to compute the *baseline* vector from a

reference point to the user, the most convenient way to express the baseline is the local *east-north-up* (ENU) frame. A point **x** in the ECEF frame is transferred to the local ENU frame of the reference point by rotating the baseline vector according to the latitude λ and longitude φ of **x0**:

$$x_{ENU} = \begin{bmatrix} -\sin\varphi & \cos\varphi & 0 \\ -\sin\lambda\cos\varphi & -\sin\lambda\sin\varphi & \cos\lambda \\ \cos\lambda\cos\varphi & \cos\lambda\sin\varphi & \sin\lambda \end{bmatrix} (x - x0)$$

The coefficient matrix is the product of two direction cosine rotation matrices. The transformation from ECEF to LLA is nontrivial but the inverse transformation can be computed in closed form. Algorithms for these transformations can be found in e.g. [7].

2.3 Signals

The signals consist of three essential parts: a carrier wave, a ranging code, and a navigation message.. These are discussed in detail in the following subsections. The structure of the Gps signal is show in Figure 2.

2.3.1 Carrier Wave

The signals are modulated on a sinusoidal carrier wave. The modulation type depends on the GNSS. GPS uses binary phase shift key modulation (while GALILEO signals are modulated by binary offset code modulation). Even though the carrier wave might seem to serve only for propagating the signal, measurements based on it are actually the key to centimeter-level precision as will be seen in Chapter 4.

2.3.2 Ranging Code

Measuring the distance from the receiver to the satellite transmitting the signal is done by observing the phase of the ranging code.

Definition 2.1. The correlation of two sequences *x* and *y* with *y* delayed by *d* samples is

$$corr(x, y, d) = \sum_i x_i y_{i-d}$$

Ranging codes are binary sequences satisfying the following correlation properties:

1- Autocorrelation: The code does not correlate with a delayed or advanced copy of itself unless the delay or advance is equal to 0, i.e. $corr(x, x, d) \approx 0$ when $d \neq 0$.

2- Cross-correlation: The code does not correlate with the code of another satellite with any delay, i.e. $corr(x, y, d) \approx 0$ when $x \neq y$.

These properties are required to ensure that the satellites can be correctly identified from the code and that two copies of the same code can be uniquely aligned by maximizing the correlation. Identifying the satellites by the code sequence constitutes a code division multiple access (CDMA) system.

Property 2 does not hold for GLONASS because GLONASS satellites are identified by their carrier frequency, not the code which is actually the same for all GLONASS satellites. Thus, GLONASS is a frequency division multiple access (FDMA) system. Due to the properties 1 and 2, ranging codes are called pseudo-random noise (PRN).

Typically, Gold codes are used as ranging codes. They can easily be generated

using simple linear feedback shift registers, and they do satisfy both of the desired properties. Gold codes are maximum length sequences, i.e., the length of a Gold code is 2^n-1 where n is the length of the shift register used to generate the code.
For example, the GPS 1023-bit coarse acquisition (C/A) codes, used on the unencrypted L1 signal, are produced using 10-bit shift registers.

To measure the range, the receiver generates a replica of the ranging code of the desired satellite and aligns it with the one received by the antenna. Then, the receiver compares the transmission time tag of the signal and the time of arrival estimated by the receiver's clock. The delay needed to align the codes is the

estimated range divided by the propagation speed of the signal which is equal to the speed of light c. However, since the receiver clock cannot be synchronized with the satellites, the bias in the clock introduces an offset to the range measurements. These biased range measurements are called pseudoranges. The offset is common to all pseudoranges measured by the receiver and therefore can be solved for as an additional unknown.
The magnitude of the noise in code measurements depends on the duration of one code bit (chip). The receiver can measure the delay by some fractions of one chip. E.g. the GPS C/A code chipping rate is 1.023 MHz. Thus, one chip corresponds to approximately 300 meters in distance. Typically, the noise in these measurements is in the order of decimeters, which corresponds to a thousandth of a chip.

2.3.3 Navigation Message

The third component in the signal is the navigation message. It contains the data necessary to compute the receiver position, such as satellite ephemerides, satellite clock and atmospheric corrections, and the almanac (approximate ephemerides for determining which satellites should be in view).

To compute the receiver position, ephemeris information for few satellites must be available. In GPS, the ephemeris and satellite clock data is broadcast every 30 seconds. In other systems, this data is broadcast often as well. The whole navigation message takes 12.5 minutes to broadcast on GPS . In an Assisted GPS (AGPS) system, it may be possible to obtain the necessary navigation data from another, faster link to decrease the time needed to compute the receiver position, i.e. the *time to first fix* (TTFF).

Broadcast ephemerides are usually accurate to some meters. If the positioning is carried out in post-processing mode, post-computed precise ephemerides can be used to enhance the solution.

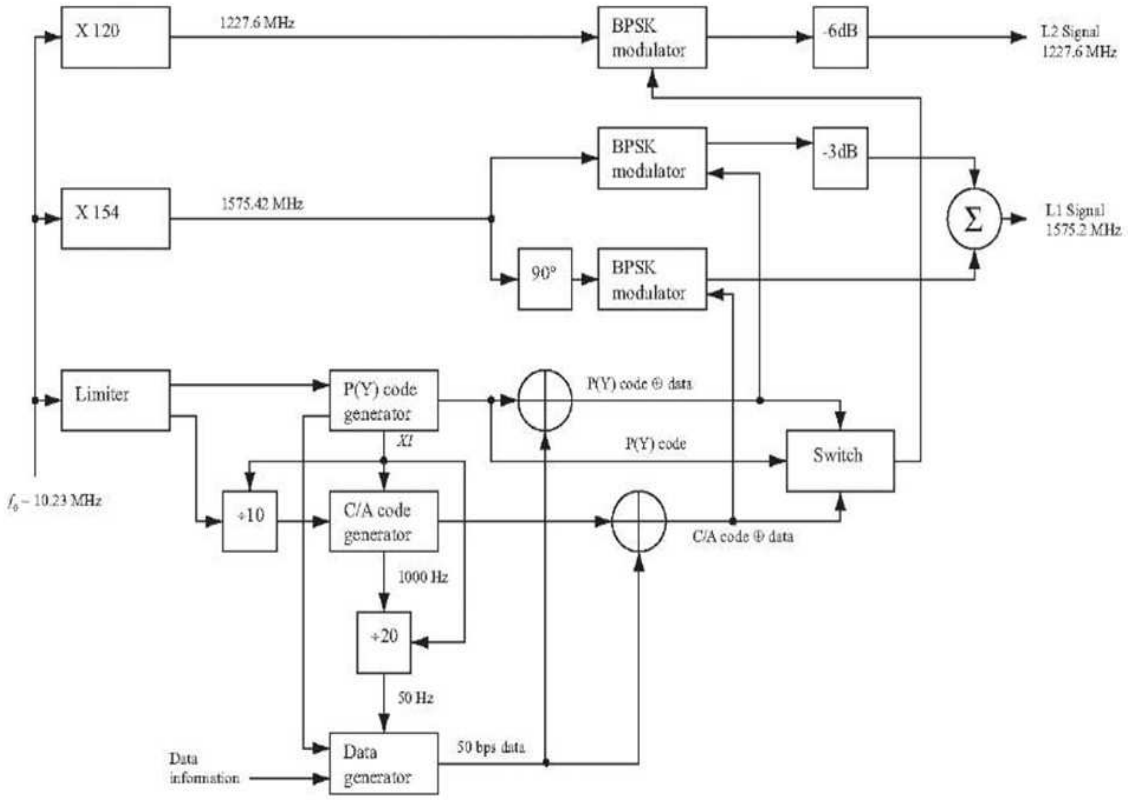

Figure 2: GPS signal structure

2.4 Measurement models

In this chapter, the basic GNSS measurements, i.e. code and carrier phase, are discussed. Error sources and the effect of satellite geometry on the solution accuracy are also reviewed.

2.4.1 Code phase observable

The purpose of the code phase measurement is to determine the distance between the satellite and the receiver. The pseudorange measurement is derived from the code phase. The pseudorange ρ for satellite *i* is the difference between the time of transmission $t_S^{(i)}$ and time of arrival $t_A^{(i)}$, shifted by the receiver clock bias δt and satellite clock bias $\delta t^{(i)}$.

The parenthesized superscripts refer to satellite ID's and are not exponents. This time difference is scaled to meters by the speed of light *c* and distorted by the measurement errors $e^{(i)}$:

$$\rho^{(i)} = c\,(\,t_A^{(i)} + \delta t - t_S^{(i)} - \delta t^{(i)}) + e^{(i)}$$

The error term $e^{(i)}$ contains some mutually independent components: atmospheric effects and unmodelable errors. The atmospheric effects for GNSS signals mainly occur in two atmospheric layers: the ionosphere and the troposphere. The ionosphere contains ions which affect electromagnetic radiation propagating through it. This results in a delay to the ranging code phase. The magnitude of the ionospheric delay depends on the amount of free electrons in the ionosphere. It is at its maximum in the afternoon when solar radiation has ionized molecules in the ionosphere and at its minimum during the night when the ions have recombined to molecules. The ionospheric effect is

dispersive, i.e. the magnitude of the delay depends on the frequency of the signal. Thus, a dual-frequency receiver is able to estimate the ionospheric delay. Ionospheric delay corrections are also broadcast in the GPS navigation message. The troposphere is the atmospheric layer where weather phenomena occur. Signal propagation speed through rain is slower than that during clear weather conditions because water has a higher refractive index than air. The tropospheric delay is not dispersive, therefore it can only be estimated by using some tropospheric model.

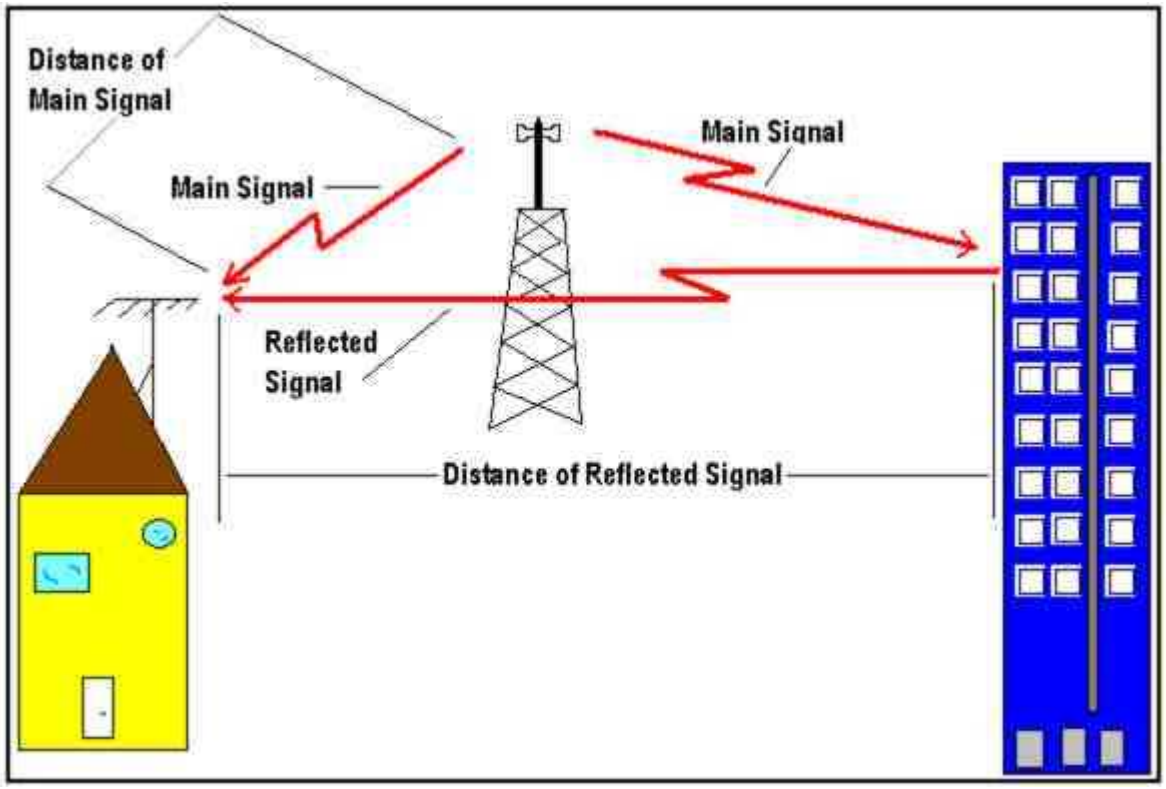

Figure 3: Multipath Propagation

Expanding the error term, the pseudorange ρ_{ir} can be modeled as:

$$\sqrt{(X_i - x_r)^2+(Y_i - y_r)^2 + (Z_i - z_r)^2} + c \cdot b_r + c \cdot b_i + I_i + T_i + M_{ir} + v_{ir}$$

Where:

- ρ_{ir} is the pseudorange calculated by the receiver *r* from the satellite *i*
- X_i , Y_i , Z_i are the components, in the ECEF system, of the vector position of the satellite *i* (obtained from the ephemeris data)
- x_r, y_r, z_r are the components, in the ECEF system, of the vector position of the receiver *r*
- c is the speed of light in the vacuum
- b_r is the receiver bias
- b_i is the satellite bias
- I_i is the ionospheric delay
- T_i is the tropospheric delay
- M_{ir} is the multipath error
- v_{ir} is the measurement error of the receiver

Using at least four pseudorange measurements, the unknowns (three components of receiver position and the receiver clock bias) can be estimated. Since the pseudorange equation is nonlinear with respect to receiver position, it is traditionally solved by linearization and least-squares methods even though closed-form solutions exist. (See Chapter 3).

2.4.2 Ionospheric delay

The ionosphere is the zone of the terrestrial atmosphere that extends itself from about 60 km until more than 2.000 km in high. As it name says, it contains a partially ionised medium, as result of the X and UV rays of Solar Radiation and the incidence of charged particles.

The propagation speed of the GNSS electromagnetic signals in the ionosphere depends on its electron density (see below), which is typically driven by two main processes: during the day, sun radiation causes ionisation of neutral atoms producing free electrons and ions. During the night, the recombination process prevails, where free electrons are recombined with ions to produce neutral particles, which leads to a reduction in the electron density.

A medium where the angular frequency ω and the wave number *k* are not proportional, is a dispersive media (i.e., the wave propagation speed and thence, the refractive index depends on the frequency). This is the case with the ionosphere where ω and k are related, in a first approximation, by[2] :

$$\omega^2 = c^2k^2 + \omega_p^2 \quad (1)$$

where[3] :

$$\omega_p = 2\pi f_p \quad \text{with} \quad f_p = 8.98\sqrt{N_e} \qquad \text{in Hz} \quad (2)$$

being N_e the ***electron density*** (in e^- / m^3). A complete derivation of this relationship can be found in [Davies, 1989] and the updated higher order terms in [McCarthy,D. and Petit,G., 2009], typically less than 0.1% of the total delay.

