Combined Doppler Time-free Positioning for Low Dynamics GNSS Receivers

Nicholas Bernard Otieno Othieno

A Thesis In the Department of Electrical and

Computer Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Applied Science

(Electrical and Computer Engineering) at

Concordia University Montreal, Quebec, Canada

April 2012

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Nicholas Bernard Otieno Othieno

Entitled: "Combined Doppler Time-free Positioning for Low Dynamics GNSS Receivers" and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

Complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair
Dr. R. Raut

_____ Examiner, External To the Program
Dr. R. Landry, ETS

_____ Examiner
Dr. N. Kharma

_____ Supervisor
Dr. S. Gleason

Approved by: _____
Dr. W. E. Lynch, Chair
Department of Electrical and Computer Engineering

_____20_____

_____
Dr. Robin A. L. Drew
Dean, Faculty of Engineering and Computer Science

Sign.png

Abstract

The research presented here has the potential to dramatically change how GNSS receivers capture and process signals, opening up new applications which are not suitable for existing navigation receivers. As opposed to existing GNSS receivers which continually process the incoming signals, this technique allows for strict management of the incoming data and position estimation outputs. This management is well suited for applications which are required to remain off or in a low power state for long and intermittent periods.

It has been demonstrated that a position can be estimated in a GPS receiver using only submillisecond code phase measurements, as opposed to complete pseudoranges. This technique is referred to as Time-free and requires a coarse estimate of time, an a priori estimate of the receiver position and GPS satellite ephemeris information. However, this technique has previously required the a priori position knowledge to be less than approximately 100 km from the user. This is often an application limiting restraint which we overcome with the new combination technique proposed here.

A combined Doppler time free navigation method which requires no knowledge of the a priori receiver position is developed. A position is first estimated with no a priori knowledge of the receiver position using only Doppler measurements. The accuracy of this estimate is not useful for navigation, but it is within the initialization requirements of the Time-free positioning method. This combined technique therefore eliminates a cumbersome requirement and improves the effectiveness of receivers being designed using short bursts of data and software radio processing techniques.

Acknowledgements

I would like to thank my supervisor Dr Scott Gleason for direction and encouragement during the masters project. I am grateful to my parents, siblings and fiancee for their financial and moral support in my arduous journey to being a research engineer. I am eternally grateful to God for life and ability to find joy in my work.

Contents

1 Introduction **1**

1.1 History of Satellite Navigation 1

1.2 GNSS Applications 4

1.3 Limitations of Traditional GPS Processing Methods 5

1.4 Contributions of this Research 7

2 Background **9**

2.1 GPS Signal Overview 9

2.2 GNSS Receivers 11

2.2.1 GNSS Receiver Structure and Operation 11

2.2.2 GNSS Receiver and Simulator Used in this Research 13

2.3 GPS Positioning 14

2.3.1 Least Squares Technique 14

2.3.2 Dilution of Precision 20

2.4 Assisted GNSS 20

2.5 Doppler Positioning 23

2.6 Time Free Positioning 24

3 Doppler Positioning **27**

3.1 Heritage Algorithm 27

3.2 Algorithm Modification 32

3.3 Test Methodology 33
3.3.1 Simulated Data 33
3.3.2 Over-the-Air GPS Data 35
3.4 Doppler Positioning Results 35
3.4.1 Simulated data 35
3.4.2 Over the Air GPS Data 40
3.5 Summary 43

4 Time-free Positioning **44**
4.1 Introduction 44
4.2 Heritage Algorithm 45
4.3 Determining the Time-free (Observed) Pseudorange 54
4.4 Correcting Integer Roll-overs 58
4.5 Test Methodology 62
4.6 Time-free Positioning Results 64
4.6.1 Stationary Receiver with Receiver Time Accurate to 2 seconds . 64
4.6.2 Monte Carlo Simulations 65
4.7 Summary 68

5 Time-free Positioning Initialized by Doppler Positioning **69**
5.1 Algorithm Development 70
5.2 Test Methodology 73
5.3 Combined Doppler Time-free Positioning Results 73
5.3.1 East North Plots of Stationary Receivers in Various Locations . 73
5.3.2 Comparison of the Least Squares and the Combined Doppler Time-free Positioning Algorithms 76

6 Conclusion **81**
6.1 Conclusion 81

6.2 Future Work 83

List of Figures

1.1 Trilateration positioning method 2

1.2 Stages in the processing of a GPS signal 6

2.1 Modulation of the GPS L1 C/A signal 10

2.2 PRN code repeats in GPS satellite transmission 10

2.3 A general GNSS receiver block diagram (Adapted from [1] with permission) 11

2.4 Determination of a GPS receiver's position using 4 satellites 17

2.5 An example assisted GNSS configuration 22

2.6 Code Phase Alignment 25

3.1 Isolines of constant Doppler frequency 30

3.2 East North Position for a Stationary Receiver Over a 24 Hour Period Starting 9:16 am April 19, 2011 36

3.3 GDOP and number of satellites visible for stationary receiver test case 1 37

3.4 Doppler Positioning Solution Error for a Slow Moving Receiver Moving Southwards as from 9:16 am on April 19, 2011 38

3.5 Doppler Positioning Solution Error for a Receiver with Increasing Velocity Moving Southwards at 9:16 am on April 19, 2011 39

3.6 East North Position for a Stationary Receiver at 6:04 pm October 28, 2010 41

3.7 East North Position for a Stationary Receiver at 6:04 pm October 28, 2010 with a Maximum Time Error of 5 Minutes 42

4.1 Incorrect determination of satellite positions due to the error in the assumed time of arrival 50
4.2 Determination of time of transmission in normal GPS 54
4.3 Determination of the Time-free pseudorange and its relationship to satellite transmission time 56
4.4 Millisecond integer roll-over 57
4.5 Parc Mont Royal Google Earth Position 64
4.6 East North Position for a Stationary Receiver at 6:04 pm October 28, 2010 65
4.7 Time-free Positioning Convergence as a Function of a Priori Receiver Position Error and Receiver Time Error 66
4.8 Time-free Positioning Regions of Convergence 67

5.1 Processing stages in Time-free Positioning initialized by Doppler Positioning 71
5.2 East North Location Plot of a Stationary Receiver in Guildford UK on May 15, 2008 74
5.3 East North Location Plot of a Stationary Receiver in Stanford USA on May 16, 2008 74
5.4 East North Location Plot of a Stationary Receiver in Savannah USA on September 23, 2009 75
5.5 East North Location Plot of a Stationary Receiver in Montreal Canada on October 20, 2010 75
5.6 Error Magnitude Plots of a Stationary Receiver in Guildford UK 77
5.7 Error Magnitude Plots of a Stationary Receiver in Stanford USA 77
5.8 Error Magnitude Plots of a Stationary Receiver in Savannah USA . . . 78

5.9 Error Magnitude Plots of a Stationary Receiver in Montreal Canada . . 78

6.1 Millisecond integer ambiguity comparison of various GNSS PRN sequences 85

Chapter 1

Introduction

1.1 History of Satellite Navigation

Navigation has been a human concern for a number of centuries. Navigation can be loosely defined as the guidance of a vehicle or person from one location to another [2] . Initially navigation was primarily for sea-faring vessels. However, the invention of aircraft and motor vehicles in the 20th century has made navigation increasingly important in air and land travel. Whereas in the past navigation was mostly based on where one had come from and where they were headed, navigation in the modern sense is based on where exactly on the Earth one is, in relation to a certain reference location.

The commencement of the space age in 1957 with the launching of the Soviet *Sputnik I* satellite brought new opportunities in the development of navigation systems [3]. It was discovered that the orbit of *Sputnik I* could be determined from a single ground station at a known location using the pattern of Doppler shifts received from the satellite. This led to the development of *Transit* by the United States which was a Doppler based satellite navigation system. Between four and seven satellites in low altitude orbit and transmitting at 150 Mz and 400 MHz where used in this system. Only one satellite was in view at a single time, and there were 100 minutes between successive passes of the satellites. Receivers recorded the satellite's Doppler shift as well as a navigation

message giving the position of the satellite for a period of between 10 and 20 minutes of the satellite pass. The Doppler shift and navigation message where then used to compute the 2D position of a stationary or slow moving user. Transit was initially used by the US Navy and was then later used in a few civil applications, namely maritime and geodesy [3]. The Soviet Union also developed two Doppler-based satellite navigation systems. *Parus* for the Soviet Navy and *Tsikada* for merchant ships [3].

The next space-based navigation system in the US was the Global Positioning System (GPS), which was borne out of the success of *Transit*. GPS benefited from advancements in technology and as a result is an entirely different system. Whereas the positioning method used by *Transit* was Doppler, the positioning method used by GPS is range-based. Range-based positioning is a kind of trilateration.

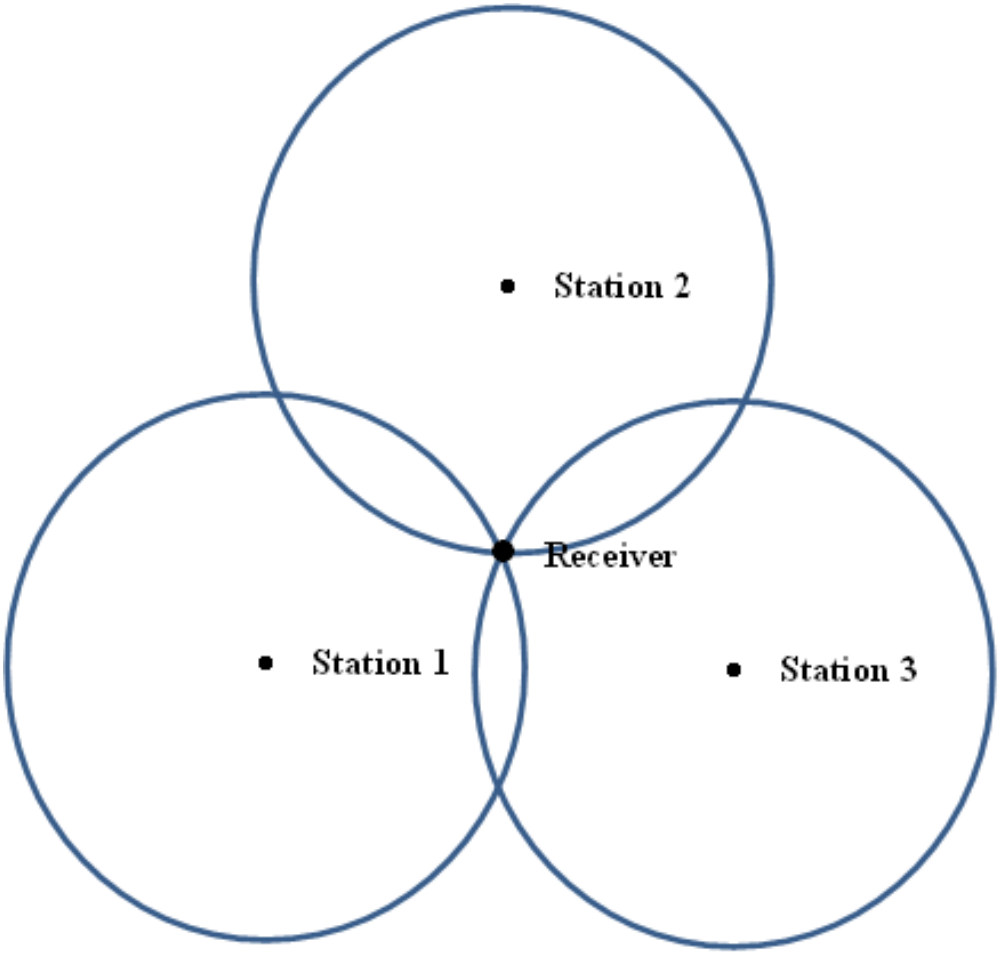

Figure 1.1: Trilateration positioning method

In trilateration, a receiver determines its position by measuring its distance from a number of signal transmitting stations. This measured distance is a radius around the transmitting station, and the intersection of the radii of multiple stations gives the receiver position. Figure 1.1 shows the 2-dimensional view of trilateration as viewed from above. The intersection of the radii of the three transmitting stations is the estimate of the receiver's position. In the case of GPS, the transmitting stations are satellites, and instead of 2-dimensional circles, we have spheres. The three spheres intersect at two points, one point above the GPS constellation and one point on or near the Earth's surface. Although two points of intersection causes some ambiguity as to where the receiver is positioned, the ambiguity is easily resolved because the receiver is assumed to be close to the Earth's surface.

The time of transmission of the signal and the time of reception of the signal are used to determine travel time of the signal. Since the speed of propagation of the signal is known, it is used in conjunction with the travel time to determine the distance from the receiver to the transmitting station. Advancements in clock technology allowed the transmission of time-synchronized signals from GPS satellites. Spread spectrum technology ensures that the signals from the satellites can be transmitted simultaneously at the same radio frequency. This is done by modulating the data from each satellite with a unique sequence of pulses called a pseudorandom number (PRN) code. The advent of integrated circuits in the 1970s enabled the design of smaller portable receivers further enhancing the usefulness of GPS as a navigation system. Medium Earth Orbit (MEO), which is between 5000km and 20000km, was chosen for the satellites as it provides an optimum balance of launch costs, number of satellites in the constellation required for proper coverage and effects of the Earth's atmosphere and gravitational field [3].

Currently (March 2012), GPS consists of a constellation of 32 satellites that orbit the Earth every 12 hours [4]. The availability of the full constellation varies from time to time due to technical issues [5]. Up to date information of the current state of the constellation can be found at the Navigation Center website that is run by the United

States Coast Guard [6]. GPS has a vast array of applications amongst both civilian and military users. Applications of Global Satellite Navigation Systems (GNSS) are discussed in 1.2.

There are similar satellite navigation systems operational or under development by other countries. GLONASS is a global satellite navigation system developed by Russia. According to Russia's Federal Space Agency Information-Analytical Center, GLONASS currently has a constellation of 31 satellites of which 24 are operational [7]. Unlike GPS which uses code division multiple access (CDMA), GLONASS uses frequency division multiple access in which all satellites transmit the same PRN code but at different frequencies [3].

The European Union is currently developing Galileo which is expected to have 27 operational satellites and 3 spares once it is fully deployed [8]. It also uses spread spectrum communications like GPS but with additional binary offset carrier (BOC) modulation of the carrier.

Other countries also have regional satellite navigation systems such as *BeiDou* developed by China, *Quasi-Zenith Satellite System* under development by Japan and the planned IRNSS by India [1].

1.2 GNSS Applications

GNSS has varied applications. Gleason and Gebre-Egziabher propose a set of categories for the applications of GNSS in [1] . GNSS is used for navigation in aviation, automobiles, space vehicles, marine transport as well as by people on foot. Other applications include remote sensing, geodesy and surveying

Pedestrians and hikers form a huge category of GNSS receiver users, with GNSS receivers being integrated into mobile phone handsets. GNSS receivers are useful to hikers who can use them to find their way in unknown terrain. City pedestrians also use GNSS receivers with regional datums to find their way through city streets. This

latter application of GNSS receivers is also used by drivers to find their way to specific addresses in cities. The driver simply keys in an address into the receiver and the receiver uses the regional datum in conjunction with local maps and other data sources to determine the location of the address as well as provide directions of how to get there. The application is however not limited to cities but can also be used for long journeys in which the receivers give directions of highways to follow.

In marine transport, GNSS receivers are used for navigation of the vessels either by using the automatic pilot or manually. GNSS is particularly useful as a method for primary navigation in the open seas within the areas of coverage. Not only are the vessels able to find their way from one port to another, but they are also able to determine where they are at sea.

In civil aviation, GNSS receivers are used as a secondary navigation aid in both flight and landing. Though GNSS can achieve reasonable accuracy for flight, aviation has got strict requirements on reliability making it impossible to use GNSS as a primary navigation aid except in oceanic flight and remote areas where there is no ground navigation network, as discussed in chapter 10 of [1]. GNSS is therefore used in conjunction with augmentation systems to detect anomalies in the navigation solution provided.

1.3 Limitations of Traditional GPS Processing Methods

The general idea underlying GPS is finding a user's position by the solution of a system of equations with four unknowns. These unknowns are position in X, position in Y, position in Z and a receiver clock offset. Four unknowns can only be solved by at least four equations and so, at least 4 satellites have to be visible for a user to determine their position by solving the system of equations. The formation of these equations however also includes other data elements. These elements are satellite ephemeris data which is information of the satellite position and satellite clock, transmission time of the GPS signal, reception time and error elements related to the transmission and reception of

the signal. Satellite ephemeris and transmission time are carried on the GPS signal. The traditional method of obtaining this data is to first acquire the satellite signal, track the signal and finally decode the transmit time in the satellite navigation data as shown in figure 1.2 [3].

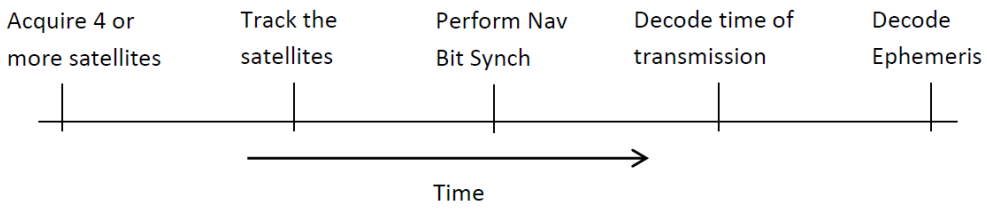

Figure 1.2: Stages in the processing of a GPS signal

The traditional method of acquisition, tracking and decoding information results in a large amount of data that has to be handled by the receiver in order to get to the point of being able to solve for the system of equations and hence perform a position fix.

As an example, consider the *fastGPS* software-defined receiver coupled with the SiGe GN3S sampler as it's RF front-end and downconverter [1]. The GN3S readme states that for 40 seconds of data, the SiGe GN3S sampler creates a file of about 625 MB [9]. It is clear that a large amount of data has to be processed for a relatively short period of captured signal. Section 1.4 proposes a technique of allowing position fixes to be obtained with shorter data sets.

Another drawback of the acquire, track and decode method in traditional GPS is the considerable amount of time it takes to obtain all the ephemeris data required to perform a position solution for the user, once acquisition and tracking have occurred. This is due to the structure of the navigation data in the GPS signal and the low transmission rate of the navigation data itself. In a worst-case scenario, the user might have to wait for as long as 36 seconds from the beginning of acquisition before all the ephemeris data required to determine an initial position fix is available. This initial position solution determination is called time to first fix (TTFF) [10].

At the moment, the two factors that lead to this wait time, low data transmission rate

and signal structure, currently cannot be eliminated for GPS L1. The structure of the GPS signal is such that it carries the satellite ephemeris. Currently, a position cannot be found without satellite ephemeris.

The solution to the requirement of obtaining ephemeris data is to acquire the ephemeris externally from a different source and then store it at the receiver. This method is called assisted-GPS (A-GPS) and is relatively easy to implement in a receiver [10]. A-GPS is discussed in section 2.4.

With the ephemeris stored at the receiver, there still exists the problem of having to decode the time from the GPS signal. This is solved by a positioning technique proposed by Peterson et al that is able to provide a position fix without having to decode time from the GPS signal [11]. This technique however has the disadvantage of requiring knowledge of the receiver position to within a certain unknown accuracy of the truth. This challenge will be addressed in section 1.4.

1.4 Contributions of this Research

In traditional GPS receivers, assistance data can be stored at the receiver hence allowing the receiver to perform position estimation without having to decode ephemeris from the satellite and by extension shortening the data set required to perform the position fix. The length of this data set can also be further shortened when Time-free Positioning is used, since the receiver does not require to decode the time from the GPS signal.

This research aims at using Time-free Positioning while mitigating its drawbacks by initializing it using the Doppler Positioning Algorithm proposed by Hill [12]. The main drawback of the Time-free Positioning Algorithm is its requirement of an a priori receiver position knowledge to a restrictive degree of accuracy. Lannelongue and Pablos state that this accuracy should be to within 100 km of the truth [13]. This research therefore seeks to discover what this accuracy is, by performing experiments. The a priori receiver position requirement is then solved by first obtaining the receiver position

estimate using a Doppler Positioning technique and feeding this position obtained into the Time-free Positioning module. At low receiver dynamics, the Doppler Positioning Algorithm is able to provide a position estimate that should be accurate to within 100 km as required by the Time-free Positioning Algorithm. Therefore this research proposes a method to remove the a priori position requirement from the Time-free Positioning Algorithm.

Another contribution is the investigation and documentation of the conditions for and limitations of both the Doppler Positioning and Time-free Positioning Algorithms. In this research, the performance of the Doppler Positioning algorithm will be investigated with respect to satellite geometry for the whole GPS constellation by simulating over a complete day. The effect of receiver dynamics on Doppler Positioning will also be investigated and a receiver velocity limit determined. The a priori time error degradation of Doppler Positioning accuracy will also be investigated.

The a priori coarse time error and a priori position error effect on Time-free Positioning will be investigated. Currently only one published work does analysis of time and position error effect on Time-free Positioning [13]. This research will investigate convergence of the algorithm based on the two aforementioned parameters.

The application of this research will target GNSS users whose dynamics are low. This category includes animal tracking, pedestrians, hikers, bikers and low velocity surface marine vessels.

It should be noted that the techniques developed in this research can find use in high receiver dynamics scenarios when the longer new PRN codes are considered and when the Doppler Positioning algorithm is implemented with inertial aiding. These cases are however not investigated in this research.

