CALCULATIONS FOR POSITIONING WITH THE GLOBAL NAVIGATION SATELLITE SYSTEM

A Thesis Presented to

The Faculty of the

Fritz J. and Dolores H. Russ
College of Engineering and Technology

Ohio University

In Partial Fulfillment

Of the Requirement for the Degree

Master of Science

by

Chao-heh Cheng

August, 1998

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

Satellite-based navigation systems have the ability to provide three-dimensional position, velocity and time information to an unlimited numbers of users on a global basis.

Currently, two satellite navigation systems are available to civilian users; the Russian Global Orbiting Navigation Satellite System (GLONASS) and the United States Global Positioning System (GPS). Positioning with GLONASS and GPS is based on satellite ranging. Distances are measured between the receiver and the position of four or more satellites. If the positions of the satellites are known, the location of the receiver can be calculated using trilateration concepts. The satellites are controlled and monitored from ground stations (the Ground Segment). The ground stations monitor the satellites for health and accuracy. Maintenance commands, orbital parameters and timing correction data are uploaded to the satellites from the Ground Segment on a periodic basis.

GLONASS and GPS employ the concept of time-of-arrival (TOA) ranging to determine the user's position. This concept entails measuring the time it takes for a signal transmitted by a satellite, at a known location, to reach the user receiver. This time interval, referred to as the signal transit time, is then multiplied by the speed of light to obtain the emitter-to-receiver distance. By measuring the transit time of signals broadcast from multiple satellites at known locations, the receiver position can be determined. Unlike previous navigation systems using ground-based transmitters, satellite-based transmitters are used to cover the entire Earth.

Both GLONASS and GPS provide military as well as civilian navigation signals. The military signals have a higher level of accuracy than the civilian signals and are also less susceptible to interference and spoofing (intentional misleading information). The civilian signals are available at no cost to users worldwide. Each system is designed to use 24 satellites that circle the earth in precisely determined orbits, with an orbital period of approximately 12 hours. Each satellite broadcasts its navigation data and timing signals to report the satellite location and time to the user.

Independent of weather conditions and time of day, satellite navigation provides position determination with accuracies on the order of 20-100 meters (horizontal, 95%). High-accuracy positioning at the sub-meter level is possible through the use of differential methods. Differential satellite positioning requires the use of a reference receiver in a known location which broadcasts corrections to nearby users. The user receiver applies the corrections to remove errors that are common between the reference receiver and the user receiver.

Due to the high precision and the rapidly falling prices of GPS receivers, a multitude of opportunities for application of GPS in private, commercial and scientific fields have arisen during the past decade. GLONASS and GPS users might have to operate in an environment with limited possibilities for receiving signals from navigation satellites, e.g. in urban or mountainous areas, during aircraft maneuvers, or in the presence of interference. In such situations combined use of GLONASS and GPS navigation signals would significantly improve the quality of navigation. The combined use of GLONASS and GPS has been recognized by the International Civil Aviation Organization (ICAO) and is referred to as the Global Navigation Satellite System or GNSS.

When GLONASS and GPS are combined, the following must be taken into account:

(i) Different structures of the GLONASS and GPS navigation data;

(ii) Differences between the coordinate systems used for GLONASS and GPS;

(iii) Time scales offset between GLONASS and GPS.

Information on the processing of GPS data is widely available in the literature [1,2], but little information is available on the processing details of GLONASS. Therefore, this thesis focuses on the processing of GLONASS measurement data. Chapter 2 provides an overview of GLONASS, its satellite constellation, signal structure, error sources and a comparison with GPS. Detailed information on the GLONASS navigation data message and orbital parameters is provided in Chapter 3. GLONASS satellite position calculations are contained in Chapter 4. Next, Chapter 5 presents position calculations for GLONASS, GPS and combined GPS/GLONASS. Experimental results for static, car and aircraft flight tests are presented in Chapter 6. Chapter 7 contains conclusions and recommendations based on the results presented in this thesis.

