Basics of the GPS Technique: Observation Equations[§]

Geoffrey Blewitt

Department of Geomatics, University of Newcastle
Newcastle upon Tyne, NE1 7RU, United Kingdom

geoffrey.blewitt@ncl.ac.uk

Table of Contents

Appears in the textbook "Geodetic Applications of GPS," published by the Swedish Land Survey.

1. INTRODUCTION

The purpose of this paper is to introduce the principles of GPS theory, and to provide a background for more advanced material. With that in mind, some of the theoretical treatment has been simplified to provide a starting point for a mathematically literate user of GPS who wishes to understand how GPS works, and to get a basic grasp of GPS theory and terminology. It is therefore not intended to serve as a reference for experienced researchers; however, my hope is that it might also prove interesting to the more advanced reader, who might appreciate some "easy reading" of a familiar story in a relatively short text (and no doubt, from a slightly different angle).

2. GPS DESCRIPTION

In this section we introduce the basic idea behind GPS, and provide some facts and statistics to describe various aspects of the Global Positionining System.

2.1 THE BASIC IDEA

GPS positioning is based on trilateration, which is the method of determining position by measuring distances to points at known coordinates. At a minimum, trilateration requires 3 ranges to 3 known points. GPS point positioning, on the other hand, requires 4 "pseudoranges" to 4 satellites.

This raises two questions: (a) "What are pseudoranges?", and (b) "How do we know the position of the satellites?" Without getting into too much detail at this point, we address the second question first.

2.1.1 How do we know position of satellites?

A signal is transmitted from each satellite in the direction of the Earth. This signal is encoded with the "Navigation Message," which can be read by the user's GPS receivers. The Navigation Message includes orbit parameters (often called the "broadcast ephemeris"), from which the receiver can compute satellite coordinates (X,Y,Z). These are Cartesian coordinates in a geocentric system, known as WGS-84, which has its origin at the Earth centre of mass, Z axis pointing towards the North Pole, X pointing towards the Prime Meridian (which crosses Greenwich), and Y at right angles to X and Z to form a right-handed orthogonal coordinate system. The algorithm which transforms the orbit parameters into WGS-84 satellite coordinates at any specified time is called the "Ephemeris Algorithm," which is defined in GPS textbooks [e.g., Leick, 1991]. We discuss the Navigation Message in more detail later on. For now, we move on to "pseudoranges."

2.1.2 What are pseudoranges?

Time that the signal is transmitted from the satellite is encoded on the signal, using the time according to an atomic clock onboard the satellite. Time of signal reception is recorded by receiver using an atomic clock. A receiver measures difference in these times:

$$\text{pseudorange} = (\text{time difference}) \times (\text{speed of light})$$

Note that pseudorange is almost like range, except that it includes clock errors because the receiver clocks are far from perfect. How do we correct for clock errors?

2.1.3 How do we correct for clock errors?

Satellite clock error is given in Navigation Message, in the form of a polynomial. The unknown receiver clock error can be estimated by the user along with unknown station coordinates. There are 4 unknowns; hence we need a minimum of 4 pseudorange measurements.

2.2 THE GPS SEGMENTS

There are four GPS segments:

- the Space Segment, which includes the constellation of GPS satellites, which transmit the signals to the user;
- the Control Segment, which is responsible for the monitoring and operation of the Space Segment,
- the User Segment, which includes user hardware and processing software for positioning, navigation, and timing applications;
- the Ground Segment, which includes civilian tracking networks that provide the User Segment with reference control, precise ephemerides, and real time services (DGPS) which mitigate the effects of "selective availability" (a topic to be discussed later).

Before getting into the details of the GPS signal, observation models, and position computations, we first provide more information on the Space Segment and the Control Segment.

2.2.1 Orbit Design

The satellite constellation is designed to have at least 4 satellites in view anywhere, anytime, to a user on the ground. For this purpose, there are nominally 24 GPS satellites distributed in 6 orbital planes. So that we may discuss the orbit design and the implications of that design, we must digress for a short while to explain the geometry of the GPS constellation.

According to Kepler's laws of orbital motion, each orbit takes the approximate shape of an ellipse, with the Earth's centre of mass at the focus of the ellipse. For a GPS orbit, the eccentricity of the ellipse is so small (0.02) that it is almost circular. The semi-major axis (largest radius) of the ellipse is approximately 26,600 km, or approximately 4 Earth radii.

The 6 orbital planes rise over the equator at an inclination angle of 55^{o} to the equator. The point at which they rise from the Southern to Northern Hemisphere across the equator is called the "Right Ascension of the ascending node". Since the orbital planes are evenly distributed, the angle between the six ascending nodes is 60^{o}.

Each orbital plane nominally contains 4 satellites, which are generally not spaced evenly around the ellipse. Therefore, the angle of the satellite within its own orbital plane, the "true anomaly", is only approximately spaced by 90°. The true anomaly is measured from the point of closest approach to the Earth (the perigee). (We note here that there are other types of "anomaly" in GPS terminology, which are angles that are useful for calculating the satellite coordinates within its orbital plane). Note that instead of specifying the satellite's anomaly at every relevant time, we could equivalently specify the time that the satellite had passed perigee, and then compute the satellites future position based on the known laws of motion of the satellite around an ellipse.

Finally, the argument of perigee is the angle between the equator and perigee. Since the orbit is nearly circular, this orbital parameter is not well defined, and alternative parameterisation schemes are often used.

Taken together (the eccentricity, semi-major axis, inclination, Right Ascension of the ascending node, the time of perigee passing, and the argument of perigee), these six parameters define the satellite orbit. These parameters are known as Keplerian elements. Given the Keplerian elements and the current time, it is possible to calculate the coordinates of the satellite.

GPS satellites do not move in perfect ellipses, so additional parameters are necessary. Nevertheless, GPS does use Kepler's laws to its advantage, and the orbits are described in parameters which are Keplerian in appearance. Additional parameters must be added to account for non-Keplerian behaviour. Even this set of parameters has to be updated by the Control Segment every hour for them to remain sufficiently valid.

2.2.2 Orbit design consequences

Several consequences of the orbit design can be deduced from the above orbital parameters, and Kepler's laws of motion. First of all, the satellite speed can be easily calculated to be approximately 4 km/s relative to Earth's centre. All the GPS satellites orbits are prograde, which means the satellites move in the direction of Earth's rotation. Therefore, the relative motion between the satellite and a user on the ground must be less than 4 km/s. Typical values around 1 km/s can be expected for the relative speed along the line of sight (range rate).

The second consequence is the phenomena of "repeating ground tracks" every day. It is straightforward to calculate the time it takes for the satellite to complete one orbital revolution. The orbital period is approximately T = 11 hr 58 min. Therefore a GPS satellite completes 2 revolutions in 23 hr 56 min. This is intentional, since it equals the sidereal day, which is the time it takes for the Earth to rotate 360°. (Note that the solar day of 24 hr is not 360°, because during the day, the position of the Sun in the sky has changed by 1/365.25 of a day, or 4 min, due to the Earth's orbit around the Sun).

Therefore, every day (minus 4 minutes), the satellite appears over the same geographical location on the Earth's surface. The "ground track" is the locus of points on the Earth's surface that is traced out by a line connecting the satellite to the centre of the Earth. The

ground track is said to repeat. From the user's point of view, the same satellite appears in the same direction in the sky every day minus 4 minutes. Likewise, the "sky tracks" repeat. In general, we can say that the entire satellite geometry repeats every sidereal day (from the point of view of a ground user).

As a corollary, any errors correlated with satellite geometry will repeat from one day to the next. An example of an error tied to satellite geometry is "multipath," which is due to the antenna also sensing signals from the satellite which reflect and refract from nearby objects. In fact, it can be verified that, because of multipath, observation residuals do have a pattern that repeats every sidereal day. As a consequence, such errors will not significantly affect the precision, or repeatability, of coordinates estimated each day. However, the accuracy can be significantly worse than the apparent precision for this reason.

Another consequence of this is that the same subset of the 24 satellites will be observed every day by someone at a fixed geographical location. Generally, not all 24 satellites will be seen by a user at a fixed location. This is one reason why there needs to be a global distribution of receivers around the globe to be sure that every satellite is tracked sufficiently well.

We now turn our attention to the consequences of the inclination angle of 55^o. Note that a satellite with an inclination angle of 90^o would orbit directly over the poles. Any other inclination angle would result in the satellite never passing over the poles. From the user's point of view, the satellite's sky track would never cross over the position of the celestial pole in the sky. In fact, there would be a "hole" in the sky around the celestial pole where the satellite could never pass. For a satellite constellation with an inclination angle of 55^o, there would therefore be a circle of radius at least 35^o around the celestial pole, through which the sky tracks would never cross. Another way of looking at this, is that a satellite can never rise more than 55^o elevation above the celestial equator.

This has a big effect on the satellite geometry as viewed from different latitudes. An observer at the pole would never see a GPS satellite rise above 55^o elevation. Most of the satellites would hover close to the horizon. Therefore vertical positioning is slightly degraded near the poles. An observer at the equator would see some of the satellites passing overhead, but would tend to deviate from away from points on the horizon directly to the north and south. Due to a combination of Earth rotation, and the fact that the GPS satellites are moving faster than the Earth rotates, the satellites actually appear to move approximately north-south or south-north to an oberver at the equator, with very little east-west motion. The north component of relative positions are therefore better determined than the east component the closer the observer is to the equator. An observer at mid-latitudes in the Northern Hemisphere would see satellites anywhere in the sky to the south, but there would be a large void towards the north. This has consequences for site selection, where a good view is desirable to the south, and the view to the north is less critical. For example, one might want to select a site in the Northern Hemisphere which is on a south-facing slope (and visa versa for an observer in the Southern Hemisphere).

2.2.3 Satellite Hardware

There are nominally 24 GPS satellites, but this number can vary within a few satellites at any given time, due to old satellites being decommissioned, and new satellites being launched to

replace them. All the prototype satellites, known as Block I, have been decommissioned. Between 1989 and 1994, 24 Block II (1989-1994) were placed in orbit. From 1995 onwards, these have started to be replaced by a new design known as Block IIR. The nominal specifications of the GPS satellites are as follows:

- Life goal: 7.5 years
- Mass: ~1 tonne (Block IIR: ~2 tonnes)
- Size: 5 metres
- Power: solar panels 7.5 m^2 + Ni-Cd batteries
- Atomic clocks: 2 rubidium and 2 cesium

The orientation of the satellites is always changing, such that the solar panels face the sun, and the antennas face the centre of the Earth. Signals are transmitted and received by the satellite using microwaves. Signals are transmitted to the User Segment at frequencies L1 = 1575.42 MHz, and L2 = 1227.60 MHz. We discuss the signals in further detail later on. Signals are received from the Control Segment at frequency 1783.74 Mhz. The flow of information is a follows: the satellites transmit L1 and L2 signals to the user, which are encoded with information on their clock times and their positions. The Control Segment then tracks these signals using receivers at special monitoring stations. This information is used to improve the satellite positions and predict where the satellites will be in the near future. This orbit information is then uplinked at 1783.74 Mhz to the GPS satellites, which in turn transmit this new information down to the users, and so on. The orbit information on board the satellite is updated every hour.

2.2.4 The Control Segment

The Control Segment, run by the US Air Force, is responsible for operating GPS. The main Control Centre is at Falcon Air Force Base, Colorado Springs, USA. Several ground stations monitor the satellites L1 and L2 signals, and assess the "health" of the satellites. As outlined previously, the Control Segment then uses these signals to estimate and predict the satellite orbits and clock errors, and this information is uploaded to the satellites. In addition, the Control Segment can control the satellites; for example, the satellites can be maneuvered into a different orbit when necessary. This might be done to optimise satellite geometry when a new satellite is launched, or when an old satellite fails. It is also done to keep the satellites to within a certain tolerance of their nominal orbital parameters (e.g., the semi-major axis may need adjustment from time to time). As another example, the Control Segment might switch between the several on-board clocks available, should the current clock appear to be malfunctioning.

2.3 THE GPS SIGNALS

We now briefly summarise the characteristics of the GPS signals, the types of information that is digitally encoded on the signals, and how the U.S. Department of Defense implements denial of accuracy to civilian users. Further details on how the codes are constructed will be presented in Section 3.

2.3.1 Signal Description

The signals from a GPS satellite are fundamentally driven by an atomic clocks (usually cesium, which has the best long-term stability). The fundamental frequency is 10.23 Mhz. Two carrier signals, which can be thought of as sine waves, are created from this signal by multiplying the frequency by 154 for the L1 channel (frequency = 1575.42 Mhz; wavelength = 19.0 cm), and 120 for the L2 channel (frequency = 1227.60 Mhz; wavelength = 24.4 cm). The reason for the second signal is for self-calibration of the delay of the signal in the Earth's ionosphere.

Information is encoded in the form of binary bits on the carrier signals by a process known as phase modulation. (This is to be compared with signals from radio stations, which are typically encoded using either frequency modulation, FM, or amplitude modulation, AM). The binary digits 0 and 1 are actually represented by multiplying the electrical signals by either $+1$ or -1, which is equivalent to leaving the signal unchanged, or flipping the phase of the signal by 180^{o}. We come back later to the meaning of phase and the generation of the binary code.