Chapter 2

Background

2.1 GPS Signal Overview

The GPS signal is described in the ISP-GPS-200D specification [14] . The components of the signal described are carrier, ranging code and navigation data. The GPS carrier signals are transmitted on L-band frequencies denoted L1 at 1.57542 GHz, L2 at 1.2276 GHz and L5 at 1.17645 GHz. The L1 carrier transmits two signals, one for civil users and another for military users. The L2 carrier contains a military signal only. The L5 signal is a safety of life signal whose carrier signal is located in the protected Aeronautical Radionavigation Service (ARNS) band [15].

The L1 civil signal is modulated with a navigation message which has been modulated by a coarse acquisition code. Figure 2.1 shows how the civil GPS L1 signal is created.

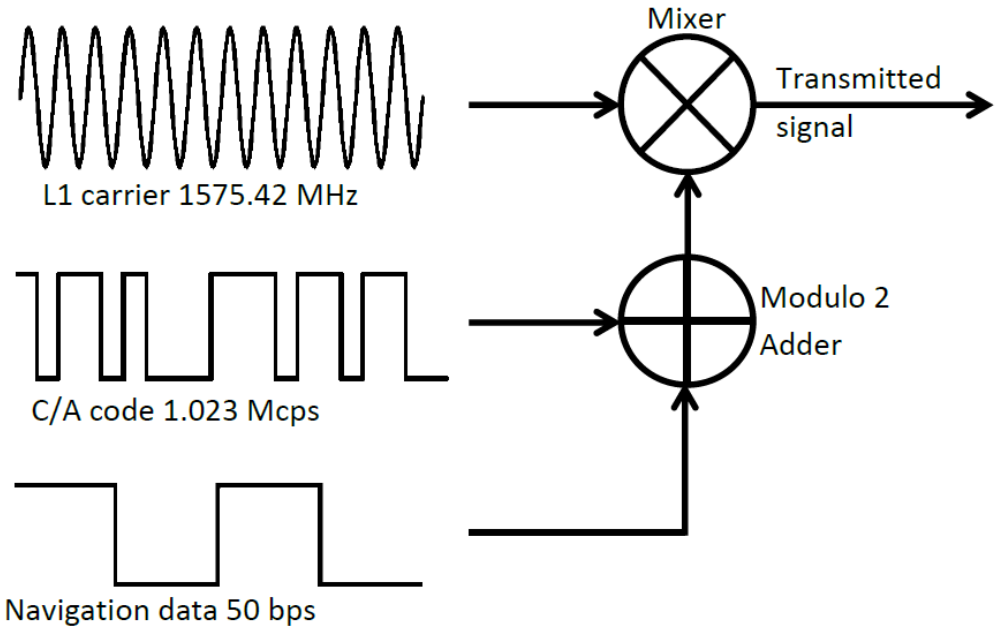

Figure 2.1: Modulation of the GPS L1 C/A signal

The code transmitted on the L1 frequency is a pseudorandom sequence called a coarse acquisition (PRN C/A) code, which is actually a Gold code. Each satellite has got its own unique PRN C/A code. The codes ensure that all satellites can transmit at the same frequency and still be identified by receivers. The PRN codes 1023 bits long and are repeated every millisecond hence giving them a frequency of 1023 MHz.

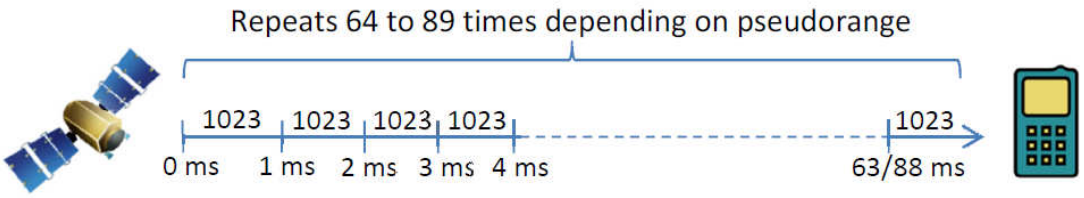

Figure 2.2: PRN code repeats in GPS satellite transmission

When a GPS signal is received by a receiver, it will usually not be at the L1 frequency of 1.57542GHz but will appear to be at a slightly higher or slightly lower frequency. This is due to the movement of the satellites causing Doppler shift of the signal. The received signal C/A code will also not be at an integer code phase boundary but

will have shifted by a fractional value to a non-integer boundary. This is due to the distance and clock dependent time delay between the time of transmission of the signal and its reception.

2.2 GNSS Receivers

2.2.1 GNSS Receiver Structure and Operation

GNSS receivers can either be hardware receivers or software-defined receivers. Hardware receivers are normally based on application specific integrated circuits (ASICs) but they can be based on field-programmable gate arrays (FPGAs). ASICs have a fixed hardware configuration whereas FPGAs have some flexibility in the hardware configuration. Figure 2.3 shows the block diagram of a generic GNSS receiver.

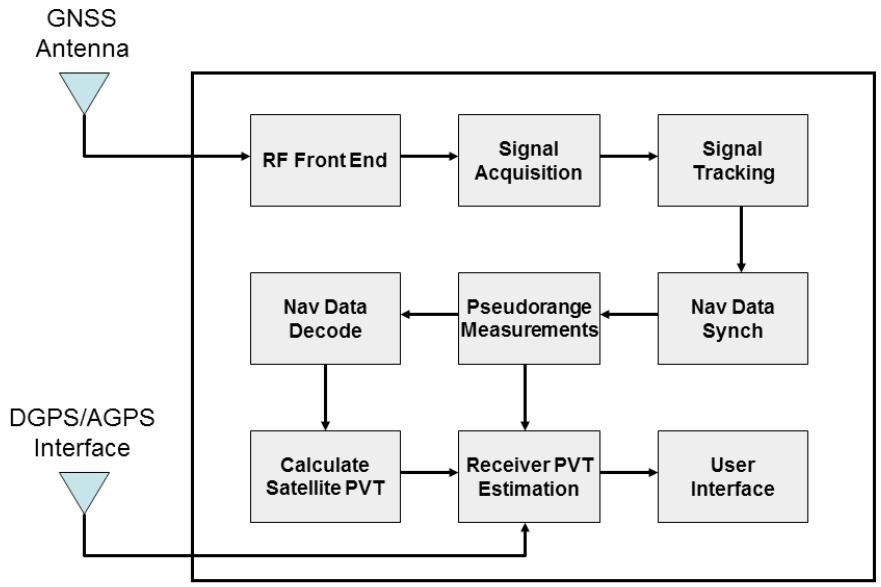

Figure 2.3: A general GNSS receiver block diagram (Adapted from [1] with permission)

The first stage is the capturing of the signals by a suitable antenna. This antenna is normally a right hand circularly polarized (RHCP) antenna in the case of GPS since GPS signals are RHCP [14]. Once the signals are captured, they are amplified and downconverted to a suitable intermediate frequency (IF) by the RF frontend.

Once the signal is sampled at IF, each individual satellite's transmission has to be discovered in a process called acquisition. Acquisition involves doing a correlation of the signal with each satellite's PRN C/A code. As mentioned in section 2.1 , each satellite PRN code is a unique sequence. Therefore, during acquisition these PRN codes allow for currently visible satellites to be "detected". The aim of acquisition is to find the frequency and phase at which each satellite signal is located. In GPS, a search range of 2 kHz is searched in steps (of 500 Hz for example) to find the Doppler frequency of the satellite. Each frequency step is then searched for all code phase delays.

The output of acquisition is then sent to a tracking unit which as its name suggests tracks the satellite signal in frequency, carrier phase and code phase delay. Acquisition only gives a rough estimate of the signal Doppler and code phase, but tracking provides further accuracy. This involves determining the Doppler frequency, carrier phase and code delay of a signal and adjusting the receiver accordingly so that the receiver is locked onto the received signal. The receiver generated signal and the received signal have to be locked in frequency, phase and code delay.

Once this lock has been achieved, the "in-phase prompt correlator accumulations of the tracking module are used to decode the signal navigation data" [1]. In order to achieve this, the position of a navigation data bit is first determined by looking attempting to determine a phase change in the data samples from the tracking module. Once a navigation data bit has been found, the receiver tries to determine the start of a subframe.

The next step after determining the start of a subframe is determining the time of transmission of the subframe. This also is decoded from the subframe, as each satellite embeds a time of transmission of the next subframe before transmission of a subframe.

With the time of transmission, the satellite can now start measuring pseudoranges by subtracting the time of transmission from the receiver time and multiplying by the speed of light. A pseudorange is the estimated distance between a satellite and a receiver. This measurement will have errors in it which include: satellite clock error, receiver clock error, ionospheric error, tropospheric error and other unmodelled errors.

The receiver then decodes the satellite ephemeris from the navigation data. In the GPS case, satellite ephemeris are Keplerian elements which are used to determine the position, velocity and time (PVT) of a satellite. The ephemeris are used in conjunction with the measured pseudoranges to determine the receivers position which is then output to a user interface. This is explained in more detail in section 2.3.

2.2.2 GNSS Receiver and Simulator Used in this Research

In this thesis, the series of blocks from signal acquisition to receiver PVT estimation of the general GNSS receiver of figure 2.3 form the open source fastGPS software-defined GPS receiver. This receiver was developed by Morgan Quiqley, Pieter Abbeel and Scott Gleason. It is an open-source software-defined GPS receiver that provides a suitable platform on which to test and implement the ideas in this thesis.

It should be noted that there are other software-defined GNSS receivers. Akos et al designed a MATLAB based software-defined GPS receiver that performs positioning using the GPS L1 captured by a simple front-end [16].Heckler and Garrison designed the C++ based Purdue software receiver (PSR) for research and teaching purposes [17]. Ledvina, Psiaki et al also designed a real-time L1 software receiver that is interoperable GPS and Galileo [18]. *fastGPS* is chosen for use in this research because of its ease of reconfiguration by changing the source code which allows easy and quick testing of algorithms. It also has faster execution speed compared to MATLAB-based software defined GNSS receivers. *fastGPS* also allows access to raw measurements such as Doppler and code phase enabling easy investigation of various aspects of GPS receiver

operation.

Two choices for the front-end are available: the SiGe GN3S Sampler and the USRP2. The SiGe GN3S Sampler was developed by the GNSS Lab at the University of Colorado and SiGe [19]. It is an ASIC-based RF front end which downconverts GPS signals to an intermediate frequency (IF) of 4.1304MHz at a sampling rate of 16.3676 Megasamples/second. The USRP2 developed by Ettus Research LLC is an FPGA-based RF front-end with adjustable downconversion and and sampling rates [20].

A GPS simulator is also used to test algorithms developed in this thesis. The GNSS Simulator developed by Scott Gleason and found in the DVD that accompanies reference [1] is able to simulate GPS satelllites with either a terrestrial or space GPS receiver. The simulator is useful for investigation of the effect of receiver dynamics on the algorithms developed, since varied scenarios of receiver dynamics can be tested.

One other notable GNSS simulator is the Software Defined Navigator (SDN) developed at Ecole de Technologie Superieure (ETS) in Montreal Canada [21]. It was created for rapid prototyping of hybrid GPS/Galileo receivers.

2.3 GPS Positioning

In this research, vector quantities are represented in **bold** font while scalar quantities are represented in normal font. In addition, estimated values are represented as a letter overlaid with a caret (ˆ).

2.3.1 Least Squares Technique

Having done an overview of general GNSS receivers, the specific case of GPS position estimation will now be described. This section thus describes the ”Least Squares technique” of position, velocity and time (PVT) estimation.

As was mentioned in section 1.1 distances from transmitting satellites are used to find the location of a receiver. These distances are known as pseudoranges in GNSS.

According to Misra and Enge, the pseudorange measurement of a satellite can be modeled as [3]:

$$\rho^i = r^i(t, t-\tau) + c[\delta t_u - \delta t^i(t-\tau)] + I^i(t) + T^i(t) + \varepsilon^i(t) \tag{2.1}$$

Where:

ρ^i	=	Pseudorange measurement for satellite i in meters
t	=	Time of signal reception in seconds
$r^i(t, t-\tau)$	=	True pseudorange in meters between the receiver at time t and the satellite at time of transmission $t-\tau$
τ	=	Delay in seconds between the time of transmission of the signal from a satellite and the time of reception at the receiver
c	=	Speed of light in meters/second
δt_u	=	Receiver clock bias from GPS time in meters
$\delta t^i(t-\tau)$	=	Satellite clock offset in meters
$I^i(t)$	=	Propagation delay due to the Earth's ionosphere in meters
$T^i(t)$	=	Propagation delay due to the Earth's troposphere in meters
$\varepsilon^i(t)$	=	Other errors in meters

As suggested by Gleason and Gebre-Egziabher, equation 2.1 can be simplified by applying the known satellite clock error correction, grouping all the error terms into one term and omitting the references to time[1]:

$$\rho_c^i = r^i + c\delta t_u + \varepsilon_A^i \tag{2.2}$$

The true pseudorange is defined as the magnitude of the vector between the receiver and the satellite:

$$r^i = ||\mathbf{S_c^i} - \mathbf{U_c}|| \tag{2.3}$$

Where:

$\mathbf{U_c}$ = True receiver position in Cartesian coordinates

$\mathbf{S_c^i}$ = True satellite position in Cartesian coordinates

Equation 2.3 is then substituted into equation 2.2 and the receiver clock error term is renamed:

$$\rho_c^i = ||\mathbf{S_c^i} - \mathbf{U_c}|| + b + \varepsilon_A^i \tag{2.4}$$

Where:

b = Receiver clock error

Equation 2.4 shows how the corrected pseudorange is related to the receiver's position and clock error. The equation has 4 unknowns: the receiver position in Cartesian coordinates (x,y and z) and the receiver clock error b. Therefore, at least 4 equations are required to solve for these unknowns. This means at least 4 corrected pseudorange equations are required which translates to at least 4 satellites in view as shown in figure 2.4.

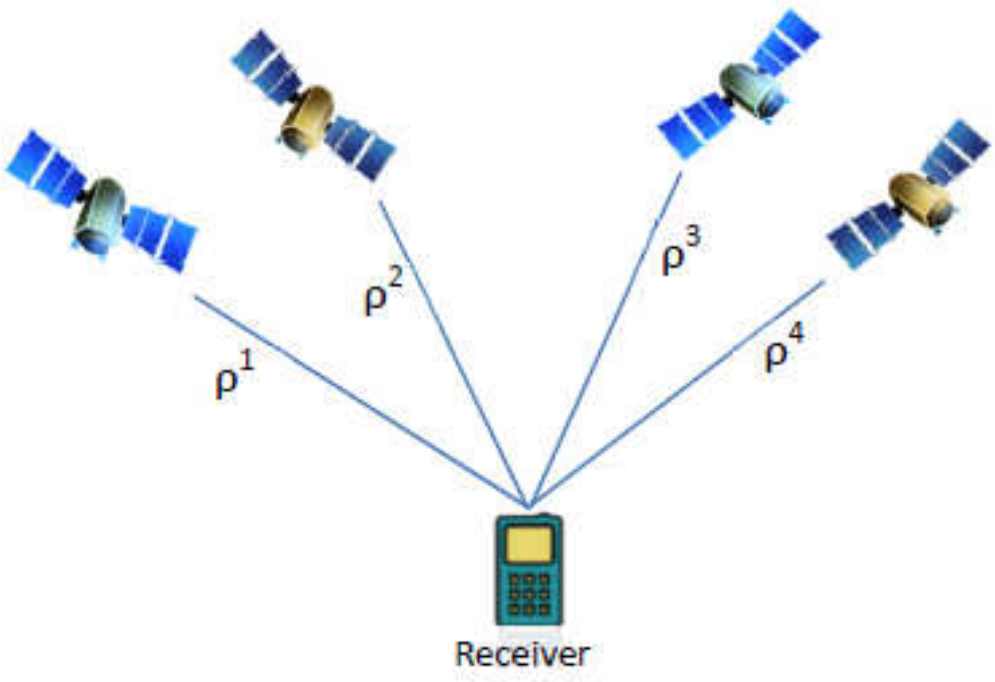

Figure 2.4: Determination of a GPS receiver's position using 4 satellites

The estimation of the receiver position and clock error is done by linearizing the pseudorange measurements around an a priori guess of the receiver position and receiver clock offset. The estimated pseudorange based on the receiver position guess and clock offset guess is given by:

$$\widehat{\rho}^{i} = ||\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c}|| + \widehat{b} \tag{2.5}$$

Where:

$\widehat{\mathbf{U}}_\mathbf{c}$ = Receiver position guess

$\widehat{b}$ = Receiver clock error guess

The estimated pseudorange in equation 2.5 is then subtracted from the corrected measured pseudorange:

$$\delta\rho^i = \rho_c^i - \widehat{\rho}^i \tag{2.6a}$$

$$\delta\rho^i = ||\mathbf{S_c^i} - \mathbf{U_c}|| + b + \varepsilon_A^i - (||\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c}|| + \widehat{b}) \tag{2.6b}$$

Equation 2.6 is further modified by considering the fact that the difference between $||\mathbf{S_c^i} - \mathbf{U_c}||$ and $||\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c}||$ is an unknown receiver position correction $\delta\mathbf{U_c}$ and the difference between b and $\widehat{b}$ is an unknown clock bias correction δb:

$$\delta\rho^i = ||\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c} - \delta\mathbf{U_c}|| + \delta b + \varepsilon_A^i - ||\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c}|| \tag{2.7}$$

Taylor series expansion is then applied to equation 2.7 and only up to the first order terms taken to give:

$$\delta\rho^i = \frac{\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c}}{||\mathbf{S_c^i} - \widehat{\mathbf{U}}_\mathbf{c}||} \bullet \delta\mathbf{U_c} + \delta b + \varepsilon_A^i \tag{2.8a}$$

$$\delta\rho^i = -\mathbf{L}^i \bullet \delta\mathbf{U_c} + \delta b + \varepsilon_A^i \tag{2.8b}$$

Where:

L = Line of sight unit vector between satellite i and the a priori receiver position

As was previously mentioned, at least 4 satellites are required to solve for the position of a receiver. Therefore, 4 or more of equation 2.8 are required. These equations are used to build a system of equations represented in matrix form as:

$$\delta\rho = \begin{vmatrix} \rho^0 - \widehat{\rho}^0 \\ \rho^1 - \widehat{\rho}^1 \\ \vdots \\ \rho^i - \widehat{\rho}^i \end{vmatrix} = \begin{vmatrix} -L_x^0 & -L_y^0 & -L_z^0 & 1 \\ -L_x^1 & -L_y^1 & -L_z^1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -L_x^i & -L_y^i & -L_z^i & 1 \end{vmatrix} \begin{vmatrix} \delta x \\ \delta y \\ \delta z \\ \delta b \end{vmatrix} + \begin{vmatrix} \varepsilon_A^0 \\ \varepsilon_A^1 \\ \vdots \\ \varepsilon_A^i \end{vmatrix} \tag{2.9}$$

Equation 2.9 is of the form $\mathbf{G}\widetilde{\mathbf{x}} = \delta\rho$ and can now be solved as follows:

$$\widetilde{\mathbf{x}} = (\mathbf{G}^{\mathbf{T}}\mathbf{G})^{-\mathbf{1}}\mathbf{G}^{\mathbf{T}}\delta\rho \tag{2.10}$$

Where:

$$\mathbf{G} = \begin{vmatrix} -L_x^0 & -L_y^0 & -L_z^0 & 1 \\ -L_x^1 & -L_y^1 & -L_z^1 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -L_x^i & -L_y^i & -L_z^i & 1 \end{vmatrix}$$

$$\widetilde{\mathbf{x}} = \begin{vmatrix} \delta x \\ \delta y \\ \delta z \\ \delta b \end{vmatrix}$$

It should be noted that the error terms in equation 2.9 are included for mathematical completeness and hence only exist as hidden terms in the measurements in the solution in equation 2.10. Equation 2.10 is in a form that can allow determination of solutions for an overdetermined case. An overdetermined case is when more than the required four satellites are available to perform a position estimate. This is normally advantageous as it allows for better satellite geometry. Equation 2.10 is then solved iteratively in order to produce an accurate solution for the receiver position and clock bias estimates.

2.3.2 Dilution of Precision

The accuracy of the position estimate of a GPS receiver is dependent on the satellite geometry of the visible satellites. The satellite geometry is the arrangement in the sky of the visible satellites. The geometry matrix **G** of the GPS solution formula is used to create the dilution of precision (DOP) matrix **H** whose major axis is used to determine the quality of the satellite geometry [1].

$$\mathbf{H} = (\mathbf{G^T G})^{-1} = \begin{vmatrix} H_{11} & - & - & - \\ - & H_{22} & - & - \\ - & - & H_{33} & - \\ - & - & - & H_{44} \end{vmatrix} \tag{2.11}$$

The measure of the satellite geometry, geometric dilution of precision (GDOP), is thus defined:

$$H_{GDOP} = \sqrt{H_{11} + H_{22} + H_{33} + H_{44}} \tag{2.12}$$

In this research, GDOP will be used in conjunction with mean position error to give a better understanding of the performance of the algorithms developed in terms of position accuracy. GDOP greatly affects position accuracy and hence has to be included as part of the analysis.

2.4 Assisted GNSS

Assisted GNSS, as its name suggests, is a method of aiding the operation of a GNSS receiver. Several kinds of assistance techniques exist for acquisition, tracking and navigation, but this research concentrates only on assistance techniques for navigation.

Navigation assistance is shown in figure 2.3. It has the effect of eliminating the need for the ”Nav Data Synch” block and the ”Nav Data Decode” block.