[2] Crawford, 1968
[3] Davies, 1989

The previous equation is named the *Relation of Dispersion* of the ionosphere, and ω_p is called the critical frequency of the ionospheric plasma, in the sense that signals with $\omega < \omega_p$ will be reflected and signals with $\omega > \omega_p$ will cross through the plasma.

The electron density in the ionosphere changes with the height having a maximum of $10^{11} - 10^{12}$ on 300-500 km. Thence, according to the expression, electromagnetic signals with $f > f_p \sim 10^6$ Hz will be able to cross the ionosphere. This is the case of GNSS signals which frequencies are at the order of 1 GHz = 10^9 Hz. Radio signals with frequencies under f_p will be reflected in the ionosphere.

From equation (1), and taking into account that $\omega = 2\pi f$ and the definition of phase and group velocity

$$v_{ph} = \frac{\omega}{k} \qquad v_{gr} = \frac{d\omega}{dk} \qquad (3)$$

it follows:

$$v_{ph} = \frac{c}{\sqrt{1 - \left(\frac{f_p}{f}\right)^2}} \qquad (4)$$

Thence,[4]

$$n_{ph} = \frac{c}{v_{ph}} \qquad n_{gr} = \frac{c}{v_{gr}} \qquad (5)$$

[4] [Hernandez-Pajares et al., 2007] Hernandez-Pajares, M., Juan, J. and Sanz, J.,2007. Second-order ionospheric term in GPS: Implementation and impact on geodetic estimates . Journal of Geophysical Research. 112, pp. 1-16.

the phase refractive index of the ionosphere can be approximated by:

$$n_{ph} = \sqrt{1 - \left(\frac{f_p}{f}\right)^2} \sim 1 - \frac{1}{2}\left(\frac{f_p}{f}\right)^2 = 1 - \frac{40.3}{f^2} N_e \quad (6)$$

At the frequency of GNSS signals, the previous approximation accounts for more than the 99.9% of the refractivity (first order ionospheric effect). That is, with less than a 0.1% of error, it can be assumed:

$$n_{ph} = 1 - \frac{40.3}{f^2} N_e \quad (7)$$

Differentiating the equation (1) with respect to k and taking into account (3), (5) and the approximation done before , yields the group refractive index:

$$n_{gr} = 1 + \frac{40.3}{f^2} N_e \quad (8)$$

Finally, the error on the pseudorange estimate is obtained integrating, along the path L of the signal in the ionosphere, the difference between the refractive index (8) and the the refractive index of the speed of light in the vacuum,1.

$$I_i = \int (n_{gr} - 1) dl = \frac{40.3}{f^2} \int N_e \, dl = \frac{40.3}{f^2} \cdot TEC$$

Total electron content (or **TEC**) is an important descriptive quantity for the ionosphere of the Earth. TEC is the total number of electrons present along a path between two points, with units of electrons per square meter, where 10^{16} electrons/m² = 1 TEC unit (TECU).

TEC is significant in determining the scintillation and group delay of a radio wave through a medium. Ionospheric TEC is characterized by observing carrier phase delays of received radio signals transmitted from satellites located above the ionosphere, often using Global Positioning System satellites. TEC is strongly affected by solar activity.

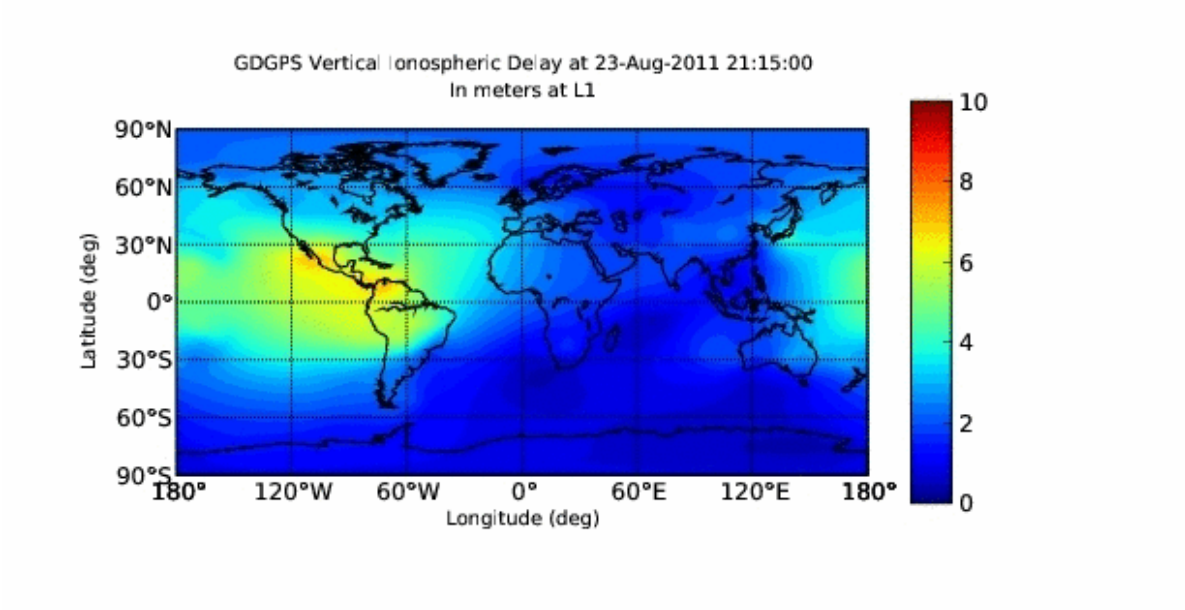

Figure 4: Ionospheric delay

2.4.3 Carrier Phase observable

We now introduce the carrier phase observable, which is used for high precision applications.

We start with the basic concepts, starting with the meaning of “phase”. We then go on to describe the process of observing the carrier phase, and develop an observation model. Fortunately, most of the model can be reduced to what we have learned so far for the pseudorange. Unlike most textbooks, we take the approach of presenting the model in the “range formulism”, where the carrier

phase is expressed in units of metres, rather than cycles. However, there are some fundamental differences between the carrier phase and the pseudorange observables, as we shall see when we discuss “phase ambiguity” and the infamous problem of “cycle slips”.

The Meaning of Phase and Frequency

“**Phase**” is simply “angle of rotation,” which is conventionally in units of “cycles” for GPS analysis. Consider a point moving anti-clockwise around the edge of a circle, and draw a line from the centre of the circle to the point. As illustrated in Figure 5, the “phase” $\phi(t)$ at any given time t can be defined as the angle through which this line has rotated.

Phase is intimately connected with our concept of time, which is always based on some form of periodic motion, such as the rotation of the Earth, the orbit of the Earth around the Sun (“dynamic time”), or the oscillation of a quartz crystal in a wristwatch (“atomic time”). Even our reprentation of time is often based on rotation, such as the angle of the hands on the face of a clock. Angles of rotation give us our measure of “time.” In this way, phase can be thought of as a measure of time (after conversion into appropriate units). We can write this formally as:

$$T(t) = k(\,\phi(t) - \phi_0)$$

- T is the time according to our clock at time t (whatever the clock may be)
- $\phi_0 = \phi(0)$ is so that the clock reads zero when t = 0
- k is a calibration constant , converting units of cycles into units of seconds

Indeed, we can take the above equation as the *definition* of clock time. Whether of not this clock time is useful depends on the constancy of rate of change of phase. This brings us to the concept of frequency.

The "**frequency**," expressed in units of "cycles per second," is the number of times the line completes a full 360o rotation in one second (which of course, is generally a fractional number). This definition is somewhat lacking, since it seems to assume that the rotation is steady over the course of one second. One can better define frequency instantaneously as the first derivative of phase with respect to time; that is, the angular speed.

$$f = \frac{d\phi(t)}{dt}$$

We chose to treat phase as a fundamental quantity, and frequency as a derived quantity. For example, we can say that frequency is a constant, if we observe the phase as changing linearly in time. Constant frequency is the basis of an ideal clock. If the frequency can be written as a constant, f_0, then we can write the phase of an ideal clock as:

$$\phi_{ideal} = f_0 t + \phi_0$$

therefore

$$T_{ideal} = k f_0 t$$

Since we want our a clock second to equal a conventional second (T_{ideal}=t), we see that an appropriate choice for the calibration constant is $k = 1/f_0$, where f_0 is the nominal frequency of the oscillator. Going back to our original equation for clock time, we can now define clock time as:

$$T(t) = \frac{\phi(t) - \phi_0}{f_0}$$

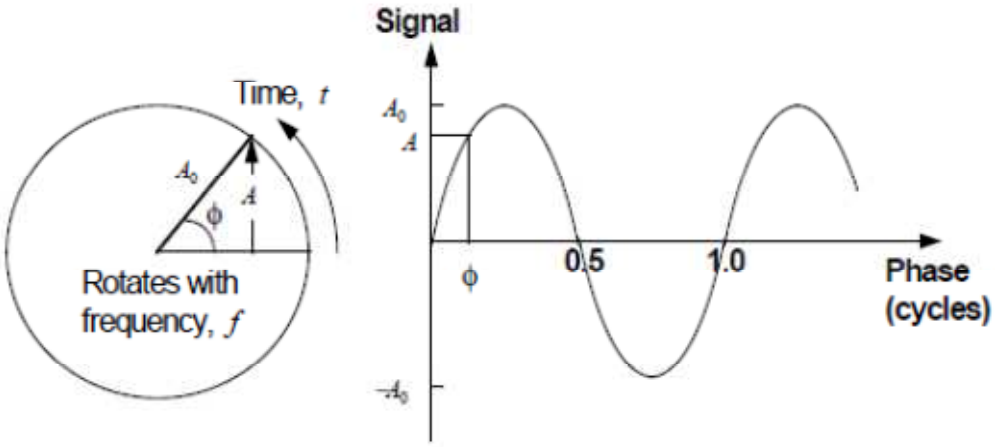

Figure 5: The meaning of phase

At time *t*, the height of point $A(t)$ above the centre of the circle in figure 5 is given by:

$$A(t) = A_0 \sin(2\pi\phi(t))$$

Where A_0 is the radius of the circle. Since the concept of phase is often applied to periodic signals, we can call $A(t)$ the "signal" and A_0 the "amplitude of the signal".

For example, in the case of radio waves $A(t)$ would be the strength of the electric field, which oscillates in time as the wave passes by. Inverting the above formula, we can therefore determine the phase $\phi(t))$ if we measure the signal $A(t)$ (and similarly, we could infer the clock time).

Note that, for an ideal clock, the signal would be a pure sinusoidal function of time:

$$\begin{aligned} A_{ideal} &= A_0 \sin(2\pi\phi_{ideal}) \\ &= A_0 \sin(2\pi f_0 t + 2\pi\phi_0) \\ &= (A_0 \cos(2\pi\phi_0)) \sin(2\pi f_0 t) + (A_0 \sin(2\pi\phi_0)) \cos(2\pi f_0 t) \\ &= A_0^s \sin(\omega_0 t) + A_0^c \cos(\omega_0 t) \end{aligned}$$

where the "angular frequency" $\omega_0 = 2\pi f_0 t$ has units of radians per second. For a real clock, the signal would the same sinusoidal function of its own "clock time," (but would generally be a complicated function of true time):

$$A(T) = A_0^s \sin(\omega_0 T) + A_0^c \cos(\omega_0 T)$$

We note that the nominal GPS signal takes on the above form, except that the signal is modulated by "chips", formed by multiplying the amplitudes A_0^s (for C/A code) and A_0^c (for P code) by a pseudorandom sequence of +1 or -1. The underlying sinusoidal signal is called the "carrier signal." It is the phase of the carrier signal that gives us precise access to the satellite clock time; therefore we can use this phase for precise positioning.

The carrier beat signal

The GPS carrier signal *G(t)* from the satellite is "mixed" (multiplied) with the receiver's own replica carrier signal *R*(*t*). The result of this mixing is shown in Figure 6.

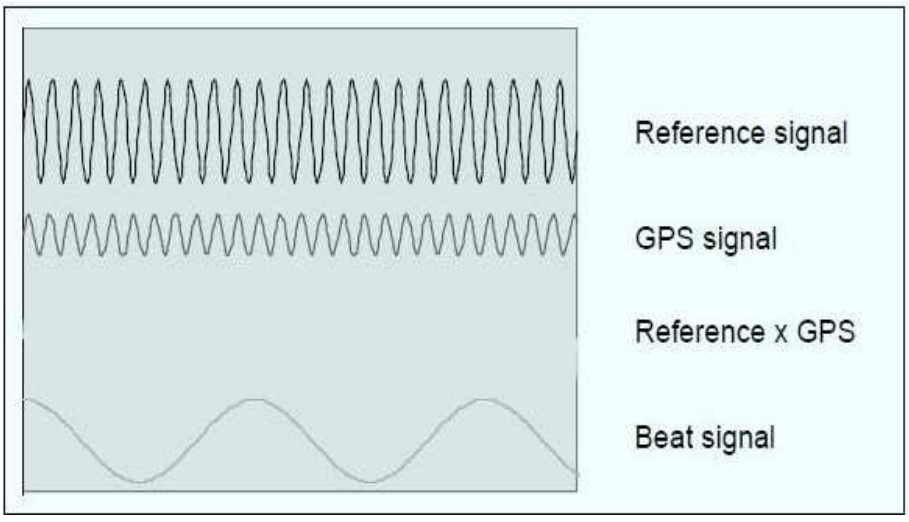

Figure 6: Producing a beat signal by mixing the carrier and replica signals

Mathematically, one can show that one would expect the result to be the difference between a low frequency signal and a high frequency signal:

$$R(t) \otimes G(t) = G_0 \sin(2\pi\phi_G(t)) \times R_0 \sin(2\pi\phi_R(t))$$
$$= \frac{G_0 R_0}{2}\left[cos2\pi\left(\phi_R(t) - \phi_G(t)\right) - cos2\pi\left(\phi_R(t) + \phi_G(t)\right)\right]$$

The high frequency component can be easily filtered out by the receiver electronics, leaving only the carrier beat signal.

$$B(t) = Filter\{R(t) \otimes G(t)\}$$
$$= \frac{G_0 R_0}{2} cos2\pi\left(\phi_R(t) - \phi_G(t)\right)$$
$$= B_0 cos2\pi\left(\phi_B(t)\right)$$

where we have introduced the carrier beat phase $\phi_B(t)$, which is defined to be equal to the difference in phase between the replica signal and the GPS signal.

$$\phi_B(t) = \phi_R(t) - \phi_G(t)$$

By differentiating the above equation with respect to time, we find that the "beat frequency" is equal to the difference in frequencies of the two input signals.

$$f_B = \frac{d\phi_B}{dt} = f_R - f_G$$

We note that the above formulas apply even when the carrier phase is modulated with codes, provided the replica signal is also modulated (because the values of -1 will cancel when multiplying the two signals). If the codes are not known, it is possible to square both the incoming signal and the replica signal prior to mixing. The problem with this is that squaring amplifies the noise, thus introducing larger random measurement errors.