There are three types of code on the carrier signals:

- The C/A code
- The P code
- The Navigation Message

The C/A ("course acquisition") code can be found on the L1 channel. As will be described later, this is a code sequence which repeats every 1 ms. It is a pseudo-random code, which appears to be random, but is in fact generated by a known algorithm. The carrier can transmit the C/A code at 1.023 Mbps (million bits per second). The "chip length", or physical distance between binary transitions (between digits $+1$ and -1), is 293 metres. The basic information that the C/A code contains is the time according to the satellite clock when the signal was transmitted (with an ambiguity of 1 ms, which is easily resolved, since this corresponds to 293 km). Each satellite has a different C/A code, so that they can be uniquely identified.

The P ("precise") code is identical on both the L1 and L2 channel. Whereas C/A is a courser code appropriate for initially locking onto the signal, the P code is better for more precise positioning. The P code repeats every 267 days. In practice, this code is divided into 7 day segments; each weekly segment is designated a "PRN" number, and is designated to one of the GPS satellites. The carrier can transmit the P code at 10.23 Mbps, with a chip length of 29.3 metres. Again, the basic information is the satellite clock time or transmission, which is identical to the C/A information, except that it has ten times the resolution. Unlike the C/A code, the P code can be encrypted by a process known as "anti-spoofing" , or "A/S" (see below).

The Navigation Message can be found on the L1 channel, being transmitted at a very slow rate of 50 bps. It is a 1500 bit sequence, and therefore takes 30 seconds to transmit. The Navigation Message includes information on the Broadcast Ephemeris (satellite orbital parameters), satellite clock corrections, almanac data (a crude ephemeris for all satellites), ionosphere information, and satellite health status.

2.3.2 Denial of Accuracy

The U.S. Department of Defense implements two types of denial of accuracy to civilian users: Selective Availability (S/A), and Anti-Spoofing (A/S). S/A can be thought of as intentional errors imposed on the GPS signal. A/S can be thought of as encryption of the P code.

There are two types of S/A: epsilon, and dither. Under conditions of S/A, the user should be able to count on the position error not being any worse than 100 metres. Most of the time, the induced position errors do not exceed 50 metres.

Epsilon is implemented by including errors in the satellite orbit encoded in the Navigation Message. Apparently, this is an option not used, according to daily comparisons made between the real-time broadcast orbits, and precise orbits generated after the fact, by the International GPS Service for Geodynamics (IGS). For precise geodetic work, precise orbits are recommended in any case, and therefore epsilon would have minimal impact on precise users. It would, however, directly impact single receiver, low-precision users. Even then, the effects can be mitigated to some extent by using technology known as "differential GPS", where errors in the GPS signal are computed at a reference station at known coordinates, and are transmitted to the user who has appropriate radio receiving equipment.

Dither is intentional rapid variation in the satellite clock frequency (10.23 MHz). Dither, therefore, looks exactly like a satellite clock error, and therefore maps directly into pseudorange errors. Dither is switched on at the moment (1997), but recent U.S. policy statements indicate that it may be phased out within the next decade. As is the case for epsilon, dither can be mitigated using differential GPS. The high precision user is minimally effected by S/A, since relative positioning techniques effectively eliminate satellite clock error (as we shall see later).

Anti-Spoofing (A/S) is encryption of the P-code. The main purpose of A/S is prevent "the enemy" from imitating a GPS signal, and therefore it is unlikely to be switched off in the foreseeable future. A/S does not pose a signficant problem to the precise user, since precise GPS techniques rely on measuring the phase of the carrier signal itself, rather than the pseudoranges derived from the P code. However, the pseudoranges are very useful for various algorithms, particularly in the rapid position fixes required by moving vehicles and kinematic surveys. Modern geodetic receivers can, in any case, form 2 precise pseudorange observables on the L1 and L2 channels, even if A/S is switched on. (We briefly touch on how this is done in the next section). As a consequence of not having full access to the P code, the phase noise on measuring the L2 carrier phase can be increased from the level of 1 mm to the level of 1 cm for some types of receivers. This has negligible impact on long sessions for static positioning, but can have noticeable effect on short sessions, or on kinematic positioning. Larger degradation in the signal can be expected at low elevations (up to 2 cm) where signal strength is at a minimum.

3. THE PSEUDORANGE OBSERVABLE

In this section, we go deeper into the description of the pseudorange observable, an give some details on how the codes are generated. We develop a model of the pseudorange observation, and then use this model to derive a least-squares estimator for positioning. We discuss formal errors in position, and the notion of "Dilution of Precision", which can be used to assess the effect of satellite geometry on positioning precision.

3.1 CODE GENERATION

It helps to understand the pseudorange measurement if we first take a look at the actual generation of the codes. The carrier signal is multiplied by a series of either +1 or -1, which are seperated by the chip length (293 m for C/A code, and 29.3 m for P code). This series of +1 and -1 multipliers can be interpreted as a stream of binary digits (0 and 1).

How is this stream of binary digits decided? They are determined by an algorithm, known as a linear feedback register. To understand a linear feedback register, we must first introduce the XOR binary function.

3.1.1 XOR: The "Exclusive OR" Binary Function

A binary function takes two input binary digits, and outputs one binary digit (0 or 1). More familiar binary functions might be the "AND" and "OR" functions. For example, the AND function gives a value of 1 if the two input digits are identical, that is (0,0), or (1,1). If the input digits are different, the AND function gives a value of 0. The OR function gives a value of 1 if either of the two input digits equals 1, that is (0,1), (1,0), or (1,1).

The XOR function gives a value of 1 if the two inputs are different, that is (1,0) or (0,1). If the two inputs are the same, (0,0) or (0,1), then the value is 0.

What is XOR(A,B)? Remember this: *Is A different to B? If so, the answer is 1.*

- If $A \neq B$, then XOR(A,B) = 1
- If $A = B$, then XOR(A,B) = 0

The XOR function can be represented by the "truth table" shown in Table 1.

Input A	**Input B**	**Output XOR(A,B)**
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**0**

Table 1. Truth table for the XOR function.

3.1.2 Linear Feedback Registers

Linear feedback registers are used to generate a pseudorandom number sequence. The sequence is pseudorandom, since the sequence repeats after a certain number of digits (which, as we shall see, depends on the size of the register). However, the statistical properties of the sequence are very good, in that the sequence appears to be white noise. We return to these properties later, since they are important for understanding the measurement process. For now, we look at how the register works.

Cycle, N	$A_N = XOR(A_{N-1}, C_{N-1})$	$B_N = A_{N-1}$	$C_N = B_{N-1}$
1	initialise: 1	1	**1**
2	XOR(1,1) = 0	1	**1**
3	XOR(0,1) = 1	0	**1**
4	XOR(1,1) = 0	1	**0**
5	XOR(0,0) = 0	0	**1**
6	XOR(0,1) = 1	0	**0**
7	XOR(1,0) = 1	1	**0**
8 (=1)	XOR(1,0) = 1 (pattern repeats)	1	**1**

Table 2. A 3 stage linear feedback register. The output is in column C.

Table 2 illustrates a simple example: the "3 stage linear feedback register." The "state" of the register is defined by three binary numbers (A, B, C). The state changes after a specific time interval. To start the whole process, the intial state of a feedback register is always filled with 1; that is, for the 3 stage register, the initial state is (1, 1, 1). The digits in this state are now shifted to the right, giving (blank, 1, 1). The digit (1) that is pushed off the right side is the output from the register. The blank is replaced by taking the XOR of the other two digits (1,1). The value, in this case, equals 0. The new state is therefore (0, 1, 1). This process is then repeated, so that the new output is (1), and the next state is (1, 0, 1). The next output is (1) and the next state is (1, 1, 0). The next output is (0), and the next state is (0, 1, 1), and so on.

In the above example, the outputs can be written (1, 1, 1, 0, ...). This stream of digits is known as the "linear feedback register sequence." This sequence will start to repeat after a while. It turns out that during a complete cycle, the feedback register will produce every possible combination of binary numbers, except for (0, 0, 0). We can therefore easily calculate the length of the sequence before it starts to repeat again. For a 3 stage register, there are 8 possible combinations of binary digits. This means that the sequence will repeat after 7 cycles. The sequence length is therefore 7 bits. More generally, the sequence length is:

$$L(N) = 2^N - 1$$

where N is the size of the register (number of digits in the state). For example, a 4 state linear feedback register will have a sequence length of 15 bits.

3.1.3 C/A Code

The C/A code is based on the 10 stage linear feedback register sequence, for which the sequence length is $L(10) = 2^{10}-1 = 1023$ bits. The C/A code really has a repeating sequence of 1023 bits; however the design is slightly more complicated than presented above. The C/A code is actually a "Gold code", which is derived by taking the XOR of the output from 2 linear feedback registers. Unique C/A codes can be generated for each satellite by selecting different pairs of cells from each register to define the output.

In summary, the C/A code is a unique Gold code on each satellite, which is a pseudorandom sequence of bits with a repeating sequence length of 1023. C/A bit transitions occur at 1.023 Mhz. Note that the fundamental frequency in the satellite is 10.23 Mhz, so this represents one transition every 10 cycles. At this rate of bit transitions, the full sequence of 1023 bits is transmitted in 1 ms. Therefore, the sequence repeats 1000 times per second. The chip length (distance between bit transitions) is 293 m. Therefore, the sequence repeats every 300 km.

3.1.4 P Code

The P code is also generated from a combination of two different registers, in such a way that it repeats every 266.4 days. Each 7 day section is assigned a "PRN code." Satellites are often identified by their PRN number; however, the user should beware that any given satellite can have its PRN code changed. Therefore, PRN codes should not be used in place of Satellite Vehicle Numbers (SVN) when talking about particular satellites. (For example, if someone writes software which identifies satellites using PRN numbers, there might be a problem in orbit modelling, for example, PRN 2 is assigned to a Block II satellite now, but to a Block IIR satellite next year). There are 38 possible PRN codes; given that there are 24 nominal satellites, some PRN codes are left unused. The PRN sequence is reset at Saturday midnight, defining the start of "GPS week."

3.1.5 GPS signal transmission and reception

Let us now summarise how the GPS signal is transmitted from space, and then received on the ground. The GPS signal starts in the satellite as a voltage which oscillates at the fundamental clock frequency of 10.23 Mhz. (If selective availability is on, this signal is then "dithered" so that the frequency varies unpredictably). This signal is then separately multiplied in frequency by the integers 154 and 120, to create the L1 and L2 carrier signals. The signals are then multiplied by +1 and −1 according the algorithms described above to generate the C/A code (on L1) and the P code (on both L1 and L2). These codes are unique to each satellite. Finally, the Navigation Message is encoded onto the signal. The signals are boosted by an amplifier, and then sent to transmitting antennas, which point towards the Earth. These antennas are little more than exposed electrical conductors which radiate the signal into space in the form of electromagnetic waves.

These electromagnetic waves pass through space and the Earth's atmosphere, at very close to the speed of light in a vacuum, until they reach the receiver's antenna. The waves create a minute signal in the antenna, in the form of an oscillating voltage. The signal is now pre-amplified at the antenna, to boost the signal strength, so that it is not overcome by noise by the time it gets to the other end of the antenna cable. The signal then enters the receiver, which

then measures it using a process known as "autocorrelation." It is beyond the scope of this paper to go into the details of receiver design, so our description will be kept at the level required to understand how the observable model can be developed.

3.2 AUTOCORRELATION TECHNIQUE

We have described how the GPS satellites construct the GPS signals. Actually, the receiver also generate GPS-like signals internally. The receiver knows precisely what the transmitted GPS signal is supposed to look like at any given time, and it generates an electronic replica, in synchronisation with the receiver's own clock. The receiver then compares the replica signal with the actual signal. Since the GPS signal was actually created in the satellite some time previously (about 0.07 seconds ago, due to the speed of light), the receiver's replica signal must be delayed in to match up the incoming signal with the replica signal. This time delay is actually what the receiver is fundamentally measuring. Clearly, this represents the time taken for the signal to pass from the satellite to the receiver, but it also includes any error in the satellite clock, and any error in the receiver clock. One can see that the time delay is therefore related to the range to the satellite. We return to this model later, and now turn our attention to how the receiver matches the two signals.

The time difference is computed by autocorrelation. The first bit from signal one is multiplied by the first bit of signal two. For example, if the first bits from the two signals both have values −1, then the result is $(-1) \times (-1) = +1$. Similarly, if both bits have values +1, then the result is +1. On the other hand, if the two bits disagree, the result is $(+1) \times (-1) = -1$. This process is repeated for the second pair of bits, and so on. The result can be written as a sequence of +1 (where the bits agree) and -1 (where the bits disagree). This sequence is then summed, and divided by the total number of bits in each signal. For example, if signal A can be written (+1, −1, −1, +1, −1), and signal B can be written (+1, +1, −1, −1, +1), then multiplication gives (+1, −1, +1, −1, −1); the sum of which gives −1; then dividing by the number of bits (5) gives −0.2. Note that if the two signals matched perfectly, the result would be +1. If the two signals were completely random, we should expect a result close to zero.