In normal GPS navigation, the receiver has to decode satellite ephemeris from the signal received from the satellite. This ephemeris is contained in the first 3 subframes of a GPS frame. Satellite ephemeris contains orbital parameters that can be used to calculate the satellite position and velocity at a given epoch. Satellite clock error terms are also included in the ephemeris and are used in calculating satellite clock error at the time epoch. Satellite ephemeris is needed in the navigation equations for receiver PVT estimation. However, decoding ephemeris from the satellite signal takes time because the receiver must not only determine the beginning of a subframe, but must also capture the signal at subframe 1. If the receiver captures the signal at subframe 3, it must wait another 2 subframes (4 and 5) for the next subframe 1 to be transmitted.

Therefore, to save time required to decode ephemeris from the satellite signal, the receiver can obtain the ephemeris from a different source and store it at the receiver. Figure 2.5 shows an example of one such assisted GNSS (AGNSS) configuration.

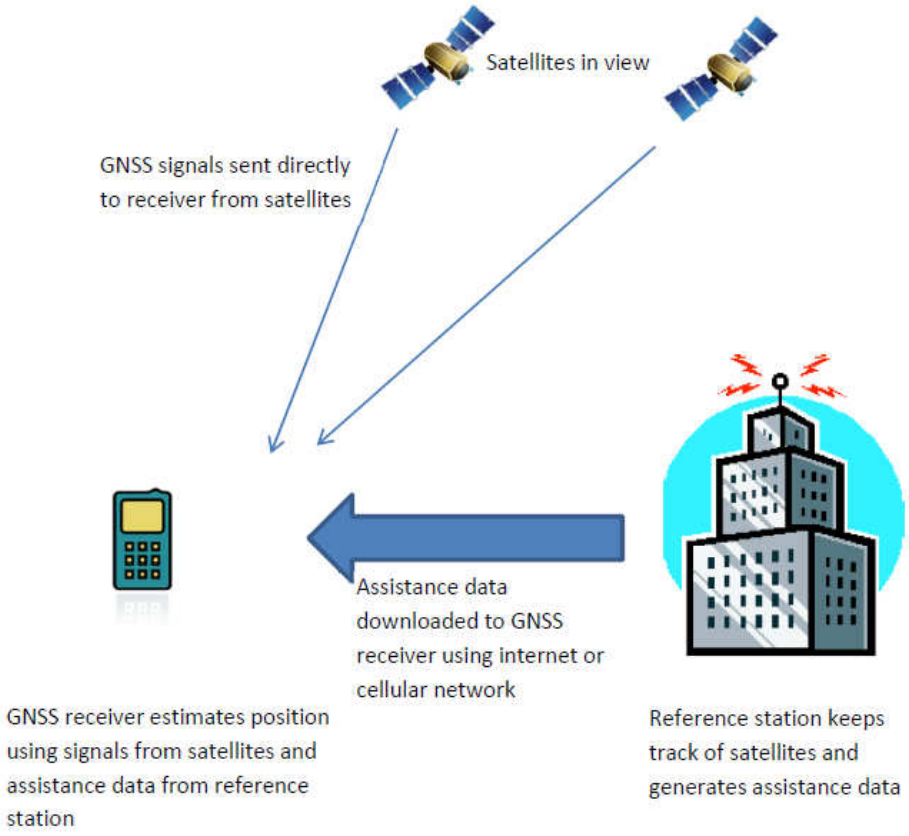

Figure 2.5: An example assisted GNSS configuration

In this configuration, the receiver receives assistance data, which includes the satellite ephemeris and satellite clock correction parameters, from a location other than the satellite itself. In the case of GPS receivers in mobile phones, it is common for the receiver to receive the assistance data via the mobile phone network. GPS receivers can also be connected to the internet to download satellite assistance data from such sites as the International GNSS Service (IGS) [22]. Ephemeris downloaded from such sources tends to be valid for a limited time.

Once the receiver has assistance data locally stored, it is freed from the burden of having to decode ephemeris and satellite clock correction parameters from the satellite. The receiver needs only to decode time of week (TOW) and measure the C/A code phase from the satellite signal in order to determine the time of transmission of a sample. This time of transmission is then used to obtain the satellite positions and velocities

from the locally stored ephemeris. It should be noted that the assistance data provides satellite orbital parameters at regular time intervals and hence most often the parameters have to be propagated for times of transmission which are between these intervals. In the IGS case, the assistance data is actually provided in actual satellite positions and velocities every 15 minutes [23]. These values have to be propagated for times that are in between these 15 minute intervals. The receiver therefore has to have some kind of orbit propagator. Assistance data from the IGS can be propagated using the algorithm by Schenewerk [24].

2.5 Doppler Positioning

As mentioned in section 1.4, Doppler positioning can be a used to estimate a receiver position without having a priori knowledge of the receiver position. It's main drawback is that it is inaccurate especially for receivers with high dynamics. However the method proposed by Jonathan Hill provides a position estimate of a receiver that is accurate enough to initiate Time-free positioning [12].

This technique is based on determining Doppler offsets of the various visible satellites and using these Doppler measurements with satellite velocity, satellite position and an estimated initial receiver position to build a system of equations. The system of equations has four unknowns, which are receiver X position, receiver Y position, receiver Z position and receiver clock rate error. This system of equations is solved recursively as shown in chapter 3 to find the values of the unknowns.

It should be noted that Doppler positioning can operate without decoding ephemeris from the satellite. However, the requirement of the satellite position and satellite velocity suggests that ephemeris is still needed. There is where the assisted GPS comes in. Ephemeris data obtained from a different location must be available at the receiver. It is from this assistance data that the GNSS receiver will obtain the satellite locations and velocities.

In order to obtain the correct satellite locations and velocities, the GNSS receiver should have a reasonable time accuracy. Time accuracy is important because if the receiver has an incorrect clock, it will compute erroneous satellite locations and velocities from the assistance data. This will in turn lead to incorrect calculated pseudoranges as the satellites are not where they are supposed to be in the sky.

The time accuracy requirement must not be too stringent. If it is too stringent, it renders the method difficult to perform. This is because one of the strategies of making receivers cheap and hence available to many users is by using low cost clocks. Therefore a time accuracy limit that can be considered ”reasonable” for use with Doppler Positioning must be determined. This accuracy limit must also be well within the reach of modern low cost clocks that can be found in general purpose receivers.

As was mentioned before, Doppler Positioning is known to be inaccurate, especially when the receiver is moving [12]. It is therefore necessary to investigate the effects of receiver velocity. This will enable a threshold velocity to be determined such that when the receiver is moving at velocities above the threshold, the position estimation results are considered untrustworthy.

The derivation and further discussion of Doppler Positioning is found in chapter 3.

2.6 Time Free Positioning

Time-free positioning, as the name suggests, is a method of estimating a GPS receiver's position without having access to the true GPS time of transmission from the satellite. This implies that just like Doppler Positioning, Time-free Positioning does not require to decode transmitted satellite ephemeris. It too uses the assisted GPS technique of having satellite ephemeris stored at the receiver and requires a reasonable time accuracy to make use of this assistance data.

Time-free Positioning uses code phase measurements to reconstruct observed pseudoranges. Code phase is the ”distance” by which a replica PRN code at the receiver is

shifted in order to align it with the PRN code received from the satellite. This is shown in figure 2.6 below.

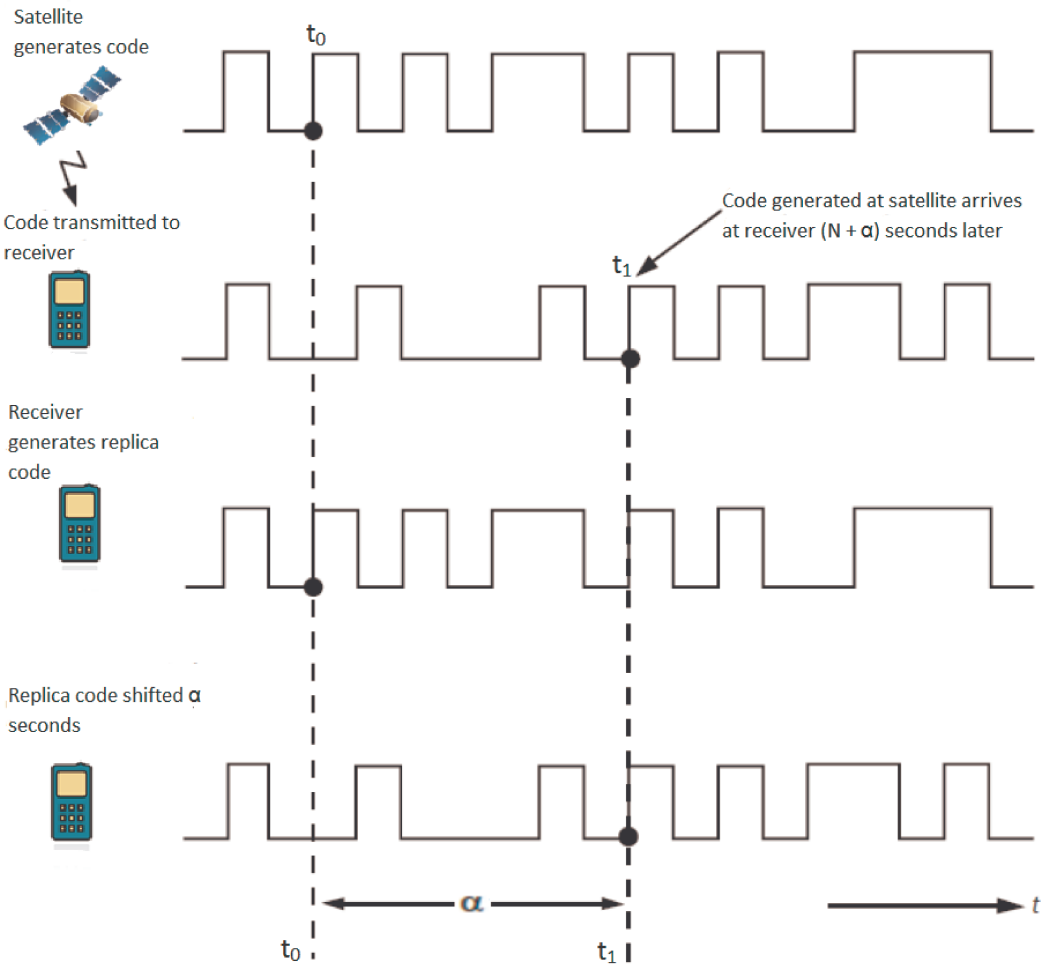

Figure 2.6: Alignment of replica PRN code at the receiver with received PRN code from the satellite (Based on figure in [25])

A signal from a GPS satellite takes between 64 ms and 89 ms to travel to a receiver on the Earth's surface, depending on whether the satellite is directly overheard or at the horizon [10]. As was shown in figure 2.2, the PRN code consists of 1023 bits which repeat every 1 ms. When a code phase alignment is done as in figure 2.6, the alignment is done by comparing the 1023 bits in a single PRN epoch. The alignment α is the code phase measurement obtained by the receiver. The code phase measurement is therefore

a submillisecond value which only provides a fractional part of the full observed pseudorange. The rest of the observed pseudorange is an unknown integer value between 64 and 89 ms. The determination of the full observed pseudorange in Time-free Positioning involves guessing the value of this integer millisecond. The integer millisecond count is an ambiguity in the exact observed pseudorange determination. A technique therefore has to be found to predict the millisecond integer count accurately, and if this prediction is not sufficiently accurate, measures have to be put in place to identify and correct it.

Just like in Doppler Positioning, the time estimate in Time-free Positioning is used to determine satellite locations from the assistance data. These satellite locations are used in conjuction with an estimated receiver position to estimate pseudoranges. According to Lannelongue and Pablos, this estimated receiver location must be within 100 km of the truth while the initialization coarse time should be within 60 seconds of the truth for the algorithm to work [13]. The validity of these two limits will be investigated in chapter 4.

Once observed and estimated pseudoranges have been determined, the receiver position can be estimated iteratively using a least squares method as shown in section 2.3. The derivation and further discussion of Time-free Positioning is found in chapter 4.

Chapter 3

Doppler Positioning

In this work, vector quantities will be represented in **bold** font while scalar quantities will be represented in normal font. In addition, estimated values are represented as a letter overlaid with a caret (ˆ) and a dot product with a bold dot (•).

3.1 Heritage Algorithm

In creating a Doppler positioning method, the key is to develop a system of equations that can be used to find the unknowns of position (in x, y and z) and clock frequency. Doppler frequency is a quantity that is represented in hertz. However, it can also be represented in meters per second (velocity). The relationship between satellite velocity and Doppler frequency is exploited to develop this system of equations. In this chapter, the Doppler positioning method proposed by Hill is first derived, a modification is made to his method, and the resulting system of equations presented [12].

Jonathan Hill starts by defining $\mathbf{V^i}$ to be the velocity difference between satellite i and the receiver [12] :

$$\mathbf{V^i} = \mathbf{S_v^i} - \mathbf{U_v} \tag{3.1}$$

Where:

$\mathbf{V^i}$ = Velocity difference between satellite i and the receiver

$\mathbf{S_v^i}$ = Velocity of satellite i obtained from A-GPS data at time of signal reception

$\mathbf{U_v}$ = Velocity of the receiver

The velocity of the receiver is assumed to be zero to eliminate for three velocity unknowns in x, y and z.

Next Hill defines a normalized line of sight vector $\mathbf{L}$, pointing from the satellite position $\mathbf{S_c^i}$ to the estimated receiver position $\mathbf{U_c}$:

$$\mathbf{L^i} = \frac{\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S_c^i}}{||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S_c^i}||} \tag{3.2}$$

Where:

$\mathbf{L^i}$ = Unit vector pointing from satellite position to receiver position (Normalized line of sight vector)

$\widehat{\mathbf{U}}_\mathbf{c}$ = Estimated Receiver Position

$\mathbf{S_c^i}$ = Position of satellite i obtained from A-GPS data

Equation 3.1 and equation 4.8 are combined to get the projection of the difference between the satellite velocity and the receiver velocity along the normalized view vector, which is in fact the pseudorange rate:

$$\dot{\rho}^i = \mathbf{V^i} \bullet \frac{\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S_c^i}}{||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S_c^i}||} \tag{3.3}$$

Where:

$\dot{\rho}^i$ = Pseudorange rate

However, the Doppler shift equation is given by:

$$F_r = F_t\left(1+\frac{\dot{\rho}^i}{c}\right) = F_t + \Delta F_r \tag{3.4}$$

Where:

F_r = Frequency received by the receiver

F_t = Frequency transmitted by the satellite

ΔF_r = Doppler offset

From equation 3.4 it can be seen that the pseudorange rate is actually the Doppler frequency of the satellite in terms of speed.

$$\dot{\rho}^i = \frac{\Delta F_r}{F_t}c \tag{3.5}$$

Where:

$\dot{\rho}^i$ = Doppler frequency of the satellite in terms of speed (pseudorange rate)

c = Speed of light

This Doppler frequency can be represented in terms of θ as shown in figure 3.1 (a):

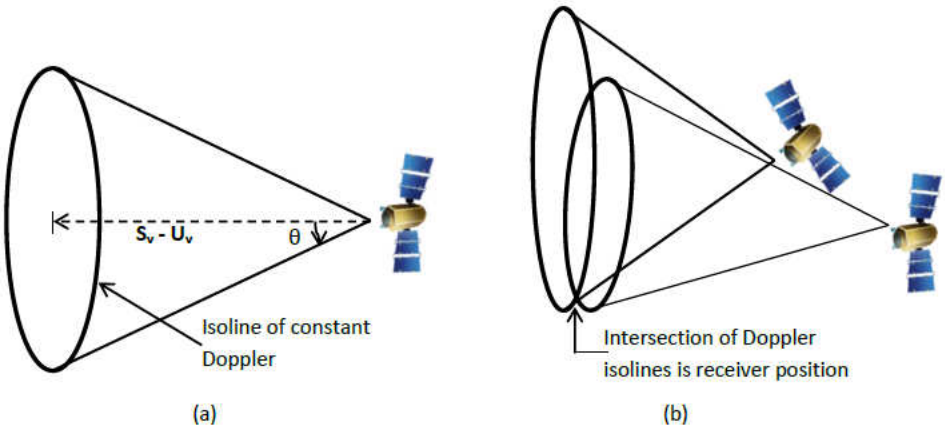

Figure 3.1: Isolines of constant Doppler frequency

θ is the angle between the direction of the Doppler frequency and the vector of the difference between the satellite velocity and receiver velocity. The equation defining figure 3.1 is given by:

$$\dot{\rho}^i = ||\mathbf{V^i}||cos\theta \tag{3.6}$$

From figure 3.1 it can be seen that $\dot{\rho}$ forms a cone of which the circumference of the circular base is a locus of points of constant Doppler frequency. The cone represents the Doppler shift of just one satellite. The intersection of the circumferences of the circular bases of a sufficient number of such cones will be an estimate of the position of the receiver. This is an example of trilateration as discussed in section 1.1.

However, satellite measurements are always plagued by errors, the dominant of which is the receiver clock. When this error is taken into account, equation 3.3 becomes:

$$\dot{\rho}^i = \mathbf{V^i} \bullet \frac{\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S^i_c}}{||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S^i_c}||} + \dot{b} \tag{3.7}$$

Where:

$\dot{b}$ = Clock frequency error

Equation 3.7 is then re-arranged as shown below

$$\dot{\rho}^i||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}|| = \mathbf{V^i}\bullet\left(\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}\right)+\dot{b}||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}|| \tag{3.8a}$$

$$\widehat{\mathbf{U}}_\mathbf{c}\bullet\mathbf{V^i} = \mathbf{S}^\mathbf{i}_\mathbf{c}\bullet\mathbf{V^i}+\dot{\rho}^i||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||-\dot{b}||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}|| \tag{3.8b}$$

At this point, Hill suggests the addition of $\dot{b}||\mathbf{S}^\mathbf{i}_\mathbf{c}||$ to both sides of equation 3.8 . This gives [12] :

$$\widehat{\mathbf{U}}_\mathbf{c}\bullet\mathbf{V^i}+\dot{b}||\mathbf{S}^\mathbf{i}_\mathbf{c}|| = \mathbf{S}^\mathbf{i}_\mathbf{c}\bullet\mathbf{V^i}+\dot{\rho}^i||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||+\dot{b}||\mathbf{S}^\mathbf{i}_\mathbf{c}||-\dot{b}||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||$$

$$\widehat{\mathbf{U}}_\mathbf{c}\bullet\mathbf{V^i}+\dot{b}||\mathbf{S}^\mathbf{i}_\mathbf{c}|| = \mathbf{S}^\mathbf{i}_\mathbf{c}\bullet\mathbf{V^i}+\dot{\rho}^i||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||+\dot{b}\left(||\mathbf{S}^\mathbf{i}_\mathbf{c}||-||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||\right) \tag{3.9}$$

Although Hill's equation (equation 3.9) now has the unknowns on the left-hand side, the clock frequency error still has a term on the right-hand side of the equation. Equation 3.9 is just for one satellite. A number of satellites define a system of equations $\mathbf{G}\widetilde{\mathbf{x}}=\mathbf{B}$ where $\mathbf{G}$ is a matrix, $\widetilde{\mathbf{x}}$ is an error vector and $\mathbf{B}$ is a vector. $\mathbf{G}$, $\widetilde{\mathbf{x}}$, and $\mathbf{B}$ are defined as follows:

$$\mathbf{G} = \begin{vmatrix} V_x^0 & V_y^0 & V_z^0 & ||\mathbf{S}^\mathbf{0}_\mathbf{c}|| \\ V_x^1 & V_y^1 & V_z^1 & ||\mathbf{S}^\mathbf{1}_\mathbf{c}|| \\ \vdots & \vdots & \vdots & \vdots \\ V_x^i & V_y^i & V_z^i & ||\mathbf{S}^\mathbf{i}_\mathbf{c}|| \end{vmatrix} \tag{3.10}$$

$$\widetilde{\mathbf{x}} = \begin{vmatrix} \widetilde{U}_{cx} \\ \widetilde{U}_{cy} \\ \widetilde{U}_{cz} \\ \widetilde{b} \end{vmatrix} \tag{3.11}$$

$$\mathbf{B}(\widetilde{\mathbf{x}}) = \mathbf{S}^\mathbf{i}_\mathbf{c}\bullet\mathbf{V^i}+\dot{\rho}^i||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||+\dot{b}\left(||\mathbf{S}^\mathbf{i}_\mathbf{c}||-||\widehat{\mathbf{U}}_\mathbf{c}-\mathbf{S}^\mathbf{i}_\mathbf{c}||\right) \tag{3.12}$$

Where:

$\widetilde{\mathbf{x}}$ = Updated position and clock error terms for each iteration of the recursive solution.

The system of equations $\mathbf{G}\widetilde{\mathbf{x}} = \mathbf{B}$ can be solved using $\widetilde{\mathbf{x}} = (\mathbf{G})^{-\mathbf{1}}\mathbf{B}$ if only 4 satellites are involved in the solution. However, for an overdetermined case (more than 4 satellites) the following equation is required:

$$\widetilde{\mathbf{x}} = (\mathbf{G^T G})^{-\mathbf{1}}\mathbf{G^T B} \tag{3.13}$$

This solution is performed recursively. The receiver position $\widehat{\mathbf{U}}_\mathbf{c}$ and magnitude of the vector pointing from the satellite position to the receiver position $||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}^\mathbf{i}_\mathbf{c}||$ change with each iteration. $\widehat{\mathbf{U}}_\mathbf{c}$ should converge closer to the true position with each iteration.

3.2 Algorithm Modification

Equations 3.10 through 3.12 are the actual implementation of Hill's algorithm [12]. This research however modifies Hill's algorithm slightly. The modification is done to produce a simpler formulation by moving terms rather than Hill's version of adding terms which was found to be slightly more complex.