Origin of the Phase Ambiguity

Our model of carrier beat phase not a complete picture, since we can arbitrarily add an integer number of cycles to the carrier beat phase, and produce exactly the same observed beat signal.

Suppose we only record the fractional phase of the first measurement. We would have no way of telling which integer N to add to this recorded phase so that it really did equal the difference in phase between the replica signal and the GPS signal. This is fundamentally because we have no direct measure of the total phase of the incoming GPS signal. We can express this as follows:

$$\Phi + N = \phi_R - \phi_G$$

where we use a capital Greek symbol Φ to emphasise that it represents the phase value actually recorded by the receiver. Provided the receiver does keep track of how many complete signal oscillations there have been since the first measurement, it can attach this number of cycles to the integer portion of the recorded beat phase. However, there will still be an overall ambiguity N that applies to all measurements. That is, we can model N as being the same (unknown) constant for all measurements. If the receiver looses count of the oscillations (e.g., because the signal is obstructed, or because of excessive noise), then a new integer parameter must be introduced to the model, starting at that time. This integer discontinuity in phase data is called a “cycle slip.”

Interpretation of Phase Ambiguity

The reader might also be wondering if there is some kind of geometrical interpretation for N. It turns out that there is, but it does require some oversimplified assumptions. By definition, the unknown value of N can be written as:

$$N = (\text{integer portion of } \phi_R - \phi_G) - (\text{integer portion of } \Phi)$$

The second term is completely arbitrary, and depends on the receiver firmware. For example, some receivers set this value to zero for the first measurement. Let us assume this is true, and drop this term. For the sake of interpretation, let us now assume that the receiver and satellite clocks keep perfect time. Under these circumstances, the first term would equal the integer portion of the number of signal oscillations that occur in the receiver from the time the signal was transmitted to the time the signal was received. We can therefore interpret N as equal to the number of carrier wavelengths between the receiver (at the time it makes the first observation), and the satellite (at the time it transmitted the signal). Of course, we made assumptions about perfect clocks and the particular nature of the firmware; so we must beware not to take this interpretation too literally.

2.4.4 The Carrier Phase Observation Model

We now move towards a more rigorous treatment of the carrier beat phase observable, building on our concepts of phase and signal mixing. Our notation will change slightly in preparation for further development.

The carrier phase can be modeled as:

$$(\phi_{jr} + N_{jr})\lambda = r_{jr} + c \cdot b_r + c \cdot b_j - I_j + T_j + m_{jr} + \mu_{jr}$$

Where

- ϕ_{jr} is the phase given from the receiver ;
- N_{jr} is the initial integer ambiguity;
- λ is the carrier wavelength;
- r_{jr} is the geometric distance between the satellite j and the receiver r;
- m_{jr} is the multipath error;
- μ_{jr} is the measurement error of the receiver

This measurement is significantly more precise than code phase measurements. Typical noise levels in code and carrier phase measurements are compared in Figure 7. The measurements shown in the figure were logged using a stationary consumer-grade receiver.

However, only the phase can be measured, not the number of full carrier cycles. The unknown integer number of carrier cycles is commonly known as the integer ambiguity. As long as the receiver tracks the signal uninterruptedly and remains locked to it, the integer ambiguity remains constant. Thus, subtracting two consecutive carrier phase measurements to the same satellite cancels the integer ambiguity and gives a precise estimate of the change in pseudorange between these measurements.

This idea is discussed further in Chapter 3. The difference of two consecutive carrier phase measurements is also known as the delta range, and the carrier phase itself is sometimes referred to as accumulated delta range.

The carrier phase measurement is corrupted by the same error sources as the code phase. However, while code phase is delayed in the ionosphere, the carrier

phase is advanced. Also, carrier phase measurements are shifted by satellite and receiver clock drifts (frequency biases), not the time biases.

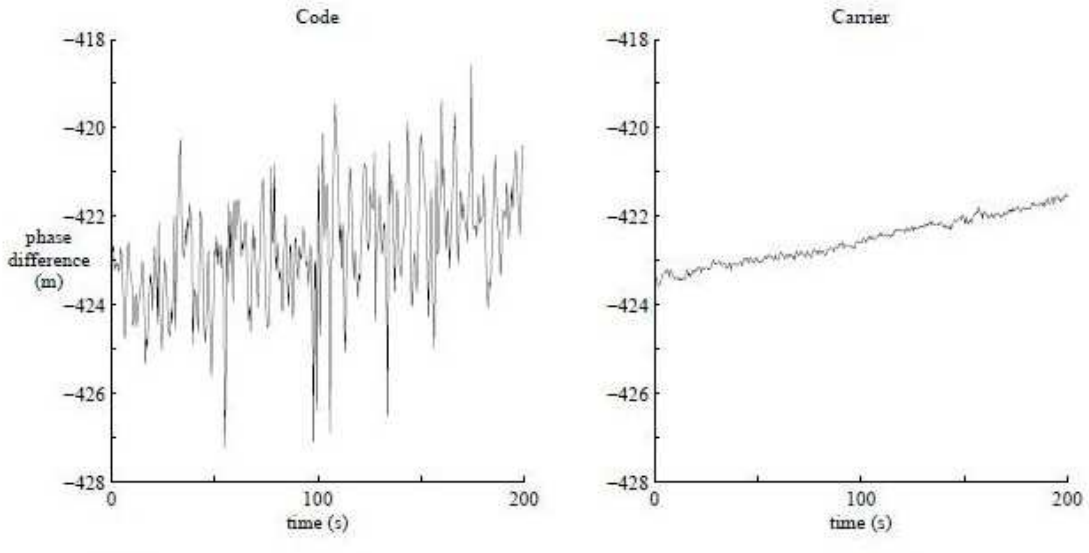

Figure 7: Noise in code and carrier phase measurements. Only differences between consecutive measurement are shown, not absolutely measurements. The carrier phase were scaled to meters by the signal wavelength for easier comparison

CHAPTER 3

NAVIGATION SOLUTION

3.1 Problem formulation

In this chapter, a simple method for positioning using carrier phase measurements is presented and discussed. The method provides information on the user position relative to the receiver location at some instant .

The major problem in using the carrier phase measurements for positioning is the integer ambiguity present in the measurement. However, it is known that the ambiguity remains constant as long as the receiver continuously tracks the signal maintaining phase lock. Therefore, the ambiguity can be cancelled by forming differences of two measurements by the same receiver to the same satellite at different epochs of time provided that no cycle slips have occurred, i.e., the tracking truly has been continuous. The difference of two carrier phase measurements yields a precise estimate of the change in range between the receiver and the satellite during the time between the measurements.

In addition to canceling out the integer ambiguity term, the effect of other time and/or location correlated error sources is reduced but not totally canceled their derivatives remain present. Thus, atmospheric and satellite clock and orbit errors are diminished. Since the derivatives of these are still present in the solution, the accuracy of the time-differenced measurements will degrade with time. Fortunately, these errors do not change rapidly in time, compared to, e.g., the clock dithering caused by SA, which was a strong motivation for the development of differential positioning methods but fortunately is deactivated at

the moment. Therefore, the trajectory of the receiver can be computed for some short time interval.

3.2 Single Difference

The purpose of “single differencing” is to eliminate satellite clock bias. The model is based on differencing the phase measurements between two receivers,(the first one is fixed and the coordinates are known), relative to the same satellite at the same epoch: if the baseline is short (less than 30 km), the common errors like ionospheric and tropospheric delay can be delete.

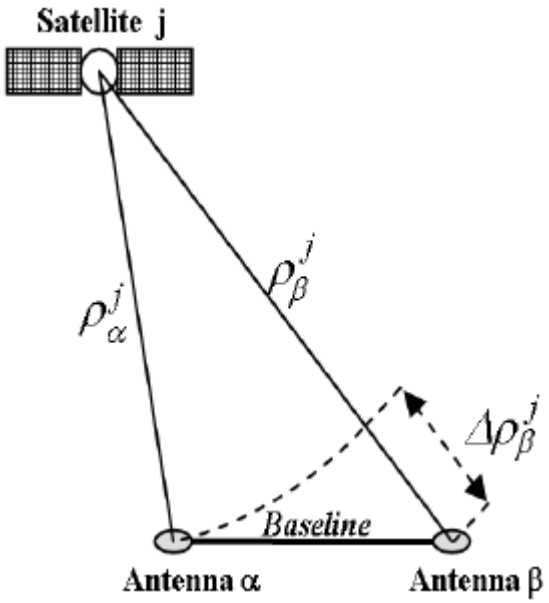

Figure 8: Single differencing geometry

Considering the phase model (2.4.4), indicating with F the measurements taking by the fixed receiver and with M the measurements by the mobile (or fixed) one, for the j-satellite the single difference model could be write as:

$$(\phi_{jF} + N_{jF})\lambda = r_{jF} + c \cdot b_F + c \cdot b_j - I_j + T_j + m_{jF} + \mu_{jF}$$

$$(\phi_{jM} + N_{jM})\lambda = r_{jM} + c \cdot b_M + c \cdot b_j - I_j + T_j + m_{jM} + \mu_{jM}$$

$$\begin{aligned}(\Delta\phi_j + \Delta N_j)\lambda &= (\phi_{jM} + N_{jM})\lambda - (\phi_{jF} + N_{jF})\lambda \\ &= r_{jM} + c \cdot b_M + m_{jM} + \mu_{jM} - (r_{jF} + c \cdot b_F + m_{jF} + \mu_{jF}) \\ &= r_{jM} - r_{jF} + c \cdot (b_M - b_{F)} + (m_{jM} - m_{jF}) + (\mu_{jM} - \mu_{jF}) \\ &= r_{jM} - r_{jF} + c \cdot \Delta b + \Delta m_j + \Delta\mu_j\end{aligned}$$

where

- $\Delta\phi_j = (\phi_{jM} - \phi_{jF})$
- $\Delta N_j = (N_{jM} - N_{jF})$
- $\Delta b = b_M - b_F$
- $\Delta m_j = m_{jM} - m_{jF}$
- $\Delta\mu_j = \mu_{jM} - \mu_{jF}$

An assumption has been made, that the satellite clock bias is effectively identical at the slightly different times that the signal was transmitted to F and to M. The difference in transmission time could be as much as a few milliseconds, either because the imperfect receiver clocks have drifted away from GPS time by that amount, or because the stations might be separated by 1,000 km or more. Since selective availability is typically at the level of 10^{-9} (variation in frequency divided by nominal frequency), over a millisecond (10^{-3}s) the
satellite clock error will differ by 10^{-12}s. This translates into a distance error of 10^{-12}c, or 0.3 mm, a negligible amount under typical S/A conditions (however, it

may not be negligible if the level of S/A were increased; but this effect could in principle be corrected if we used reference receivers to monitor S/A). Another point worth mentioning, is that the coordinates of the satellite at transmission time can easily be significantly different for receiver F and M.

The atmospheric delay terms are now considerably reduced, and vanish in the limit that the receivers are standing side by side. The differential troposphere can usually be ignored for horizontal separations less than approximately 30 km, however differences in height should be modelled. The differential ionosphere can usually be ignored for separations of 1 to 30 km, depending on ionospheric conditions. Due to ionospheric uncertainty, it is wise to calibrate for the ionosphere using dual-frequency receivers for distances greater than a few km.

Although the single difference has the advantage that many error sources are eliminated or reduced, the disadvantage is that only relative position can be estimated (unless the network is global-scale). Moreover, the receiver clock bias is still unknown, and very unpredictable. This takes us to “double differencing”.

3.3 Double Difference

The purpose of “double differencing” is to eliminate receiver clock bias. Consider the single differenced observation equations for two receivers R and M observing satellites *j* and *k*:

$$(\Delta\phi_j + \Delta N_j)\lambda = r_{jM} - r_{jF} + c \cdot \Delta b + \Delta m_j + \Delta\mu_j$$

$$(\Delta\phi_k + \Delta N_k)\lambda = r_{kM} - r_{kF} + c \cdot \Delta b + \Delta m_k + \Delta\mu_k$$

The double difference phase is defined as the difference between these two:

$$\begin{aligned}(\nabla\Delta\phi_{jk} + \nabla\Delta N_{jk})\lambda &= \left(\Delta\phi_j + \Delta N_j\right)\lambda - (\Delta\phi_k + \Delta N_k)\lambda \\ &= r_{jM} - r_{jF} + \Delta m_j + \Delta\mu_j - (r_{kM} - r_{kF} + \Delta m_k + \Delta\mu_k) \\ &= r_{jM} - r_{jF} - r_{kM} + r_{kF} + \left(\Delta m_j - \Delta m_k\right) + (\Delta\mu_j - \Delta\mu_k) \\ &= r_{jM} - r_{jF} - r_{kM} + r_{kF} + \nabla\Delta m_{jk} + \nabla\Delta\mu_{jk}\end{aligned}$$

where

- $\nabla\Delta\phi_{jk} = \Delta\phi_j - \Delta\phi_k$
- $\nabla\Delta N_{jk} = \Delta N_j - \Delta N_k$
- $\nabla\Delta m_{jk} = \Delta m_j - \Delta m_k$
- $\nabla\Delta\mu_{jk} = \Delta\mu_j - \Delta\mu_k$

As more than one double difference is needed at each measurement epoch for computing a position fix, the between-satellite differences can be constructed in many satellite combinations.

If there are k single differences available, only k -1double differences can be constructed without redundancy.

One of the satellites is chosen as the base (or reference) satellite. All double differences are formed with respect to this satellite, i.e. between satellites (1, 2), (1, 3), . . . , (1, k).

Usually, the satellite with the highest elevation angle is chosen as the base satellite because signals coming from higher elevations travel a shorter time through the atmosphere and therefore are less plagued by ionospheric and tropospheric errors. They are also less prone to multipath. The evident drawback

of all measurements depending on one satellite is that if the base satellite signal is lost, all double differences and their possibly resolved integer ambiguities will be lost.

3.4 Baseline estimation process

We now deal exclusively with double difference carrier phase measurement currently available. The carrier phase measurement at L1 from K satellites give $K-1$ double difference equations at each epoch. A dual-frequency receiver would provide another set of $K-1$ equations from the measurement at L2.

While the phase tracking is continuous, three new unknowns (new position coordinates) are introduced at each epoch if the user is in motion, and none if the user is stationary. Consider static initialization.

There are $[\,3+(K-1)\,]$ unknowns for single frequency measurements and $[\,3+2(K-1)\,]$ unknowns for dual-frequency measurements and, over time, many more equations.

We expect the estimation process to be helped by redundant measurements, good satellite geometry, dual-frequency measurements, and significant change in satellite geometry over the observation period. Unmodeled errors would hurt. The errors generally grow with the baseline length, but significant multipath at either station can pose a challenge.

Double difference equations is linear in the integers, but nonlinear in the position coordinates. In order to estimate these parameters, we will try to form an over determined system of linear equations and solve it using the least squares criterion and the LAMBDA method.

3.4.1 Linear Model for position estimation

The position of the stationary reference station F is known, and we are interesting in estimating the position of the user (M) relative to the reference station.