2. GLOBAL ORBITING NAVIGATION SATELLITE SYSTEM

2.1 SATELLITE CONSTELLATION STRUCTURE

GLONASS consists of three components: the space segment, the ground segment and the user segment. The space segment consists of the actual satellites in orbit. The ground segment consists of a master control station (MCS) at Moscow and four monitor stations in St. Petersburg, Ternopol, Eniseisk and Komsomolsk-na-Amure. The user segment consists of all the military and civilian receivers.

When fully deployed, the space segment will consist of a constellation of 24 GLONASS satellites. These satellites are arranged in three orbital planes with eight satellites in each plane. Three of these 24 satellites, one in each orbital plane, are active on-orbit spares that are used to replace any malfunctioning satellite in that plane. A 21-satellite constellation provides continuous, 4-satellite visibility over 97% of the Earth's surface, whereas a 24-satellite constellation provides continuous observation of no fewer than 5 satellites simultaneously from more than 99% of the Earth's surface.

The GLONASS satellites in the constellation are referenced by their slot number and by their frequency channel. The slot number refers to the position of the satellite in orbit. Slots 1-8 are in plane I, slots 9-16 are in plane II and slots 17-24 are in plane III. If a GLONASS satellite is referred to as 01/12, it is in slot 1 and transmitting on frequency channel 12.

The GLONASS orbital planes of the satellites are inclined at a nominal 64.8 degrees with respect to the equatorial plane and are spaced by 120 degrees in longitude. The orbital radius is 25,510 km (19,100 km above the surface of the Earth). Figure 2.1 shows the GLONASS orbital configuration. This arrangement results in an orbital period of the satellites of approximately 11.25 hours. Therefore, the orbital period is 8/17 of a sidereal day such that, after eight sidereal days, the GLONASS satellites have completed exactly 17 orbital revolutions. For an observer on the earth, a particular satellite will reappear at the same place in the sky after eight sidereal days.

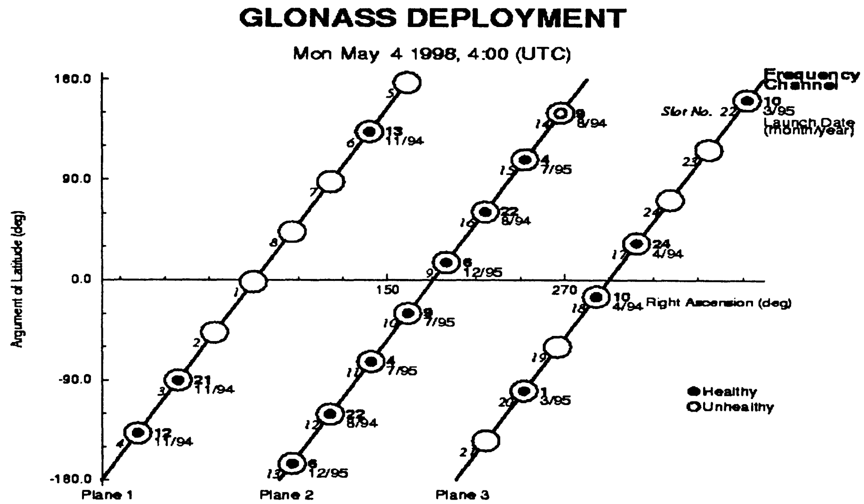


Figure 2.1 GLONASS Satellite Constellation [42]

2.2 SIGNAL STRUCTURE

The structure of the GLONASS radio signal is documented in the *Global Satellite Navigation System GLONASS, Interface Control Document*, by the Russian Institute of Space Device Engineering [17].

GLONASS uses multiple carrier frequencies to broadcast signals from the satellites, which is also referred to as *frequency division multiple access* (FDMA). Each satellite uses the same pseudorandom noise (PRN) code to produce a spread spectrum signal in space. In contrast, GPS satellites broadcast at the same carrier frequency, but each satellite uses a different PRN code, which is denoted as *code division multiple access* (CDMA). The GLONASS satellites transmit in two frequency bands; between 1602.0 and 1615.5 MHz (L1) and between 1246.0 and 1256.5 MHz (L2). The carrier frequencies on which a satellite broadcasts are defined by the following equations:

$$f_{L1} = (1602 + 0.5625k)MHz \tag{2.1}$$

$$f_{L2} = (1246 + 0.4375k)MHz \tag{2.2}$$

Where k refers to the frequency channel (1-24).