Modification of this algorithm starts from equation 3.8. The clock frequency error term $\dot{b}||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}^\mathbf{i}_\mathbf{c}||$ is moved to the left-hand side of the equation instead of adding a new term.

$$\widehat{\mathbf{U}}_\mathbf{c} \bullet \mathbf{V^i} + \dot{b}||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}^\mathbf{i}_\mathbf{c}|| = \mathbf{S}^\mathbf{i}_\mathbf{c} \bullet \mathbf{V^i} + \dot{\rho}^i||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}^\mathbf{i}_\mathbf{c}|| \tag{3.14}$$

The G matrix for a system of equations based on equation 3.14 becomes:

$$\mathbf{G} = \begin{vmatrix} V_x^0 & V_y^0 & V_z^0 & ||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}_\mathbf{c}^\mathbf{0}|| \\ V_x^1 & V_y^1 & V_z^1 & ||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}_\mathbf{c}^\mathbf{1}|| \\ \vdots & \vdots & \vdots & \vdots \\ V_x^i & V_y^i & V_z^i & ||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}_\mathbf{c}^\mathbf{i}|| \end{vmatrix} \tag{3.15}$$

The $\mathbf{x}$ vector remains the same while the $\mathbf{B}$ vector changes to:

$$\mathbf{B}(\widetilde{\mathbf{x}}) = \mathbf{S}_\mathbf{c}^\mathbf{i} \bullet \mathbf{V}^\mathbf{i} + \dot{\rho}^i ||\widehat{\mathbf{U}}_\mathbf{c} - \mathbf{S}_\mathbf{c}^\mathbf{i}|| \tag{3.16}$$

The system of equations is still solved recursively using equation 3.13. The difference between this method and Hill's method is that all clock terms are consolidated on the left hand side of the equation.

The theory of the algorithm is now fully developed. This theoretical algorithm will now be tested. Section 3.3 describes the methods used to test the algorithm.

3.3 Test Methodology

The algorithm is tested on two main types of data sets: simulated data and over-the-air (OTA) GPS data.

3.3.1 Simulated Data

From equation 3.1 it can be seen that the velocity of the receiver directly influences the derivation of the Doppler Positioning solution. Therefore the impact of this velocity on the convergence and accuracy of the solution has to be investigated. This allows us to meet the goals set out in section 1.4: to understand the limits of the Doppler Positioning algorithm and hence be able to suitably integrate it with the Time-free algorithm. To do this, three scenarios are created on which to test the Doppler Positioning Algorithm:

1. Stationary receiver

2. Slow moving receiver

3. Receiver with increasing velocity

The stationary receiver scenario enables the assessment of the accuracy of the position estimate with zero velocity. A velocity that is close to that of a person on foot or a cyclist is then chosen to see by how much the accuracy is impacted. Finally the increasing velocity scenario is used to determine how the position solution degrades with increasing velocity. A GNSS simulator provides an easy and convenient method of investigating these scenarios. *The GNSS Simulator*, which is a software GNSS simulator, is used to perform these simulations in this research [1]. It allows simulation of varied receiver dynamics over a variety of time periods. Quick analysis of the algorithm is therefore made possible.

For each scenario two-line satellite orbital elements provided by NORAD (North American Aerospace Defense Command) collected on four separate days are used to create four different test cases [26]. In the stationary receiver scenario, four sets of 24 hours are simulated using four sets of two-line elements (TLEs). This is done by downloading new TLEs for the GPS constellation everyday for four days. Testing the algorithm in this way provides varied geometries in order to capture performance of the algorithm in a variety of conditions.

The 24 hour period is chosen because it is known that GPS satellites have an orbital period of 12 hours. Therefore the 24 hours ensures that many possible satellite geometries are tested during the scenario. Best case, worst case and average performance of the algorithm can therefore be determined over 4 full days of simulation.

For the slow moving receiver and receiver with increasing velocity, the receiver is set to move in one specific compass point direction (North, South, East and West) for the four different days. The velocity of the receiver is then increased from 5 m/s up to 40 m/s in steps of 5 m/s. Each velocity is maintained for 20 minutes and data is sampled

every minute. The slow moving receiver also has a simulated time of 24 hours like the stationary receiver. However, the increasing velocity receiver only has a simulated time of 2 hours 40 minutes. This is however sufficient to investigate the effect of velocity on the algorithm.

3.3.2 Over-the-Air GPS Data

The SiGe GN3S sampler is used to capture an over the air (OTA) GPS signal for a duration of about 40 seconds. The SiGe GN3S sampler downconverts the GPS L1 to an intermediate frequency (IF) of 4.1304 MHz and then samples the signal at 16.3676 Megasamples/s. This IF GPS signal is stored in a file and subsequently processed by the *fastGPS* software receiver.

fastGPS is a single frequency software receiver programmed in C [1]. It can provide position solution with or without assistance data. This research however uses it in A-GPS mode with locally stored ephemeris. *fastGPS* is first run in normal mode to obtain a position from the OTA GP RF data using the Least Squares technique.

The *fastGPS* code is then modified to implement the version of Doppler Positioning derived in section 3.2. The position obtained by Least Squares is then chosen as the reference position and the file is once again processed in *fastGPS* in Doppler Positioning mode. The positions obtained by Doppler Positioning are then compared to the reference position.

3.4 Doppler Positioning Results

3.4.1 Simulated data

Stationary Receiver

The simulated position for the stationary receiver was chosen to be 45.499° N, 73.595° W at a height of 156.56 meters above sea level. This simulated position is a location in

Parc Mont Royal in Montreal Canada.

Once the scenario has been simulated in *The GNSS Simulator*, the generated data is fed into a MATLAB script that performs Doppler Positioning and plots the results in East North (EN) format.

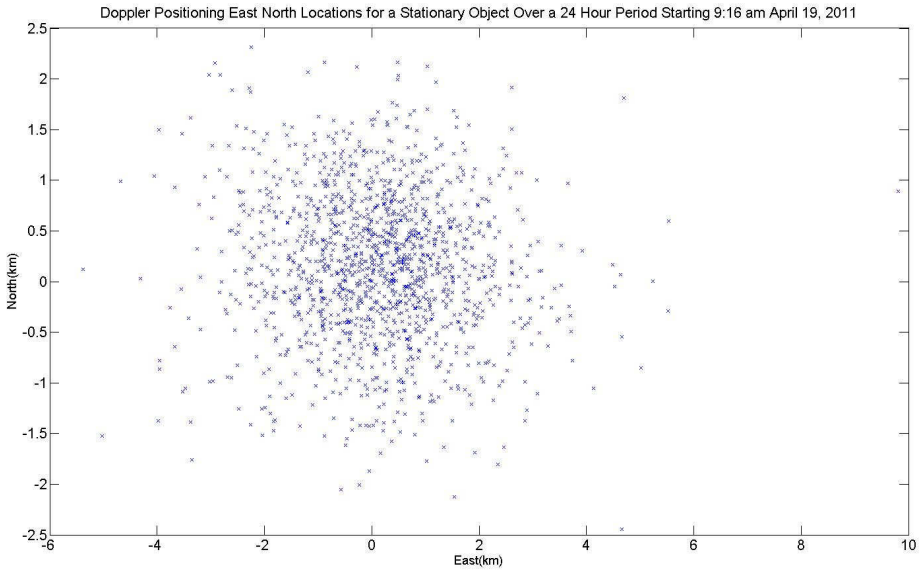

Figure 3.2: East North Position for a Stationary Receiver Over a 24 Hour Period Starting 9:16 am April 19, 2011

Figure 3.2 is the East North positions relative to the true position for data simulated on the first set of 24 hours for the stationary receiver. It can be seen that the error in the receiver position never goes above 10 km. This is well within the 100 km threshold suggested by Lannelongue and Pablos in [13] that is required for a position to be used to initialize the Time-free positioning algorithm. Throughout the 24 hour period, the error is largely less than 4 km except in a few cases. The largest error in the position was further investigated and was found to correspond to an increase in geometric dilution of precision (GDOP). GDOP is a statistical indication of how accurate a position solution is, depending on the satellite geometry. A higher GDOP value shows that the position solution is less accurate due to poor satellite geometry whereas a lower value of GDOP

shows that the position solution is more accurate due to good satellite geometry.

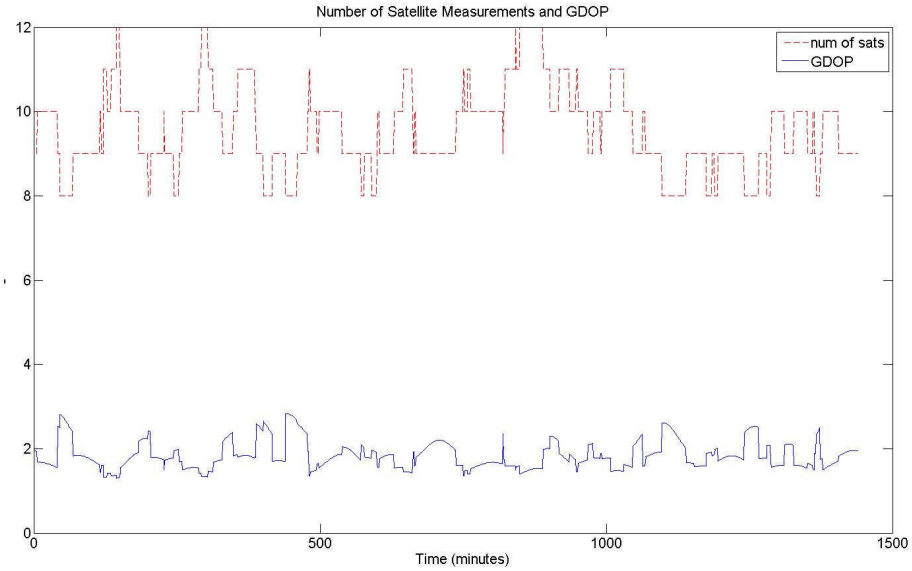

Figure 3.3: GDOP and number of satellites visible for stationary receiver test case 1

Figure 3.3 shows the relationship between number of satellites and GDOP for the stationary receiver case. In majority of the cases, a reduction in number of satellites leads to an increase in GDOP (though this may not always be the case).

Though their plots are not shown, all test case simulations for the stationary receiver showed roughly similar position solutions over their respective 24 hour periods.

Slow Moving Receiver

A starting point similar to the position of the stationary receiver was chosen for the slow moving receiver. The receiver was then simulated as moving southwards at a velocity of 1.5 m/s. These receiver dynamics correspond to a user walking on foot and using a handheld GPS receiver. A MATLAB script is again used to perform Doppler Positioning on the simulated data, and the results are plotted.

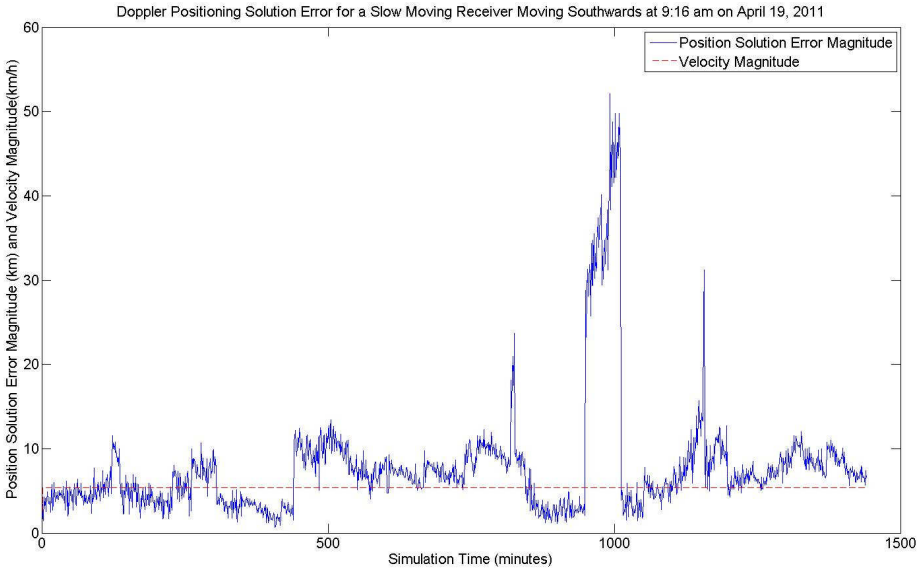

Figure 3.4: Doppler Positioning Solution Error for a Slow Moving Receiver Moving Southwards as from 9:16 am on April 19, 2011

Figure 3.4 is the error magnitude for data simulated on the first set of 24 hours for the slow moving receiver. This time period corresponds exactly to the time period during which the stationary receiver data was simulated. This means that the exact movement of satellites and hence corresponding GDOP is calculated at each time step. The error magnitudes for the slow moving receiver are however worse than those of the stationary receiver, with a maximum peak of 50 km. This peak corresponds to an increase in GDOP similar to that which was observed for the stationary receiver in figure 3.3. This is because the simulation of the stationary receiver and the slow moving receiver is over the same time period, with the exact same satellite geometries.

Increasing Velocity Receiver

The increasing velocity receiver was also simulated as starting from the same location as the stationary receiver. The initial velocity was 0 m/s but after 2 seconds of simulation time, the velocity was increased to 5 m/s and thereafter was increased by 5 m/s every 20 minutes for a period of 160 minutes of simulation time. This was to ensure that a

sufficient range of velocities was tested.

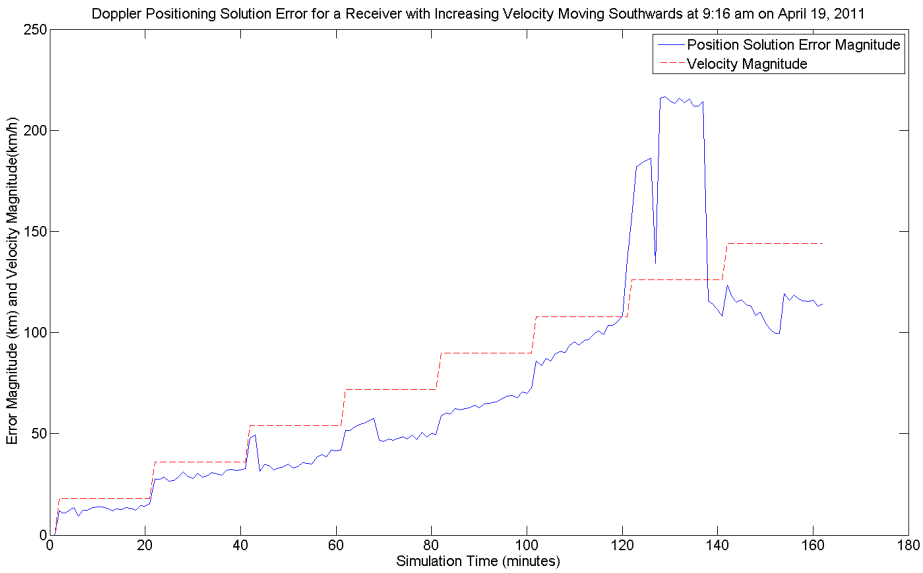

Figure 3.5: Doppler Positioning Solution Error for a Receiver with Increasing Velocity Moving Southwards at 9:16 am on April 19, 2011

From figure 3.5, it can be seen that the error magnitude increases with each increase in velocity. This is due to the fact that the algorithm used for position determination is dependent on receiver velocity, which was assumed to be zero. Therefore, each increase in velocity means that the assumed velocity of zero is more and more disparate from the true velocity.

The results also show a large degradation in position estimation 120 minutes into the simulation. This degradation is due to a low elevation satellite contributing to the Doppler position solution. Low elevation satellites tend to have larger Doppler errors than high elevation satellites. In *The GNSS Simulator*, the errors modeled are ionospheric errors, tropospheric errors, Gaussian random noise and a constant bias for each satellite [1]. These errors affect Doppler receiver positioning depending on elevation.

Hill explains how these large Doppler errors contribute to the receiver positioning using equation 3.17:

$$\Delta\theta = \frac{-\Delta\dot{\rho}^{i}}{||\mathbf{S_v^i}||sin(\theta)} \tag{3.17}$$

Where:

$\Delta\theta$ = Radial error contribution to receiver position

$\dot{\rho}^{i}$ = Error in Doppler frequency.

θ = Elevation angle.

Equation 3.17 shows that for high elevations θ, the radial error $\Delta\theta$ will be small whereas, for low elevations, the radial error will be large.

3.4.2 Over the Air GPS Data

Stationary Receiver with Receiver Time Accurate to 2 seconds

Over the air (OTA) GPS signals were captured using the SiGe GN3S Sampler at Parc Mont Royal in Montreal Canada. This data was processed in the C based *fastGPS* software receiver using the least squares positioning technique and the position obtained was used as the reference (”true”) position for the Doppler Positioning data.

The OTA GPS data was then fed into an implementation of Doppler positioning in C in *fastGPS* which generated position estimates in a text file. These position estimates where loaded into MATLAB and converted to East North positions which where then plotted as seen in figure 3.6. The position results obtained were also compared to those of a hand-held Garmin GPS receiver.

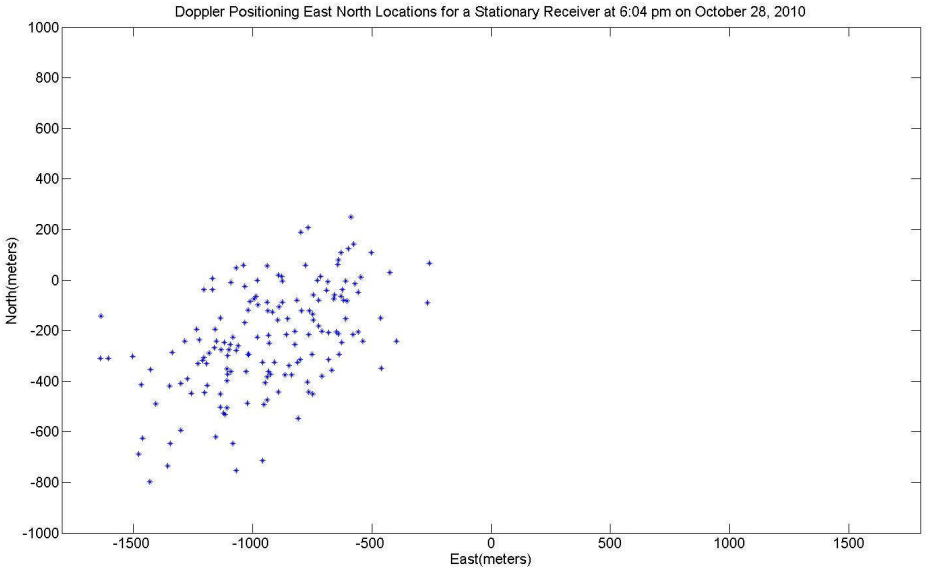

Figure 3.6: East North Position for a Stationary Receiver at 6:04 pm October 28, 2010

From figure 3.6 it can be seen that the largest error is less than 2 km. This value is obtained after the tracking loops have settled. Before the tracking loops settle, a maximum error somewhere between 20 and 25 km is obtained. A total of 173 runs of the Doppler Positioning algorithm were done by the *fastGPS* receiver as it processed data over a 35 second period. In this scenario, a priori time was accurate to within 2 seconds.

Stationary Receiver with Receiver Time Error up to 5 minutes

The same OTA GPS signal captured at Mont Royal is used to investigate the effect of a priori receiver time error on the Doppler Positioning algorithm. A Monte-Carlo simulation is run by adding a random error up to a maximum of 5 minutes to the a priori receiver time.

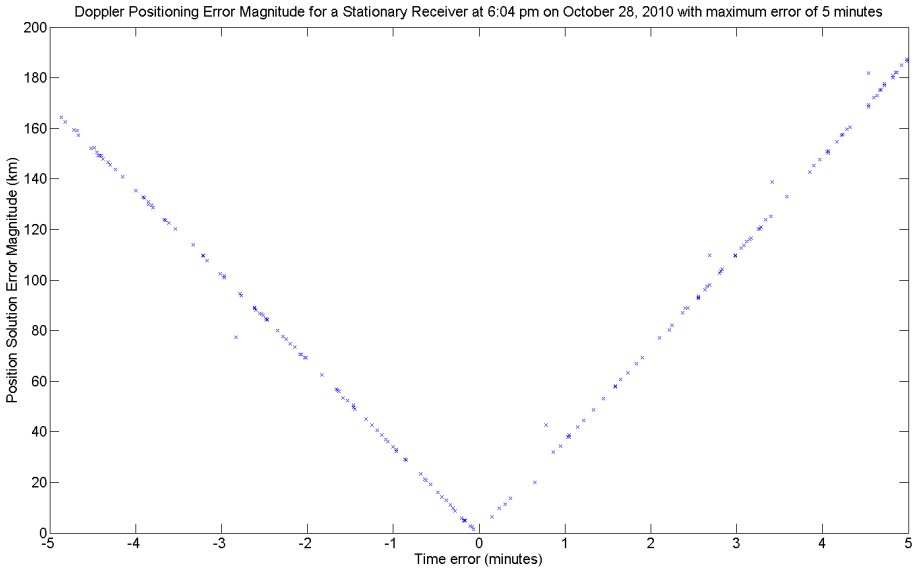

Figure 3.7: East North Position for a Stationary Receiver at 6:04 pm October 28, 2010 with a Maximum Time Error of 5 Minutes

Figure 3.6 shows the results of adding random time errors to the a priori receiver time when performing Doppler positioning. It shows that position error in kilometers increases almost linearly with a priori time error.