Consider a generic double difference equation for carrier phase. For a short baseline, we can delete the multipath error.

$$\left(\nabla\Delta\phi_{jk} + \nabla\Delta N_{jk}\right)\lambda = r_{jM} - r_{jF} - r_{kM} + r_{kF} + \nabla\Delta\mu_{jk}$$

For obtaining an estimate of the vector S_M, the relative position between the two receiver, we can linearize the equation of double difference ,given an initial estimate of the relative position S_{M0}.

Linearazing in a neighbourhood of the vector S_{M0} the model descending is:

$$\left(\nabla\Delta\phi_{jk} + \nabla\Delta N_{jk}\right)\lambda \approx r_{jM0} - r_{jF} - r_{kM0} + r_{kF} + \left(h_{jM} - h_{kM}\right)\cdot(S_M - S_{M0}) + e$$

Where

- S_M is the vector of relative position between the two receivers;
- S_{M0} is the reference vector for the linearization;

$$S_M = \begin{bmatrix} S_{xM} \\ S_{yM} \\ S_{zM} \end{bmatrix} = \begin{bmatrix} X_M - X_F \\ Y_M - Y_F \\ Z_M - Z_F \end{bmatrix} \qquad S_{M0} = \begin{bmatrix} S_{xM0} \\ S_{yM0} \\ S_{zM0} \end{bmatrix} = \begin{bmatrix} X_{M0} - X_F \\ Y_{M0} - Y_F \\ Z_{M0} - Z_F \end{bmatrix}$$

$$r_{jM0} = \sqrt{(X_j - (X_F - S_{xM0}))^2 + (Y_j - (Y_F - S_{yM0}))^2 + (Z_j - (Z_F - S_{zM0}))^2}$$

$$h_{jM} = \vec{\nabla} r_{jM}\big|_{S_{M0}}$$

$$= \left[\frac{(X_F - S_{xM0}) - X_j}{r_{jM0}} \quad \frac{(Y_F - S_{yM0}) - Y_j}{r_{jM0}} \quad \frac{(Z_F - S_{zM0}) - Z_j}{r_{jM0}} \right]$$

$$e = \nabla \Delta \mu_{jk}$$

This equation constitutes the measurement model to be considered for solving the problem. Therefore, the application of double difference method performs a relative positioning, because permits a baseline estimation.

Chosen the satellite 1 as the reference, writing n as the number of visible satellite,we regrouping terms and defining :

$$y(t) = \begin{bmatrix} \lambda \nabla \Delta \phi_{21} - r_{2M0} + r_{2F} + r_{1M0} - r_{1F} \\ \vdots \\ \lambda \nabla \Delta \phi_{n1} - r_{nM0} + r_{nF} + r_{1M0} - r_{1F} \end{bmatrix}$$

$$G(t) = \begin{bmatrix} h_{2M} - h_{1M} \\ \vdots \\ h_{nM} - h_{1M} \end{bmatrix}$$

$$N = \begin{bmatrix} \nabla \Delta N_{12} \\ \vdots \\ \nabla \Delta N_{1n} \end{bmatrix}$$

$$\varepsilon = \begin{bmatrix} \nabla \Delta \mu_{21} \\ \vdots \\ \nabla \Delta \mu_{n1} \end{bmatrix}$$

$$\delta s = S_M(t) - S_{M0}(t)$$

The linear equation derived from the double difference model is

$$y(t) = G(t) \cdot \delta s + \lambda \cdot N + \lambda \cdot \varepsilon(t)$$

We can combine all such linear equations obtained from single or dual frequency measurements at measurement epoch t_i into a generic vector matrix representation

$$y(i) = G(i) \cdot \delta s + \lambda \cdot N + \lambda \cdot \varepsilon(i) \qquad (3.1)$$

where

- $y(i)$ denotes the difference between the measured and computed carrier phase double differences for the initial position estimate;
- $G(i)$ is the observation matrix characterizing the double difference user-reference station satellite geometry;
- δs is the error in the initial position estimate;
- N is the vector of double difference integer ambiguities to be estimated;

If there are K double difference measurements, $G(i)$ is a $K \times 3$ matrix and N is a K vector. (K= n-1 for single frequency measurements, and K= 2·(n-1) for dual frequency measurements, where n is the number of satellites in view).

Consider a simple case of stationary user, the next measurement epoch brings additional K equations

$$y(i+1) = G(i+1) \cdot \delta s + \lambda \cdot N + \lambda \cdot \varepsilon(i+1) \qquad (3.2)$$

We can combine (3.1) and (3.2) as:

$$\begin{bmatrix} y(i) \\ y(i+1) \end{bmatrix} = \begin{bmatrix} G(i) \\ G(i+1) \end{bmatrix} \cdot \delta s + \lambda \begin{bmatrix} I \\ I \end{bmatrix} \cdot N + \begin{bmatrix} \varepsilon(i) \\ \varepsilon(i+1) \end{bmatrix} \qquad (3.3)$$

For a general case of measurements , perhaps for multiple epochs, we can write (3.3) as:

$$y = G \cdot \delta s + A \cdot N + \lambda \cdot \varepsilon$$

Using the least squares criterion, we look for a real-valued δs 3-vector and a K vector of integers N which minimize the cost function

$$c(\delta s, N) = \left\| y - G \cdot \delta s - A \cdot N \right\|^2 \qquad (3.4)$$

The cost function is simply the sum of the lengths of residual vectors squared. In a kinematic case, δs would change from one measurement epoch to next. There is no difficulty, however, in formulating the problem as above. Note also that if we have a basis for assigning different weights to the measurements, we can formulate (3.4) as a weighted least squares problem. We'll revisit this issue below.

Minimization of (3.4) would be straightforward were it not for the constraint that each element of N be an integer. We have an *integer least-squares* problem. We could disgregard the constraint to make the problem routine and, in

fact, this approach is commonly used. An alternative is to limit the estimates to a set of integers, and then 'search' this set for the best solution. In principle, we can obtain the solution with measurements form single epoch.

3.4.2 Least Squares

The method of least squares assumes that the best-fit curve of a given type is the curve that has the minimal sum of the deviations squared (*least square error*) from a given set of data.

The "estimated residuals" are defined as the difference between the actual observations and the new, estimated model for the observations.

Consider a generic linearised equation

$$b = B \cdot x + v$$

Where

- b is the residual observation (observed minus computed observations)
- x is the vector of the unknown
- B is the observation matrix of the unknown
- v is the matrix column vector contains all the noise terms, which are also unknown at this point

Let us consider a solution for the linearised observation equations, denoted $\hat{x}$. Using it, we can write the estimated residuals as

$$\hat{v} = b - B \cdot \hat{x}$$

The “least squares” solution can be found by varying the value of x until the following functional is minimised:

$$J(x) = \sum_{i=1}^{m} v_i^2 = v^T v = (b - B \cdot x)^T (b - B \cdot x)$$

That is, we are minimising the sum of squares of the estimated residuals. If we vary x by asmall amount, then $J(x)$ should also vary, except at the desired solution where it is stationary (since the slope of a function is zero at a minimum point). The following illustrates the application of this method to derive the least squares solution.The minimum of the sum of squares is found by setting the gradient to zero:

$$\begin{aligned}
\delta J(\hat{x}) &= 0 \\
\delta\{(b - B \cdot \hat{x})^T (b - B \cdot \hat{x})\} &= 0 \\
\delta(b - B \cdot \hat{x})^T (b - B \cdot \hat{x}) + (b - B \cdot \hat{x})^T \delta(b - B \cdot \hat{x}) &= 0 \\
(-B \cdot \delta\hat{x})^T (b - B \cdot \hat{x}) + (b - B \cdot \hat{x})^T (-B \cdot \delta\hat{x}) &= 0 \\
(-2B \cdot \delta\hat{x})^T (b - B \cdot \hat{x}) &= 0 \\
(\delta\hat{x}^T \cdot B^T)(b - B \cdot \hat{x}) &= 0 \\
\delta\hat{x}^T (B^T b - B^T B\hat{x}) &= 0 \\
B^T B\hat{x} &= B^T b
\end{aligned}$$

The last line is called the system of “normal equations”. The solution to the normal equations is therefore:

$$\boldsymbol{\hat{x} = (B^T B)^{-1} B^T b}$$

This assumes that the inverse to $B^T B$ exists. For example, $m \geq 4$ is a necessary (but not sufficient) condition. Problems can exist if, for example, a pair of satellites lie in the same line of sight, or if the satellites are all in the same orbital plane. In almost all practical situations, $m \geq 5$ is sufficient. Alternatively, one parameter could be left unestimated (e.g., the height could be fixed to sea-level for a boat).

3.4.3 Correlations among the double difference measurements

We have assumed the measurements(both code and carrier) from the satellites in view to be uncorrelated. In particular, the covariance of the carrier phase measurements at an instant is modeled as

$$\sum_{\phi} = \sigma_{\phi}^2 I$$

where σ_{ϕ} is the standard deviation of the phase measurements, and I is the identity matrix. We make this simplifying assumption in the absence of a simple, truer model. In this section, we examine the correlations among the single and double differences.

The single differences corresponding to a pair of satellites can be written in matrix notation as

$$\begin{bmatrix} \phi_{ur}^k \\ \phi_{ur}^l \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \phi_u^k \\ \phi_r^k \\ \phi_u^l \\ \phi_r^l \end{bmatrix}$$

The covariance matrix of this pair of single difference is

$$\sum_{sd} = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \sigma_\phi^2 & 0 & 0 & 0 \\ 0 & \sigma_\phi^2 & 0 & 0 \\ 0 & 0 & \sigma_\phi^2 & 0 \\ 0 & 0 & 0 & \sigma_\phi^2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix}$$

$$= 2\sigma_\phi^2 I$$

We, therefore, conclude that if the measurement are uncorrelated, so are their single differences. The common variance of the single differences is twice that for the carrier phase measurements, as we had noted previously.

Now, on to the double differences. Taking satellite l as the reference, we can write a pair of double differences corresponding to satellites k, l and m as

$$\begin{bmatrix} \phi_{ur}^{kl} \\ \phi_{ur}^{ml} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} \phi_{ur}^{k} \\ \phi_{ur}^{l} \\ \phi_{ur}^{m} \end{bmatrix}$$

The covariance matrix for this pair of double differences is

$$\sum_{dd} = 2\sigma_\phi^2 \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

The double differences are correlated even if the original measurements are not. For n satellites ,the covariance matrix of double difference will be a $(n-1) \times (n-1)$ matrix.

3.5 The LAMBDA method

LAMBDA method (Least-squares AMBiguity Decorrelation Adjustment) is a procedure for integer ambiguity estimation in carrier phase measurements. After applying a decorrelating transformation, a sequential conditional adjustment is made upon the ambiguities. As a result, integer least-squares estimates for the ambiguities are obtained.

We reformulate the problem as one of estimating $\boldsymbol{\delta x}$, a 3-vector of real numbers, and $\boldsymbol{N}$, a vector of integers, which are solutions of

$$y = G \cdot \delta x + A \cdot N + \lambda \cdot \varepsilon$$

given that the covariance of ε is $\sum_{dd}$. In other words,find δx and N which minimize a revised cost function

$$\begin{aligned} c(\delta x, N) &= \left\| y - G \cdot \delta x - A \cdot N \right\|_W^2 \\ &= (y - G \cdot \delta x - A \cdot N)^T W (y - G \cdot \delta x - A \cdot N) \end{aligned}$$

which uses the inverse of the noise covariance matrix $W = \sum_{dd}^{-1}$ to give different weights to the contributions of the residuals. The algorithm implementation comprises 3 steps.

3.5.1 Step 1

The first step is obtain float solutions: disgregard the constraint of the ambiguities and obtain solutions for δx and N which minimizie the cost function. We now use the weighted least squares criterion to account for the correlations among the double differences.
In LAMBDA this is only the first step, which ends with the float solutions for the position and ambiguities and their covariance matrix . We now show the mathematical process of this first step.

Let the vector $\boldsymbol{\delta x}$ contains the three components of the baseline and the vector $\boldsymbol{N}$ contain ambiguities for the L1 frequency and possibly for the L2 frequency. The double differenced observations are collected in the vector $\boldsymbol{y}$.

$$[G \quad A]\begin{bmatrix}\delta x \\ N\end{bmatrix} = y + errors$$

We shall not not derive the weighted least squares estimator here,but for completeness, the solution is given here:

$$\hat{\boldsymbol{x}} = (\boldsymbol{B^T}\sum\boldsymbol{B})^{-1}\boldsymbol{B^T}\sum\boldsymbol{b}$$

where $\sum$ is the covariance matrix of the residual.
If

- $B = [G \quad A]$
- $\hat{x} = \begin{bmatrix}\widehat{\delta x} \\ \widehat{N}\end{bmatrix}$
- $b = y$

Applying this result to our method, the formal solution is

$$\begin{bmatrix}\widehat{\delta x}\\ \widehat{N}\end{bmatrix} = \left[\begin{bmatrix}G^T\\ A^T\end{bmatrix}\sum_{dd}[G \quad A]\right]^{-1}\cdot\begin{bmatrix}G^T\\ A^T\end{bmatrix}\cdot\sum_{dd}\cdot y$$

$$= \begin{bmatrix}G^T\sum_{dd}G & G^T\sum_{dd}A\\ A^T\sum_{dd}G & A^T\sum_{dd}A\end{bmatrix}^{-1}\cdot\begin{bmatrix}G^T\sum_{dd}\\ A^T\sum_{dd}\end{bmatrix}\cdot y$$

$$= \begin{bmatrix}Q_{\widehat{\delta x}} & Q_{\widehat{\delta x},\widehat{N}}\\ Q_{\widehat{\delta x},\widehat{N}}{}^T & Q_{\widehat{N}}\end{bmatrix}\cdot\begin{bmatrix}G^T\sum_{dd}\\ A^T\sum_{dd}\end{bmatrix}\cdot y \qquad (3.5)$$

Where

- $\begin{bmatrix}\widehat{\delta x}\\ \widehat{N}\end{bmatrix}$ is the float solution, $Q_{\widehat{\delta x}}$ and $Q_{\widehat{N}}$ the corresponding covariance matrices;
- $Q_{\widehat{\delta x},\widehat{N}}$ gives the cross-correlations between the two.

3.5.2 Step 2

In the second step we have to find the integer vector N which minimize the cost function

$$c(N) = \left(N - \widehat{N}\right)^T W_N \left(N - \widehat{N}\right) \qquad (3.6)$$

where $\widehat{N}$ is the float solution from step 1 and the weight matrix W_N is the inverse of its covariance matrix $W_N = Q_{\widehat{N}}{}^{-1}$.