This is why the GPS signals are designed to look random. When the two signals are not properly matched in time, the result of autocorrelation gives an answer close to zero; if the signals are matched in time, the result is close to +1 (but not exactly, since a real signal also has noise, so some bits are incorrect). One can see that the larger the number of bits that are compared, the better the resolution. This is because the random bits will average to zero better, the more bits we compare.

The Gold codes have the property that the autocorrelation is constant until we get to within one chip of the correct answer. Within that window of ±1 chip, the autocorrelation function looks like an equilateral triangle, with a value of 1 at its peak (assuming no noise). We can therefore use the known triangular shape as a model to help us find the time displacement that maximises the autocorrelation. (More sophisticated receivers account for the fact that multipath distorts the shape of this triangle, and can thus reduce the effect of multipath).

Now that we have found the peak autocorrelation, the inferred time displacement between the two signals is multiplied by the speed of light. This observation is called the pseudorange. The pseudorange measurement is described schematically in Figure 1.

3.3 PSEUDORANGE OBSERVATION EQUATIONS

3.3.1 Simplified Pseudorange Model

Receivers record data at regular, specified intervals (say, every 30 seconds, as instructed by the receiver user). It is the reading of the receiver clock time T, which is used to say exactly when the measurement is sampled. Therefore, the value of T at a measurement epoch is known exactly, and is written to the data file along with the observation. (What is not known, is the true time of measurement). The actual observation to satellite s can be written:

$$P^S = (T - T^S)\, c$$

where T is the known reading of the receiver clock when signal is received, T^S is the reading of the satellite clock when the signal was transmitted, and c is the speed of light (in a vacuum) = 299792458 m/s.

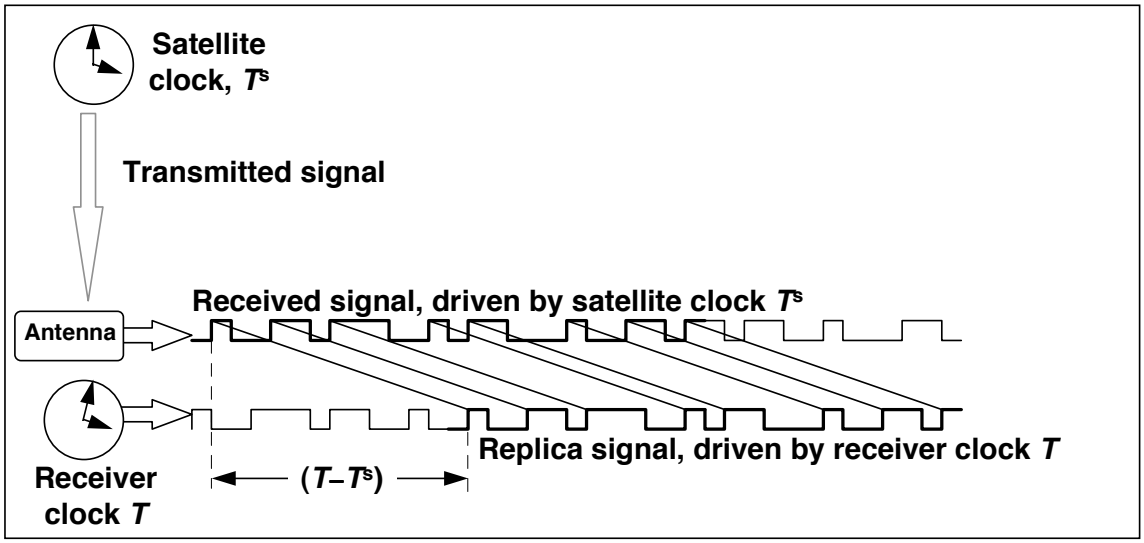

Figure 1: A schematic diagram showing how the GPS pseudorange observation is related to the satellite and receiver clocks.

The modelled observation can be developed by setting the clock time T equal to the true receive time t plus a clock bias τ, for both the receiver and satellite clocks:

$$T = t + \tau$$
$$T^S = t^S + \tau^S$$

Substitution gives the pseudorange as a function of the true time the signal was received:

$$\begin{aligned} P^S(t) &= ((t+\tau) - (t^S + \tau^S))c \\ &= (t - t^S)c + c\tau - c\tau^S \\ &= \rho^S(t, t^S) + c\tau - c\tau^S \end{aligned}$$

where $\rho^S(t, t^S)$ is the range from receiver (at receive time) to the satellite (at transmit time). This model is simplified; for example, it assumes the speed of light in the atmosphere is c, and it ignores the theory of relativity; but this simplified model is useful to gain insight into the principles of GPS. From Pythagoras Theorem, we can write:

$$\rho^S(t,t^S) = \sqrt{\left(x^S(t^S) - x(t)\right)^2 + \left(y^S(t^S) - y(t)\right)^2 + \left(y^S(t^S) - y(t)\right)^2}$$

The Navigation message allows us to compute the satellite position (x^S, y^S, z^S) and the satellite clock bias τ^S. Therefore we are left with 4 unknowns, the receiver position (x, y, z) and the receiver clock bias τ.

We note here one complication: that the satellite position must be calculated at transmission time, t^S. This is important, because the satellite range can change as much as 60 metres from the time the signal was transmitted, to the time the signal was received, approximately 0.07 seconds later. If the receive time were used instead, the error in computed range could be tens of metres. Starting with the receive time, t, the transmit time can be computed by an iterative algorithm known as "the light time equation," which can be written as follows:

$$\begin{aligned} t^S(0) &= t = \left(T - \tau\right) \\ t^S(1) &= t - \frac{\rho^S(t,t^S(0))}{c} \\ t^S(2) &= t - \frac{\rho^S(t,t^S(1))}{c} \\ &\vdots \end{aligned}$$

where the satellite position (and hence the range $\rho^S(t, t^S)$) is calculated at each step using the Keplerian-type elements from the Navigation Message, and the algorithm is stopped once the computed range converges (i.e., don't change by more than a negligible amount). Although more rapidly converging methods have been implemented, the above method is probably the easiest to understand.

Note that the above algorithm starts with the true receive time, which requires the receiver clock bias. We usually don't know in advance what the bias is, but for most receivers it never gets larger than a few milliseconds (beyond which, the receiver will reset its clock). If we assume it is zero in the above computation, the error produced is a few metres, which is much smaller than typical point positioning precision of approximately 50 metres with S/A switched on. We can therefore safely ignore this effect for now, and return to it later when we discuss the more precise carrier phase observable.

We now look at our system of simplified observation equations from 4 satellites in view of the receiver. Using the above notation, we can write the pseudoranges to each satellite as:

$$P^1 = \left((x^1 - x)^2 + (y^1 - y)^2 + (z^1 - z)^2 \right)^{1/2} + c\tau - c\tau^1$$

$$P^2 = \left((x^2 - x)^2 + (y^2 - y)^2 + (z^2 - z)^2 \right)^{1/2} + c\tau - c\tau^2$$

$$P^3 = \left((x^3 - x)^2 + (y^3 - y)^2 + (z^3 - z)^2 \right)^{1/2} + c\tau - c\tau^3$$

$$P^4 = \left((x^4 - x)^2 + (y^4 - y)^2 + (z^4 - z)^2 \right)^{1/2} + c\tau - c\tau^4$$

(Note that in this and subsequent equations, the superscripts next to the satellite coordinates are meant to identify the satellite, and should not be confused with exponents). In the following section, we proceed to solve this system of equations for the 4 unknowns, (x, y, z, τ) using familiar least squares methods. Although this is not strictly necessary for 4 unknowns with 4 parameters, it does generalise the solution to the case where we have $m \geq 4$ satellites in view.

4. POINT POSITIONING USING PSEUDORANGE

4.1 LEAST SQUARES ESTIMATION

4.1.1 Linearised Model

We solve the point positioning problem by first linearising the pseudorange observation equations, and then using the familiar methods of least-squares analysis. For completeness, we summarise the linearisation procedure and the development of the least squares method specifically for the GPS point positioning problem. First, we assume we can write the actual observation to be the sum of a modelled observation, plus an error term:

$$\begin{aligned} P_{\text{observed}} &= P_{\text{model}} + \text{noise} \\ &= P(x, y, z, \tau) + v \end{aligned}$$

Next, we apply Taylor's theorem, where we expand about the model computed using provisional parameter values (x_0, y_0, z_0, τ_0), and ignore second and higher order terms.

$$\begin{aligned} P(x, y, z, \tau) &\cong P(x_0, y_0, z_0, \tau_0) + \left(x - x_0\right)\frac{\partial P}{\partial x} + \left(y - y_0\right)\frac{\partial P}{\partial y} + \left(z - z_0\right)\frac{\partial P}{\partial z} + \left(\tau - \tau_0\right)\frac{\partial P}{\partial \tau} \\ &= P_{\text{computed}} + \frac{\partial P}{\partial x}\Delta x + \frac{\partial P}{\partial y}\Delta y + \frac{\partial P}{\partial z}\Delta z + \frac{\partial P}{\partial \tau}\Delta \tau \end{aligned}$$

Note that the partial derivatives in the above expression are also computed using provisional values (x_0, y_0, z_0, τ_0). The residual observation is defined to be the difference between the actual observation and the observation computed using the provisional parameter values:

$$\begin{aligned}\Delta P &\equiv P_{\text{observed}} - P_{\text{computed}} \\ &= \frac{\partial P}{\partial x}\Delta x + \frac{\partial P}{\partial y}\Delta y + \frac{\partial P}{\partial z}\Delta z + \frac{\partial P}{\partial \tau}\Delta\tau + v\end{aligned}$$

This can be written in matrix form:

$$\Delta P = \begin{pmatrix} \frac{\partial P}{\partial x} & \frac{\partial P}{\partial y} & \frac{\partial P}{\partial z} & \frac{\partial P}{\partial \tau} \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta\tau \end{pmatrix} + v$$

We get such an equation for each satellite in view. In general, for m satellites, we can write this system of m equations in matrix form:

$$\begin{pmatrix} \Delta P^1 \\ \Delta P^2 \\ \Delta P^3 \\ \vdots \\ \Delta P^m \end{pmatrix} = \begin{pmatrix} \frac{\partial P^1}{\partial x} & \frac{\partial P^1}{\partial y} & \frac{\partial P^1}{\partial z} & \frac{\partial P^1}{\partial \tau} \\ \frac{\partial P^2}{\partial x} & \frac{\partial P^2}{\partial y} & \frac{\partial P^2}{\partial z} & \frac{\partial P^2}{\partial \tau} \\ \frac{\partial P^3}{\partial x} & \frac{\partial P^3}{\partial y} & \frac{\partial P^3}{\partial z} & \frac{\partial P^3}{\partial \tau} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial P^m}{\partial x} & \frac{\partial P^m}{\partial y} & \frac{\partial P^m}{\partial z} & \frac{\partial P^m}{\partial \tau} \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta\tau \end{pmatrix} + \begin{pmatrix} v^1 \\ v^2 \\ v^3 \\ \vdots \end{pmatrix}$$

The equation is often written using matrix symbols as:

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{v}$$

which expresses a linear relationship between the residual observations **b** (i.e., observed minus computed observations) and the unknown correction to the parameters **x**. The column matrix **v** contains all the noise terms, which are also unknown at this point. We call the above matrix equation the "linearised observation equations".

4.1.2 The Design Matrix

The linear coefficients, contained in the "design matrix" **A**, are actually the partial derivatives of each observation with respect to each parameter, computed using the provisional parameter values. Note that **A** has the same number of columns as there are parameters, $n = 4$, and has the same number of rows as there are data, $m \geq 4$. We can derive the coefficients of **A** by partial differentiation of the observation equations, producing the following expression:

$$\mathbf{A} = \begin{pmatrix} \frac{x_0 - x^1}{\rho} & \frac{y_0 - y^1}{\rho} & \frac{z_0 - z^1}{\rho} & c \\ \frac{x_0 - x^2}{\rho} & \frac{y_0 - y^2}{\rho} & \frac{z_0 - z^2}{\rho} & c \\ \frac{x_0 - x^3}{\rho} & \frac{y_0 - y^3}{\rho} & \frac{z_0 - z^3}{\rho} & c \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_0 - x^m}{\rho} & \frac{y_0 - y^m}{\rho} & \frac{z_0 - z^m}{\rho} & c \end{pmatrix}$$

Note that **A** is shown to be purely a function of the direction to each of the satellites as observed from the receiver.