An a priori time error of 3 minutes or more induces a 100 km or more position error in the position estimate. This a priori time error will be compared to the a priori time error requirements of Time-free positioning in chapter 4.

The results in figure 3.7 show that the relationship between a priori time error and position error magnitude is nearly linear. This could be due to the fact that the Doppler positioning algorithm is directly dependent on clock frequency rather than on clock error itself.

3.5 Summary

The simulations and collected OTA GPS data show that Doppler Positioning provides a position estimate with an error in the order of tens of kilometers for stationary and slow moving receivers. The Monte Carlo a priori time error simulations show that a position accuracy to the same order can be maintained by keeping the a priori time accurate to within 3 minutes. In chapter 4, the a priori position and time accuracy required for Time-free positioning will be assessed to better understand how to implement a combined algorithm.

Chapter 4

Time-free Positioning

4.1 Introduction

As was mentioned in section 1.4, a receiver normally requires a large amount of data to be processed before a navigation solution can be performed. Doppler Positioning can provide a position estimate of a receiver soon after acquisition of satellites has been accomplished. Unfortunately Doppler Positioning is not accurate enough to be of use on its own. In the tests carried out in this research, Doppler Positioning yielded position estimates that were accurate to several hundreds of meters at best but often significantly more in non-ideal cases. Therefore a method is required to improve accuracy and yet still maintain the reduced processing workload for a GPS receiver. One such candidate is Time-free Positioning which was first proposed in 1995 by Peterson, Hartnett and Ottoman in [11], expounded in 1998 by Lannelongue and Pablos in [13] and further developed in 2009 by Van Diggelen in [10]. Notations of various quantities are borrowed from both [13] and [10] in this research.

In Time-free Positioning, the position of a receiver is estimated without knowledge of the precise time of transmission of a GPS signal. This automatically removes the need to extract the time of week (TOW) from the navigation data. An a priori guess of both receiver position and time and assistance ephemeris have to be provided to the

algorithm to provide a position solution.

This research will developed a Time-free positioning algorithm and demonstrate what estimated a priori position and time accuracies are required for the convergence of the algorithm.

4.2 Heritage Algorithm

Peterson, Hartnett and Ottman start by defining a time measure which they call time of arrival (TOA) [11]. Time of arrival is the time at which a GPS signal arrives at a receiver. The pseudorange equation for Time-free positioning is based on the difference between the observed time of arrival TOA and the assumed time of arrival TOA_g, modulo 1ms. Lannelongue and Pablos relate TOA to and assumed TOA to true time and coarse time using the equation:

$$TOA - TOA_g = t - t_g \text{ [1 ms]} \tag{4.1}$$

Where:

TOA = Observed time of arrival

TOA_g = Assumed time of arrival

t = True time

t_g = Coarse time, estimate of t

The observed TOA is obtained from the millisecond code phase measurements of the receiver whereas the assumed TOA is the receiver time at reception of the signal. Coarse time t_g was a term created by Peterson, Hartnett and Ottman, which will be explained later. The observed TOA is a fractional measurement and not a complete one. This is the code phase measurement and is used as a fractional pseudorange. This fractional pseudorange is described as a submillisecond pseudorange by Van Diggelen in [10] since it is a fractional value of a millisecond for GPS L1 C/A. This research

employs both fractional and submillisecond terminologies.

To obtain a full observed pseudorange, the submillisecond pseudorange is added to an integer millisecond count as follows:

$$\rho = N + \alpha \tag{4.2}$$

Where:

ρ = Complete observed pseudorange in milliseconds

N = Millisecond integer count

α = fractional pseudorange/submillisecond pseudorange

The millisecond integer count is however not available and has to be estimated, given only the fractional pseudorange. The method employed is dependent on the estimated pseudorange which is obtained using coarse time. If the fractional pseudorange is close to a millisecond integer boundary, additional bias can lead to the estimation of the wrong millisecond integer count due to a roll-over of the integer. The additional bias is termed common bias and a method of mitigating their effects will be presented. The estimated pseudoranges will therefore be defined first and the full observed pseudoranges developed thereafter.

The requirements for the Time-free algorithm are assistance ephemeris and satellite clock terms and a priori receiver position and coarse time estimates. To determine the estimated pseudoranges, satellite positions at the a priori guess of coarse time are extracted from the assistance ephemeris data and then subtracted from the a priori receiver position estimate.

$$\widehat{\rho}^{i}(\widehat{t_g}) = ||\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g}) - \widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c}}(\widehat{t_g})|| \tag{4.3}$$

Where:

$\widehat{\rho}^{i}(\widehat{t_g})$ = Estimated pseudorange of satellite i at estimated coarse time

$\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g})$ = Estimated receiver position at estimated coarse time

$\widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c}}(\widehat{t_g})$ = Satellite Position at obtained from A-GPS data at estimated coarse time

However, this estimated pseudorange is not accurate enough for use without applying corrections. A few manipulations have to be done on it in order to produce a more accurate predicted pseudorange. First, an estimated time of transmission is determined from the estimated coarse time $\widehat{t_g}$:

$$\widehat{t_{tx}} = \widehat{t_g} - \frac{\widehat{\rho}^{i}(\widehat{t_g})}{c} \tag{4.4}$$

Where:

$\widehat{t_{tx}}$ = Estimated time of transmission

c = Speed of light

This estimated time of transmission $\widehat{t_{tx}}$ is used to again extract satellite positions from ephemeris data. The accuracy of these extracted satellite positions is then further improved by accounting for the Earth's rotation during signal transmission. To compensate for the Earth's rotation, the time of flight of the signal is determined:

$$\tau = \frac{||\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g}) - \widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c}}(\widehat{t_{tx}})||}{c} \tag{4.5}$$

Where:

$\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g})$ = Estimated receiver position

$\widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c}}(\widehat{t_{tx}})$ = Satellite Position obtained from A-GPS data at estimated time of transmission

The time of flight of the signal is then combined with the rate of right ascension of Earth rotation in a transformation matrix which is applied to the satellite positions:

$$\widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c\_corrected}}(\widehat{t_{tx}}) = \begin{vmatrix} \cos\tau\dot{\Omega} & \sin\tau\dot{\Omega} & 0 \\ \text{-sin}\ \tau\dot{\Omega} & \cos\tau\dot{\Omega} & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \widehat{S^i_{cx}}(\widehat{t_{tx}}) \\ \widehat{S^i_{cy}}(\widehat{t_{tx}}) \\ \widehat{S^i_{cz}}(\widehat{t_{tx}}) \end{vmatrix} \tag{4.6}$$

Where:

$\widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c\_corrected}}(\widehat{t_{tx}})$ = Position of satellite i at estimated transmission time and corrected for the Earth's rotation

$\dot{\Omega}$ = Rate of right ascension

The predicted pseudorange is calculated thus:

$$\widehat{\rho}^i = ||\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g}) - \widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c\_corrected}}(\widehat{t_{tx}})|| \tag{4.7}$$

This predicted pseudorange takes into account the receiver position at the estimated coarse time, satellite positions at estimated time of transmission, and the Earth's rotation during the transmission of the GPS signal.

Next, just like in regular navigation using the least squares technique, the normalized line of sight vector $\mathbf{L}$ is defined , pointing from the estimated satellite position $\widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c\_corrected}}(\widehat{t_{tx}})$ to the estimated receiver position $\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g})$:

$$\mathbf{L}^{\mathbf{i}} = \frac{\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g}) - \widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c\_corrected}}(\widehat{t_{tx}})}{||\widehat{\mathbf{U}}_{\mathbf{c}}(\widehat{t_g}) - \widehat{\mathbf{S}}^{\mathbf{i}}_{\mathbf{c\_corrected}}(\widehat{t_{tx}})||} \tag{4.8}$$

Where:

$\mathbf{L}^{\mathbf{i}}$ = Unit vector pointing from satellite position to the a priori receiver position (Normalized line of sight vector)

Line of sight vectors from the satellites and difference between observed pseudorange and estimated pseudoranges are used to create an equation that linearizes pseudoranges around a rough estimate of receiver position in order to provide a position solution. This

equation is the basic GPS position solution equation and is shown below [1]:

$$L_x^i \delta x + L_y^i \delta y + L_z^i \delta z + \delta b = \rho^i - \widehat{\rho}^i \tag{4.9}$$

Where:

δx = Difference between true receiver x position and estimated receiver x position: $x - \widehat{x}$

δy = Difference between true receiver y position and estimated receiver y position: $y - \widehat{y}$

δz = Difference between true receiver z position and estimated receiver z position: $z - \widehat{z}$

δb = Common bias (receiver clock error and other common delays)

ρ^i = Observed pseudorange from satellite i

$\widehat{\rho}^i$ = Estimated pseudorange from satellite i

However, equation 4.9 is not complete. Coarse time t_g is used to determine an estimated pseudorange, but equation 4.9 does not account for the error in the estimated pseudorange due to the coarse time t_g. Coarse time t_g is a guess of the GPS time and is used to determine the estimated time of transmission $\widehat{t_{tx}}$. Since t_g is a guess, it is not accurate. It therefore follows that estimated time of transmission $\widehat{t_{tx}}$ is also not accurate, leading to incorrect determination of the satellite position and the satellite clock error [10].

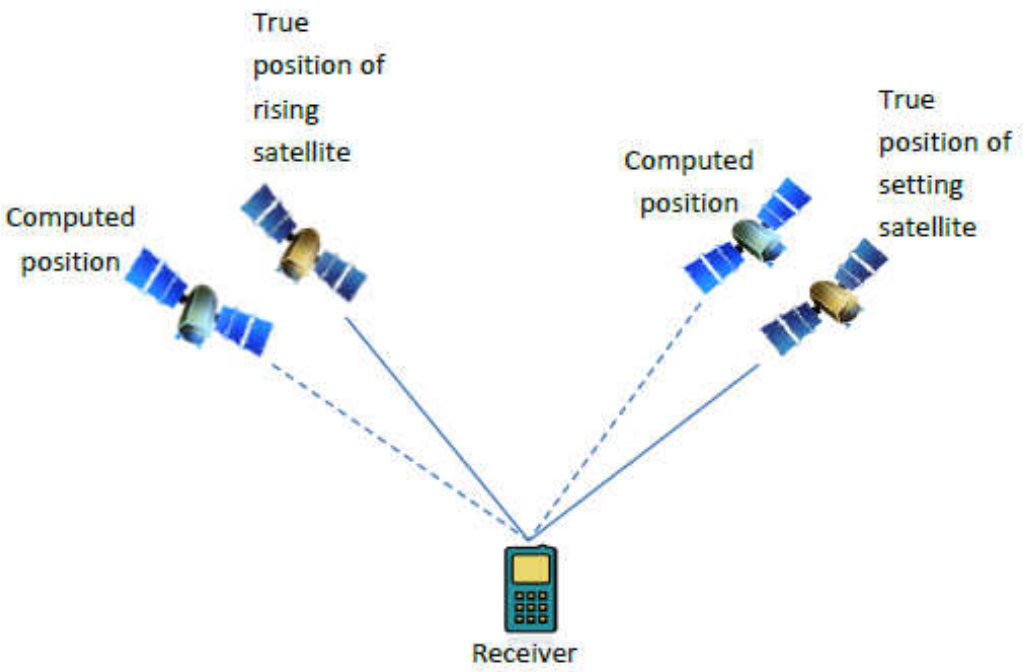

Figure 4.1: Incorrect determination of satellite positions due to the error in the assumed time of arrival

From figure 4.1 it can be seen that the error in t_g causes satellites to appear in positions where they are not. Due to the position of the satellites in the sky and their movement, each satellite is affected by t_g by differing amounts leading to each satellite having its own value of $\widehat{t}_{tx}$ that is different for the others. Both Peterson, Hartnett and Ottoman in [11] and Lannelongue and Pablos in [13] concur that the error in the coarse time (assumed time of arrival) t_g is accounted for by the pseudorange rate from the satellite to the receiver. Van Diggelen goes one step further to mathematically derive how the contribution of the pseudorange rate comes about from the error in the coarse time [10].

$$\begin{aligned} \widehat{\rho}^i(\widehat{t}_{tx}) - \widehat{\rho}^i(t_{tx}) &= \widehat{\rho}^i(\widehat{t}_{tx}) - \widehat{\rho}^i(\widehat{t}_{tx} + \delta t_g) \\ &= \dot{\rho}^i \delta t_g \end{aligned} \tag{4.10}$$

Where:

δt_g^i = Difference between estimated coarse time and true coarse for satellite i

t_{tx} = True time of transmission

$\widehat{t_{tx}}$ = Estimated time of transmission

$\dot{\rho}^i$ = Pseudorange rate of satellite i

It has already been shown from equation 3.3 to equation 3.5 that pseudorange rate can be derived from satellite Doppler frequency. Satellite Doppler frequencies are used in calculating the pseudorange rates since they are already available from the receiver tracking loops.

Equation 3.5 is thus restated:

$$\dot{\rho}^i = \frac{\Delta F_r}{F_t} c \tag{4.11}$$

Where:

$\dot{\rho}^i$ = Pseudorange rate

ΔF_r = Doppler frequency received by the receiver

F_t = Frequency transmitted by the satellite

Equation 4.10 is then combined with equation 4.9 to create the pseudorange difference equation for Time-free positioning.

$$L_x^i \delta x + L_y^i \delta y + L_z^i \delta z + \delta b + \dot{\rho}^i \delta t_g = \rho^i - \widehat{\rho}^i \tag{4.12}$$

Where:

δx = Difference between true receiver x position and estimated receiver x position: $x - \widehat{x}$

δy = Difference between true receiver y position and estimated receiver y position: $y - \widehat{y}$

δz = Difference between true receiver z position and estimated receiver z position: $z - \widehat{z}$

δb = Common bias (receiver clock error and other common delays)

$\dot{\rho}^i$ = Pseudorange rate of satellite i

δt_g^i = Difference between estimated coarse time and true coarse time

ρ^i = Observed pseudorange from satellite i

$\widehat{\rho}^i$ = Estimated pseudorange from satellite i

For several satellites, equation 4.12 forms a system of equations which is compactly represented as $\mathbf{Gx} = \delta\rho$ and where G, x and $\delta\rho$ are defined as follows:

$$\mathbf{G} = \begin{vmatrix} L_x^0 & L_y^0 & L_z^0 & 1 & \dot{\rho}^0 \\ L_x^1 & L_y^1 & L_z^1 & 1 & \dot{\rho}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ L_x^i & L_y^i & L_z^i & 1 & \dot{\rho}^i \end{vmatrix} \tag{4.13a}$$

$$\mathbf{x} = \begin{vmatrix} \delta x \\ \delta y \\ \delta z \\ \delta b \\ \delta t_g \end{vmatrix} \tag{4.13b}$$

$$\delta\rho = \begin{vmatrix} \rho^0 - \widehat{\rho}^0 \\ \rho^1 - \widehat{\rho}^1 \\ \vdots \\ \rho^i - \widehat{\rho}^i \end{vmatrix} \tag{4.13c}$$

It should be noted that equation 4.12 is in units of meters unlike equation 4.1 which is in millisecond integers. Addition of the coarse time term t_g means that the Time-free positioning pseudorange difference equation has 5 unknowns which necessitates that 5 satellites instead of 4 are required to perform a position solution.

Equation 4.12 in the form $\mathbf{Gx} = \delta\rho$ and can therefore be solved iteratively until the correction values δx, δy, δz are less than a certain threshold value. The solution for the overdetermined case is given by:

$$\mathbf{x} = (\mathbf{G^T G})^{-1}\mathbf{G^T}\delta\rho \tag{4.14}$$

4.3 Determining the Time-free (Observed) Pseudorange

Although equation 4.12 is complete, one part of its derivation has not yet been discussed. This is the determination of the Time-free (observed) pseudorange. As shown in equation 4.2, the full observed pseudorange is the addition of the submillisecond pseudorange and the integer millisecond count. Since the GPS L1 C/A repeates between 64 and 89 times for a pseudorange, the integer millisecond count N is not known and hence has to be determined. This is a tricky task to accomplish since in Time-free positioning, the time of transmission of the satellite signal is not available. Also, Time-free positioning requires a submillisecond pseudorange, whereas the GPS receiver provides a submillisecond code phase. The submillisecond code phase available from the receiver has to be reversed to obtain the submillisecond pseudorange. Consider figure 4.2 that shows how time of transmission of a sample is determined in traditional GPS.

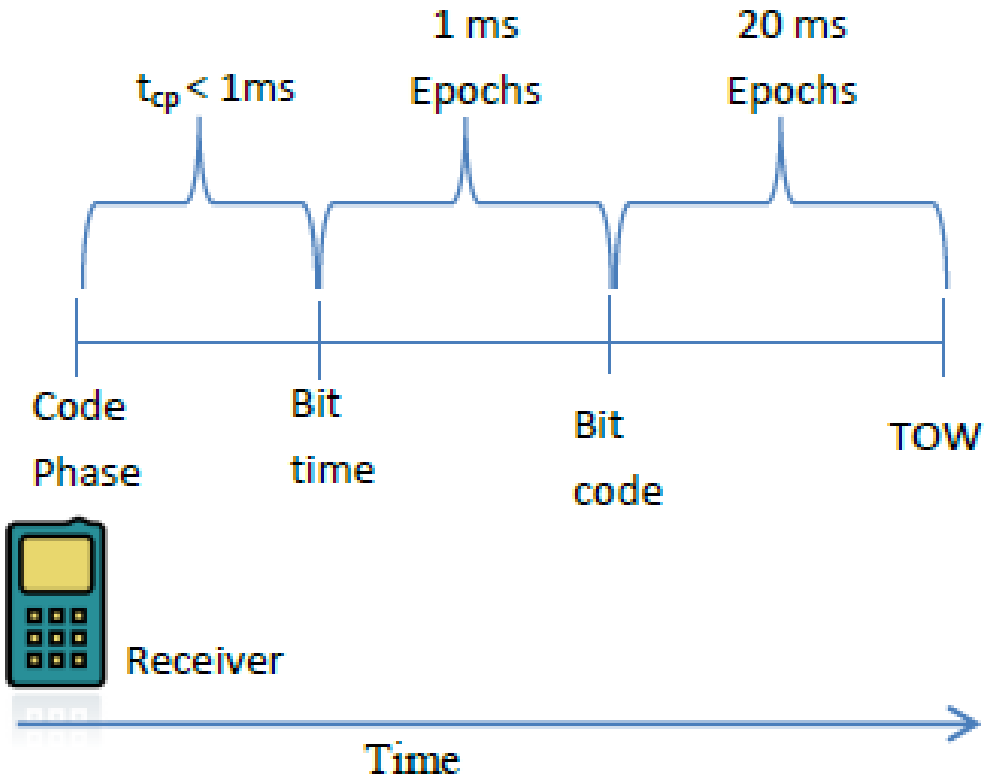

Figure 4.2: Determination of time of transmission in normal GPS

The earliest received sample in figure 4.2 is at time TOW (rightmost side of the diagram) whereas the current sample has its time of transmission at time code phase (leftmost side of the diagram). Therefore in order to calculate the time of transmission of the current sample, TOW is used as a reference point in the past. A number of 20 ms epoch bit times, 1 ms epoch bit codes and a submillisecond code phase are then added to the TOW in order to obtain time of transmission.

$$t_{tx}{}^{i} = t^{i}_{TOW} + Xt^{i}_{20ms} + Yt^{i}_{1ms} + t^{i}_{cp} \tag{4.15}$$

Where:

$t_{tx}{}^{i}$ = Time of transmission of sample

t^{i}_{TOW} = Time of Week (TOW). Time of transmission of subframe

t^{i}_{20ms} = 20 ms epoch

X = Number of 20 ms epoch repeats

t^{i}_{1ms} = 1 ms epoch

Y = Number of 1 ms epoch repeats

t^{i}_{cp} = Submillisecond code phase

The observed pseudorange is then calculated once the time of transmission of the sample has been determined

$$\rho^{i} = (t_{rx} - t_{tx}{}^{i})c \tag{4.16}$$

Where:

ρ^{i} = Observed pseudorange

t_{rx} = Time of signal reception

c = Speed of light

Consider now figure 4.3 which shows the components of a Time-free pseudorange in milliseconds, as well as how time of reception of a sample would normally be calculated.