Step 2 is the heart of LAMBDA method. The measure of distance of an integer vector $\widehat{N}$ is given by (3.6). The contour of points with a constant value of the cost function is an ellipse in two dimensions and an ellipsoid in higher

dimensions, centered at $\widehat{N}$. The search space is delimited by selecting the size of the ellipsoid to be searched via a parameter value $d > 0$. The inequality

$$\left(\boldsymbol{N} - \widehat{\boldsymbol{N}}\right)^{\boldsymbol{T}} \boldsymbol{W}_{\boldsymbol{N}} \left(\boldsymbol{N} - \widehat{\boldsymbol{N}}\right) \leq d$$

defines the integer vectors $\boldsymbol{N}$ which are candidates for the solution. The search space consists of the integer grid points inside an ellipsoid.
Clearly, this search space must be large enough to contain the right answer and small enough to be searched quickly.

In practice, the constant-cost ellipsoids can be very elongated, longer by orders of magnitude in one direction that in another, This is specially the case when the measurements are limited to a single epoch or only few epochs.
The result is that points which appear much farther away from $\widehat{\boldsymbol{N}}$ may have lower values of the cost function than those which appear nearby.
Brute force search, therefore, would be inefficient.
What's needed is a change of variable which would turn the elongated ellipsoid into a sphere so that the search can be limited to the neighbors of $\widehat{\boldsymbol{N}}$.

If the weight matrix $\boldsymbol{W}_{\boldsymbol{N}}$ is diagonal, the minimization of the cost function is trivial. The best estimate of the integer ambiguity is the corresponding float estimate rounded off to the nearest integer.
A diagonal $\boldsymbol{W}_{\boldsymbol{N}}$ would mean that the integer ambiguity estimates in the float solution are all uncorrelated. In general, $\boldsymbol{W}_{\boldsymbol{N}}$ would not be diagonal, and the objective of step 2 is to introduce a change of variables so that the resultant correlation matrix is diagonal.

$\boldsymbol{W_N}$ is a positive semi-definite matrix and it would appear that diagonalizing it would not be a problem. We can use the matrix of its eigenvectors to transform the variables. Actually, this approach will not work here because the transformation will not preserve the integer nature of ambiguities. We have to restrict the transformations to those that take integers into integers. Actually, the inverse transformation must do the same, too, so that we can find the solution of the original problem. The required transformation $\boldsymbol{Z}$ must satisfy the following conditions

- $\boldsymbol{Z}$ must have integer entries;
- $\boldsymbol{Z}$ must be invertible;
- $\boldsymbol{Z^{-1}}$ must have integer entries

These conditions ensure that there is a one-to-one relationship between integers in the original and transformed spaces.
Consider a hypothetical transformation $\boldsymbol{Z}$ in this restricted class of transformations which diagonalizes $\boldsymbol{W_N}$. Let

$$\boldsymbol{M = ZN} \quad \textit{and} \quad \boldsymbol{\widehat{M} = Z\widehat{N}}$$

The cost function in the transformed space is

$$\boldsymbol{\left(M - \widehat{M}\right)^T (Z^{-T} W_N Z^{-1})(M - \widehat{M})}$$

Since $\boldsymbol{Z^{-T} W_N Z^{-1}}$ is diagonal, we find the solutions for $\boldsymbol{M}$ right away by rounding off each element of $\boldsymbol{\widehat{M}}$**.** We now transform the problem back and find $\boldsymbol{N}$ from

$$\boldsymbol{N = Z^{-1}M}$$

LAMBDA would be a simple algorithm if $\boldsymbol{W_N}$ could be diagonalized using our restricted class of transformations. Unfortunately, that's almost never the case and the integer ambiguities are not decorrelated fully. LAMBDA involves many subtle steps to transform $\boldsymbol{W_N}$ into a matrix that is nearly diagonal as possible [Teunissen(1996)].

For a mathematical procedure of the method see[5].

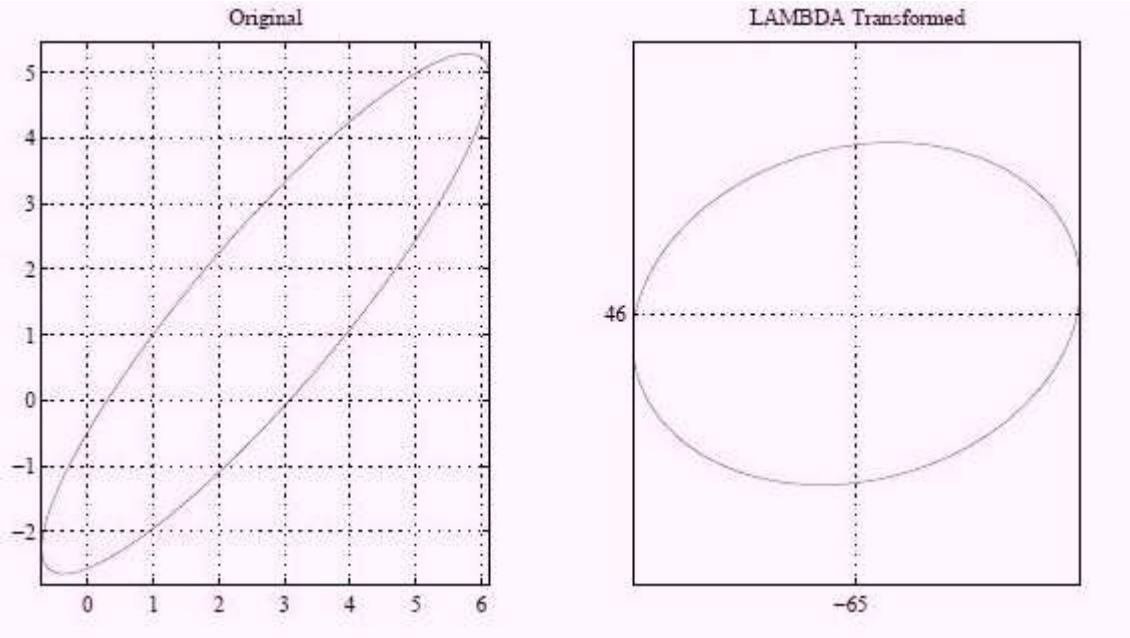

Figure 9: The confidence ellipse for two double differences

[5] Paul de Jonge and Christian Tiberius(1996). The LAMBDA method for integer ambiguity estimation:implementation aspects.

3.5.3 Step 3

After the integer solution $\breve{N}$ is found, we substitute it for $\hat{N}$. Consequently, the solution $\widehat{\delta x}$ changes to $\breve{\delta x}$. In order to determine $\breve{\delta x}$ we multiply the lower block in (3.5) by $Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}$, and subtract from the upper block.
Calling

$$R = G^T\sum_{dd}y \qquad L = A^T\sum_{dd}y$$

equation (3.5) became:

1. $\widehat{\delta x} = Q_{\widehat{\delta x}}R + Q_{\widehat{\delta x},\hat{N}}L$

   $\hat{N} = Q_{\widehat{\delta x},\hat{N}}{}^T R + Q_{\hat{N}}L$
2. $\widehat{\delta x} = Q_{\widehat{\delta x}}R + Q_{\widehat{\delta x},\hat{N}}L$

   $Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}\hat{N} = Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}Q_{\widehat{\delta x},\hat{N}}{}^T R + Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}Q_{\hat{N}}L$

Remebering that $Q_{\hat{N}}^{-1}Q_{\hat{N}} = I$ and subtracting ,we obtained

$$\widehat{\delta x} - Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}\hat{N} = \left(Q_{\widehat{\delta x}} - Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}Q_{\widehat{\delta x},\hat{N}}{}^T\right)R$$

Rewriting the expression for $\breve{N}$ and $\breve{\delta x}$

$$\breve{\delta x} - Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}\breve{N} = \left(Q_{\widehat{\delta x}} - Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}Q_{\widehat{\delta x},\hat{N}}{}^T\right)R$$

Subtracting each other,

$$\widehat{\delta x} - Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}\hat{N} - \left(\breve{\delta x} - Q_{\widehat{\delta x},\hat{N}}Q_{\hat{N}}^{-1}\breve{N}\right) = 0$$

$$\widetilde{\boldsymbol{\delta x}} = \widehat{\boldsymbol{\delta x}} - \boldsymbol{Q}_{\widehat{\boldsymbol{\delta x}},\widehat{\boldsymbol{N}}}\boldsymbol{Q}_{\widehat{\boldsymbol{N}}}^{-\mathbf{1}}(\widehat{\boldsymbol{N}} - \breve{\boldsymbol{N}})$$

The right side is known, and $\widetilde{\delta x}$ is quickly found.

CHAPTER 4

EXPERIMENTAL TEST

4.1 Setup description and equipment used

The field tests were done using one kind of receiver. The time-difference method was evaluated using a commercial-grade ProPak-V3 receiver. Double-differenced solutions were computed from data measured by a pair of dual-frequency capable NovAtel ProPak-V3 receivers .

Figure 10: Novatel ProPak-V3

NovAtel's ProPak-V3 is a durable, triple-frequency GNSS receiver that tracks GPS + GLONASS as well as L-Band and SBAS. When combined with one of NovAtel’s rugged GPS-700 series antennas, the ProPak-V3 provides superior tracking performance, positioning accuracy and reliability. It also supports USB communications and Inertial Measurement Unit (IMU) technology .We take up next the ProPak V-3 main aspects.

Features

- L1, L2, L5, L-Band and SBAS tracking
- GPS only for GPS + GLONASS
- Rt-2™, RT20®, ALIGN®, API, GL1DE® and 50 Hz firmware options
- Aluminum enclosure

Benefits

- Multi-constellation tracking yields higher solution availability and reliability
- Durable metal enclosure ensures reliable positioning in harsh environments and EMI conditions
- Same easy-to-use interface as the ProPak-G2 plus
- Upgradeable receiver firmware ensures easy upgrading to future signals as soon as they are available
- Supported by industry’s highest level of customer service

Attributes

System type	Enclosed	
General info	Length (mm)	185.00
	Width/diameter (mm)	160.00
	Height (mm)	71.00
	Weight (g)	1000.00
	Typical power consumption (W)	2.80
Constellation	GPS	
	GLONASS	
Tracking	Max Num of Frequency	Triple
	L-Band	
	SBAS	
Performance	Accuracy	RMS
	Single Point L1	1.5 m
	Single Point L1/L2	1.2 m
	SBAS	0.6 m
	DGPS	0.4 m

Table 1: Novatel attributes

Measurement Precision

	GPS	GLO
L1 C/A Code	4 cm	15 cm
L1 Carrier Phase	0.5 mm	1.5 mm
L2 P(Y) Code	8 cm	8 cm
L2 Carrier Phase	1.0 mm	1.5 mm

Advanced multipath mitigation

The ProPak-V3 provides superior multipath rejection close to the antenna and in high multipath environments.

Supports NovAtel SPAN Technology

A single cable from the ProPak-V3 to an IMU creates a robust GNSS/INS system that provides continuous 3D position, velocity and attitude, even during periods when satellite signals are blocked. The system delivers measurements at 100 Hz data rate.

The test was a static one, in which both receivers remain stationary at precisely known positions, to verify the quality of the proposed algorithm.
Data, as said before, were collected by two ProPak V-3 receivers, with 1 Hz of sampling rate.
Base receiver was placed on a reference landmark with coordinates 44° 12’0.36’’N, 12°3'45.72"E and [4478920.0484 m; 957152.4415 m; 4424113.3286 m] in ECEF system;

The user receiver, has coordinates 44°11’58.92’’N , 12° 3'49.68"E, and [4478933.3172 m; 957241.0112 m; 4424076.2166 m] in ECEF system;

Differencing the two ECEF coordinates of the user and the receiver we can estimate the baseline ECEF coordinates between the two.

$$\begin{bmatrix} 4478933.3172\ \text{m} \\ 957241.0112\ \text{m} \\ 4424076.2166\ \text{m} \end{bmatrix} - \begin{bmatrix} 4478920.0484\ \text{m} \\ 957152.4415\ \text{m} \\ 4424113.3286\ \text{m} \end{bmatrix} = \begin{bmatrix} 13.2688\ \text{m} \\ 88.5697\ \text{m} \\ -37.1119\ \text{m} \end{bmatrix}$$

The norm of the baseline is the distance ,in meter, between the two receivers.

$$norm\left(\begin{bmatrix} 13.2688\ \text{m} \\ 88.5697\ \text{m} \\ -37.1119\ \text{m} \end{bmatrix}\right) = 96.9430\ m$$

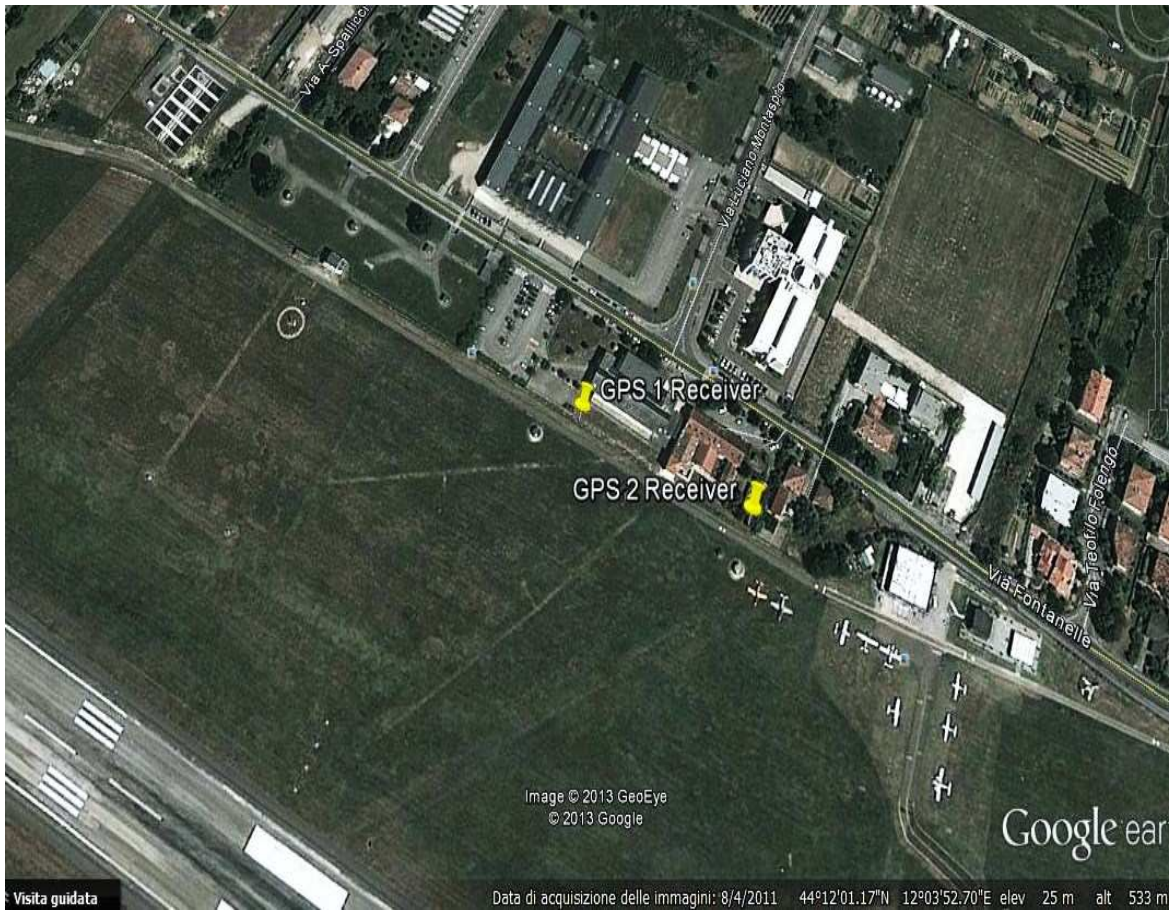

Figure 11: Google Earth image of the receivers position

The solution was attained through a least-squares method with a priori information . The standard deviation for phase measurement was set to 0.001 m. Baseline components were calculated epoch by epoch with LAMBDA methodology, described in Chapter 3, using float ambiguity values from measurement processing in each epoch by least-squares.