4.1.3 The Least Squares Solution

Let us consider a solution for the linearised observation equations, denoted $\hat{\mathbf{x}}$. The "estimated residuals" are defined as the difference between the actual observations and the new, estimated model for the observations. Using the linearised form of the observation equations, we can write the estimated residuals as:

$$\hat{\mathbf{v}} = \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}$$

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

$$J(\mathbf{x}) \equiv \sum_{i=1}^{m} v_i^{\,2} = \mathbf{v}^{\mathrm{T}}\mathbf{v} = \left(\mathbf{b} - \mathbf{A}\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{b} - \mathbf{A}\mathbf{x}\right).$$

That is, we are minimising the sum of squares of the estimated residuals. If we vary **x** by a small amount, then $J(\mathbf{x})$ 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:

$$\begin{aligned}
\delta J(\hat{\mathbf{x}}) &= 0 \\
\delta\left\{\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right)^{\mathrm{T}}\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right)\right\} &= 0 \\
\delta\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right)^{\mathrm{T}}\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right) + \left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right)^{\mathrm{T}}\delta\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right) &= 0 \\
\left(-\mathbf{A}\delta\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right) + \left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right)^{\mathrm{T}}\left(-\mathbf{A}\delta\mathbf{x}\right) &= 0 \\
\left(-2\mathbf{A}\delta\mathbf{x}\right)^{\mathrm{T}}\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right) &= 0 \\
\left(\delta\mathbf{x}^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\right)\left(\mathbf{b} - \mathbf{A}\hat{\mathbf{x}}\right) &= 0 \\
\delta\mathbf{x}^{\mathrm{T}}\left(\mathbf{A}^{\mathrm{T}}\mathbf{b} - \mathbf{A}^{\mathrm{T}}\mathbf{A}\hat{\mathbf{x}}\right) &= 0 \\
\mathbf{A}^{\mathrm{T}}\mathbf{A}\hat{\mathbf{x}} &= \mathbf{A}^{\mathrm{T}}\mathbf{b}
\end{aligned}$$

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

$$\hat{\mathbf{x}} = \left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{b}$$

This assumes that the inverse to $\mathbf{A}^T\mathbf{A}$ 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).

4.2 ERROR COMPUTATION

4.2.1 The Covariance and Cofactor Matrices

If the observations **b** had no errors, and the model were perfect, then the estimates $\hat{\mathbf{x}}$ given by the above expression would be perfect. Any errors **v** in the original observations **b** will obviously map into errors $\mathbf{v_x}$ in the estimates $\hat{\mathbf{x}}$. It is also clear that this mapping will take exactly the same linear form as the above formula:

$$\mathbf{v_x} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{v}$$

If we have (a priori) an expected value for the error in the data, σ, then we can compute the expected error in the parameters. We discuss the interpretation of the "covariance matrix" later, but for now, we define it as the (square) matrix of expected values of one error multiplied by another error; that is, $C_{ij} \equiv \mathrm{E}(v_i v_j)$. A diagonal element C_{ii} is called a "variance," and is often written as the square of the standard deviation, $C_{ii} \equiv \mathrm{E}(v_i^2) = \sigma_i^2$. We can concisely define the covariance matrix by the following matrix equation:

$$\mathbf{C} \equiv \mathrm{E}(\mathbf{v}\mathbf{v}^T).$$

Let us for now assume we can characterise the error in the observations by one number, the variance $\sigma^2 = \mathrm{E}(v^2)$, which is assumed to apply to all m observations. Let us also assume that all observations are uncorrelated, $\mathrm{E}(v_i v_j) = 0$ (for $i \neq j$). We can therefore write the covariance matrix of observations as the diagonal matrix, $\mathbf{C}_\sigma = \sigma^2\mathbf{I}$, where **I** is the $m \times m$ identity matrix:

$$\mathbf{C}_\sigma = \begin{pmatrix} \sigma^2 & 0 & \cdots & 0 \\ 0 & \sigma^2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \sigma^2 \end{pmatrix}_{m \times m}$$

Under these assumptions, the expected covariance in the parameters for the least squares solution takes on a simple form:

$$
\begin{aligned}
\mathbf{C}_{\mathbf{x}} &= \mathrm{E}\left(\mathbf{v}_{\mathbf{x}}\mathbf{v}_{\mathbf{x}}{}^{\mathbf{T}}\right) \\
&= \mathrm{E}\left(\left(\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathbf{T}}\mathbf{v}\right)\left(\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathbf{T}}\mathbf{v}\right)^{\mathbf{T}}\right) \\
&= \mathrm{E}\left(\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathbf{T}}\mathbf{v}\mathbf{v}^{\mathbf{T}}\mathbf{A}\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\right) \\
&= \left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathbf{T}}\,\mathrm{E}\left(\mathbf{v}\mathbf{v}^{\mathbf{T}}\right)\mathbf{A}\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1} \\
&= \left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathbf{T}}C_{\sigma}\mathbf{A}\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1} \\
&= \left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathbf{T}}\left(\sigma^2\mathbf{I}\right)\mathbf{A}\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1} \\
&= \sigma^2\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1} \\
&= \sigma^2 \times \langle \text{cofactor matrix} \rangle
\end{aligned}
$$

Note that the "cofactor matrix" $(\mathbf{A}^{\mathbf{T}}\mathbf{A})^{-1}$ also appears in the formula for the least squares estimate, $\hat{\mathbf{x}}$. The "cofactor matrix" is also sometimes called the "covariance matrix," where it is implicitly understood that it should be scaled by the variance of the input observation errors. Since GPS observation errors are a strong function of the particular situation (e.g., due to environmental factors), it is common to focus on the cofactor matrix, which, like **A**, is purely a function of the satellite-receiver geometry at the times of the observations. The cofactor matrix can therefore be used to assess the relative strength of the observing geometry, and to quantify how the level of errors in the measurements can be related to the expected level of errors in the position estimates.

It should therefore be clear why **A** is called the "design matrix"; we can in fact compute the cofactor matrix in advance of a surveying session if we know where the satellites will be (which we do, from the almanac in the Navigation Message). We can therefore "design" our survey (in this specific case, select the time of day) to ensure that the position precision will not be limited by poor satellite geometry.

4.2.2 Interpreting the Covariance Matrix

The covariance matrix for the estimated parameters can be written in terms of its components:

$$
\begin{aligned}
\mathbf{C}_{\mathbf{x}} &= \sigma^2\left(\mathbf{A}^{\mathbf{T}}\mathbf{A}\right)^{-1} \\
&= \sigma^2\begin{pmatrix}
\sigma_x{}^2 & \sigma_{xy} & \sigma_{xz} & \sigma_{x\tau} \\
\sigma_{yx} & \sigma_y{}^2 & \sigma_{yz} & \sigma_{y\tau} \\
\sigma_{zx} & \sigma_{zy} & \sigma_z{}^2 & \sigma_{z\tau} \\
\sigma_{\tau x} & \sigma_{\tau y} & \sigma_{\tau z} & \sigma_{\tau}{}^2
\end{pmatrix}
\end{aligned}
$$

As an example of how to interpret these components, if the observation errors were at the level of $\sigma = 1$ metre, the error in *y* coordinate would be at the level of σ_y metres; if the observation errors were $\sigma = 2$ metres, the error in *y* would be $2\sigma_y$ metres, and so on.

The off-diagonal elements indicate the degree of correlation between parameters. If σ_{yz} were negative, this means that a positive error in *y* will probably be accompanied by a negative error in *z*, and visa versa. A useful measure of correlation is the "correlation coefficient," which is defined as

$$\rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_i^2 \sigma_j^2}}$$

The correlation coefficient is only a function of the cofactor matrix, and is independent of the observation variance, σ^2. Its value can range between −1 to +1, where 0 indicates no correlation, and +1 indicates perfect correlation (i.e., the two parameters are effectively identical). Several textbooks show that the "error ellipse" in the plane defined by the (*z*, *y*) coordinates (for example) can be computed using the elements σ_z^2, σ_y^2, and ρ_{zy} .

4.2.3 Local Coordinate Errors

Applications tend to focus on horizontal and vertical position. Also, height, h, tends to have largest error than horizontal coordinates. It is therefore more convenient to look at errors in local geodetic coordinates; that is to transform geocentric coordinates *(u,v,w)* to local topocentric coordinates *(n, e, h)*. For this, we have to transform the covariance matrix, using the laws of error propogation. Consider the rotation matrix *G* which takes us from small relative vector in geocentric system into the local system at latitude φ and longitude λ:

$$\Delta\mathbf{L} = \mathbf{G}\Delta\mathbf{X}$$

$$\begin{pmatrix} \Delta n \\ \Delta e \\ \Delta h \end{pmatrix} = \begin{pmatrix} -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ -\sin\lambda & \cos\lambda & 0 \\ \cos\varphi\cos\lambda & \cos\varphi\sin\lambda & \sin\varphi \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}$$

Obviously, matrix **G** would also transform the errors in $\Delta\mathbf{X}$ into errors in $\Delta\mathbf{L}$:

$$\mathbf{v_L} = \mathbf{G}\mathbf{v_X}$$

We now derive how to transform the covariance matrix of coordinates from geocentric system to the local system. This procedure is sometimes referred to as the "law of propogation of errors":

$$\begin{aligned} \mathbf{C_L} &= \mathrm{E}\left(\mathbf{v_L}\mathbf{v_L^T}\right) \\ &= \mathrm{E}\left(\left(\mathbf{G}\mathbf{v_X}\right)\left(\mathbf{G}\mathbf{v_X}\right)^{\mathbf{T}}\right) \\ &= \mathrm{E}\left(\mathbf{G}\mathbf{v_X}\mathbf{v_X^T}\mathbf{G^T}\right) \\ &= \mathbf{G}\,\mathrm{E}\left(\mathbf{v_X}\mathbf{v_X^T}\right)\mathbf{G^T} \\ &= \mathbf{G}\mathbf{C_X}\mathbf{G^T} \end{aligned}$$

For future reference, the general form of the resulting equation $\mathbf{C_L} = \mathbf{G}\mathbf{C_X}\mathbf{G^T}$ is applicable to any problem involving an affine transformation (i.e., multiplication of a column vector by any

rectangular matrix, **G**). Note that for this particular problem, $\mathbf{C_X}$ is really the 3×3 submatrix taken from the original 4×4 matrix (which also included coefficients for the clock parameter τ). The covariance matrix in the local system $\mathbf{C_L}$ can be written in terms of its components:

$$\mathbf{C}_L = \sigma^2 \begin{pmatrix} \sigma_n^{\,2} & \sigma_{ne} & \sigma_{nh} \\ \sigma_{en} & \sigma_e^{\,2} & \sigma_{eh} \\ \sigma_{hn} & \sigma_{he} & \sigma_h^{\,2} \end{pmatrix}$$

We could then use this covariance, for example, to plot error ellipses in the horizontal plane.

4.2.4 Dilution of Precision

We can now define the various types of "dilution of precision" (DOP) as a function of diagonal elements of the covariance matrix in the local system:

$$\begin{aligned} VDOP &\equiv \sigma_h \\ HDOP &\equiv \sqrt{\sigma_n^2 + \sigma_e^2} \\ PDOP &\equiv \sqrt{\sigma_n^2 + \sigma_e^2 + \sigma_h^2} \\ TDOP &\equiv \sigma_\tau \\ GDOP &\equiv \sqrt{\sigma_n^2 + \sigma_e^2 + \sigma_h^2 + c^2\sigma_\tau^2} \end{aligned}$$

where, for example, *VDOP* stands for "vertical dilution of precision," H stands for horizontal, P for position, T for time, and G for geometric. As an example of how to interpret DOP, a standard deviation of 1 metre in observations would give a standard deviation in horizontal position of *HDOP* metres, and a standard deviation in the receiver clock bias of *TDOP* seconds. If *VDOP* had a value of 5, we could expect pseudorange errors of 1 metre to map into vertical position errors of 5 metres, and so on. As we have seen, the cofactor matrix and therefore the DOP values are purely a function of satellite geometry as observed by the receiver. A "good geometry" therefore gives low DOP values. A "bad geometry" can give very high DOP values. As a general rule, *PDOP* values larger than 5 are considered poor. If there are fewer than a sufficient number of satellites to produce a solution, or if 2 out of 4 satellites lie in approximately the same direction in the sky, then the cofactor matrix becomes singular, and the DOP values go to infinity. The above formulas assume that all 4 parameters (x, y, z, τ) are being estimated. Of course, if fewer than these are estimated, for example if height is not estimated, then the modified DOP values would get smaller, and they would no longer be generally infinity for only 3 satellites in view.

4.2.5 Mission Planning

Mission planning is the term used to describe the pre-analysis of the satellite geometry in advance of a survey. Essentially, it typically involves using commercial software to plot the DOP values as a function of time at a given geographical location. Since most applications involve local to regional distances, it is not too important which station's location is used for this analysis, since the satellites will appear approximately in the same position in the sky for

all stations. One thing that can vary a lot from station to station is the "elevation mask". Most software allow the user to specify which parts of the sky obstruct the view of the satellites (e.g., due to trees, buildings, or mountains). The elevation mask can substantially change the DOP values, so careful attention should be paid to this. Even if the elevation mask went down to the horizon, the user may wish to set it to 15 degrees all around, as research shows that data below 15 degrees is usually plagued by multipathing errors and other problems, such as cycle slips, and a low signal to noise ratio. As mentioned previously, the user might only be interested in horizontal position, where the height is known adequately in advance (e.g., for a boat at sea). Most software allow for DOP values to be computed under the assumption that height is fixed.

5. THE CARRIER PHASE OBSERVABLE

5.1 CONCEPTS

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", the principles of interferometry, and the Doppler effect. 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".