Because the upper portion of the GLONASS L1 band overlaps with one of the radio astronomy bands, several changes will be made to the GLONASS frequency plan over a 10-year period which started in 1998. The final frequency plan will have channels ranging from –7 to 6 instead of the current range from 1 to 24. Antipodal satellites will share the same frequency. This change will not affect users on the Earth's surface, since they are not able to receive signals from antipodal satellites at the same time.

Using the FDMA technique results in better interference rejection for narrow-band interference signals compared to CDMA techniques. A narrow band interference source that disrupts only one FDMA signal would disrupt all CDMA signals simultaneously. Furthermore, FDMA eliminates the need to consider the interference effect between multiple signal codes (cross-correlation). However, the use of multiple frequencies does introduce the possibility for hardware delays that are different for each channel.

Each GLONASS carrier frequency is bi-phase modulated by the modulo-2 summation of either a 511 kHz or 5.11 MHz PRN ranging code sequence and a 50 bits per second (BPS) navigation message signal. Similar to GPS satellites, GLONASS satellites transmit two PRN codes, one is a coarse acquisition (C/A) code and one is a precise (P) code. The C/A code is present on the L1 frequency only, whereas the P code is present on both the L1 and L2 frequencies.

In the GLONASS system, the C/A code is a 511-bit binary sequence that is modulated onto the carrier frequency at a chipping rate of 0.511 MHz and thus repeats every millisecond. The P code is a 5.11 million bits long binary sequence that is modulated onto the carrier frequency at a chipping rate of 5.11 MHz and thus repeats every second.

The navigation message is broadcast from a GLONASS satellite at a rate of 50 BPS. The navigation message contains ephemeris information in terms of position, velocity, and acceleration of the satellite and the offset of the satellite clock from GLONASS system time, together with almanac information for the entire satellite constellation.

2.3 SATELLITE POSITIONING ERROR SOURCES

Inaccuracies in the determination of the user position using GLONASS or GPS are caused by the environment, satellite orbit and clock errors, and by the process carried out by the user equipment to estimate the user position. Generally, the error sources for GLONASS and GPS can be divided into two categories: correctable errors and non-correctable errors. Correctable errors are errors that are correlated for two receivers that are closely separated, while non-correctable errors are uncorrelated between receivers.

Correctable Errors

Sources of correctable errors include satellite clock and ephemeris data, and ionospheric and tropospheric delays.

Clock errors and ephemeris errors originate with the Ground Segment. A clock error is usually a slowly changing clock offset that appears as a slowly changing bias on the range measurements made by a receiver. An ephemeris error is the error in the data used by a receiver to calculate the position of a satellite.

Ionospheric delay errors and tropospheric delay errors are caused by atmospheric conditions. The ionosphere is the upper portion (between 70km and 1000km above the Earth's surface) of the Earth's atmosphere and ionospheric delay is a function of the total number of electrons in the ionosphere along the signal path. Since the ionosphere has fewer electrons at night, the ionospheric delay is more pronounced during the day than during the night. The ionospheric delay depends on the location of the satellite with respect to the receiver. When viewing satellites at zenith, the delay ranges from 3m to 15m. At low satellite viewing angles (about 5 degrees), the delay ranges from 9m to 45m [4].

The troposphere is the lower part of the Earth's atmosphere. Tropospheric delay is related to humidity, temperature, and pressure along the signal path. Usually, a tropospheric delay is smaller than an ionospheric delay. These delays increase when satellite signals must travel longer distances through the troposphere, and can become quite large for low-elevation satellites. The tropospheric delay ranges from 1 to 30 meters [4].