4.2 Test analysis

The data were collected in tridimensional matrix, called OBS2 (data for the GPS receiver 2) and OBS1(data for the GPS receiver 1). The characteristic of the observation matrix are

$$OBS = time \times satellites \times parameters$$

Where

- time is the number of epochs of samples;
- satellites is the number of GPS satellites (32);
- parameters are 11 data recording by the GPS receiver: in our tests, the only parameters that we use are the 3^{rd} and the 5^{th} ,respectively, the phase on the L1 frequency and the phase on the L2 frequency;

In our test the observations matrix were

$$OBS1 = 1117 \times 32 \times 11$$

$$OBS2 = 1117 \times 32 \times 11$$

Clearly, at the moment ,not all the 32 satellites were in view, so we try to built up a 2D matrix for the carrier phase. In fact, if the time column has a 0 on the first epoch, the satellite is clearly not in view. The result was that both the receivers were observed 10 satellites. We eliminate (see Matlab code) the zero value sample epochs, and obtain four phase matrix, two for the L1, and two for the L2,for both receivers. The first row is the number of the satellite.

1	3	6	11	14
1.1126e+08	1.2077e+08	1.2640e+08	1.0631e+08	1.1637e+08
1.1126e+08	1.2077e+08	1.2640e+08	1.0631e+08	1.1637e+08
1.1125e+08	1.2078e+08	1.2640e+08	1.0631e+08	1.1637e+08
1.1125e+08	1.2078e+08	1.2641e+08	1.0631e+08	1.1637e+08
1.1125e+08	1.2078e+08	1.2641e+08	1.0631e+08	1.1637e+08
1.1125e+08	1.2079e+08	1.2641e+08	1.0631e+08	1.1638e+08
1.1125e+08	1.2079e+08	1.2642e+08	1.0631e+08	1.1638e+08
1.1124e+08	1.2080e+08	1.2642e+08	1.0630e+08	1.1638e+08
1.1124e+08	1.2080e+08	1.2643e+08	1.0630e+08	1.1638e+08

Table 2: First receiver phases on the L1 frequency for the first 5 satellites (9 epochs)

19	20	22	28	32
1.1176e+08	1.2000e+08	1.2208e+08	1.2694e+08	1.1044e+08
1.1177e+08	1.2000e+08	1.2208e+08	1.2694e+08	1.1044e+08
1.1177e+08	1.2000e+08	1.2208e+08	1.2693e+08	1.1043e+08
1.1177e+08	1.2000e+08	1.2209e+08	1.2693e+08	1.1043e+08
1.1177e+08	1.1999e+08	1.2209e+08	1.2693e+08	1.1043e+08
1.1178e+08	1.1999e+08	1.2209e+08	1.2693e+08	1.1043e+08
1.1178e+08	1.1999e+08	1.2209e+08	1.2693e+08	1.1043e+08
1.1178e+08	1.1998e+08	1.2210e+08	1.2693e+08	1.1042e+08
1.1178e+08	1.1998e+08	1.2210e+08	1.2693e+08	1.1042e+08

Table 3: First receiver phases on the L1 frequency for the others 5 satellites (9 epochs)

Dual-frequency capability is not the sole advantage of high-end receivers. The probably largest tradeoff for cheaper hardware cost, the receiver clock, causes problems. Although the receiver clock biases cancel during double differencing, the measurement epochs are not necessarily perfectly synchronized between receivers. Naturally the receiver's sensitivity and ability to maintain phase lock affect RTK performance. Frequent cycle slips are of course unfavorable for carrier phase positioning.

Only the satellites tracked by both the reference and rover receivers can be used in differential positioning, so the receiver should be able to acquire and track as many satellites as possible.

As we said in Chapter 3, we try to choose the satellite with the highest elevation angle in order to as the base satellite for the double difference, because the signal coming from higher elevations travel a shorter time through the

atmosphere and therefore are less plagued by ionospheric and tropospheric errors. It is also less prone to multipath.

We transform the coordinates of the satellites into NED system coordinates, Nord, East and Down, and then calculate the angle of elevation, also called the altitude, determined by first finding the compass bearing on the horizon relative to true north, and then measuring the angle between that point and the object, from the reference frame of the observer. Elevation angles for objects above the horizon range from 0 (on the horizon) up to 90 degrees (at the zenith). Sometimes the range of the elevation coordinate is extended downward from the horizon to -90 degrees (the nadir). This is useful when the observer is located at some distance above the surface, such as in an aircraft. Figure 12 shows the elevation angle.

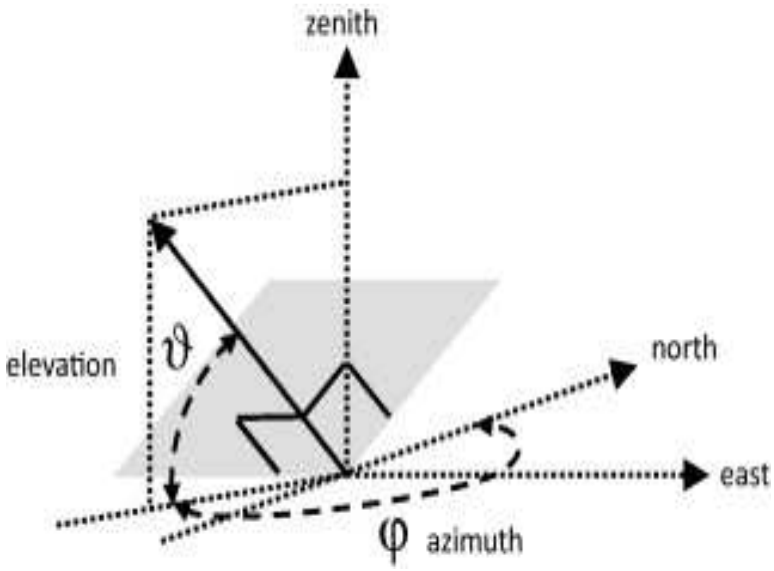

Figure 12: Elevation angle

Elevation angle ϑ can be calculated (in degrees) as:

$$\vartheta^{\circ} = \arcsin\left(\frac{-NED_{down}}{norm(NED)}\right) \cdot \frac{180}{\pi}$$

Where:

- arcsin is the inverse of the sin function;
- NED_{down} is the third component of the NED coordinates of a satellite, or rather the down component;
- $norm(NED)$ is the norm of the NED coordinates of a satellite;

In our test the results are:

Theta (ϑ°)	Satellite
51.435432679421858	1
28.993728674215742	3
16.686907535583959	6
72.060670142940268	11
39.828897730343051	14
57.011048271965080	19
28.310027015568377	20
25.850776498545546	22
18.522182305961120	28
55.947753125882187	32

Table 4: Elevation angle of the satellites

Since the satellite 11 has the major elevation angle, we have taken it as the reference one for the double difference implementation.

4.3 Experimental results

The sampling rate was 1 Hz in all measurements, which suffices for detecting such motion. For more accurate trajectory reconstruction, a higher sampling frequency would be beneficial.

All computations were carried out in post-processing mode. This enabled a priori verification of the data to be cycle slip free. No runtime cycle slip detection was attempted. The verification was done by hand and was not rigorous, so basically small cycle slips could be present in the data. However, no evident a posteriori signs of cycle slips were observed.

Using two separate antennas decorrelates multipath. In the figure below is show the difference between the computed position and the calculate one(baseline slightly less than 100 meters), and the errors in 100 seconds of sample. The antennas were not located in very multipath-prone locations, so the resulting multipath error is not expected to be large.. The baseline solutions do not coincide with the reference baseline computed but the discrepancies are in centimeter level in the horizontal plane and a couple of centimeters in the vertical direction.

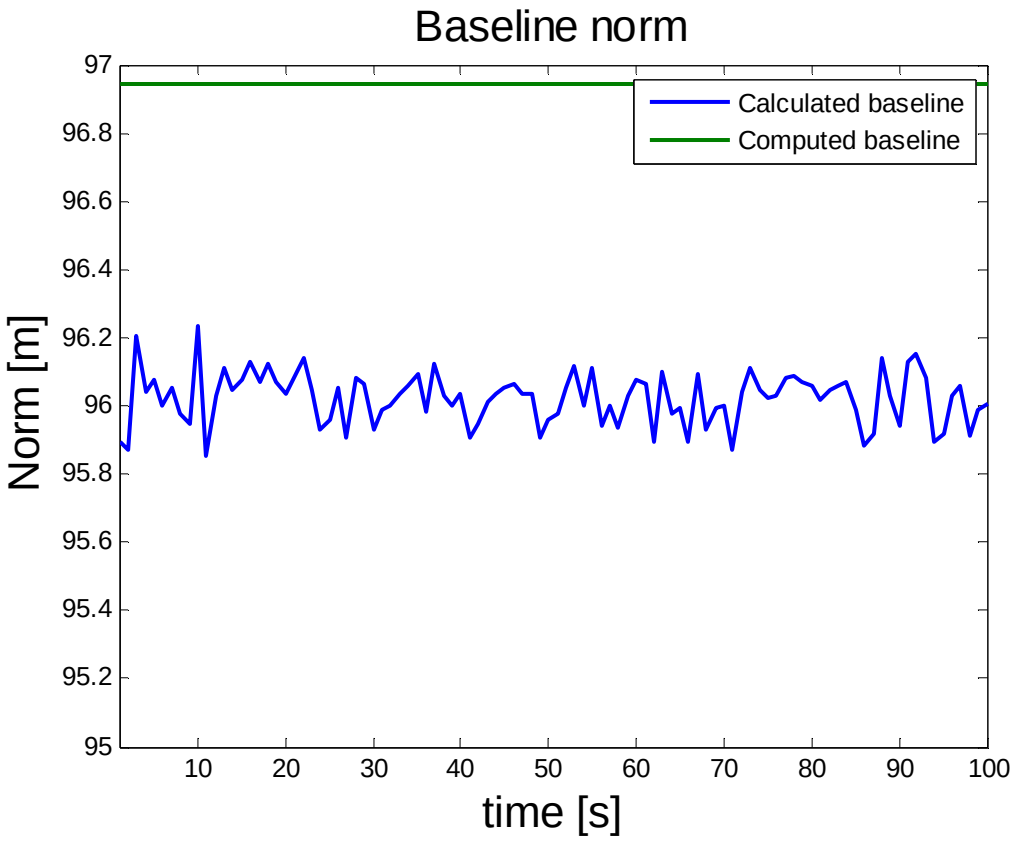

Figure 13: Computed and calculated baseline in 100 seconds

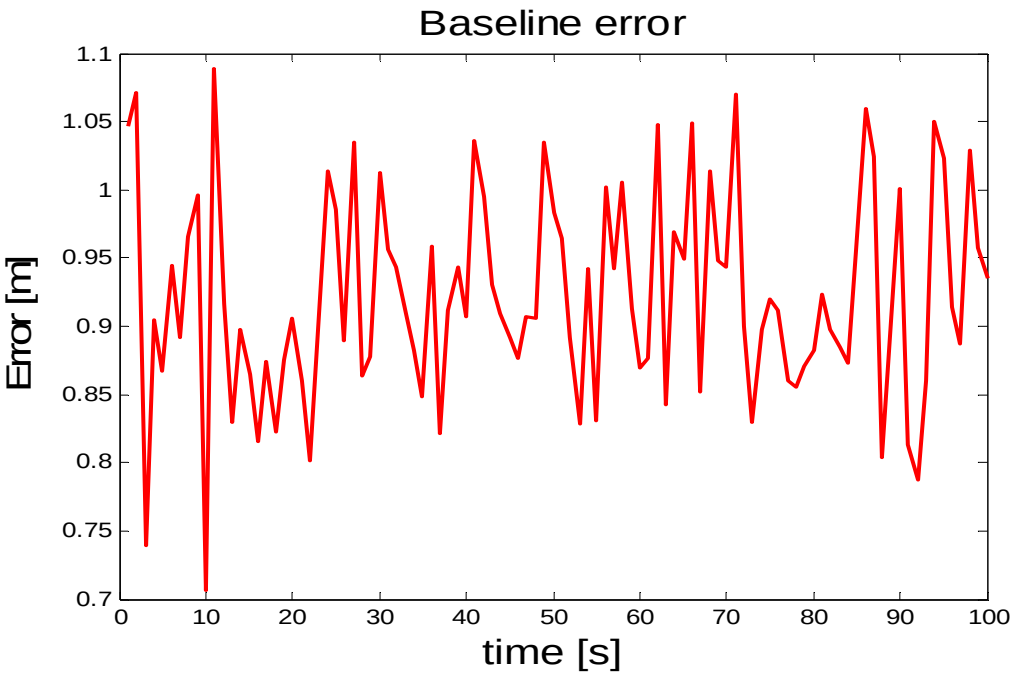

Figure 14: Baseline component error for static test

Figure 14 shows the evolution of the baseline error in 100 seconds. The average error is 0,92083 m, and could be consider a good estimate for our purpose, considering the Pro Pak features and our simplified model. (No cycle slips, no multipath errors).

The errors of the NED component, Nord, Est and Down are shown below.