5.1.1 The Meaning of "Phase," "Frequency" and "Clock Time"

"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 2, the "phase" $\varphi(t)$ 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:

$$T(t) = k\big(\varphi(t) - \varphi_0\big)$$

where $T(t)$ is the time according to our clock at time t (whatever the clock may be), $\varphi_0 = \varphi(0)$ is so that the clock reads zero when $t = 0$, and k is a calibration constant, converting the 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 360° 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.

$$f \equiv \frac{d\varphi(t)}{dt}$$

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:

$$\varphi_{\text{ideal}} = f_0 t + \varphi_0$$

therefore

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

Since we want our a clock second to equal a conventional second ($T_{\text{ideal}}=t$), 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:

$$T(t) = \frac{\varphi(t) - \varphi_0}{f_0}$$

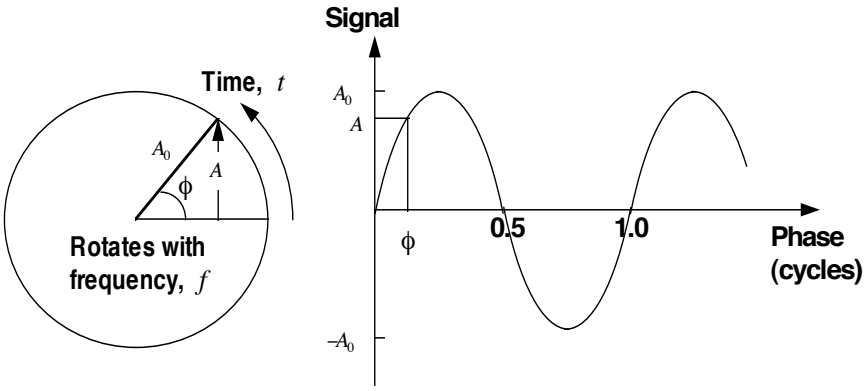

Figure 2: The meaning of phase.

5.1.2 How phase is related to a periodic signal

At time t, the height of point $A(t)$ above the centre of the circle in figure 2 is given by:

$$A(t) = A_0 \sin\left[2\pi\varphi(t)\right]$$

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 $\varphi(t)$ 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:

$$\begin{aligned} A_{\text{ideal}} &= A_0 \sin 2\pi\varphi_{\text{ideal}} \\ &= A_0 \sin\left(2\pi f_0 t + 2\pi\varphi_0\right) \\ &= \left(A_0 \cos 2\pi\varphi_0\right) \sin 2\pi f_0 t + \left(A_0 \sin 2\pi\varphi_0\right) \cos 2\pi f_0 t \\ &= A_0^S \sin\omega_0 t + A_0^C \cos\omega_0 t \end{aligned}$$

where the "angular frequency" $\omega_0 \equiv 2\pi f_0$ 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.

5.1.3 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 3.

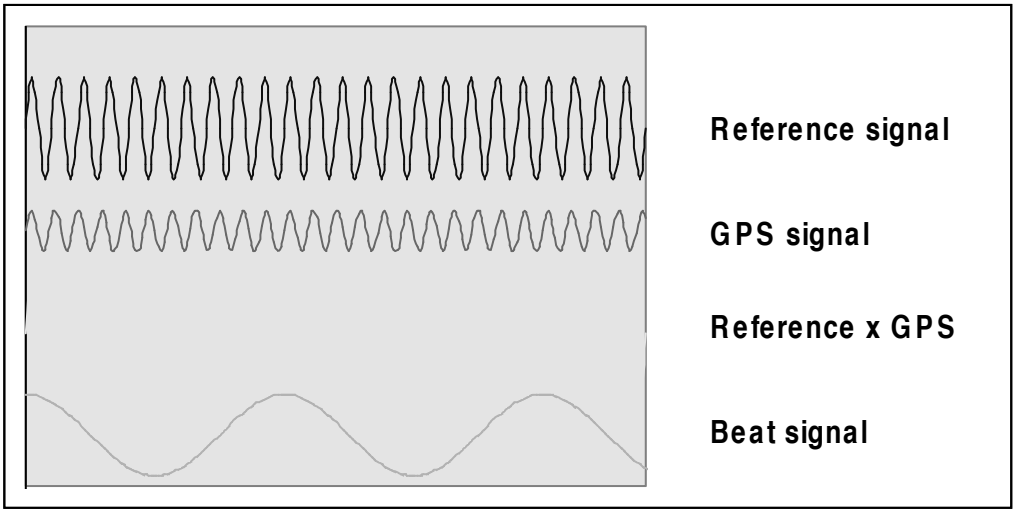

Figure 3: Producing a beat signal by mixing the carrier and replica signals

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

$$\begin{aligned} R(t) \otimes G(t) &= G_0 \sin 2\pi\varphi_G(t) \times R_0 \sin 2\pi\varphi_R(t) \\ &= \frac{G_0 R_0}{2}\Big[\cos 2\pi\big(\varphi_R(t) - \varphi_G(t)\big) - \cos 2\pi\big(\varphi_R(t) + \varphi_G(t)\big)\Big] \end{aligned}$$

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

$$\begin{aligned} B(t) &= \text{Filter}\big\{R(t) \otimes G(t)\big\} \\ &= \frac{G_0 R_0}{2}\cos 2\pi\big(\varphi_R(t) - \varphi_G(t)\big) \\ &\equiv B_0 \cos 2\pi\big(\varphi_B(t)\big) \end{aligned}$$

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

$$\varphi_B(t) \equiv \varphi_R(t) - \varphi_G(t)$$

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.

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

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.

5.1.4 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:

$$\Phi + N = \varphi_R - \varphi_G$$

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."

5.1.5 Interpretation of the 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:

$$N = (\text{integer portion of } \varphi_R - \varphi_G) - (\text{integer portion of } \Phi)$$

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.

5.1.6 Intuitive Model: The Doppler Effect

How can phase be used to measure distance? One way hinted at above is that the phase essentially tells you the clock time. As we shall see in the next section, we can develop phase in almost the same way as the pseudorange model. Another intuitive way of looking at it is to consider the Doppler effect. We are all familiar with the acoustic version of the Doppler effect, as we hear a vehicle's at a higher pitch when it is approaching, and a lower pitch when receding. Can we use the Doppler effect to design a distance measuring device?

Imagine two perfect clocks; one is at a fixed point, the other is approaching in a vehicle. Let both clocks be generating a sinusoidal signal. The frequency difference between the reference signal, and the approaching signal, increases with the vehicle's speed of approach. Let us build a receiver to mix the two signals and measure the beat signal. The beat frequency would be a measure of the speed.

Let us count the cycles of the beat signal; or better yet, let us measure the phase (cycles plus fractional cycles) of the beat signal. Clearly, the beat phase would be measures the change in distance to vehicle. We can therefore (after appropriate unit conversion) write the intuitive equation:

Beat phase = distance to vehicle + constant

This demonstrates that, although beat phase can be used to precisely measure change in distance from one time to another, there is an unknown constant which prevents us from knowing the full distance. This can be seen by considering moving the reference observer 10 metres away from the original position, and then repeating the experiment. The Doppler effect is clearly exactly the same, and the number of cycles passing by would not change. The very first value of the measured beat phase will indeed be different, but this single measurement cannot be used to infer distance. For example, we have already discussed that don't know what integer number of cycles to attribute to the first beat phase measurement.

5.2 CARRIER PHASE OBSERVATION MODEL

5.2.1 Carrier Beat Phase 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.

To summarise what we know already, the satellite carrier signal (from antenna) is mixed with reference signal generated by receiver's clock. The result, after high pass filtering, is a "beating" signal. The phase of this beating signal equals the reference phase minus the incoming GPS carrier phase from a satellite; however, it is ambiguous by an integer number of cycles. From this point on, "carrier beat phase" will be simply called "carrier phase" (but it should not be confused with the phase of the incoming signal!).

Observation of satellite S produces the carrier phase observable Φ^S:

$$\Phi^S(T) = \varphi(T) - \varphi^S(T) - N^S$$

where φ is the replica phase generated by the receiver clock, and φ^S is the incoming signal phase received from GPS satellite *S*. The measurement is made when the receiver clock time is *T*.

Now take the point of view that the phase of the incoming signal received at receiver clock time *T* is identical to the phase that was transmitted from the satellite at satellite clock time T^S.

$$\varphi^S(x,y,z,T) = \varphi^S_{\text{transmit}}(x^S, y^S, z^S, T^S_{\text{transmit}})$$

Of course, if we adopt this point of view, then we shall eventually have to consider the model of how long it takes a wavefront of constant phase to propagate from the satellite to the receiver, so that we may model the appropriate satellite clock time at the time of signal transmission, T^S. We return to that later.

As discussed previously, we can write clock time as a function of phase and nominal frequency:

$$T(t) = \frac{\varphi(t) - \varphi_0}{f_0}$$

We can therefore substitute all the phase terms with clock times:

$$\varphi(T) = f_0 T + \varphi_0$$
$$\varphi^S_{\text{transmit}}(T^S) = f_0 T^S_{\text{transmit}} + \varphi^S_0$$

Therefore, the carrier phase observable becomes:

$$\begin{aligned}\Phi^S(T) &= f_0 T + \varphi_0 - f_0 T^S - \varphi^S_0 - N^S \\ &= f_0\left(T - T^S\right) + \varphi_0 - \varphi^S_0 - N^S\end{aligned}$$

where we implicitly understand that the clock times refer to different events (reception and transmission, respectively).

We note that any term containing the superscript *S* are different for each satellites, but all other terms are identical. Receivers are designed and calibrated so that the phase constant φ_0 is identical for all satellites; that is, there should be no interchannel biases. Receivers should also sample the carrier phase measurements from all satellites at exactly the same time. (If the receivers have multiplexing electronics to save on cost, then the output should have been interpolated to the same epoch for all satellites). The time T^S will vary slightly from satellite to satellite, since the satellite transmission time must have been different for all signals to arrive at the same time. We also note that the last three terms are constant, and cannot be separated from each other. We can collectively call these terms the "carrier phase bias," which is clearly not an integer.

In preparation for multi-receiver and multi-satellite analysis, we now introduce the subscripts *A, B, C,* etc. to indicate quantities specific to receivers,, and we introduce superscripts *j, k, l,* etc. to identify satellite-specific quantities. We write the carrier phase observed by receiver *A* from satellite *j*:

$$\Phi_A^j(T_A) = f_0\left(T_A - T^j\right) + \varphi_{0A} - \varphi_0{}^j - N_A^j$$

Note that data should be sampled at exactly the same values of clock time (called "epochs") for all receivers, so all values of T_A are identical at a given epoch. However receivers clocks do not all run at exactly the same rate, therefore the true time of measurement will differ slightly from receiver to receiver. Also, note that each receiver-satellite pair has a different carrier phase ambiguity.

5.2.2 Range Formulation

It is convenient to convert the carrier phase model into units of range. This simplifies concepts, models, and software. In the range formulation, we multiply the carrier phase equation by the nominal wavelength.

$$\begin{aligned} L_A^j(T_A) &\equiv \lambda_0 \Phi_A^j(T_A) \\ &= \lambda_0 f_0\left(T_A - T^j\right) + \lambda_0\left(\varphi_{0A} - \varphi_0{}^j - N_A^j\right) \\ &= c\left(T_A - T^j\right) + \lambda_0\left(\varphi_{0A} - \varphi_0{}^j - N_A^j\right) \\ &\equiv c\left(T_A - T^j\right) + B_A^j \end{aligned}$$

where we still retain the name "carrier phase" for $L_A^j(T_A)$, which is in units of metres. We see immediately that this equation is identical to that for the pseudorange, with the exception of the "carrier phase bias," B_A^j which can be written (in units of metres):

$$B_A^j \equiv \lambda_0\left(\varphi_{0A} - \varphi_0{}^j - N_A^j\right)$$

Note that the carrier phase bias for (undifferenced) data is not an integer number of wavelengths, but also includes unknown instrumental phase offsets in the satellite and receiver.

We have not mentioned yet about any differences between carrier phase on the L1 and L2 channel. Although they have different frequencies, in units of range the above equations take on the same form. Actually, the clock bias parameters would be identical for both L1 and L2 phases, but the carrier phase bias would be different. The main difference comes when we develop the model in terms of the propagation delay, which is a function of frequency in the Earth's ionosphere.

5.2.3 Observation Model

We note that the first term in the carrier phase model is simply the pseudorange, and the second term is a constant. We have already developed a simplified model for pseudorange, so we can therefore write a model for carrier phase as follows:

$$\begin{aligned} L_A^j(T_A) &= c\left(T_A - T^j\right) + B_A^j \\ &= \rho_A^j(t_A, t^j) + c\tau_A - c\tau^j + Z_A^j - I_A^j + B_A^j \end{aligned}$$

In the above expression, we have explicitly included the delay on the signal due to the troposphere Z_A^j and the ionosphere $-I_A^j$ (the minus sign indicating that the phase velocity actually increases). Models for the atmospheric delay terms are beyond the scope of this text.

The model for pseudorange can be similarly improved, with the small difference that the ionospheric delay has a positive sign.

$$\begin{aligned} P_A^j(T_A) &= c\left(T_A - T^j\right) \\ &= \rho_A^j(t_A, t^j) + c\tau_A - c\tau^j + Z_A^j + I_A^j \end{aligned}$$

This is because, from physics theory, any information, such as the +1 and −1 "chips" which are modulated onto the carrier wave, must travel with the "group velocity" rather than "phase velocity". According to the theory of relativity, information can not be transmitted faster than c. From the physics of wave propagation in the ionosphere, it can be shown that the group delay is (to a very good first order approximation) precisely the same magnitude, but opposite sign of the phase delay (which is really a phase "advance").