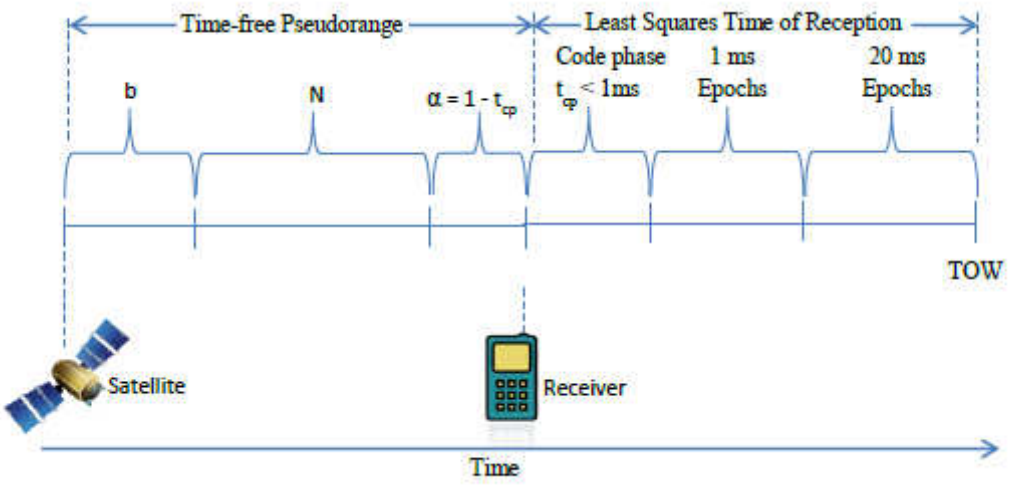

Figure 4.3: Determination of the Time-free pseudorange and its relationship to satellite transmission time

From figure 4.3, it can be seen that the measured pseudorange includes an unknown common bias b, a millisecond integer count N, and a submillisecond pseudorange α. However, in Time free positioning, only the code phase measurement is available at the receiver. The Time-free submillisecond pseudorange is inversely related to the tracked code phase as shown in 4.3. The Time-free submillisecond pseudorange is therefore determined by subtracting the tracked code phase from one.

$$\alpha^i = 1 - t^i_{cp} \tag{4.17}$$

Where:

t^i_{cp} = Submillisecond code phase

α^i = Fractional pseudorange/submillisecond pseudorange from satellite i to the receiver

From 4.3 it should also be noted that there is a common bias b in the full observed pseudorange determination whose effects will be discussed later

Now that the code phase has been obtained in the correct format, the integer count N has to be calculated. An easy approach to obtain N is by using the predicted pseu-

dorange $\widehat{\rho}^i$ as was previously mentioned. If the a priori conditions have been met, then the predicted pseudorange should be at most half a code chip (or 150 km) in error when compared to the true pseudorange. Therefore a logical way to go about determination of the millisecond integer count N for each satellite is:

$$N^i = \text{round}\ \left(\widehat{\rho}^i - \alpha^i\right) \tag{4.18}$$

Unfortunately, there is a hidden pitfall in this technique. If the true value of a fractional pseudorange is close to a millisecond integer, then the addition of the unknown commmon bias to the full pseudorange (N + fractional pseudorange) has the effect of rolling over the full pseudorange to the next millisecond integer (N + 1 + fractional pseudorange) [10].

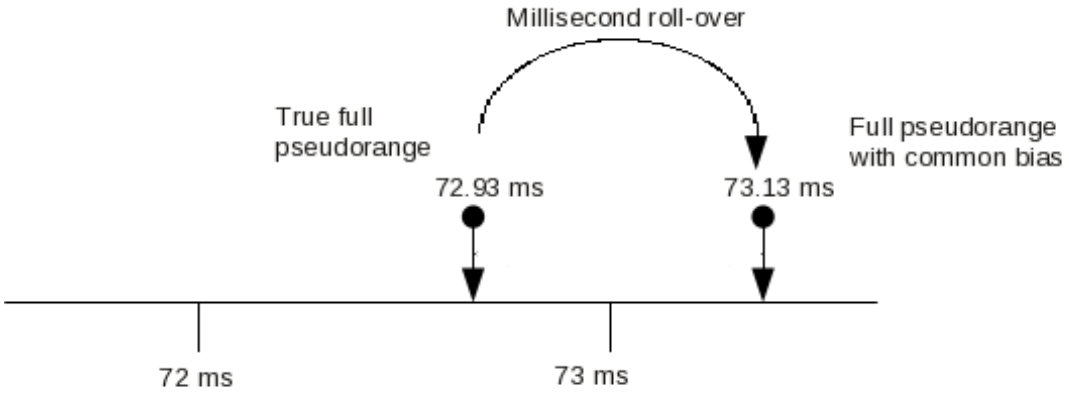

Figure 4.4: Occurence of a millisecond integer roll-over due to addition a common bias to a true fractional pseudorange lying close to an integer millisecond

Consider an example as shown in figure 4.4 in which the true pseudorange is 72.93 ms. The millisecond integer count is 72 and the submillisecond pseudorange is 0.93 ms. It can clearly be seen that the the submillisecond pseudorange is close to an integer value. If the unknown common bias in the measured pseudorange is 0.2 ms, then the measured full pseudorange will be 72.93 ms + 0.2 ms = 73.13 ms. The millisecond integer count is now 73 instead of 72 and hence a millisecond integer roll-over is said to

have occurred. A millisecond rollover in the measured pseudorange introduces an error in the position solution that is in the range of about 300 km.

Once the integer millisecond count N has been estimated, the observed pseudorange for each satellite is determined as the last required piece of equation 4.12:

$$\rho^i = N^i + \alpha^i + \Delta t^i_{sat} \tag{4.19}$$

Where:

Δt^i_{sat} = Clock error for satellite i

The pseudorange difference equation for Time-free Positioning is now complete and is restated again in equation 4.20:

$$L^i_x \delta x + L^i_y \delta y + L^i_z \delta z + c\delta b + \dot{\rho}^i \delta t_g = \rho^i - \widehat{\rho}^i \tag{4.20}$$

4.4 Correcting Integer Roll-overs

The occurence of millisecond integer roll-overs can lead to incorrect Time-free pseudoranges being applied in equation 4.20. When this occurs, the Time-free pseudorange differs with the unknown true pseudorange between the satellite and the receiver by one code chip, which is about 300 km. Lannelongue and Pablos in [13] and Van Diggelen in [10] propose techniques of dealing with this problem.

To mitigate the effect of the millisecond integer roll-over, Lannelongue and Pablos suggest that the Time-free pseudorange from one satellite be chosen to be a reference point about which all other Time-free pseudoranges are clustered such that all the pseudoranges are within half a code chip (0.5 ms) of each other [13].

Van Diggelen also suggests using the pseudorange of one of the satellites as a reference point. However, he uses the reference satellite to determine integer millisecond

count N of all other satellites, after which he determines the Time-free pseudoranges using equation 4.19. A formal equation to do this is derived in [10].

This research takes an approach similar to that used by Lannelongue and Pablos [13], but with some differences. In our approach, equation 4.12 is used as a starting point, then corrective action is taken on resulting values, and finally the equation is solved. Below is the summary steps used in our approach:

1. Determine the Time-free pseudoranges using $\rho^i = N^i + \alpha^i + \Delta t^i_{sat}$.

2. Obtain the differences between the Time-free pseudoranges and the predicted pseudoranges.

3. Sort the pseudorange differences in ascending order.

4. Compare all the pseudorange differences against the first pseudorange difference.

5. Adjust any pseudorange differences that have values greater than 150,000 km less than or more than the reference pseudorange difference.

Initially, all the Time-free pseudoranges are determined using equation 4.19. The difference of the observed and predicted pseudoranges $\delta\rho^i$ is then determined. Below is an example of a set of pseudorange difference values in meters:

$\delta\rho^i = \rho^i - \widehat{\rho}^i$
149035.31759698
448592.40845498
448964.30594639
449103.36740229
449052.55073474
148875.77063919
148723.57033195
448791.53666085
149142.52695797

The pseudorange difference values are then sorted in ascending order.

148723.57033195
148875.77063919
149035.31759698
149142.52695797
448592.40845498
448791.53666085
448964.30594639
449052.55073474
449103.36740229

After sorting the values, the smallest pseudorange difference value is selected as a reference value denoted $\delta\rho_{ref}$. In the case of the example values, this is 148723.57033195 meters. This pseudorange difference value corresponds to the satellite that has the least error in its Time-free pseudorange.

Each of the other pseudorange differences $\delta\rho^i$ is compared against the reference value $\delta\rho_{ref}$. It is expected that the difference between $\delta\rho^i$ and $\delta\rho_{ref}$ should be no more than half a code chip.

$$||\delta\rho^i - \delta\rho_{ref}|| < \frac{r_{code}}{2} \tag{4.21}$$

Where:

r_{code} = Length of 1ms PRN code, which is about 300km

If $||\delta\rho^i - \delta\rho_{ref}||$ is more than half a code chip, then the absolute value of $\delta\rho^i$ is adjusted by one chip as shown in equation 4.22 below:

$$\begin{aligned} &\text{if } (\delta\rho^i - \delta\rho_{ref} > \frac{r_{code}}{2}) \text{ then} \\ &\quad \delta\rho^i = \delta\rho^i - r_{code} \\ &\text{if } (\delta\rho^i - \delta\rho_{ref} < -\frac{r_{code}}{2}) \text{ then} \\ &\quad \delta\rho^i = \delta\rho^i + r_{code} \end{aligned} \tag{4.22}$$

From our example values, the last 5 values of the sorted $\delta\rho^i$s clearly show that a millisecond integer roll-over occurred in the determination of observed pseudoranges of the satellites associated with these measurements.

148723.57033195
148875.77063919
149035.31759698
149142.52695797
448592.40845498
448791.53666085
448964.30594639
449052.55073474
449103.36740229

These $\delta\rho^i$s are adjusted downwards by one code chip to give values that are in agreement with the reference measurement.

148723.57033195
148875.77063919
149035.31759698
149142.52695797
148592.408454980
148791.536660850
148964.305946390
149052.550734740
149103.367402290

Once all the $\delta\rho^i$ values are correctly adjusted, the Time-free equation 4.20 which is of the form $\mathbf{Gx} = \delta\rho$ can be solved recursively for the overdetermined case as follows:

$$\mathbf{x} = (\mathbf{G^T G})^{-1}\mathbf{G^T}\delta\rho \tag{4.23}$$

It should be noted however, that the pseudorange difference adjustment technique herein developed can still work for a case in which the rolled-over pseudorange difference is used as a reference and all other pseudorange differences are aligned to it. In the example values provided, if 449103.36740229 meters was used as a reference value and all other values aligned to it, equation 4.23 would still converge to the correct solution. This is because any extra bias is common and corrected as a clock error.

The Time-free positioning algorithm is now considered to be complete. In the succeeding sections, tests are developed and run on the algorithm to investigate its performance.

4.5 Test Methodology

The algorithm is implemented in *fastGPS* and tested on the same over the air (OTA) GPS data for a stationary receiver in Parc Mont Royal Montreal, Canada that was used to test the Doppler Positioning algorithm. It is also tested on data sets from the UK and the USA, supplied with [1]. For these data sets, the accuracy of the position solution

is also compared against GDOP to show how geometry impacts performance of the algorithm.

Monte-Carlo simulations are then programmed and run in *fastGPS* to investigate the behavior of the algorithm over a range of simultaneous a priori receiver position errors and receiver time errors. *fastGPS* is used since it is C-based and hence has much faster execution than MATLAB, making it suitable for performing a large number of Mont Carlo simulations. The Monte Carlo setup involves using the available OTA data sets as a source of real code phase measurements and using the mean position of the Least Squares position solution of the data set as the reference position. Random position errors measuring upto 200 km are added to the reference position in the Earth centered Earth fixed (ECEF) reference system and the resulting values are used as initialization position inputs to the algorithm. Random time errors of upto 2 minutes and 30 seconds are also added to the initialization coarse time and fed into the Time-free algorithm. About 2,000 Monte Carlo simulation runs are done and the results of both convergent and non-convergent solutions plotted.

4.6 Time-free Positioning Results

4.6.1 Stationary Receiver with Receiver Time Accurate to 2 seconds

Once again the OTA GPS signal captured at Parc Mont Royal in Montreal Canada was used. Figure 4.5 shows the position of the receiver within the park.

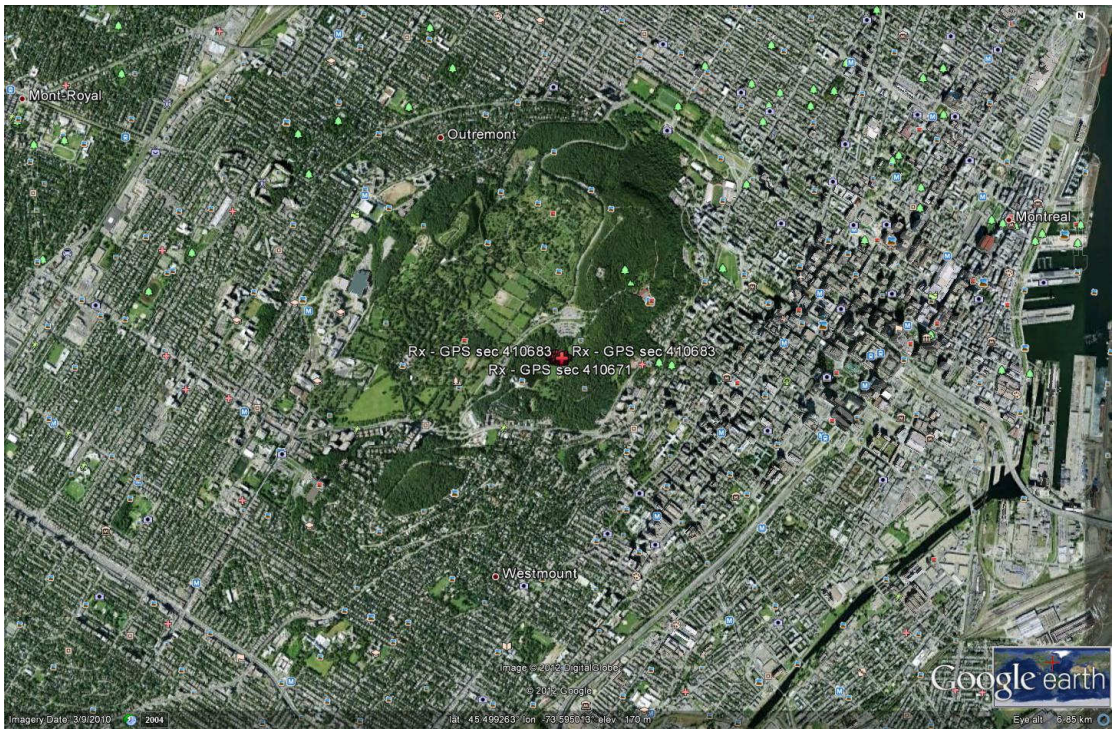

Figure 4.5: Google Earth Position for a Stationary Receiver at 6:04 pm on October 28, 2010 at Parc Mont Royal

This data was processed normally and the position obtained was used as the reference position for the Time-free Positioning East North data as seen in figure 4.6.

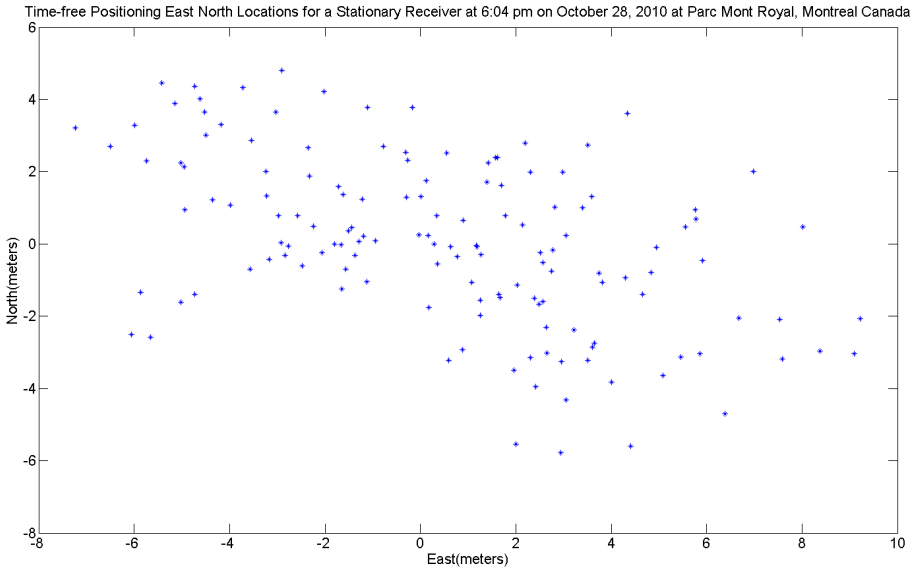

Figure 4.6: East North Position for a Stationary Receiver at 6:04 pm October 28, 2010

Additional OTA GPS data sets for a stationary receiver were fed into the Time-free Positioning algorithm and the results of the mean position error and average GDOP are tabulated in table 4.1.

Location	**Date**	**Mean Error (meters)**	**Avg GDOP**
Guildford UK	May 15 2008	3.8883	2.3517
Stanford USA	May 16 2008	13.6723	5.9972
Savannah USA	September 23 2009	6.6460	2.1125
Mont Royal Canada	October 20 2010	8.2625	3.5807

Table 4.1: Mean position error and GDOP for OTA GPS data sets used with the *fastGPS* software receiver

From the results in figure 4.6, and table 4.1, it can be seen that acceptable accuracy of less than 10 meters is achieveable with the Time-free Positioning algorithm.

4.6.2 Monte Carlo Simulations

In this experiment, random errors were added to the initial receiver position and receiver time estimates in *fastGPS*. The Time-free algorithm was run and the results dumped in a text file. A plot of the results was then made in MATLAB to illustrate when the

algorithm had converged for the approximately 2,000 test cases. The mean of the Least Squares position solution was used as the truth. The results are shown in figure 4.7.

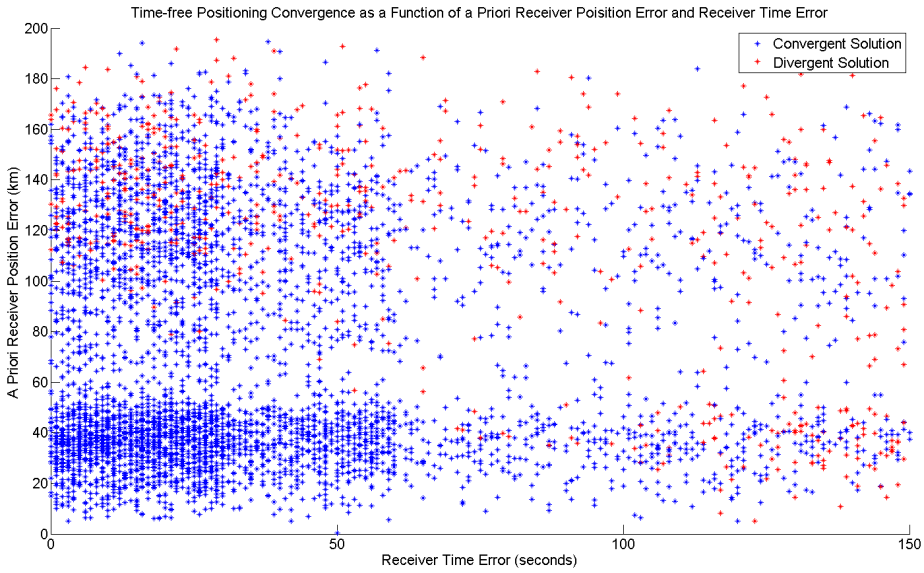

Figure 4.7: Time-free Positioning Convergence as a Function of a Priori Receiver Position Error and Receiver Time Error

In figure 4.7, the blue spots indicate convergence for the specified a priori receiver position error and receiver time error while the red spots indicate divergence of the Time-free algorithm. The results show that the Time-free Positioning algorithm converged with an a priori receiver position that is in error to the tune of 100 km, as long as the receiver time has very small error. Alternatively, the algorithm converged with a time error of up to 2 minutes as long as the a priori receiver position was highly accurate.

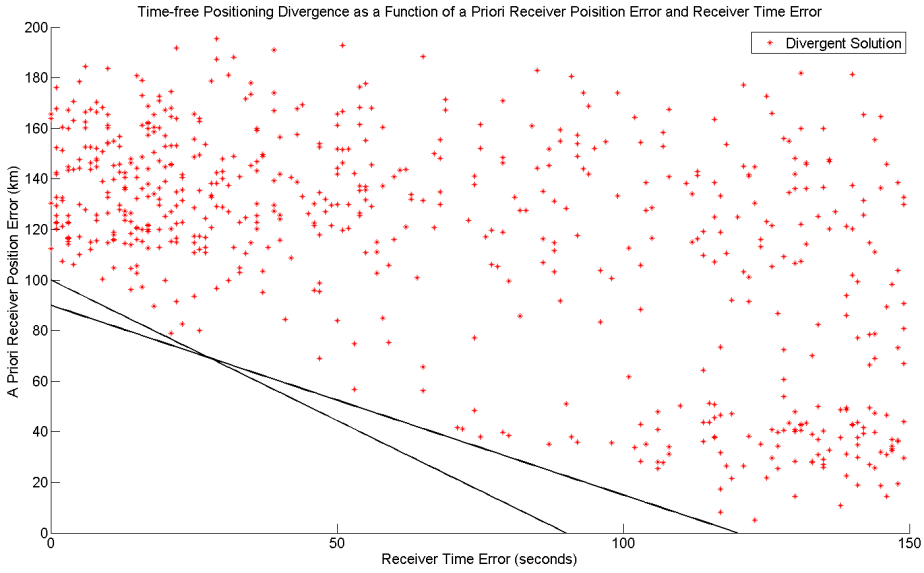

Figure 4.8: A priori error conditions and successful Time-free positioning estimates using Monte Carlo Simulations

Figure 4.8 defines possible regions of convergence of the Time-free algorithm. These regions of convergence are defined by drawing lines that demarcate the areas where no divergence of the algorithm occurs. Two lines are drawn: one with a steep slope and one with a gentle slope. The steep slope line defines the region of convergence for cases where the receiver has high a priori position uncertainty but can obtain highly accurate a priori coarse time estimates. The gentle slope line on the other hand is for receivers that require less accurate a priori coarse time estimates but can have more reliable a priori position estimates.

4.7 Summary

In this chapter, the Time-free positioning algorithm was developed. A technique for mitigating integer roll-overs was presented and the resulting algorithm was tested on live GPS signals collected in various locations in North America and Europe. Monte Carlo simulations where then performed on the Time-free algorithm to determine its region of convergence.