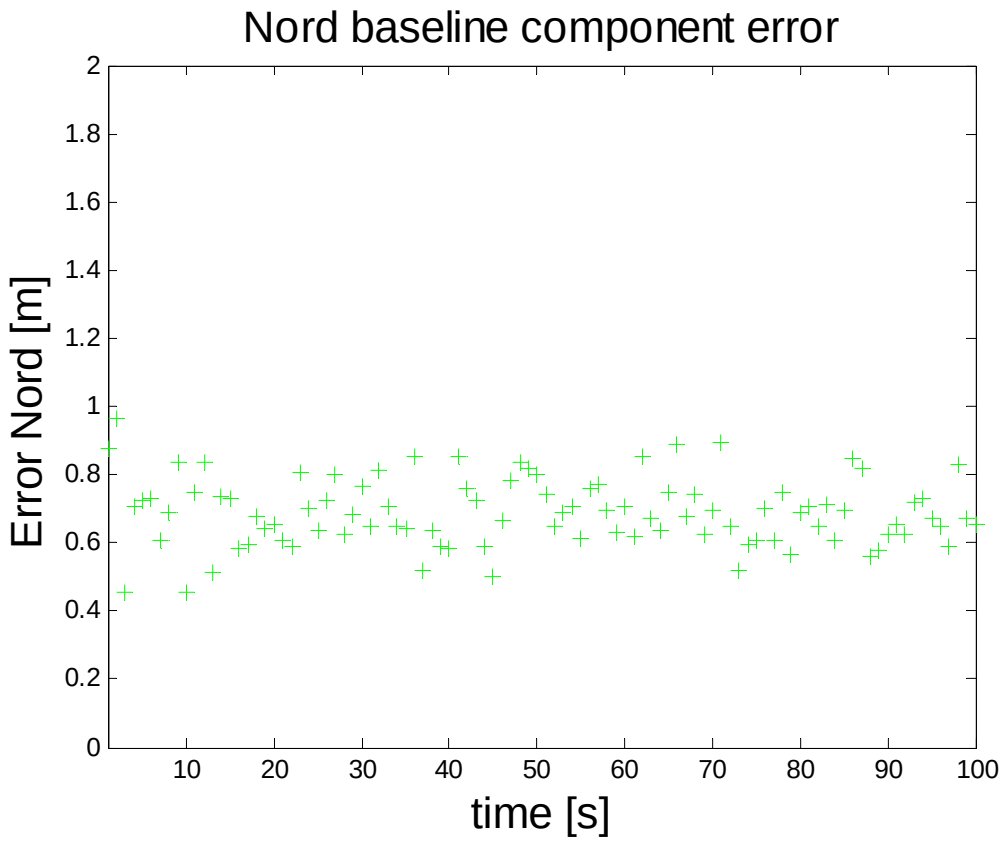

Figure 15: Nord component error

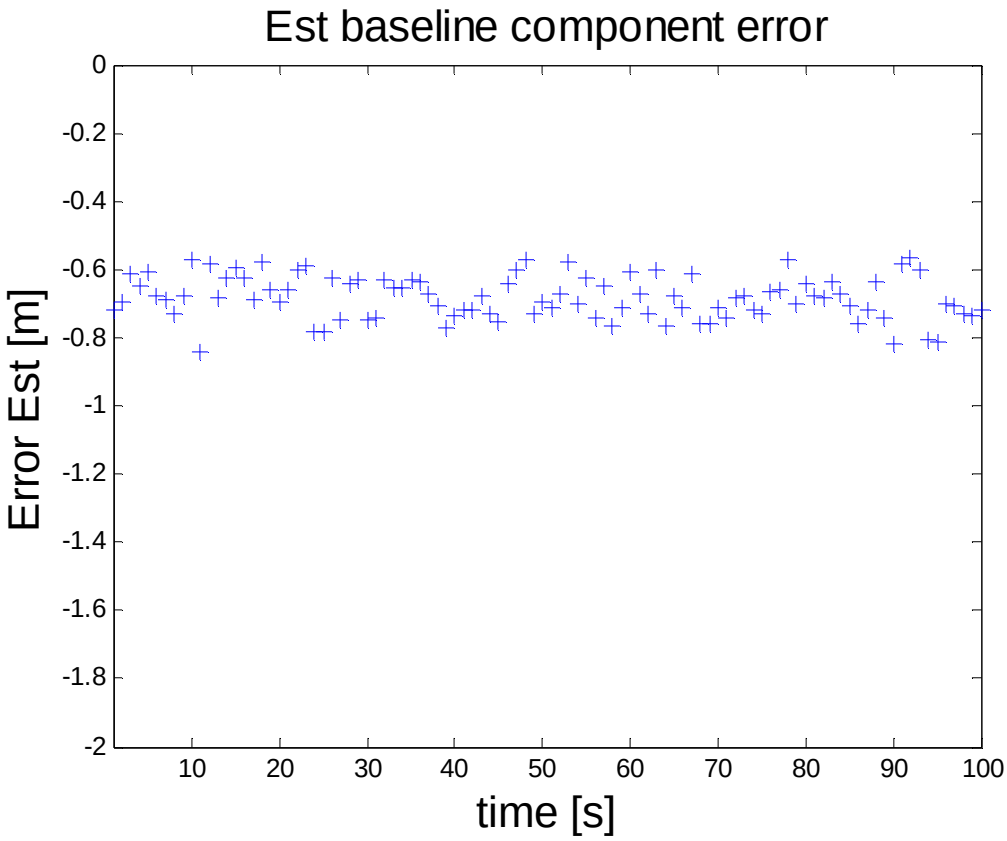

Figure 16: Est component error

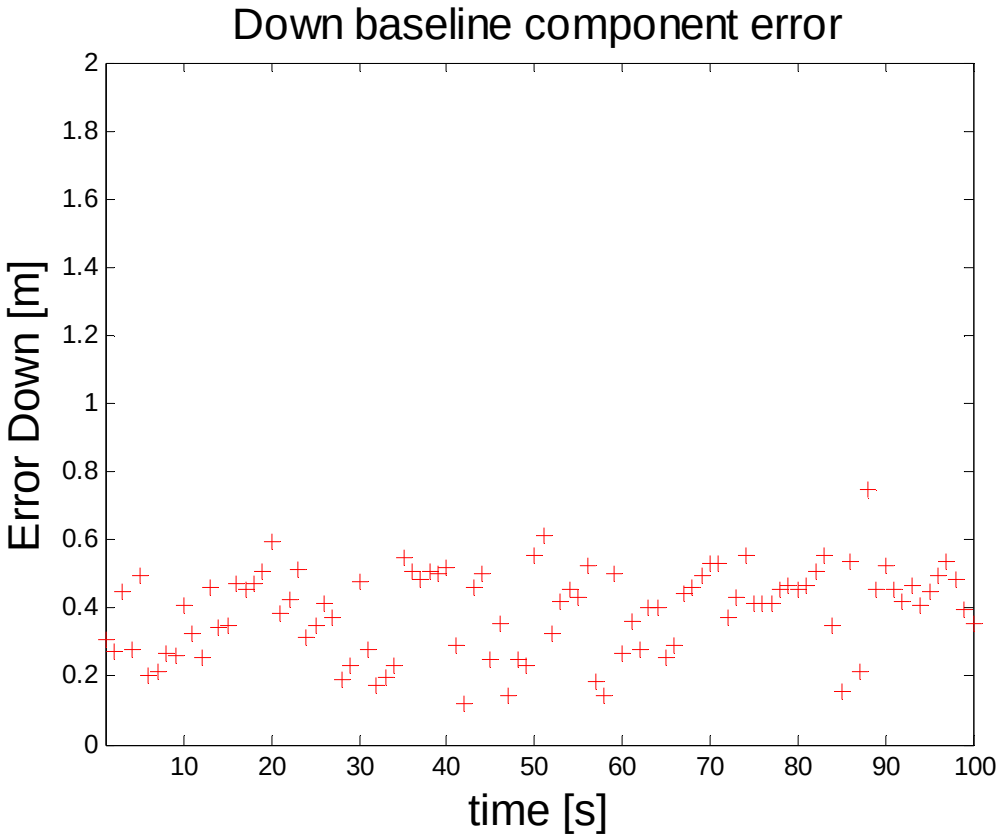

Figure 17: Down component error

Figures show that in 100 seconds of samples, the medium errors for the 3 components are:

Nord medium error [m]	0,689724017
Est medium error [m]	-0,6824275994
Down medium error [m]	0,3941270872

Tabella 5: Medium error component

The next figure show the grapich of the Nord and the Est component, and the difference between the computed and the calculated one.

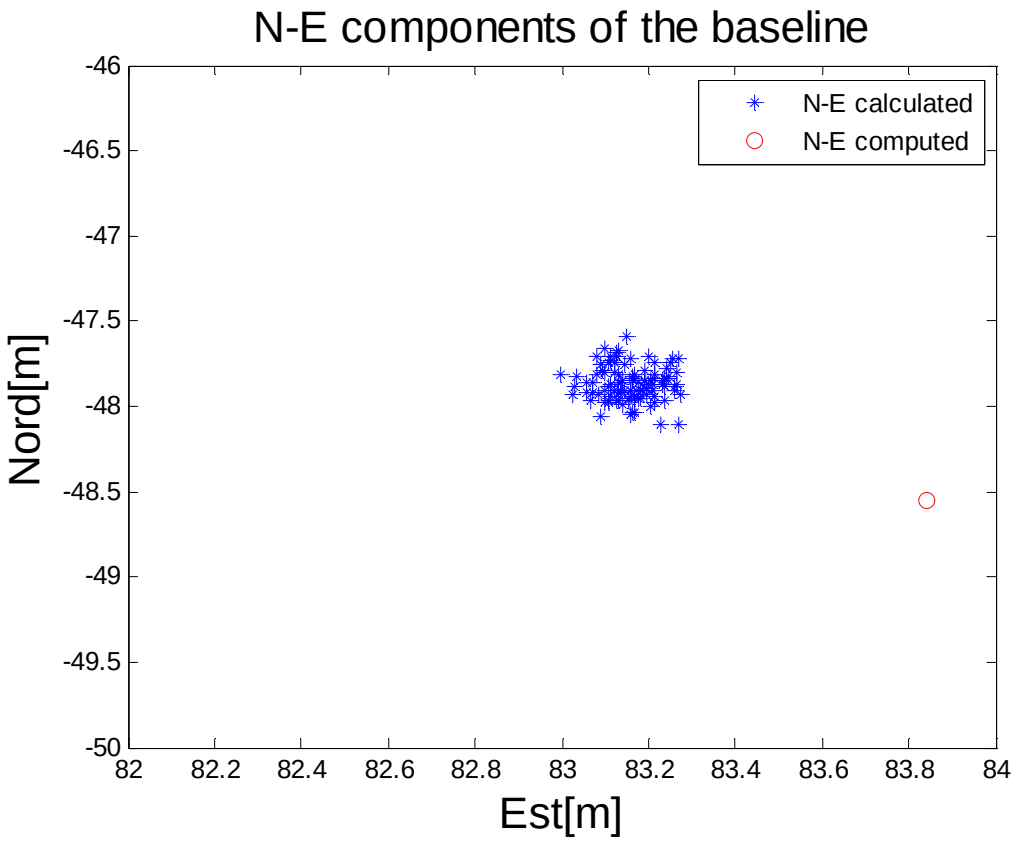

Figure 18: Computed and calculated N-E grapich

Another important parameter to be estimated is the standard deviation of the baseline. This parameter will give us the goodness of the valuation.

In statistics and probability theory, **RMS**, root mean square (represented by the symbol sigma, σ) shows how much variation or "dispersion" exists from the average (mean, or expected value). A low RMS indicates that the data points tend to be very close to the mean; high RMS indicates that the data points are spread out over a large range of values.

The RMS of a random variable, statistical population, data set, or probability distribution is the square the square root of its variance. It is algebraically simpler through practically less robust than the average absolute deviation, and a useful property of standard deviation is that it, unlike variance, it is expressed in the same units as the data.

In the case where X takes random values from a finite data set x_1, x_2,….x_N, with each value having the same probability, the RMS is

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2} \quad \text{where } \mu = \frac{1}{N}\sum_{i=1}^{N} x_i$$

In our test, RMS results $\sigma = 0{,}0763$ m. This means that most baseline solutions have a length within 0,0763 m in of the mean (96,022 m).

CHAPTER 5

CONCLUSIONS

In this thesis the positioning techniques were carried out through phase measurements processing, using the LAMBDA approach.
Double-differenced baseline solutions were precise and mostly consistent with the reference baselines. Only high-quality receivers were used in these measurements. Using low-cost receivers for RTK measurements as well would have been natural, but could not be carried out due to hardware problems. Thus, the effect of asynchronous measurements and higher noise were not studied.
The LAMBDA-method is capable of correctly estimating the integer ambiguities very fast and efficiently. The results in this paper show that in order to successfully estimate the integer ambiguities, data of only a short time span are required. The method therefore enables instantaneous precise navigation and very rapid static surveying.

The double-difference positioning implementation was also post-processing only and for stationary baselines. Therefore, the term “real-time kinematic” does not directly apply to it. However, neither data from future epochs nor post-processed precise ephemerides or other corrections were used, so in principle the algorithm would have worked in real-time as well.

A relatively high number of satellites was needed to obtain a reliable integer ambiguity fix. The baseline computations were done using ten satellites. This may become a problem if the rover and reference receivers do not have that

many common satellites in track. However, most receivers have at least 12 tracking channels which should be enough for relatively short baselines, but with the number of available GPS's and satellites growing in the future, more channels may be necessary to make sure both receivers choose the same satellites to track.

The test results show that centimeter-level precision is attainable using double frequency measurements only with relatively simple algorithms.

As no pre-surveyed reference baselines were available, absolute accuracies could not be assessed rigorously.

The discrepancies from the reference computations imply that some inaccuracy is present, but in most cases the errors were in centimeter magnitude.

Once the measurement information relaying issue is solved by some standard, RTK is expected to bring the positioning performance attained using consumer-level equipment to centimeter level.

Logically, further developments consider:

(i) include better dynamics in the estimation process;
(ii) a better filter tunning for carrier phase measurements;
(iii) add other measurements combination, widelane combination;
(iv) validate ambiguities after resolving;
(v) cycle slips detection and correction;

Acknowledgements

To begin with, I would like to thank my family for their love and support for all this years. Nothing would have been possible without them. I would also like to thank professor Matteo Zanzi and his research group for inspiring and encouraging me with their advice during the whole work with this dissertation. Thanks for your support and availability.