In the GPS system, clock and ephemeris errors can be intentionally introduced through a process called Selective Availability (SA). SA is used by the United States

Department of Defense (DoD) to degrade the performance of the Standard Positioning Service (SPS) GPS signals. When active, SA typically introduces ranging errors on the order of 30 meters on each of the satellites. These errors are different for each of the satellites, and have a correlation time of approximately 2 minutes [30].

Non-Correctable Errors

Sources of non-correctable errors include thermal noise, multipath and receiver measurement errors. A typical receiver experiences high-frequency thermal noise on the order of several meters. Since the thermal noise has a high-frequency, it can be reduced through filtering. Multipath is caused by signal reflections from nearby objects. High-frequency multipath errors can be reduced by filtering, but low-frequency errors can pose a serious concern for high-precision applications of GPS or GLONASS. Typical multipath errors are on the order of a few meters, but in extreme situations, multipath errors can be on the order of tens of meters. For modern receivers, the contribution of receiver measurement errors can generally be neglected.

2.4 COMPARISON OF GLONASS AND GPS

This section provides a comparison of GPS and GLONASS. For applications where GPS and GLONASS are combined, it is important to understand the differences between the two systems. Tables 2.1 and 2.2 summarize key parameters of GPS and GLONASS [11,30].

Both GPS and GLONASS are designed to have 24 operational satellites. However, the orbital arrangement of the satellites will not be the same, as can be seen from Table 2.2. GLONASS has three orbital planes with eight satellites evenly distributed in each plane. The planes have a nominal inclination of 64.8 degrees with respect to the Earth's equatorial plane and their intersections with the equatorial plane are spaced by 120 degrees in longitude. The orbital radius is about 25,510 kilometers. This orbital radius yields an orbital period of 8/17 of a sidereal day such that; after eight sidereal days, the GLONASS satellites have completed exactly 17 orbital revolutions. A sidereal day is the time it takes for the Earth to complete a full revolution and is approximately equal to a calendar day minus four minutes. For an observer on the earth, a particular satellite will reappear at the same place in the sky after eight sidereal days. Because each orbital plane contains eight equally spaced satellites, one of the satellites will be at the same spot in the sky at the same sidereal time each day.

The GPS constellation, however, has six orbital planes with four satellites unevenly distributed in each plane. These planes are inclined by 55 degrees with respect to the Earth's equatorial plane and their intersections with the equatorial plane are separated by 60 degrees in longitude, The satellite orbits are near-circular with a radius of about 26,560 kilometers. The GPS orbital period is one half of a sidereal day. Therefore, after one sidereal day the geometric relationship between fixed locations on the earth and the satellites repeats.

Table 2.1 Comparison of GPS and GLONASS

Parameter	GLONASS	GPS	Comments
Number of satellites	24	24	
Number of orbital planes	3	6	
Satellites per orbital plane	8, evenly spaced	4, unevenly spaced	
Orbital inclination (degrees)	64.8	55	
Orbital radius (km)	25,510	26,560	
Orbital period	11 hr 15 min	11 hr 58 min	
Time reference	GLONASS time scale based on UTC(SU)	GPS time scale based on UTC (US Naval Observatory)	GLONASS leads UTC(SU) time by 3 hours GPS leads UTC (USNO) by a number of leap seconds.
Geodetic datum	PZ-90	WGS84	Coordinate systems are very similar but no accurate transformation parameters have yet been determined
C/A code (L1) code rate (MHz)	0.511	1.023	
C/A code (L1) chip length (m)	587	293	
P code (L1) code rate (MHz)	5.11	10.23	
P code (L1) chip length (m)	58.7	29.3	
Chipping rate	L1: 0.511 MHz L2: 5.11 MHz	L1: 1.023 MHz L2: 10.23 MHz	
Signal separation	Frequency Division Multiple Access (FDMA)	Code Division Multiple Access (CDMA)	GLONASS satellites transmit the same pseudo random code (PRN) at different carrier frequencies GPS satellites transmit different PRN code at the same carrier frequency

Table 2.2 Comparison of GPS and GLONASS (Cont.)