Regions of convergence for a receiver using this algorithm can be defined by deciding which quantity between a priori coarse time and a priori position estimate, the receiver can accurately obtain. The recommended rule of thumb is to select compromise limits. For example, figure 4.8 has two possible regions of convergence whose boundaries intersect at a point. This intersection point corresponds to a priori position estimate that is accurate to within 70km of the truth, and an a priori time that is accurate to within 20 seconds of the truth. These are good compromise limits for many receivers.

Now that the Time-free algorithm is developed and tested, integration with the Doppler positioning algorithm will be done in the next chapter. Integration of these two algorithms will be based on results from chapters 3 and 4.

Chapter 5

Time-free Positioning Initialized by Doppler Positioning

In chapter 4, it was determined that Time-free Positioning requires an a priori position estimate that is accurate to within 70 km of the true position and a receiver time that is accurate to within 20 seconds, to have assured algorithm convergence. If this position estimate error goes beyond 100 km, there is very high probability of divergence of the algorithm even with accurate receiver time. Although a priori coarse time accuracy for Time-free Positioning is theoretically expected to be at least 60 seconds [13], a coarse time accuracy threshold of 20 seconds is selected for the version of Time-free Positioning in this research based on the results of the Monte Carlo simulations in chapter 4.

In chapter 3, it was discovered that Doppler Positioning gives an estimated position that is accurate to within 70 km when the receiver is moving at speeds of 100 km/h or less (see figure 3.5). It was also determined that Doppler Positioning gives position estimates accurate to within 70 km from the true position for stationary receivers when the time accuracy is maintained to within 3 minutes of the true time (see figure 3.7). These limits where however found to be GDOP and satellite elevation dependent. Low elevation satellites or poor GDOP where found to adversely affect the position solution

causing poor Doppler positioning results even at low receiver dynamics.

Now that both Doppler Positioning and Time-free Positioning have been developed and tested individually, a new method that combines the two can now be created.

5.1 Algorithm Development

Figure 5.1 shows the processing stages when Time-free Positioning is initialized by Doppler Positioning.

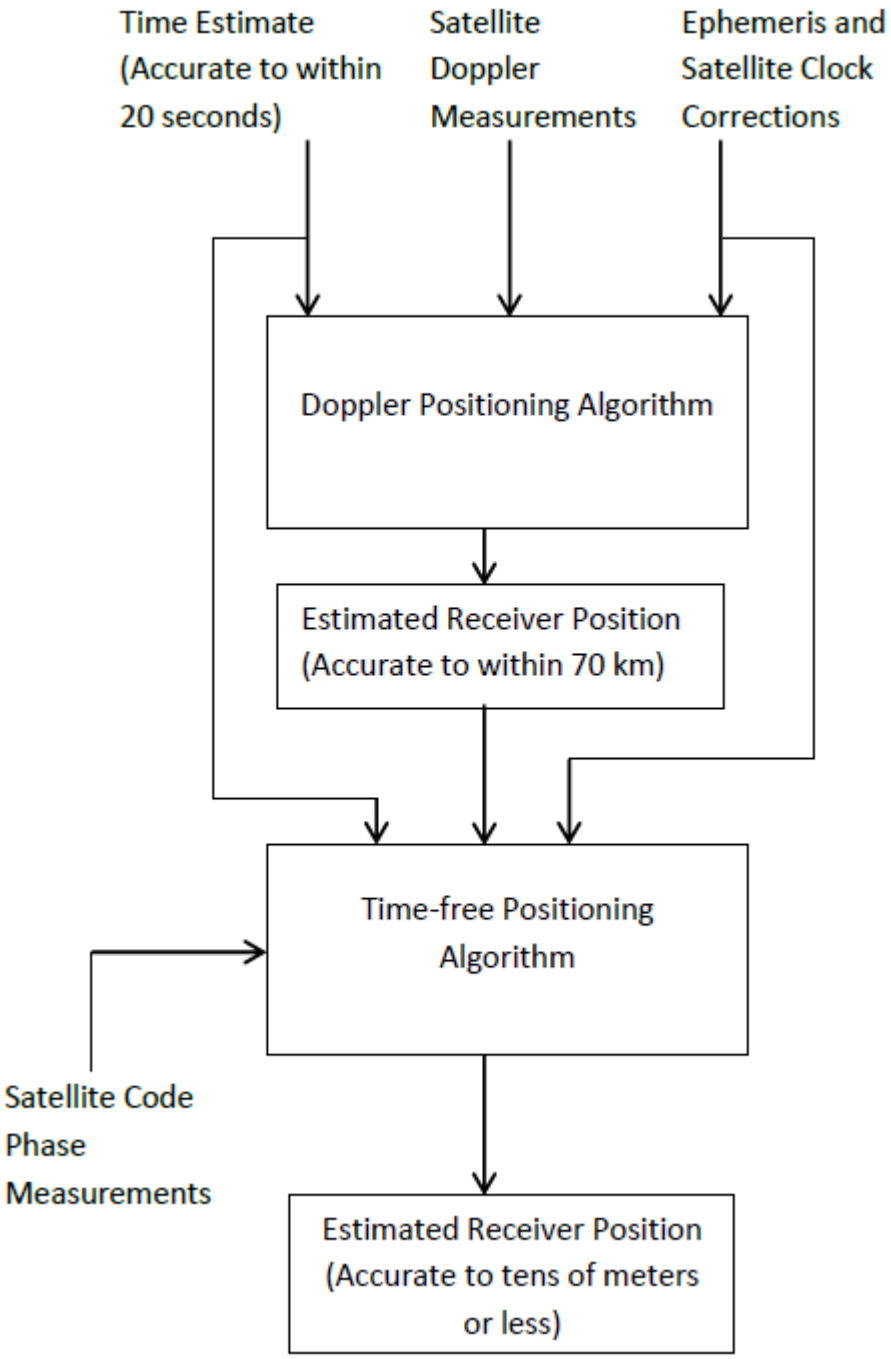

Figure 5.1: Processing stages in Time-free Positioning initialized by Doppler Positioning

From figure 5.1, it can be seen that initially 3 sets of input are required for the Doppler Positioning algorithm: a priori time estimate, satellite Doppler measurements, Ephemeris and satellite clock corrections. The satellite Dopplers are provided by the

tracking function of the software receiver. The time estimate is obtained from the receivers clock whose accuracy should be within 20 seconds of the true GPS time. This time constraint is more than sufficient for most receivers, but is chosen to allow for complete confidence in the position solution obtained. The ephemeris and satellite clock corrections are externally sent to the receiver to ensure that ephemeris from the satellite signal is not needed.

Once the Doppler Positioning module has obtained a position solution, this position solution is fed into the Time-free Positioning module. This position solution should be accurate to within 70 km of the true position, for the Time-free Positioning Algorithm to work consistently. Figure 3.5 shows that Doppler Positioning can produce a position estimate accurate to 70 km if receiver speeds are below 100 km/h. However, cases of poor satellite geometry can cause an accuracy poorer than 100 km to be obtained at lower receiver velocities.

The Time-free Positioning Algorithm also requires an a priori coarse time estimate, satellite ephemeris and code phase measurements. Just like in Doppler Positioning, the a priori coarse time is obtained from the receiver clock. The same stringent time accuracy is maintained for Time-free Positioning in order to ensure that there is a high level of confidence in the position solution convergence. Ephemeris and satellite clock corrections that are locally stored at the receiver also play the same role as in Doppler Positioning.

The code phase measurements are the last inputs into the Time-free Positioning module. They are obtained from the tracking module. It is from the code phase measurements that the Time-free Positioning algorithm generates an observed pseudorange as discussed in chapter 4.

Once the Time-free Positioning Algorithm is run once, its position solution can be reused in subsequent runs of the algorithms. Checks are however put in place to ensure that if the Time-free Positioning Algorithm diverges, its position estimate is not fed back into the next run of the algorithm. Such checks include comparing the magnitude of the

correction values of the Time-free algorithm and ensuring that the position magnitude is not a very large value.

5.2 Test Methodology

The algorithm is tested on the OTA GPS data for a stationary receiver in Parc Mont Royal Montreal, Canada that was used to test both the Doppler Positioning algorithm and Time-free Algorithm. It is also tested on data sets from the UK and the USA, supplied in [1]. Most of the data sets used are of at least 37 seconds. This would normally be sufficient data to perform positioning with the least squares GPS positioning algorithm using ephemeris data from the satellite signal itself. However, in all test cases, ephemeris stored locally at the receiver is used.

East North (EN) graphs are plotted for all the data sets to assess the performance of the combined algorithm. The EN locations are a method of quickly visually assessing the performance of the algorithm. Statistical measures of mean, variance and standard deviation are then calculated to assess the overall performance and accuracy of this technique.

The combined algorithm error magnitudes are also compared against the normal least squares algorithm. The least squares algorithm which is the most commonly used in receivers provides a good benchmark against to compare the performance of the combined Doppler Time-free positioning algorithm.

5.3 Combined Doppler Time-free Positioning Results

5.3.1 East North Plots of Stationary Receivers in Various Locations

Figure 5.2 to figure 5.5 show east north (EN) plots generated by the combined Doppler Time-free positioning algorithm for a stationary receiver at various locations in North America and Europe.

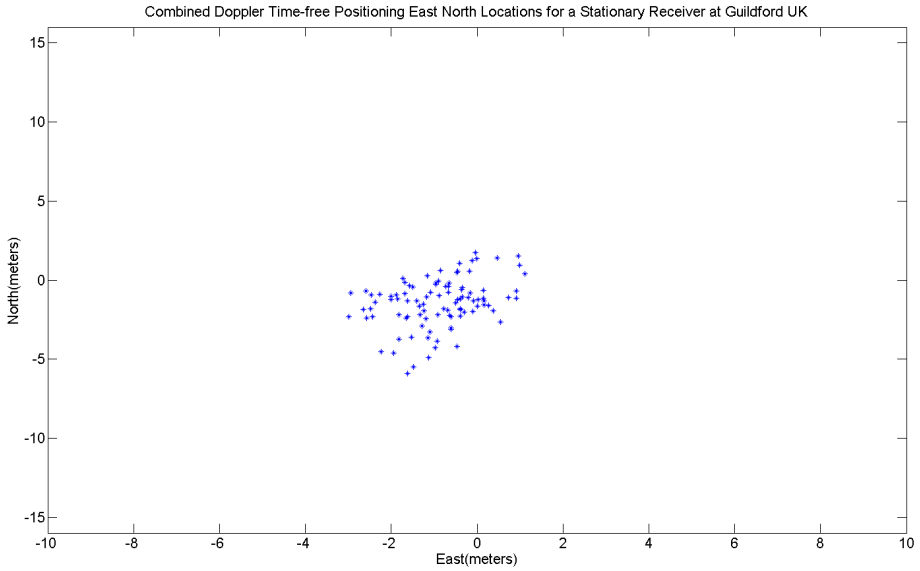

Figure 5.2: Combined Doppler Time-free Positioning East North Location Plot of a Stationary Receiver in Guildford UK on May 15, 2008

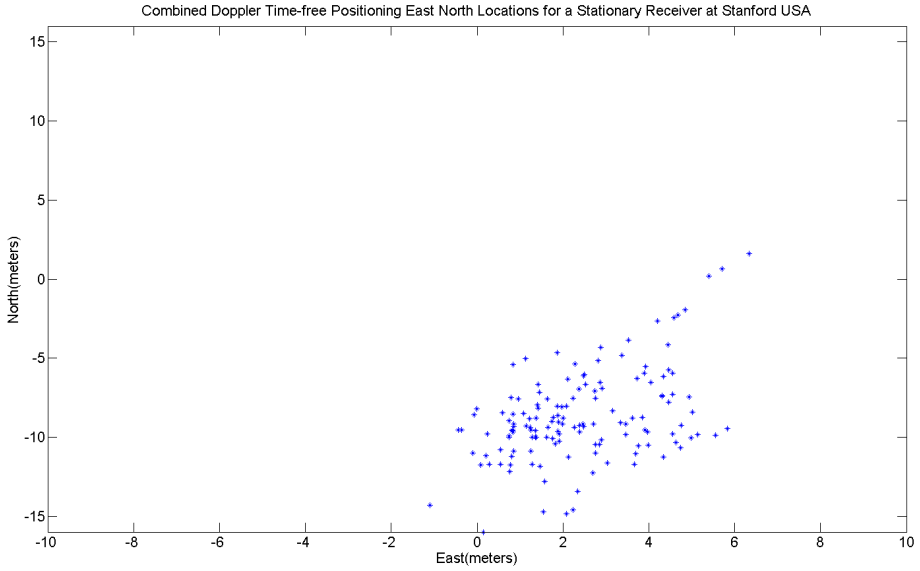

Figure 5.3: Combined Doppler Time-free Positioning East North Location Plot of a Stationary Receiver in Stanford USA on May 16, 2008

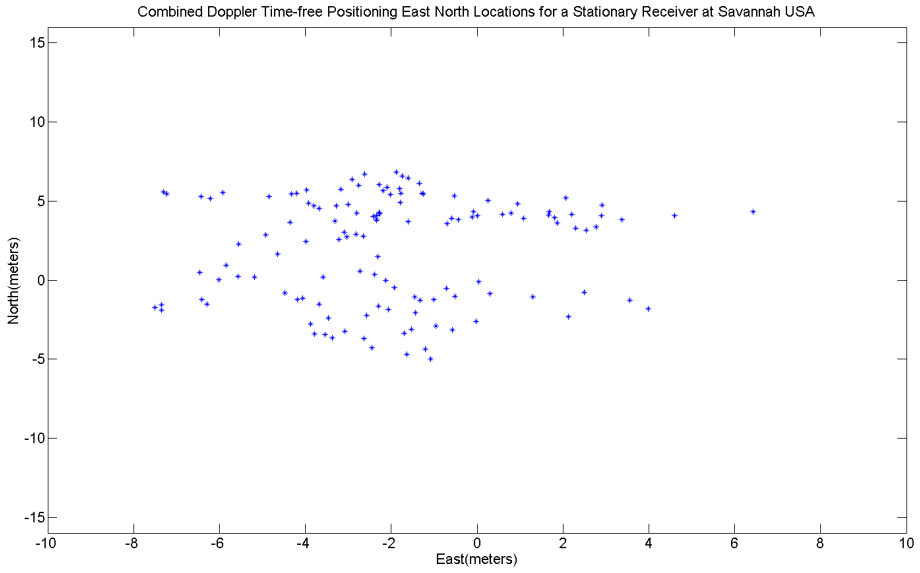

Figure 5.4: Combined Doppler Time-free Positioning East North Location Plot of a Stationary Receiver in Savannah USA on September 23, 2009

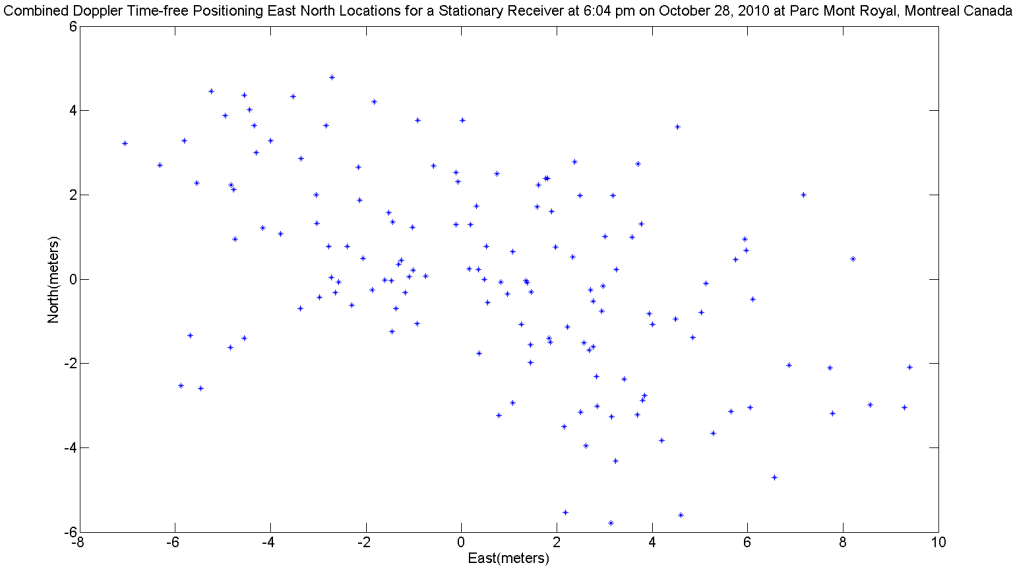

Figure 5.5: Combined Doppler Time-free Positioning East North Location Plot of a Stationary Receiver in Montreal Canada on October 20, 2010

In all the data sets, the true position was determined by taking the mean of the

Location	Mean (meters)	Variance	Standard Deviation (meters)
Guildford UK	3.8883	2.8342	1.6835
Stanford USA	13.6723	13.8037	3.7153
Savannah USA	6.6460	7.2819	2.6985
Mont Royal Canada	8.2625	16.0656	4.0082

Table 5.1: Statistical measures of the position errors for the combined Doppler Time-free positioning algorithm for the data sets used in *fastGPS*

positions obtained from the least squares solutions. The Guildford data set in figure 5.2 shows the best performance, with a mean position error of 3.8883 meters and a standard deviation of 1.6835 over a 34 second period.

The worst performance on the other hand is observed in the Stanford data set in figure 5.3 with a mean position error of 13.6723 meters and a standard deviation of 3.7153. In the Stanford results the majority of the receiver position estimations in the EN plot are also offset to one side of the true position (0,0). This suggests poor satellite geometry. The worse the satellite geometry, the higher the GDOP. The average GDOP for the Stanford data is 5.9972 whereas the average GDOP for the Guildford data set is 2.3517.

5.3.2 Comparison of the Least Squares and the Combined Doppler Time-free Positioning Algorithms

The position solution error magnitude performance of the Combined Doppler Time-free Positioning algorithm is then compared to that of the least squares positioning algorithm. It should be noted that the Combined Doppler Time-free Positioning is not compared to carrier phase positioning or other high precision techniques used in geodesy and surveying since those are not the intended applications of the algorithm.

The error magnitudes of all data sets are plotted to compare the Combined Doppler Time-free Positioning algorithm to the normal GPS Positioning Algorithm in figure 5.6 to figure 5.9.

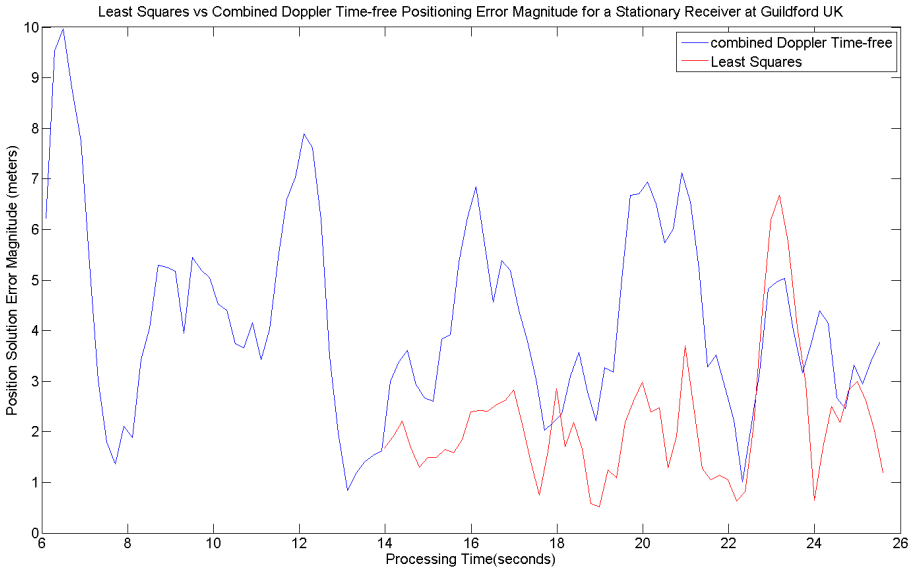

Figure 5.6: Combined Doppler Time-free Positioning versus Assisted Least Squares Error Magnitude Plots of a Stationary Receiver in Guildford UK

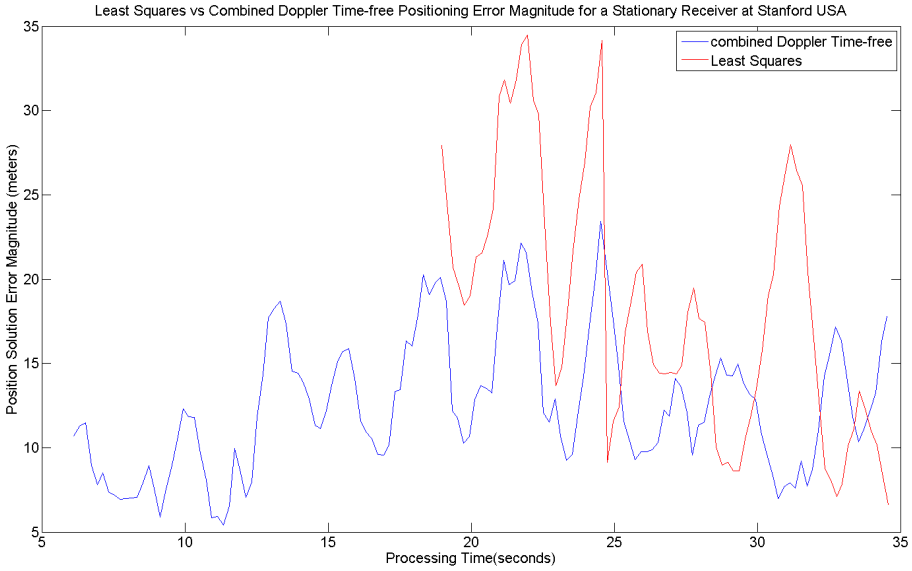

Figure 5.7: Combined Doppler Time-free Positioning versus Assisted Least Squares Error Magnitude Plots of a Stationary Receiver in Stanford USA