Bibliography

- Tiberius, de Jonge , “ The LAMBDA method for integer ambiguity estimation: implementation aspects”, Delft University of Technology, 1996.
- Teunissen, P. J. G., “Least-squares estimation of the integer GPS ambiguities”, Proceedings of IAG GENERAL MEETING, Beijing, China, IAG, 1993.
- Teunissen, P. J. G., “A new method for fast carrier phase ambiguity estimation", Proceedings of IEEE POSITION, LOCATION AND NAVIGATION SYMPOSIUM, april 11-15, Las Vegas, 1994.
- Strang, G. and Borre, K., “Linear Algebra, Geodesy, and GPS”, Cambridge Press, Wellesley, 1997.
- Geoffrey Blewitt,” Basics of the GPS technique: Observation Equations", Department of Geomatics, University of Newcastle.
- Gene H. Golub, Charles F. Van Loan , “Matrix computations: third edition”.
- http://kom.aau.dk/~borre/easy/

Appendix

Following, the Matlab file of the software.

```
clc
clear
close all
format long

%%%%%% Script di definizione variabili %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

GPSdefns;      % Definizione dei parametri numerici

%%%%%% Analisi files di testo
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

reply = input('Vuoi ricavare i dati dai files di testo originali? [y/n] ',
's'); % Richiesta per l'estrazione di nuovi dati dai files di testo
if reply == 'y'


% Nominativi files di testo RANGE+EPHEM da leggere
fileinput = ['prova_s1.txt'; 'prova_s2.txt'];

% Creazione files di testo SOLO RANGE da riempire
range = ['range2.txt'; 'range3.txt'];

% Creazione files di testo SOLO EPHEM da riempire
gpsephem = ['gpsephem2.txt'; 'gpsephem3.txt'];

% Inserimento dei dati forniti da ogni stazione (3) nei corrispondenti
files RANGE/EPHEM (2x3)

for i=1:2
    fileinput_temp = fileinput(i,:);
    range_temp = range(i,:);
    gpsephem_temp = gpsephem(i,:);
    Text_Split(fileinput_temp,range_temp,gpsephem_temp);
end;

%%%%% Creazione matrici range e ephem %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Lettura dati dal file RANGE + EPHEM delle stazioni  e creazione di una
matrice 3D (t_range x 32 x 11)
range2 = 'range2.txt';
RANGE2 = Lettura_OBS(range2);
```

```
range3 = 'range3.txt';
RANGE3 = Lettura_OBS(range3);
ephem2 = 'gpsephem2.txt';
EPHEM2 = Lettura_EPHEM(ephem2);
ephem3 = 'gpsephem3.txt';
EPHEM3 = Lettura_EPHEM(ephem3);

save('range_ephem.mat', 'EPHEM2', 'EPHEM3', 'RANGE2', 'RANGE3');

else load('range_ephem.mat');

end;

%%%%%% Matrici comuni %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

% Individuazione delle matrici RANGE comuni (in base ai tempi assoluti di
campionamento) fra le tre stazioni
[comune1, comune2, comune3] = Matrici_Comuni(RANGE2, RANGE2, RANGE3);

%%%%%% Elaborazione posizione satelliti e sincronizzazione con OBS %%%%%%%%

% Calcolo posizioni satelliti rilevati dalle stazione  e sincronizzazione
con misurazioni di pseudodistanza
[OBS2, XYZ_SAT2] = Info_Satelliti(comune2,EPHEM2);
[OBS1, XYZ_SAT1] = Info_Satelliti(comune3,EPHEM3);


 [tempi,sat,parametri]=size(comune3);

    %Stazione 3: terrazzo ingegneria( coordinate R1);
    %Stazione 2: locale pompe (coordinate R2);

    %Matrice contenente le misure di fase misurate sulla portante L1 per la
    stazione 3

    z=1;
    for j=1:sat
         if (comune3(1,j,3)~=0)

            FASI1D(1,z)=j;
            FASI1D(2:tempi+1,z)=comune3(1:tempi,j,3);
            z=z+1;

         else
              end
    end

    %Matrice contenente le misure di fase misurate sulla portante L1 per la
    stazione 2
```

```
    l=1;
    for j=1:sat
        if (comune2(1,j,3)~=0)

           FASI2D(1,l)=j;
           FASI2D(2:tempi+1,l)=comune2(1:tempi,j,3);
           l=l+1;

        else
             end
    end



%Matrice contenente le misure di fase misurate sulla portante L2 per la
stazione 3



   z=1;
   for j=1:sat
        if (comune3(1,j,5)~=0)

           FASI1D_2(1,z)=j;
           FASI1D_2(2:tempi+1,z)=comune3(1:tempi,j,5);
           z=z+1;

        else
             end
   end

%Matrice contenente le misure di fase misurate sulla portante L2 per la
stazione 2

   z=1;
   for j=1:sat
        if (comune2(1,j,5)~=0)

           FASI2D_2(1,z)=j;
           FASI2D_2(2:tempi+1,z)=comune2(1:tempi,j,5);
           z=z+1;

        else
             end
   end



   m=1;
```

```
%Calcolo posizioni dei satelliti visti dalla stazione 2 ed eliminazione
tempi nulli

       for j=1:32
           if (XYZ_SAT2(1,j,1)~=0)
               XYZ_SAT2D_1(1:tempi,m,1:3)=XYZ_SAT2(1:tempi,j,1:3);
               m=m+1;

         else
         end
       end

       i=1;
       k=1;

      for i=1:tempi

          if (XYZ_SAT2D_1(i,:,1)~=0)
              XYZ_SAT2D(k,:,1:3)=XYZ_SAT2D_1(i,:,1:3);
              k=k+1;
          else


          end
      end



%Calcolo posizione dei satelliti visti dalla stazione 3 ed eliminazione
tempi nulli

    m=1;
    for j=1:32
           if (XYZ_SAT1(1,j,1)~=0)
               XYZ_SAT1D_1(1:tempi,m,1:3)=XYZ_SAT1(1:tempi,j,1:3);
               m=m+1;

         else
         end
       end

       i=1;
       k=1;

      for i=1:tempi

          if (XYZ_SAT1D_1(i,:,1)~=0)
              XYZ_SAT1D(k,:,1:3)=XYZ_SAT1D_1(i,:,1:3);
              k=k+1;
          else
```

```
        end
    end


nsat=size(FASI2D,2);
R1=[4478920.0484;957152.4415;4424113.3286];
R2vera = [4478933.3172; 957241.0112; 4424076.2166];
R2=R1; Posizione iniziale del ricevitore R2 per la linearizzazione
theta=zeros(nsat,1);

%Calcolo del satellite con la maggior elevazione e spostamento dei dati
%di quest ultimo sulla prima colonna

for t=1:nsat

pos_sat(1,1)=XYZ_SAT1D(1,t,1);
pos_sat(2,1)=XYZ_SAT1D(1,t,2);
pos_sat(3,1)=XYZ_SAT1D(1,t,3);
NED_sat= Trasf_ECEF_to_NED(pos_sat,R1,1);
theta(t,1)=asin(-NED_sat(3,1)/norm(NED_sat))*180/pi;
end

for i=1:nsat

    FASI1D_mod(:,i)=FASI1D(:,i);
    FASI2D_mod(:,i)=FASI2D(:,i);
    FASI1D_mod2(:,i)=FASI1D_2(:,i);
    FASI2D_mod2(:,i)=FASI2D_2(:,i);
    XYZ_SAT2D_mod(:,i,:)=XYZ_SAT2D(:,i,:);

end

FASI1D_mod(:,1)=FASI1D(:,4);
FASI1D_mod(:,4)=FASI1D(:,1);

FASI2D_mod(:,1)=FASI2D(:,4);
FASI2D_mod(:,4)=FASI2D(:,1);

FASI1D_mod2(:,1)=FASI1D_2(:,4);
FASI1D_mod2(:,4)=FASI1D_2(:,1);

FASI2D_mod2(:,1)=FASI2D_2(:,4);
FASI2D_mod2(:,4)=FASI2D_2(:,1);

XYZ_SAT2D_mod(:,1,:)=XYZ_SAT2D(:,4,:);
XYZ_SAT2D_mod(:,4,:)=XYZ_SAT2D(:,1,:);
```

```
%Procedura per il calcolo della posizione col metodo delle doppie
differenze

   DDphi_L1=zeros(nsat-1,100);
   DDphi_L2=zeros(nsat-1,100);
   S1_R1=zeros(3,100);
   S1_R2=zeros(3,100);


   for q=1:100
     for t=2:nsat
       DDphi_L1(t-1,q)=(FASI2D_mod(q+1,t)-FASI2D_mod(q+1,1)-
       FASI1D_mod(q+1,t)+FASI1D_mod(q+1,1))*lambda_1;
       DDphi_L2(t-1,q)=(FASI2D_mod2(q+1,t)-FASI2D_mod2(q+1,1)-
       FASI1D_mod2(q+1,t)+FASI1D_mod2(q+1,1))*lambda_2;
     end


   S1_R1(1,q)=XYZ_SAT2D_mod(q,1,1);
   S1_R1(2,q)=XYZ_SAT2D_mod(q,1,2);
   S1_R1(3,q)=XYZ_SAT2D_mod(q,1,3);


   S1_R2(1,q)=XYZ_SAT2D_mod(q,1,1);
   S1_R2(2,q)=XYZ_SAT2D_mod(q,1,2);
   S1_R2(3,q)=XYZ_SAT2D_mod(q,1,3);

   end


   G1=zeros(nsat-1,3);
   tt=1;
   Sk_R1=zeros(3,100);
   Sk_R2=zeros(3,100);


   m1 = nsat-1;
   N = zeros(3+2*m1,3+2*m1);
   rs=zeros(3+2*m1,100);
   %X=zeros(3+2*m1,100);
   DDPhi=zeros(2*(nsat-1),100);
   DDRho=zeros(2*(nsat-1),100);
   X=zeros(3+2*(nsat-1),100);
   for q=1:100

   for t=2:nsat

       for tt=1:3
```

```
    Sk_R2(1,q)=XYZ_SAT2D_mod(q,t,1);
    Sk_R2(2,q)=XYZ_SAT2D_mod(q,t,2);
    Sk_R2(3,q)=XYZ_SAT2D_mod(q,t,3);

    G1(t-1,tt)=[((-R2(tt)+XYZ_SAT2D_mod(q,t,tt))/norm(R2-Sk_R2(:,q)))-(
    (-R2(tt)+XYZ_SAT2D_mod(q,1,tt))/norm(R2-S1_R2(:,q)))];

    DDrho(t-1,q)=norm(S1_R2(:,q)-R1)-norm(S1_R2(:,q)-R1)-
    norm(Sk_R2(:,q)-R1)+norm(Sk_R2(:,q)-R1);
    end
end


 % Creazione matrice di covarianza delle doppie differenze

 D = [ones(m1,1) -eye(m1) -ones(m1,1) eye(m1)];
 Sigma =D*D';
 A_modi = eye(m1);
 A_modi(:,1)=-ones(m1,1);


 A_aug = [G1 lambda_1*A_modi 0*eye(m1);G1 0*eye(m1) A_modi*lambda_2];
 DDPhi(:,q)=[DDphi_L1(:,q);DDphi_L2(:,q)];
 DDRho(:,q)=[DDrho(:,q);DDrho(:,q)];

 N = A_aug'*kron(eye(2),Sigma)*A_aug;
 rs(:,q) = rs(:,q)+A_aug'*kron(eye(2),Sigma)*(DDPhi(:,q)-DDRho(:,q));


 PP = pinv(N);


% X contiene le the componenti float della baseline e le ambiguità reali

 X(:,q)= PP*rs(:,q);



[a,sqnorm,Sigma_afixed,Z] = lambda(X(4:4+2*m1-1,q),PP(4:4+2*m1-
1,4:4+2*m1-1));


% Correzione vettore baseline come conseguenza del cambio delle
Ambiguità da float a fisse


 X(1:3,q) = X(1:3,q)-PP(1:3,4:4+2*m1-1)*inv(PP(4:4+2*m1-1,4:4+2*m1-
 1))*(X(4:4+2*m1-1,q)-a(:,1));  %primo set di candidati
```

```
    X(4:4+2*m1-1,q) = a(:,1);


   end


%Calcolo della baseline media(media fra tutti gli istanti di tempo)

   X_tot=zeros(3,1);
   norm_tot=0;
   for i=1:100
       X_tot=X_tot+X(1:3,i);
       norm_tot=norm_tot+norm(X(1:3,i));
   end

   X_medio=X_tot/100
   norm_media=norm_tot/100;
   X_vero=R2vera-R1



  %Calcolo dell' RMS(root mean square) e della deviazione standard (St_dev)
   B=0;
   T=0;
   for n=1:100
       B=B+(norm(X(1:3,n))-norm_media)^2;
       T=T+(norm(X(1:3,n)))^2;
   end

   St_dev=sqrt(B/100);
   RMS=sqrt(T/100);

   %Grafico 1 di come varia la norma di X in funzione del tempo

   Y=zeros(1,100);


   for i=1:100
       Y(1,i)=norm(X(1:3,i));

   end

   X_vero=norm(R2vera-R1)*ones(1,100);


   hold on
   figure(1);
   plot(1:100,Y,1:100,X_vero,'linewidth',2);
   title('Baseline norm','fontsize',16);
   xlabel('time [s]','fontsize',16);
```

```
ylabel(' Norm [m]','fontsize',16);
legend('Calculated baseline','Computed baseline');
axis([ 1 100 95 97])


figure(2);
plot(1:100,X_vero-Y,'r','linewidth',2);
title('Baseline error','fontsize',16);
xlabel('time [s]','fontsize',16);
ylabel(' Error [m]','fontsize',16);



%Trasformazione baseline in coordinate NED e grafici N-E

for i=1:100
    S2(1:3,i)=X(1:3,i)+R1;
end


for i=1:100
NED(1:3,i) = Trasf_ECEF_to_NED(S2(1:3,i),R1,1);
end

Nord=zeros(1,100);
Est=zeros(1,100);
Down=zeros(1,100);
for i=1:100
    Nord(1,i)=NED(1,i);
    Est(1,i)=NED(2,i);
    Down(1,i)=NED(3,i);
end



L=0;
NED_tot=0;
for i=1:100
     L=L+NED(1:3,i);
     NED_tot=NED_tot+norm(NED(1:3,i));
 end
 NED_medio=L/100
 E_NED=NED_tot/100;
 NED_vero = Trasf_ECEF_to_NED(R2vera,R1,1)

 Est_vero=NED_vero(2,1)*ones(1,100);
 Nord_vero=NED_vero(1,1)*ones(1,100);

 %Grafico 3 N-E
```

```
figure(3)
plot(Est,Nord,'*');
title('N-E components of the baseline','fontsize',16);
axis([ 82 84 -50 -46]);
xlabel('Est[m]','fontsize',16);
ylabel('Nord[m]','fontsize',16);
hold on
plot(Est_vero,Nord_vero,'ro');
legend('N-E calculated','N-E computed');


figure(4);
plot(1:100,[(Est-Est(1))' (Nord-Nord(1))' (Down-
Down(1))'],'linewidth',2);
title('Differential Position Estimates From Phase
Observations','fontsize',16);
ylabel('Corrections to Initial Position [m]','fontsize',16);
xlabel('Epochs [1 s interval]','fontsize',16);
legend('East','North','Down');
set(gca,'fontsize',16);
legend;

error=zeros(1,100);
for i=1:100

  error_N(1,i)=NED(1,i)-NED_vero(1,1);

  error_E(1,i)=NED(2,i)-NED_vero(2,1);

  error_D(1,i)=NED(3,i)-NED_vero(3,1);

end

figure(5)
plot(1:100,error_N,'g+')
axis([1 100 0 2]);
title('Nord baseline component error','fontsize',16);
xlabel('time [s]','fontsize',16);
ylabel('Error Nord [m]','fontsize',16);

figure(6)
plot(1:100,error_E,'b+')
axis([1 100 -2 0]);
title('Est baseline component error','fontsize',16);
xlabel('time [s]','fontsize',16);
ylabel('Error Est [m]','fontsize',16);

figure(7)
plot(1:100,error_D,'r+')
axis([1 100 0 2]);
title('Down baseline component error','fontsize',16);
xlabel('time [s]','fontsize',16);
ylabel('Error Down [m]','fontsize',16);
```