Parameter	GLONASS	GPS	Comments
Carrier frequency (MHz)	L1: 1602.0 +0.5625k L2: 1246.0 +0.4375k k=0 ...12	L1: 1575.42 L2: 1227.6	GLONASS stage 2 frequency plan (1998-2005)
Selective Availability (SA)	No SA activated	SA activated	GPS SA will be set to minimal levels within 4 – 10 years (US Presidential address 29/3/96)
Encryption (anti-spoofing)	Full civilian access to P code	P code encrypted	
Ephemeris broadcast	Ephemeris broadcast in Cartesian coordinates	Ephemeris broadcast in Keplerian elements	GPS ephemeris parameters updated every hour GONASS ephemeris parameters updated every 30 minutes

Contained in the ephemeris data, the broadcast GLONASS clock and clock frequency offsets yield the difference between the individual GLONASS satellite time and the GLONASS system time, which is related to Universal Time, Coordinated (UTC) as kept in the Soviet Union; *UTC (SU).* In contrast, GPS clock data are transmitted in terms of clock offset, clock frequency offset and clock frequency rate coefficients, and allow for the calculation of the difference between the individual GPS satellite time and the GPS system time. GPS time is related to UTC as kept by the U.S. Naval Observatory; *UTC (USNO).*

Each satellite in both systems transmits ephemeris data at a rate of 50 bits per second. For GLONASS, each satellite directly broadcasts its three-dimensional ECEF position, velocity and acceleration referenced to the S*oviet Geocentric System*

1990 (PZ90). The data is refreshed every 30 minutes. For a measurement time somewhere between these half-hour epochs, the user calculates the satellite's co-ordinates using orbital equations combined with position, velocity and acceleration data from the half-hour marks [17].

GPS has a different way of transmitting satellite orbit information. The satellite ephemeris broadcast by the satellites contain the parameters of the satellite orbit in terms of a linearly varying ellipse, plus small correction terms accounting for irregularities in the orbit. The ephemeris data are updated every hour and the user can compute ECEF coordinates of the satellite for a particular measurement time using well-known equations. The resulting ECEF coordinates are referenced to the World Geodetic System 1984 (WGS84).

3. GLONASS NAVIGATION DATA

3.1 NAVIGATION MESSAGE CONTENT AND STRUCTURE

The GLONASS navigation message contains both immediate and non-immediate information. The immediate information includes satellite time marks, onboard satellite time-scale shift with respect to GLONASS time, relative differences between carrier frequencies of the transmitted signals and their nominal values, and satellite position, velocity and acceleration data. The non-immediate information contains the GLONASS almanac. The immediate data offers accurate satellite information that is valid over a short period of time, while the almanac data is less accurate and can be used over a long period of time [17]. In this thesis, the focus is on using GLONASS ephemeris data to calculate the GLONASS satellite positions. The calculated satellite positions will be used to determine the user receiver position for static and dynamic applications.

As shown in Figure 3.1, the immediate and non-immediate information for a given satellite is transmitted within a superframe. The superframe has a duration of 2.5 minutes and consists of 5 frames. Each frame has a duration of 30 seconds and consists of 15 strings. Each string has a duration of 2 seconds. The structure of frames 1 to 4 is identical. The last two strings of the 5-th frame are not used for data transmission; they contain spare bits. The immediate information is contained in strings 1 to 4 of each frame. Strings 6 to 15 of each frame contain non-immediate information for 24 GLONASS satellites. Frames 1 to 4 contain the almanac for 20 satellites (5 satellite per frame) and frame 5 contains the almanac for 4 satellites. The almanac for one satellite occupies two strings of each frame [17].