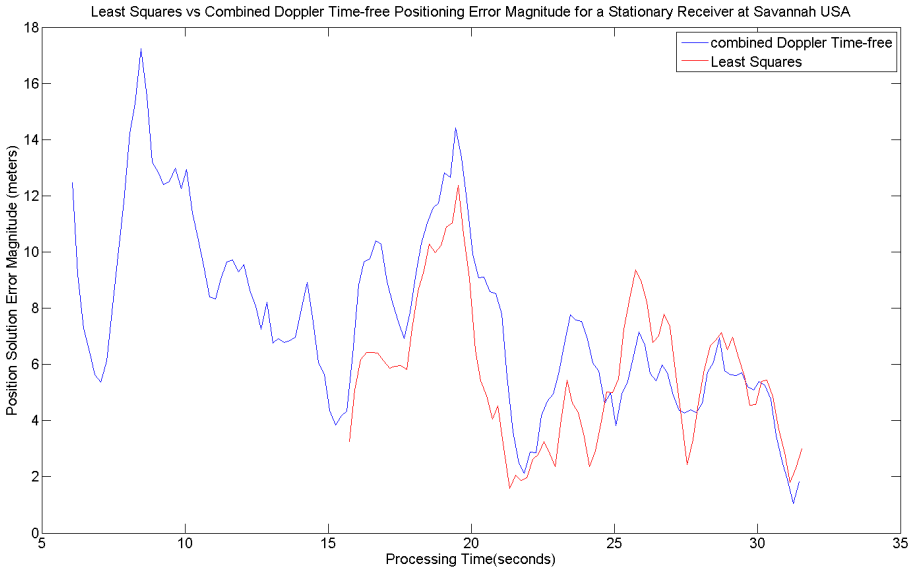

Figure 5.8: Combined Doppler Time-free Positioning versus Assisted Least Squares Error Magnitude Plots of a Stationary Receiver in Savannah USA

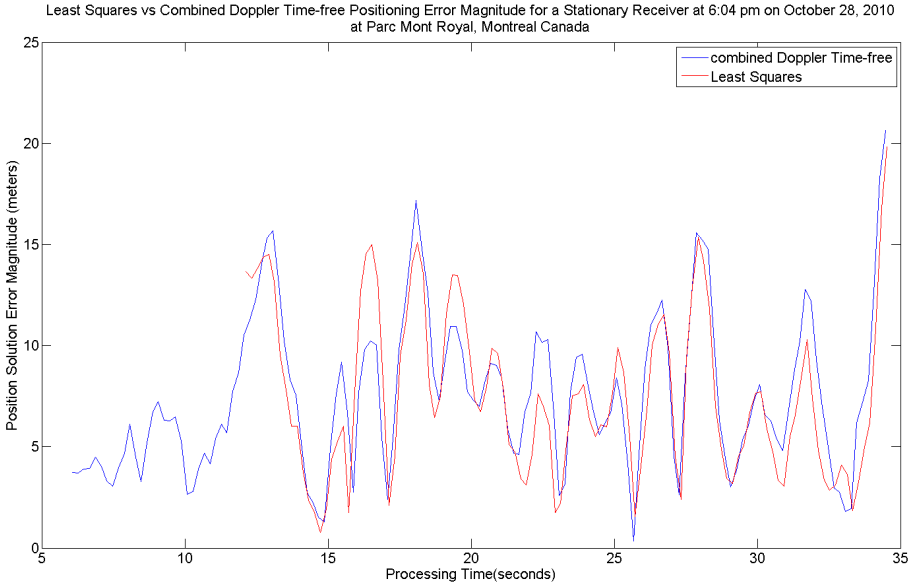

Figure 5.9: Combined Doppler Time-free Positioning versus Assisted Least Squares Error Magnitude Plots of a Stationary Receiver in Montreal Canada

The results in figure 5.6 to figure 5.9 instantly make obvious an advantage of the

combined Doppler Time-free positioning algorithm over the Least Squares positioning algorithm. This advantage is, being able to estimate the receiver position earlier. The combined Doppler Time-free algorithm provides the first position estimate 8 seconds earlier than the least squares positioning algorithm in the Guildford data set and 12 seconds earlier in the Stanford data set. However in all the data sets, combined Doppler Time-free position determination begins at least 5 seconds after satellite acquisition. This could be due to the non-optimized tracking loops in *fastGPS* that do not settle quickly enough.

The estimated error magnitudes between the two algorithms are comparable in all of the cases except in the Stanford data set case (figrue 5.7). In the Stanford case, a difference in position error of almost 15 meters is occasionally observed between the combined Doppler Time-free and the Least Squares positioning algorithms. The comparability of performance of the two algorithms is further illustrated using statistical measures pertinent to GPS as shown in tables 5.2 and 5.3.

Location	**DRMS (meters)**	**2DRMS (meters)**	**CEP (meters)**	**R95 (meters)**
Guildford UK	2.002	4.0041	1.6684	3.3367
Stanford USA	6.6725	13.3449	5.5604	11.1208
Savannah USA	4.2614	8.5227	3.5511	7.1023
Mont Royal Canada	6.36	12.72	5.3	10.6

Table 5.2: GPS Statistical measures of the position errors for the combined Doppler Time-free positioning algorithm for the data sets used in *fastGPS*

Location	**DRMS (meters)**	**2DRMS (meters)**	**CEP (meters)**	**R95 (meters)**
Guildford UK	2.0819	4.1638	1.7349	3.4698
Stanford USA	18.1742	36.3485	15.1452	30.2904
Savannah USA	3.4741	6.9482	2.8951	5.7902
Mont Royal Canada	6.2555	12.5110	5.2129	10.4259

Table 5.3: GPS Statistical measures of the position errors for the least squares positioning algorithm for the data sets used in *fastGPS*

As seen in tables 5.2 and 5.3, the performances of the two algorithms are comparable except in the Stanford data set case. The statistical parameters shown are described in [27] as follows:

Distance root mean square (drms)

If the distribution is close to a circle, it is the probability that 63% of the horizontal error is within a circle of drms radius. If the distribution is elliptical (elongated), then the probability increases to 69%

Twice the distance root mean square (2drms)

The probability that a horizontal error is within a circle of 2drms is between 95% and 98% depending on whether the scatter points form a circle or an ellipse

Circular error probable (CEP)

The radius of a circle, centered at the true position, that contains 50% of the points in the horizontal scatter plot.

Horizontal 95 percent accuracy (R95)

The radius of a circle, centered at the true position, that contains 95% of the points in the horizontal scatter plot.

The positive results show that the Combined Doppler Time-free positioning works, hence allowing time free positioning without having a priori knowledge of the receiver position.

Chapter 6

Conclusion

6.1 Conclusion

The live (OTA) signal results in chapter 5 show that the combined Doppler Time-free Algorithm has acceptable performance for stationary receivers while the simulation results in chapter 3 show that it also works with low dynamics receivers. The algorithm provides position estimates with error magnitudes of a few meters as shown by the results in figures 5.6 to 5.9. Figures 5.6 to 5.9 also show that the algorithm has performance (in terms of position error magnitude) comparable to that of the least squares algorithm for stationary receivers. Figures 3.4 and 3.5 show that at low GPS receiver dynamics, Doppler positioning can provide position estimates that are well within the limits of position initialization for Time-free positioning.

The EN scatter plots in figures 5.2 to 5.5 and the statistical measures in table 5.1 show that there is also comparable accuracy of position estimates of a stationary receiver. In all the figures, position solutions of a stationary receiver performed over an approximate 30 second period are closely clustered together and the standard deviations listed in table 5.1 are reasonably low. This shows that the algorithm has consistent performance, albeit contingent on satellite geometry.

It should however be noted that to achieve this kind of performance, previously

determined a priori time estimate and the receiver velocity limits must be respected. The a priori time estimate should have an error no greater than 20 seconds whereas the receiver velocity should be no more than 100 km/h.

It was determined from figure 3.5 that Doppler Positioning can provide position estimates with accuracy of 70 km or less if the receiver dynamics are less than 100 km/h. With 70km a priori position initialization, the Time-free Algorithm was found to converge with a maximum a priori time initialization error of 20 seconds. Though these were the limits chosen for this research, the results show that other combinations of limits are possible. For example, with zero receiver dynamics, Doppler Positioning can produce a position estimate that is less than 10 km in error. This allows Time-free Positioning to converge with an a priori coarse time initialization of accuracy upto 2 minutes. A range of receiver dynamics and a priori coarse time initialization accuracy limits are thus available for the implementation of the combined Doppler Time-free Positioning Algorithm. These limits can be chosen based on the particular application.

The combined Doppler Time-free Positioning Algorithm has three notable advantages. The first advantage is that it does not require receiver a priori position knowledge like the heritage Time-free Positioning Algorithm. This opens up applications to the combined Doppler Time-free Positioning Algorithm that had previously been impractical for Time-free positioning due to the a priori receiver position knowledge requirement.

The second advantage was observed in section 5.3 where the combined Doppler Time-free Positioning Algorithm provided position estimates at least 8 seconds before the normal GPS least squares positioning algorithm with ephemeris assistance. This is due to the need for the normal assisted GPS positioning to decode time from the GPS signal even though it has ephemeris available.

The lack of need for the combined Doppler Time-free Algorithm to decode time from the GPS signal results in the third advantage that allows the meeting of one of the main goals of the research as stated in section 1.4. This is the reduction of processing

burden on software receivers. Not a single subframe has to be be processed in order to obtain ephemeris.

Table 6.1 shows a summary of advantages and disadvantages of the Combined Doppler Time-free Positioning Algorithm as compared to the normal GPS Least Squares Positioning Algorithm.

Advantages	**Disadvantages**
Does not require a priori position knowledge of the receiver	Requires 5 satellites to estimate receiver position whereas the normal GPS Positioning Algorithm requires 4 satellites
Provides an initial position solution faster than the normal GPS Positioning Algorithm	The algorithm accuracy and convergence is dependent on receiver dynamics
Reduces processing burden on the GPS receiver	

Table 6.1: Advantages and disadvantages of the combined Doppler Time-free Positioning Algorithm

6.2 Future Work

The combined Doppler Time-freee Positioning Algorithm can be further improved and expanded. The accuracy of the Doppler Positioning section of the algorithm can be improved by providing receiver velocity assistance to the algorithm. In the current implementation of the algorithm, the receiver velocity is assumed to be zero and hence accuracy of the position solution of the receiver degrades with higher receiver dynamics. Inertial instruments can therefore be integrated into the receiver to provide the necessary rate information as presented in chapter 1 of [28].

The Time-free Positioning section of the algorithm can be expanded to include other GNSS signals. The current GNSS signals in use and those expected to be in operation in the near future all use spread spectrum communications techniques. In many cases pseudorandom codes of epochs longer than that of GPS L1 C/A code are favored. The

length of a PRN epoch in GPS L1 C/A code is 1 ms which translates to about 300 km. The algorithm developed in this research concluded that for GPS L1 C/A, a maximum a priori receiver position error of 70 km is expected for the algorithm to work, with a reasonable receiver time accuracy. Table 6.2 below shows the lengths of the pseudorandom sequences of GNSS signals.

GNSS Signal	**Code Length (ms)**
GPS L1 C/A	1
GPS L2	20
GPS L5	10
Galileo E1-B	4
Galileo E1-C	100
Galileo E5a-I	20
Galileo E5a-Q	100
Galileo E5b-I	4
Galileo E5b-Q	100

Table 6.2: Code lengths in milliseconds of various GNSS signals [8]

From table 6.2, we can see that the other GNSS signals have significantly longer codes. This means that the a priori receiver initialization requirement for Time-free Positioning can be increased to an accuracy of hundreds of kilometers. Figure 6.1 further illustrates the advantage of having a PRN sequence with a longer repeat interval.

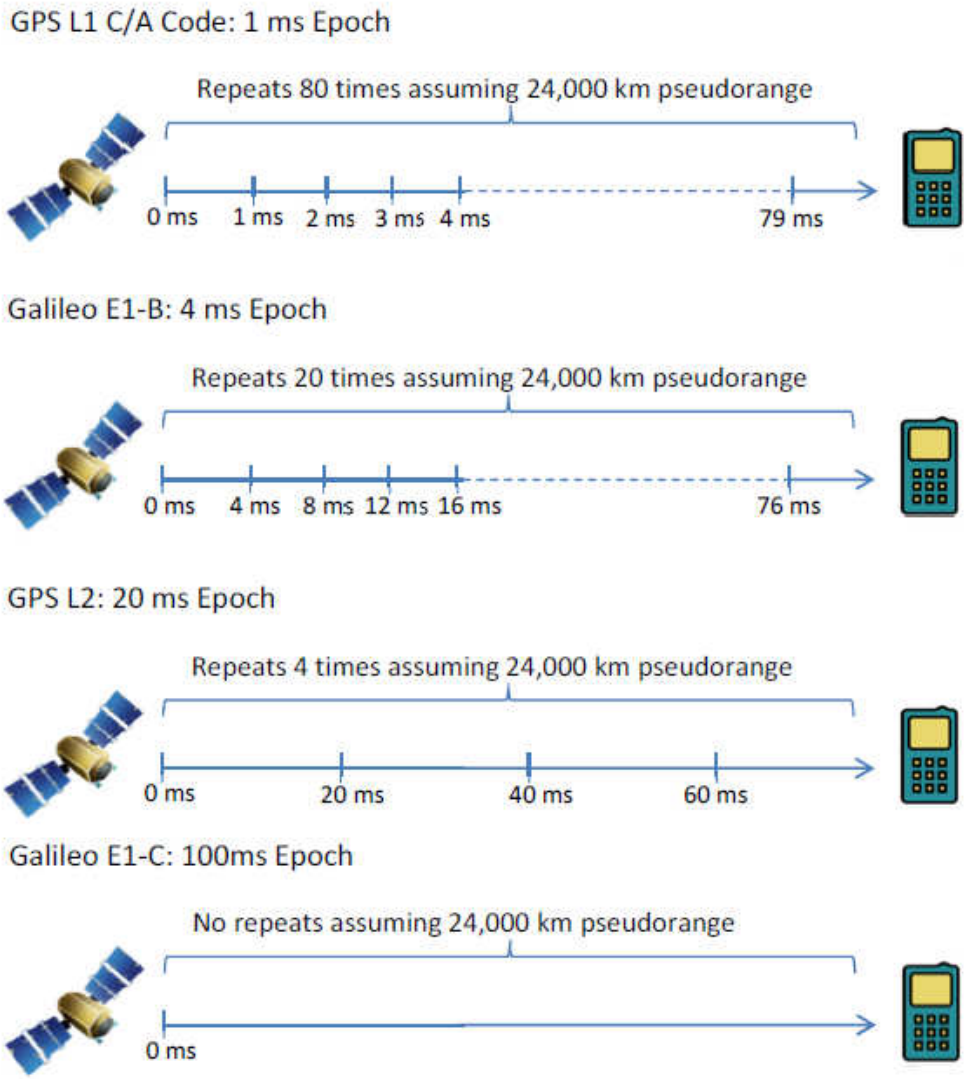

Figure 6.1: Millisecond integer ambiguity comparison of various GNSS PRN sequences

The GPS L1 C/A code with a 1 ms epoch has got an integer ambiguity of about 80 for a 24,000 km pseudorange. This ambiguity reduces to 20 with Galileo E1-B which has a 4 ms epoch code, to 4 with GPS L2 which has a 20 ms epoch code and finally to no ambiguity with Galileo E1-C which has a 100 ms epoch code. The Galileo E1-C PRN code has a length of approximately 30,000 km which is longer than the maximum pseudorange. Therefore for signals with long PRN sequence repeat intervals, the combined Doppler Time-free Positioning Algorithm can be expanded to include receivers with higher dynamics.

Further work can also be done in expanding the combined Doppler Time-free Positioning Algorithm to hybrid GPS/Galileo receivers. Work done by Constantinescu, Landry and Ilie in [29] and Eissfeller, Hein et al in [30] show that the accuracy of GNSS positioning is increased by using a GPS/Galileo receiver. A hybrid receiver might have the effect of increased accuracy for the combined Doppler Time-free Positioning Algorithm. A combined GPS/Galileo constellation would also be advantageous for satellite availability for the algorithm as it requires at least 5 satellites in view.

References

[1] Gleason S. and Gebre-Egziabher D., editors. *GNSS Applications and Methods*, pages 6–7, 55–86, 121–146, 245–252. Artech House, 2009.

[2] Wikipedia. Navigation. `http://en.wikipedia.org/wiki/Navigation` accessed 3/10/2011.

[3] Misra P. and Enge P. *Global Positioning System: Signals Measuerment and Performance*, pages 9,19–25,67–89,128,200. Ganga-Jamuna Press, 2006.

[4] USNO GPS. Current GPS Constellation. `http://tycho.usno.navy.mil/gpscurr.html`.

[5] US Government. GPS User Support. `http://www.gps.gov/support/user/` accessed 20/11/2011.

[6] US Coast Guard. GPS Status and Outage Information. `http://www.navcen.uscg.gov/?pageName=GPSSOI` accessed 20/11/2011.

[7] Federal Space Agency Information Analytical Center. GLONASS Status. `http://www.glonass-ianc.rsa.ru/en/GLONASS/`accessed 11/1/2012.

[8] European Space Agency. *European GNSS (Galileo) Open Service: Signal in Space Interface Control Document, Issue 1.1*, 2010.

[9] University of Colorado and SiGe. GN3S README. `http://ccar.colorado.edu/gnss` accessed 14/6/2011.

[10] van Diggelen F. *A-GPS: Assisted GPS, GNSS and SBAS*, pages 1,61–101. Artech House, 2009.

[11] Hartnett R. Peterson B. and Ottoman G. *GPS Receiver Structures for the Urban Canyon*, pages 1323–1332. Proceedings of the 8th International Technical Meeting of the Satellite Division of The Institute of Navigation, Palm Springs, CA, September 1995.

[12] Hill J. *The Principle of Snapthot Navigation Solution Based on Doppler Shift*, pages 3044–3051. Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation, Salt Lake City, UT, September 2001.

[13] Lannelongue S. and Pablos P. *Fast Acquisition Techniques For GPS Receivers*, pages 261–269. Proceedings of the 54th Annual Meeting of The Institute of Navigation, Denver, CO, June 1998.

[14] Navstar GPS Joint Progam Office. *Interface Specification ISP-GPS-200, Revision D,Navstar GPS Space Segment/Navigation User Interfaces*, 2006.

[15] Tholert S. Montenbruck O. Hauschild A. Langley R.B. Meurer M., Erker S., editor. *GPS L5 First Light*, pages 49–58. GPS World, June 2009.

[16] Borre K. and Akos D. *A Software-Defined GPS and Galileo Receiver: Single-Frequency Approach*, pages 1632–1637. Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation, Long Beach, CA, September 2005.

[17] Heckler G. and Garrison J. *Architecture of a Reconfigurable Software Receiver*, pages 1632–1637. Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation, Long Beach, CA, September 2004.

[18] Humphreys T.E. Powell S.P. Jr. Kintner P.M. Ledvina B.M., Psiaki M.L. *A Real-Time Software Receiver for the GPS and Galileo L1 Signals*, pages 2321–2333. Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation, Fort Worth, TX, September 2006.

[19] Sparkfun Electronics. SiGe GN3S Sampler v2. `http://www.sparkfun.com/products/8238` accessed 14/6/2011.

[20] Ettus Research LLC. USRP Family Products and Daughter Boards. `http://www.ettus.com/products` accessed 2/11/2010.

[21] Constantinescu A Dionne B, Landry R.Jr. *Implementation Methodologies of a Software Defined Navigator (SDN) Allowing the Conception of a Real Time Robust Hybrid GPS/ Galileo Receiver*. International Symposium European Radio Navigation EURAN, Munich, Germany, June 2004.

[22] IGS. International GNSS Service. `http://igscb.jpl.nasa.gov/` accessed 3/11/2010.

[23] IGS. International GNSS Service. `http://igscb.jpl.nasa.gov/igscb/data/format/sp3c.txt` accessed 3/11/2010.

[24] Schenewerk M. *A Brief Review of Basic GPS Orbital Interpolation Strategies*, pages 265–267. GPS Solution, Volume 6 Number 4, 2003.

[25] Sabbi Babu Rao. Masters Thesis: Design and Implementation of a GPS Receiver Channel and Multipath Delay Estimation Using Teager-Kaiser Operator. Indian Institute of Science.

[26] Celes Trak. NORAD Two Line Element Sets from the Center for Space Standards and Innovation. `http://www.celestrak.com/NORAD/elements/` accessed 3/11/2010.

[27] Kaplan D., editor. *Understanding GPS Principles and Applications*, pages 278–282. Artech House, 1996.

[28] Groves D. P. *Principles of GNSS, Inertial and Multisensor Integrated Navigation Systems*, page 7. Artech House, 2008.

[29] Ilie I. Constantinescu A., Landry R.Jr. *Performance Evaluation and Analysis of a Hybrid Version of a Software Defined GPS/Galileo GNSS Receiver for Dynamic Scenarios*, pages 2697–2708. Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation, Long Beach, CA, September 2004.

[30] Kaniuth R. Posfay A. Pany T. Eissfeller B., Hein G. *Implementation and Simulation of a Mass-Market GPS/Galileo Single Point Positioning Receiver*, pages 799–809. Proceedings of the 2005 National Technical Meeting of The Institute of Navigation, San Diego, CA, January 2005.