5.2.4 Accounting for Time-Tag Bias

Before proceeding, we return to the problem posed in our discussion of the pseudorange model, that we typically do not know the true time of signal reception t_A which we need to calculate the satellite-receiver range term $\rho_A^j(t_A, t^j)$ precisely. From section 3.3.1, the true time of reception can be written:

$$t_A = T_A - \tau_A$$

where the epoch T_A is known exactly, as it is the receiver clock time written into the data file with the observation (and hence called the "time-tag"). However, the receiver clock bias τ_A is not known initially, but could be as large as milliseconds. The problem is that, due to satellite motion and Earth rotation, the range will change by several metres over the period of a few milliseconds, so we must be careful to account for this for precision work (especially when using the carrier phase observable). For precision work (1 mm), we should use a value τ_A that is accurate to 1 μs.

There are various approaches to dealing with this in GPS geodetic software, which typically use some combination of the following methods:

- use values of the receiver clock bias computed in a first step using a pseudorange point position solution at each epoch;
- iterate the least-squares procedure, processing both carrier phase and pseudorange data simultaneously, and using estimates of the clock bias to compute the true receive time, and therefore the new range model;
- use an estimate $\hat{t}^j$ of the true transmit time t^j to compute the satellite position.

$$\begin{aligned}\hat{t}^j &= \hat{T}^j - \tau^j \\ &= \left(T_A - P_A^j/c\right) + \tau^j\end{aligned}$$

  where the satellite clock bias τ^j is obtained from the Navigation Message. One can then directly compute the range term and true receive time with sufficient precision, provided the approximate station coordinates are known to within 300 m (corresponding to the 1 μs timing requirement). Interestingly, this is the basis for "time transfer," since it allows one to compute the receiver clock bias using pseudorange data from only one GPS satellite. (For precise time transfer, two GPS satellites are always in operation with no S/A switched on.) As a method for computing range for precise positioning, this is not often used, perhaps for the reason that it is not a pure model, as it depends on pseudorange data and approximate positions.
- one can take a modelling "short cut" to avoid iteration by expanding the range model as a first order Taylor series. Since this method often appears in the development of the observation equation in textbooks, we discuss it in more detail here.

5.2.5 A Note on the Range-Rate Term

The observation equation can be approximated as follows:

$$\begin{aligned}L_A^j(T_A) &= \rho_A^j(t_A, t^j) + c\tau_A - c\tau^j + \mathrm{Z}_A^j - I_A^j + B_A^j \\ &= \rho_A^j(T_A - \tau_A, t^j) + c\tau_A - c\tau^j + \mathrm{Z}_A^j - I_A^j + B_A^j \\ &\approx \rho_A^j(T_A, t'^j) - \dot{\rho}_A^j\tau_A - c\tau^j + \mathrm{Z}_A^j - I_A^j + B_A^j \\ &= \rho_A^j(T_A, t'^j) + \left(c - \dot{\rho}_A^j\right)\tau_A - c\tau^j + \mathrm{Z}_A^j - I_A^j + B_A^j\end{aligned}$$

where we see that the effect can be accounted for by introducing the modelled range rate (i.e., the relative speed of the satellite in the direction of view). The "prime" for the satellite transmit time t'^j (which is used to compute the satellite coordinates) is to indicate that it is not the true transmit time, but the time computed using the nominal receive time T_A. A first order Taylor expansion has been used. The higher order terms will only become significant error sources if the receiver clock bias is greater than about 10 ms, which does not usually happen with modern receivers. In any case, clock biases greater than this amount would result in a worse error in relative position due to the effect of S/A (see section 5.3.1).

Textbooks sometimes include a "range rate" term in the development of the phase observation model, even though, strictly speaking, it is unnecessary. After all, the first line line of the above equation is correct, and the lack of a priori knowledge of the receiver clock bias can easily be dealt with by least-squares iteration, or prior point positioning using the pseudorange. On the other hand, it is nevertheless instructional to show the above set of

equations, since it does illustrate that it is more correct to use $\left(c - \dot{\rho}_A^j\right)$ as the partial derivatives with respect to the receiver clock in the design matrix, rather than simply using c (section 4.1.2). This is crucial if one is not initialising clocks using point position solutions or iteration (as is typical, for example, with the GIPSY OASIS II software). It is not important if initialisation of τ_A is achieved with 1 μs accuracy.

In the expressions to follow, we shall not explicitly include the range rate term on the assumption that time-tag bias has been handled one way or another.

5.3 DIFFERENCING TECHNIQUES

5.3.1 Single Differencing

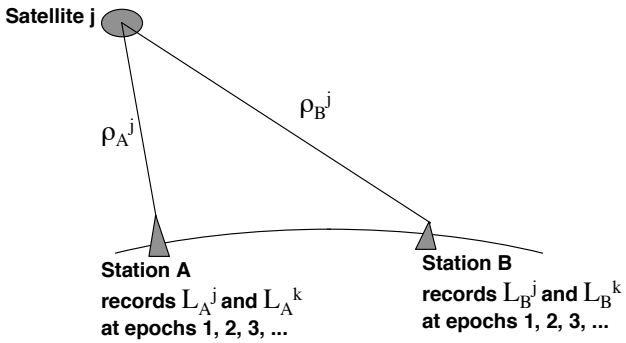

Figure 4: Single differencing geometry

The purpose of "single differencing" is to eliminate satellite clock bias. Consider the observation equations for two receivers, *A* and *B* observing same satellite, *j*:

$$L_A^j = \rho_A^j + c\tau_A - c\tau^j + Z_A^j - I_A^j + B_A^j$$
$$L_B^j = \rho_B^j + c\tau_B - c\tau^j + Z_B^j - I_B^j + B_B^j$$

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

$$\begin{aligned}\Delta L_{AB}^j &\equiv L_A^j - L_B^j \\ &= \left(\rho_A^j + c\tau_A - c\tau^j + Z_A^j - I_A^j + B_A^j\right) - \left(\rho_B^j + c\tau_B - c\tau^j + Z_B^j - I_B^j + B_B^j\right) \\ &= \left(\rho_A^j - \rho_B^j\right) + \left(c\tau_A - c\tau_B\right) - \left(c\tau^j - c\tau^j\right) + \left(Z_A^j - Z_B^j\right) - \left(I_A^j - I_B^j\right) - \left(B_A^j - B_B^j\right) \\ &= \Delta\rho_{AB}^j + c\Delta\tau_{AB} + \Delta Z_{AB}^j - \Delta I_{AB}^j + \Delta B_{AB}^j\end{aligned}$$

where we use the double-subscript to denote quantities identified with two receivers, and the triangular symbol as a mnemonic device, to emphasise that the difference is made between two points on the ground. The geometry of single differencing is illustrated in Figure 4.

An assumption has been made, that the satellite clock bias τ^j is effectively identical at the slightly different times that the signal was transmitted to *A* and to *B*. 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 receivers A and B, and this should be remembered when computing the term $\Delta\rho^j_{AB}$.

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".

5.3.2 Double Differencing

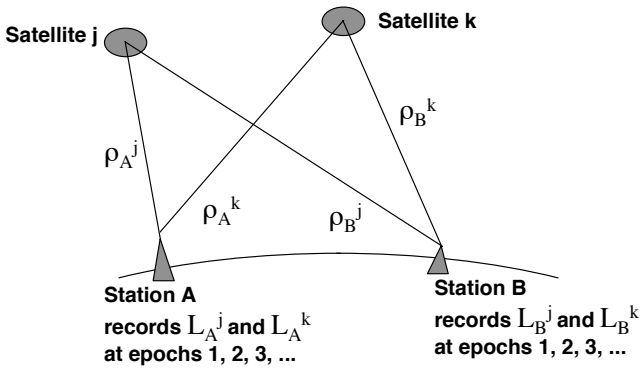

Figure 5: Double differencing geometry.

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

$$\Delta L^j_{AB} = \Delta\rho^j_{AB} + c\Delta\tau_{AB} + \Delta Z^j_{AB} - \Delta I^j_{AB} + \Delta B^j_{AB}$$
$$\Delta L^k_{AB} = \Delta\rho^k_{AB} + c\Delta\tau_{AB} + \Delta Z^k_{AB} - \Delta I^k_{AB} + \Delta B^k_{AB}$$

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

$$\begin{aligned}\nabla\Delta L_{AB}^{jk} &\equiv \Delta L_{AB}^{j} - \Delta L_{AB}^{k} \\ &= \left(\Delta\rho_{AB}^{j} + c\Delta\tau_{AB} + \Delta Z_{AB}^{j} - \Delta I_{AB}^{j} + \Delta B_{AB}^{j}\right) - \left(\Delta\rho_{AB}^{k} + c\Delta\tau_{AB} + \Delta Z_{AB}^{k} - \Delta I_{AB}^{k} + \Delta B_{AB}^{k}\right) \\ &= \left(\Delta\rho_{AB}^{j} - \Delta\rho_{AB}^{k}\right) + \left(c\Delta\tau_{AB} - c\Delta\tau_{AB}\right) + \left(\Delta Z_{AB}^{j} - \Delta Z_{AB}^{k}\right) - \left(\Delta I_{AB}^{j} - \Delta I_{AB}^{k}\right) - \left(\Delta B_{AB}^{j} - \Delta B_{AB}^{k}\right) \\ &= \nabla\Delta\rho_{AB}^{jk} + \nabla\Delta Z_{AB}^{jk} - \nabla\Delta I_{AB}^{jk} + \nabla\Delta B_{AB}^{jk}\end{aligned}$$

where we use the double-superscript to denote quantities identified with two satellites, and the upside-down triangular symbol as a mnemonic device, to emphasise that the difference is made between two points in the sky. Figure 5 illustrates the geometry of double differencing.

A point worth mentioning, is that although the receiver clock error has been eliminated to first order, the residual effect due "time tag bias" on the computation of the range term (section 5.2.4) does not completely cancel, and still needs to be dealt with if the receiver separation is large.

Any systematic effects due to unmodelled atmospheric errors are generally increased slightly by approximately 40% by double differencing as compared to single differencing. Similarly, random errors due to measurement noise and multipath are increased. Overall, random errors are effectively doubled as compared with the undifferenced observation equation. On the other hand, the motivation for double differencing is to remove clock bias, which would create much larger errors.

One could process undifferenced or single differenced data, and estimate clock biases. In the limit that clock biases are estimated at every epoch (the "white noise clock model"), these methods become almost identical, provided a proper treatment is made of the data covariance (to be described later). It is almost, but not quite identical, because differencing schemes almost always involve pre-selection of baselines in a network to form single differences, and data can be lost by lack of complete overlap of the observations to each satellite. (This problem can be minimised by selecting the shortest baselines in the network to process, and by assuring that no more than one baseline be drawn to a receiver with a significant loss of data).

5.3.3 Double Differenced Ambiguity

The double difference combination has an additional advantage, in that the ambiguity is an integer:

$$\begin{aligned}\nabla\Delta B_{AB}^{jk} &= \Delta B_{AB}^{j} - \Delta B_{AB}^{k} \\ &= \left(B_A^j - B_B^j\right) - \left(B_A^k - B_B^k\right) \\ &= \lambda_0\left(\varphi_{0A} - \varphi_0{}^j - N_A^j\right) - \lambda_0\left(\varphi_{0B} - \varphi_0{}^j - N_B^j\right) - \lambda_0\left(\varphi_{0A} - \varphi_0{}^k - N_A^k\right) + \lambda_0\left(\varphi_{0B} - \varphi_0{}^k - N_B^k\right) \\ &= -\lambda_0\left(N_A^j - N_B^j - N_A^k + N_B^k\right) \\ &= -\lambda_0 \nabla\Delta N_{AB}^{jk}\end{aligned}$$

Hence we can write the double differenced phase observation equation:

$$\nabla\Delta L_{AB}^{jk} = \nabla\Delta\rho_{AB}^{jk} + \nabla\Delta Z_{AB}^{jk} - \nabla\Delta I_{AB}^{jk} - \lambda_0 \nabla\Delta N_{AB}^{jk}$$

From the point of view of estimation, it makes no difference whether we use a minus or plus sign for N, so long as the partial derivative has a consistent sign (which, for the above equation, would be $-\lambda_0$).