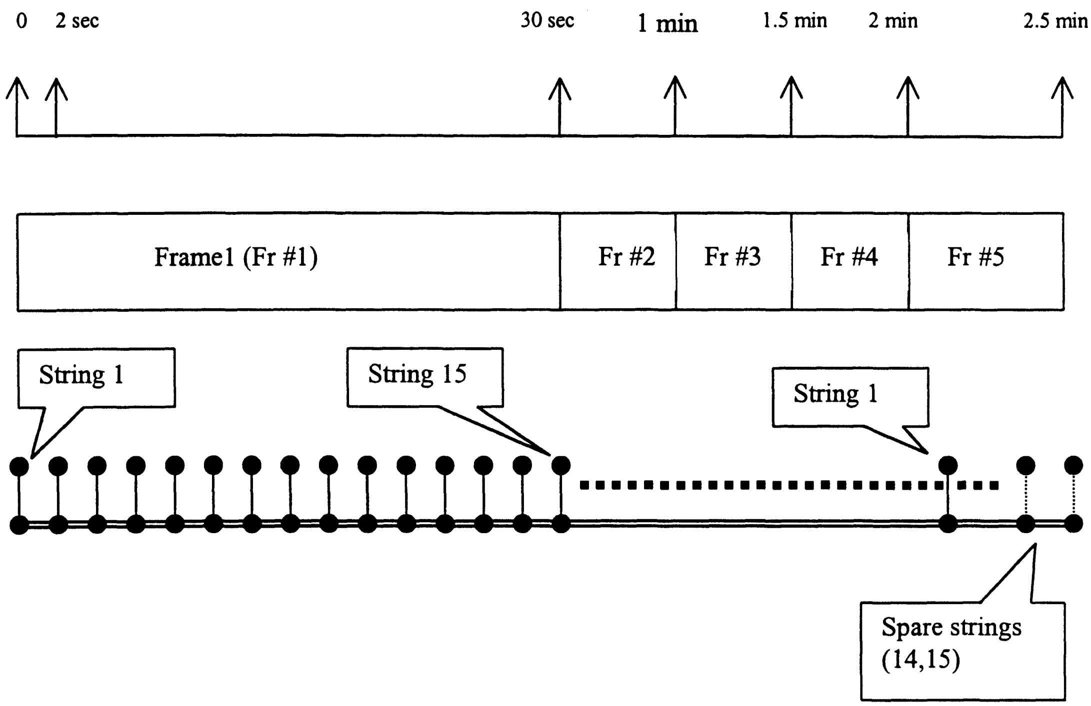


Figure 3.1 GLONASS Ephemeris data Superframe/Frame Structure

3.2 EPHEMERIS PARAMETERS

Ephemeris parameters are periodically calculated by the Ground-based Control Complex and uploaded to all GLONASS satellites twice a day. Table 3.1 summarizes the GLONASS ephemeris parameters. The meaning of these parameters is explained below.

Table 3.1 Parameters of GLONASS satellite ephemeris [17]

Word	Effective range	Units
m	1…15	dimensionless
t_k	0…23	hours
	0…59	min
	0,30	seconds
t_b	15…1425	min
$\gamma_n(t_b)$	$\pm 2^{-30}$	dimensionless
$\tau_n(t_b)$	$\pm 2^{-9}$	seconds
$X_n(t_b), Y_n(t_b), Z_n(t_b)$	$\pm 2.7\times 10^{4}$	km
$\dot{X}_n(t_b), \dot{Y}_n(t_b), \dot{Z}_n(t_b)$	± 4.3	km/s
$\ddot{X}_n(t_b), \ddot{Y}_n(t_b), \ddot{Z}_n(t_b)$	$\pm 6.2\times 10^{-9}$	km/s^2
E_n	0…31	days

- Word m is the string number within the frame.
- Word t_k is the time of the beginning of the frame within the 24-hour period, calculated in the scale of onboard satellite time.
- Word t_b is the time within the current 24-hour period according to UTC(SU) time + 03 hours 00 minutes, which includes the immediate information transmitted in the frame. The discreteness of representation of t_b is 15 minutes. The immediate information is refreshed at an update rate of 30, 45, or 60 minutes.
- Word γ_n is the carrier frequency offset of the onboard standard at time t_b

$$\gamma_n(t_b) = \frac{f_n(t_b) - f_{Hn}}{f_{Hn}} \tag{3.1}$$

where $f_n(t_b)$ is the predicted carrier frequency value of the n-th satellite, corrected for gravitational and relativistic effects at the instant t_b.

f_{Hn} is the nominal value of carrier frequency of the n-th satellite.