5.3.4 Triple Differencing

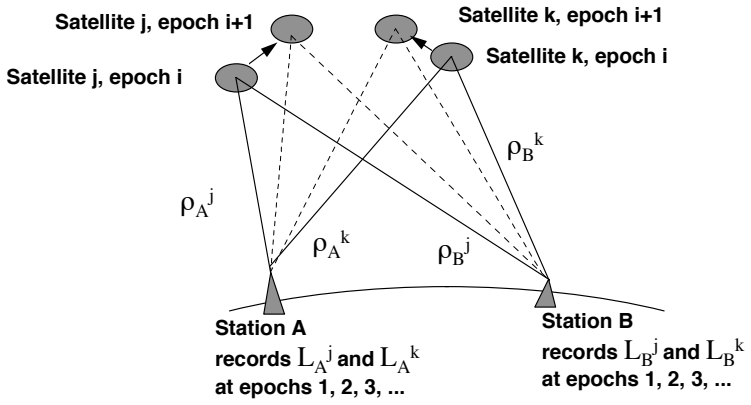

Figure 6: Triple differencing geometry

The purpose of "triple differencing" is to eliminate the integer ambiguity. Consider two successive epochs (i, i+1) of double differenced data from receivers A and B observing satellites j and k:

$$\begin{aligned}\nabla\Delta L_{AB}^{jk}(i) &= \nabla\Delta\rho_{AB}^{jk}(i) + \nabla\Delta Z_{AB}^{jk}(i) - \nabla\Delta I_{AB}^{jk}(i) - \lambda_0 \nabla\Delta N_{AB}^{jk} \\ \nabla\Delta L_{AB}^{jk}(i+1) &= \nabla\Delta\rho_{AB}^{jk}(i+1) + \nabla\Delta Z_{AB}^{jk}(i+1) - \nabla\Delta I_{AB}^{jk}(i+1) - \lambda_0 \nabla\Delta N_{AB}^{jk}\end{aligned}$$

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

$$\begin{aligned}\delta(i,i+1)\nabla\Delta L_{AB}^{jk} &\equiv \nabla\Delta L_{AB}^{jk}(i+1) - \nabla\Delta L_{AB}^{jk}(i) \\ &= \delta(i,i+1)\nabla\Delta\rho_{AB}^{jk}(i) + \delta(i,i+1)\nabla\Delta Z_{AB}^{jk}(i) - \delta(i,i+1)\nabla\Delta I_{AB}^{jk}(i)\end{aligned}$$

where we use the delta symbol to indicate the operator that differences data between epochs. Figure 6 illustrates triple differencing geometry.

The triple difference only removes the ambiguity if it has not changed during the time interval between epochs. Any cycle slips will appear as outliers, and can easily be removed by conventional techniques. This is unlike the situation with double differencing, where cycle slips appear as step functions in the time series of data.

The disadvantage of the triple difference is that it introduces correlations between observations in time. Generally, increasing correlations in data has the property of decreasing the data weights. With triple differencing, the degradation in precision is substantial; so triple differenced data are inappropriate for precise surveys. On the other hand, it is a very useful method for obtaining better nominal parameters for double differencing (to ensure linearity), and it is a robust method, due to the ease with which cycle slips can be identified and removed.

It can be shown that triple difference solution is identical to the double differenced solution, provided just one epoch double differenced equation is included for the first point in a data arc, along with the triple differences, and provided the full data covariance matrix is used to compute the weight matrix. This special approach can provide tremendous savings in computation time over straightforward double differencing, while retaining robustness.

6. RELATIVE POSITIONING USING CARRIER PHASE

6.1 SELECTION OF OBSERVATIONS

6.1.1 Linear Dependence of Observations

We can usually form many more possible combinations of double differenced observations than there are original data. This poses a paradox, since we cannot create more information than we started with. The paradox is resolved if we realise that some double differences can be formed by differencing pairs of other double differences. It then becomes obvious that we should not process such observations, otherwise we would be processing the same data more than once. This would clearly be incorrect.

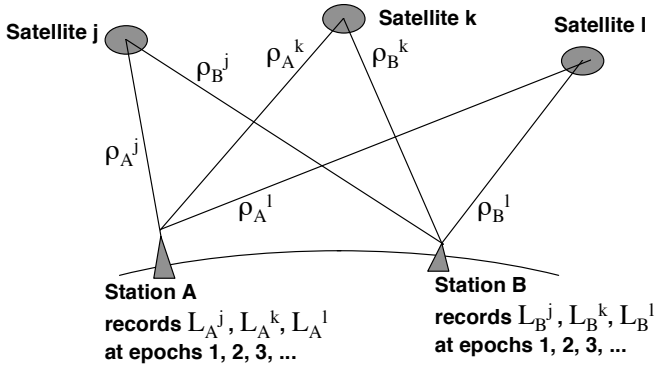

Figure 7: Double difference geometry with 3 satellites.

Figure 7 illustrates the simplest example of the problem. In this example, we have 3 satellites *j*, *k* and *l*, observed by two receivers *A* and *B*. If we ignore trivial examples (e.g., $L_{AB}^{jk} = -L_{AB}^{kj}$), there are 3 possible double differences that can be formed:

$$L_{AB}^{jk} = \left(L_A^j - L_B^j\right) - \left(L_A^k - L_B^k\right)$$
$$L_{AB}^{jl} = \left(L_A^j - L_B^j\right) - \left(L_A^l - L_B^l\right)$$
$$L_{AB}^{lk} = \left(L_A^l - L_B^l\right) - \left(L_A^k - L_B^k\right)$$

Note that we can form any one of these observations as a *linear combination* of the others:

$$L_{AB}^{jk} = L_{AB}^{jl} + L_{AB}^{lk}$$
$$L_{AB}^{jl} = L_{AB}^{jk} - L_{AB}^{lk}$$
$$L_{AB}^{lk} = L_{AB}^{jk} - L_{AB}^{jl}$$

The data set $\left\{L_{AB}^{jk}, L_{AB}^{jl}, L_{AB}^{lk}\right\}$ is therefore said to be *linearly dependent*. A linearly independent set must be used for least squares. Examples of appropriate linearly independent sets in this example are:

$$\left\{L_{AB}^{jk}, L_{AB}^{jl}\right\} = \Lambda^j \equiv \left\{L_{AB}^{ab} | a = j; b \neq j\right\}$$
$$\left\{L_{AB}^{kj}, L_{AB}^{kl}\right\} = \Lambda^k \equiv \left\{L_{AB}^{ab} | a = k; b \neq k\right\}$$
$$\left\{L_{AB}^{lj}, L_{AB}^{lk}\right\} = \Lambda^l \equiv \left\{L_{AB}^{ab} | a = l; b \neq l\right\}$$

6.1.2 The Reference Satellite Concept

The "reference satellite concept" involves using either set Λ^j, Λ^k or Λ^l throughout the data set. For example, double differences in set Λ^l all involve the satellite *l*. Any set is equally as valid, and will produce identical solutions provided the data covariance is properly constructed (see the next section). Obviously, the reference satellite itself has to have data at every epoch, otherwise data will be lost. This can cause problems for less sophisticated software. Typically, a reference satellite should be picked which has the longest period in view. A better algorithm is to select a reference satellite epoch by epoch.

Our simple example can be easily extended to more than 3 satellites. For example consider satellites 1, 2, 3, 4 and 5 in view. We can pick satellite 4 as the reference satellite; therefore our linearly independent set is:

$$\Lambda^4 \equiv \left\{L_{AB}^{ab} | a = 4; b \neq 4\right\}$$
$$= \left\{L_{AB}^{41}, L_{AB}^{42}, L_{AB}^{43}, L_{AB}^{45}\right\}$$

Note that for a single baseline (i.e. 2 receivers), the number of linearly independent double differenced observations is $s-1$, where s is the number of satellites being tracked.

6.1.3 The Reference Station Concept

However, if we have a network of more than 2 receivers, we must account for the fact that double differenced data from the set of all baselines are linearly dependent. We therefore introduce the "reference station" concept, where our set of double differences all include a common reference station. This guarantees linear independence. For example, consider satellites 1, 2, 3 and 4 being tracked by stations A, B, and C. If we pick our reference satellite to be 3, and reference station to be B, then our chosen set is:

$$\begin{aligned}\Lambda^3_B &\equiv \left\{ L^{ab}_{cd} \middle| a = 3; b \neq 3; c = B, d \neq B \right\} \\ &= \left\{ L^{31}_{BA}, L^{32}_{BA}, L^{34}_{BA}, L^{31}_{BC}, L^{32}_{BC}, L^{34}_{BC} \right\}\end{aligned}$$

Note that the number of linearly independent double differenced observations is $(s-1)(r-1)$, where s is the number of satellites being tracked, and r is the number of receivers. So, in our previous example, 3 receivers and 4 satellites gives 6 observations. This assumes that s satellites are observed by all stations. This may not be the case, either due to obstructions, receiver problems, or because the receivers are separated by such a large distance that the satellite is not above the horizon for some receivers.

If using the reference station concept, it is therefore best to choose a receiver close to the middle of a large network, with few obstructions, and no hardware problems, otherwise the set of double differences may not be as complete as it could be. The reference station concept is obviously not optimal, and is seriously problematic for large networks. A better strategy for large networks is to select short baselines that connect together throughout the entire network, being careful not to introduce linear dependent observations, by not including any closed polygons (such as triangles) in the network. In principle, there must be only one possible path between pairs of stations. An even better strategy would be to optimise this choice for every epoch.

6.1.4 Solution Uniqueness

It should be stressed that, if all stations were tracking the same set of satellites at all epochs, then the selection of reference station and reference satellite will not matter, since an identical solution will be produced whatever the selection. This assumes that the data weight matrix is properly constructed (as described below) and that no data outliers are removed.

The problem of linear dependence usually introduces a level of arbitrariness into the solutions due to violation of the above assumptions. This problem is also true even if the previously suggested improvements are made to the reference station concept, since the user typically has to make decisions on which baselines to process (even for more sophisticated software). This is somewhat unsatisfactory, since it is there generally no unique solution, However, experience shows that any reasonable selection will only produce small differences in the final solutions.

There is a way to produce a unique solution, and that is to process undifferenced observations, estimating clock parameters at each epoch. As stated previously, this will produce a solution identical to double differencing under ideal conditions. This class of software is not typically available commercially; however, it is should be stressed that double differencing software does not produce significantly inferior results for most situations. What is far more important is the quality of the observable models, the range of appropriate estimation options, and the ability to detect and deal with cycle slips and outliers.

6.2 BASELINE SOLUTION USING DOUBLE DIFFERENCES

6.2.1 Simplified Observation Equations

We now show how relative coordinates can be estimated between two receivers using the double differenced carrier phase data. We start by simplifying the observation equation, assuming that the relative atmospheric delay is negligible for short distances between receivers. We also drop the symbols "$\nabla\Delta$" of the previous section to simplify the notation. We shall therefore use the following simplified observation equation:

$$L_{AB}^{jk} = \rho_{AB}^{jk} - \lambda_0 N_{AB}^{jk}$$

6.2.2 General Procedure

Processing double differenced data from two receivers results in a "baseline solution." The estimated parameters include the vector between the two receivers, in Cartesian coordinates $(\Delta x, \Delta y, \Delta z)$ and may include parameters to model the tropospheric delay. In addition, the ambiguity parameters N_{AB}^{jk} for each set of double differences to specific satellite pairs (j, k) must be estimated.

The observation equations therefore require linearisation in terms of all these parameters (according to the process explained in section 4.1). Typically, one station is held fixed at good nominal coordinates, which quite often come from an initial pseudorange point position solution. We should mention, however, that due to S/A, point position solutions can have substantial errors (100 m) which may create significant errors in the double differenced observation model, and in the design matrix.

If we call the fixed station A, then estimating the baseline vector is equivalent to estimating the coordinates of station B. It is convenient to formulate the problem to estimate parameters (x_B, y_B, z_B). For example, consider a GPS survey between stations A and B, which observe satellites 1, 2, 3 and 4 for every epoch in the session, where we arbitrarily pick satellite 2 as the reference satellite. For every epoch i, we have the following linearly independent set of 3 double differenced observations:

$$\Lambda^2(i) \equiv \left\{ L_{AB}^{ab}(i) \middle| a = 2; b \neq 2 \right\}$$
$$= \left\{ L_{AB}^{21}(i), L_{AB}^{23}(i), L_{AB}^{24}(i) \right\}$$

We therefore have the parameter set $\left\{ x_B, y_B, z_B, N_{AB}^{21}, N_{AB}^{23}, N_{AB}^{24} \right\}$. If any cycle slips had occured and could not be corrected, then additional ambiguity parameters must be added to the list.

As in Section 3.4.1, the linearised observation equations can be expressed in the form

$$\mathbf{b} = \mathbf{A}\mathbf{x} + \mathbf{v}$$

where the residual observations are listed in the **b** matrix, which has dimensions $d \times 1$, where d is the number of linearly independent double differenced data. The design matrix **A** has dimensions $d \times p$ where p is the number of parameters, and the parameter corrections are contained in the **x** matrix, which has dimensions $p \times 1$. The observation errors are represented by the **v** matrix, which has the same dimensionality as **b**. We shall discuss the design matrix later on.

It is important to use a "weighted least squares" approach, because of correlations in the double differenced data. We shall not not derive the weighted least squares estimator here, but for completeness, the solution is given here:

$$\hat{\mathbf{x}} = \left(\mathbf{A}^{\mathrm{T}} \mathbf{W} \mathbf{A} \right)^{-1} \mathbf{A}^{\mathrm{T}} \mathbf{W} \mathbf{b}$$

where **W** is the data weight matrix, to be derived later on, and **b** is a vector containing the double-differenced residual observations.

The covariance matrix for the estimated parameters is given by:

$$\mathbf{C}_{\mathbf{x}} = \left(\mathbf{A}^{\mathrm{T}} \mathbf{W} \mathbf{A} \right)^{-1}$$

The covariance matrix can be used to judge whether the theoretically expected precision from the observation scenario is sufficient to allow ambiguities to be fixed to integer values. If ambiguity parameters can be fixed in the model, a theoretically more precise solution can be generated from the same data, but without estimating the ambiguities. This process will necessarily reduce the covariance matrix, lowering the expected errors in the station coordinates. This does not necessarily mean that the solution is better, but that it statistically ought to be better, assuming the integers were correctly fixed. The assessment of solution accuracy goes beyond the scope of this discussion, but basically one can compare results with previous results (using GPS, or even some other technique). In addition, how well the data are fit by the model is reflected in the standard deviation of the post-fit residuals.

6.2.3 The Design Matrix

The coefficients of the design matrix can be illustrated by looking at a single row, for example, corresponding to observation $L_{AB}^{24}(i)$:

$$A_{AB}^{24}(i) = \left(\frac{\partial L_{AB}^{24}(i)}{\partial x_B} \quad \frac{\partial L_{AB}^{24}(i)}{\partial y_B} \quad \frac{\partial L_{AB}^{24}(i)}{\partial z_B} \quad \frac{\partial L_{AB}^{24}(i)}{\partial N_{AB}^{21}} \quad \frac{\partial L_{AB}^{24}(i)}{\partial N_{AB}^{23}} \quad \frac{\partial L_{AB}^{24}(i)}{\partial N_{AB}^{24}} \right)$$
$$= \left(\frac{\partial \rho_{AB}^{24}(i)}{\partial x_B} \quad \frac{\partial \rho_{AB}^{24}(i)}{\partial y_B} \quad \frac{\partial \rho_{AB}^{24}(i)}{\partial z_B} \quad 0 \quad 0 \quad -\lambda_0 \right)$$

As an example of one of the partial derivatives for one of the coordinates:

$$\frac{\partial \rho_{AB}^{24}(i)}{\partial x_B} = \frac{\partial}{\partial x_B}\left(\rho_A^2(i) - \rho_B^2(i) - \rho_A^4(i) + \rho_B^4(i) \right)$$
$$= \frac{\partial \rho_A^2(i)}{\partial x_B} - \frac{\partial \rho_B^2(i)}{\partial x_B} - \frac{\partial \rho_A^4(i)}{\partial x_B} + \frac{\partial \rho_B^4(i)}{\partial x_B}$$
$$= \frac{\partial \rho_B^4(i)}{\partial x_B} - \frac{\partial \rho_B^2(i)}{\partial x_B}$$
$$= \frac{x_{B0} - x^4(i)}{\rho_B^4(i)} - \frac{x_{B0} - x^2(i)}{\rho_B^2(i)}$$

6.2.4 Minimum Data Requirements for Least Squares

For a least squares solution, a necessary condition is that the number of data exceed the number of estimated parameters

$$d \geq p$$

where we allow for the "perfect fit solution" ($d = p$). Under the assumption that all receivers track the same satellites for every epoch, the number of linearly independent double differences is

$$d = q(r-1)(s-1)$$

where q is the number of epochs, r the number of receivers, and s is the number of satellites. Assuming no cycle slip parameters:

$$p = 3 + (r-1)(s-1)$$

where there are $(r-1)(s-1)$ ambiguity parameters. Therefore,

$$q(r-1)(s-1) \geq 3 + (r-1)(s-1)$$
$$(q-1)(r-1)(s-1) \geq 3$$

Now, we know that $s \geq 2$ and $r \geq 2$ for us form double differences. Therefore, we can deduce that $q \geq 4$ if we have the minimal geometry of 2 receivers and 2 satellites (only one double difference per epoch!). Obviously, this minimal configuration is very poor geometrically, and would not be recommended as a method of precise positioning.
Note that no matter how many receivers or satellites we have, q is an integer, and therefore under any circumstance, we must have at least $q \geq 2$. That is, we cannot do single epoch relative positioning, if we are estimating integer ambiguities. If we can find out the ambiguities by some other means, then single epoch relative positioning is possible. Otherwise, we have to wait for the satellite geometry to change sufficiently in order to produce a precise solution.

For a single baseline $r = 2$ with 2 epochs of data $q = 2$ (which we should assume are significantly separated in time), the minimum number of satellites to produce a solution is condition $s \geq 4$. Interestingly, this corresponds to the minimum number of satellites for point positioning. If a tropospheric parameter were also being estimated, the condition would be $s \geq 5$. Of course, these conditions can be relaxed if we have more than 2 epochs, however it is the end-points of a data arc which are most significant, since they usually represent the maximum geometrical change which we require for a good solution. In summary, one can achieve very good results over short distances with only 4 satellites, but over longer distances where the troposphere must be estimated, a minimum of 5 satellites is recommended (at least some time during the session).

6.3 STOCHASTIC MODEL

6.3.1 Statistical Dependence of Double Differences

We have seen how double differences can be linearly dependent. The problem we now address is that double differenced observations that involve a common receiver and common satellite are *statistically dependent*. For example, at a given epoch, double differences L_{AB}^{21}, L_{AB}^{23} and L_{AB}^{24} are correlated due to the single differenced data in common, L_{AB}^{2}. Any measurement error in this single difference will contribute exactly the same error to each of the double differences. Therefore, a positive error in L_{AB}^{21} is statistically more likely to be accompanied by a positive error in L_{AB}^{23}. As another example, if we are processing a network using a reference satellite j and reference receiver A, all double differences in the linearly independent set will be statistically dependent because of the data in common, L_{A}^{j}.

6.3.2 Data Weight Matrix for Double Differences

In a situation where data are correlated, weighted least squares is appropriate. To complete our description of how to compute a relative position estimate, we therefore need to explain how to compute the appropriate data weight matrix, **W**. The construction of **W** can be generally called the "stochastic model," which describes the statistical nature of our data (as opposed to the "functional model" described so far, from which the observables can be computed deterministically.)

(As an aside for more advanced readers, some software process undifferenced observations, estimating clock biases as "stochastic parameters" at every epoch. It should be emphasised that there is a equivalence between explicit estimation of "stochastic parameters," and the use of an appropriate "stochastic model" which, in effect, accounts for the missing parameters through the introduction of correlations in the data. In principle, any parameter can either be estimated as part of the functional model, or equivalently removed using an appropriate stochastic model. To go more into this would be beyond the scope of this text.)

The weight matrix is the inverse of the covariance matrix for the double differenced data:

$$\mathbf{W} = \mathbf{C}_{\nabla\Delta}^{-1}$$

which has dimensions $q(r-1)(s-1) \times q(r-1)(s-1)$.

We start by assuming a covariance matrix for undifferenced data (i.e., the actually recorded data), which has dimensions $qrs \times qrs$. Typically, this is assumed to be diagonal, since the receiver independently measures the signals from each satellite separately. We shall, however, keep the form general. So the problem is, given a covariance matrix for undifferenced data, how do we compute the covariance matrix for double-differenced data? This is achieved using the rule of propagation of errors, which we have already seen in section 4.2.3, where geocentric coordinates were mapped into topocentric coordinates using an affine transformation. By analogy, we can deduce that the covariance of double-differenced data can be written:

$$\mathbf{C}_{\nabla\Delta} = \mathbf{D}\mathbf{C}\mathbf{D}^{\mathbf{T}}$$

where **D** is the matrix which transforms a column vector of the recorded data into a column vector of double differenced data:

$$\nabla\Delta\mathbf{L} = \mathbf{D}\mathbf{L}$$

Clearly, **D** is a rectangular matrix with the number of rows equal to the number of linearly independent double-differenced data, and the number of columns equal to the number of recorded data. Using our previous assumptions, **D** has dimensions $q(r-1)(s-1) \times qrs$. The components of **D** must have values of either +1, −1, or 0, arranged such that we produce a linearly independent set of double differences (see section 6.1.1). To complete this discussion, the double differenced data weight matrix can be written:

$$\mathbf{W} = \left(\mathbf{D}\mathbf{C}\mathbf{D}^{\mathbf{T}}\right)^{-1}$$

6.3.3 Covariance Matrix for Estimated Parameters

As we have already seen, for weighted least squares we can write the computed covariance matrix for estimated parameters as:

$$\mathbf{C}_{\mathbf{x}} = \left(\mathbf{A}^{\mathbf{T}}\mathbf{W}\mathbf{A}\right)^{-1}$$

We can now write down the full expression for the computed covariance matrix, by substituting for the double differenced data weight matrix **W**:

$$\mathbf{C_X} = \left(\mathbf{A}^{\mathrm{T}} \left(\mathbf{D}\mathbf{C}\mathbf{D}^{\mathrm{T}} \right)^{-1} \mathbf{A} \right)^{-1}$$

As mentioned above, for the (undifferenced) data covariance **C** we often use a diagonal matrix, assuming a value for the standard deviation of an observation. Typical realistic values for this are several mm. Although the receiver can usually measure the phase with better precision than a mm, the post-fit residuals typically show several mm standard deviations, due to unmodelled errors such as multipath.

Even using such an inflated value for measurement precision might not produce a realistic covariance matrix for station coordinates. This is partly due to two effects: (i) unmodelled errors can be correlated with the parameters being estimated (an "aliasing effect"), and (ii) post-fit almost always show some degree of time-correlation (e.g., due to multipath). A simple, and often surprisingly effective way to deal with this problem, is to multiply the final coordinate covariance matrix by an empirical scaling factor, inferred "by experience," according to the brand of software being used, the observation scenario, and the estimation strategy used.

7. INTRODUCING HIGH PRECISION GPS GEODESY

7.1 HIGH PRECISION SOFTWARE

The observable model discussed so far has been very basic, as it glosses over advanced features that are important for high precision software. Several software packages have been developed since the 1980's that are capable of delivering high precision geodetic estimates over long baselines. This software is a result of intensive geodetic research, mainly by universities and government research laboratories.

Typical features of such software include:

- orbit integration with appropriate force models;
- accurate observation model (Earth model, media delay...) with rigorous treatment of celestial and terrestrial reference systems;
- reliable data editing (cycle-slips, outliers);
- estimation of all coordinates, orbits, tropospheric bias, receiver clock bias, polar motion, and Earth spin rate;
- ambiguity resolution algorithms applicable to long baselines;
- estimation of reference frame transformation parameters and kinematic modelling of station positions to account for plate tectonics and co-seismic displacements.

We can summarise the typical quality of geodetic results from 24 hours of data:

- relative positioning at the level of few parts per billion of baseline length;
- absolute (global) positioning at the level of 1 cm in the IERS Terrestrial Reference Frame (ITRF);
- tropospheric delay estimated to a few mm;

- GPS orbits determined to 10 cm;
- Earth pole position determined to 1 cm;
- clock synchronisation (relative bias estimation) to 100 ps.

Two features of commercial software are sometimes conspicuously absent from more advanced packages: (i) sometimes double differencing is not implemented, but instead, undifferenced data are processed, and clock biases are estimated; (ii) network adjustment using baseline solutions is unnecessary, since advanced packages do a rigorous, one-step, simultaneous adjustment of station coordinates directly from all available GPS observations.

Some precise software packages incorporate a Kalman filter (or an equivalent formulism). This allows for certain selected parameters to vary in time, according to a statistical ("stochastic") model. Typically this is used for the tropospheric bias, which can vary as a random walk in time. A filter can also be used to estimate clock biases, where "white noise" estimation of clock bias approaches the theoretical equivalent of double differencing.

Although many more packages have been developed, there are 3 ultra high-precision software packages which are widely used around the world by researchers and are commonly referenced in the scientific literature:

- BERNESE software, developed by the Astronomical Institute, University of Berne, Switzerland;
- GAMIT software, developed by the Massachussets Institute of Technology, USA;
- GIPSY software, developed by the Jet Propulsion Laboratory, California Institute of Technology, USA

There are several other packages, but they tend to be limited to the institutions that wrote them. It should be noted that, unlike commercial software packages, use of the above software can require a considerable investment in time to understand the software and how best to use it under various circumstances. Expert training is often recommended by the distributors.

7.2 SOURCES OF DATA AND INFORMATION

For high precision work, it is important to abide by international reference system standards and use the best available sources of data and ancilliary information. We therefore summarise two especially important international sources of data information for the covenience of the interested reader:

- IERS: International Earth Rotation Service
  - Central Bureau located at the Paris Observatory, France
  - Documented IERS Conventions for observation models and reference systems
  - IERS Annual Reports
  - IERS Terrestrial Reference Frame for reference station coordinates
  - Routine publication of Earth rotation parameters

- IGS: International GPS Service for Geodynamics
  - Central Bureau located at the Jet Propulsion Laboratory, USA

- Documented IGS Standards for permanent GPS stations
- Oversees operation of global GPS network (~100 stations)
- Distributes tracking data and precise ephemerides
- Maintains on-line database with Internet access

8. CONCLUSIONS

Having read and understood this text, you should now understand the basics of GPS positioning observation models and parameter estimation. You should also have an appreciation of the difference between basic positioning, and the more advanced positioning using high precision software packages. If all has gone well, and you think the above statements are true, then you should now have a good background knowledge and an appropriate context to prepare you for more advanced material.