- Word $\tau_n(t_b)$ is the time shift between the n-th satellite time t_n and the GLONASS system time t_s at the instant of t_b:

$$\tau_n(t_b) = t_s(t_b) - t_n(t_b) \tag{3.2}$$

- Words $X_n(t_b), Y_n(t_b), Z_n(t_b)$ are ECEF (PZ-90) coordinates of the n-th satellite at the instant of t_b.
- Words $\dot{X}_n(t_b), \dot{Y}_n(t_b), \dot{Z}_n(t_b)$ are components of the velocity vector of the n-th satellite at the instant of t_b.
- Words $\ddot{X}_n(t_b), \ddot{Y}_n(t_b), \ddot{Z}_n(t_b)$ are components of the acceleration vector of the n-th satellite at the instant of t_b.
- Word E_n is the age of the immediate information of the n-th satellite at the instant of t_b.

4. GLONASS SATELLITE POSITION SOLUTION

4.1 GLONASS SYSTEM TIME

The GLONASS satellites use Cesium oscillators for their time references, which provide daily frequency stability of better than 5×10^{-13}. The accuracy of satellite time synchronization relative to GLONASS System Time is approximately 20 nanoseconds (one sigma).

GLONASS System Time (GLONASST) is based on Hydrogen clocks and has a stability better than 5×10^{-14}. The GLONASS time shift relative to Soviet Union Universal Time, Coordinated (UTC (SU)) should be less than 1 millisecond. The accuracy of the time shift should be less than 1 microsecond [3].

The fundamental time scale for all time keeping is International Atomic Time (TAI). It results from analyses by the Bureau International de l'Heure (BIH) in Paris of data from atomic standards of many countries. The fundamental unit of TAI is the International System of Units (SI) second, defined as the duration of 9,192,631,770 periods of the radiation corresponding to the transition between two hyperfine levels of the ground state of the Cesium 133 atom.

Because TAI is a continuous time scale, it has one fundamental problem in practical use: the earth's rotation with respect to the sun is slowing down by a variable amount which averages, at present about 1 second per two years. Thus TAI would eventually become inconveniently out of synchronization with the solar day. This problem has been overcome by introducing UTC, which runs at the same rate as TAI but is incremented by 1-second jumps ("leap second") when necessary, normally at the end of June or December of each year. It typically takes one, one and a half or even two years to gain a leap second, depending on the variable deceleration of the earth's rotation

UTC (SU) is maintained by the Main Meteorological Center of Russian Time and Frequency Service (VNIIFTRI) at Mendeleevo, Moscow region. When UTC (SU) is incremented by leap seconds, GLONASST is incremented too. Therefore, there is no integer-second difference between GLONASST and UTC (SU). However there is a constant offset of three hours between GLONASST and UTC (SU) due to GLONASS monitoring-specific features. That is:

$$\text{GLONASST} = \text{UTC (SU)} + 03\text{h}.00\text{min}. \quad (4.1)$$

GPS System Time (GPST) is not incremented by leap seconds; For 1998, there is a 12-second difference between GPST and UTC (USNO). The relationship between GLONASS time and GPS time is:

$$\text{GLONASST} = \text{GPST} + 03\text{h}.00\text{min} - \text{leap second} \quad (4.2)$$

As of 1 July 1997,

TAI is ahead of UTC (USNO) by 31 seconds.

TAI is ahead of GPS by 19 seconds.

GPS is ahead of UTC (USNO) by 12 seconds.

4.2 GLONASS COORDINATE SYSTEM

GLONASS satellites broadcast their positions in PZ-90 coordinates as a part of their navigation messages. The navigation message is broadcast by each satellite and usually refreshed every half-hour. Section 4.3 and Appendix A contain the method and equations to calculate the satellite positions at any instant of time.

The PZ-90 geocentric coordinate frame, specified as an abstract mathematical entity, is just one element of an overall reference system to represent the earth from geometric, gravitational, and geodetic standpoints. Like its counterpart, WGS84 for GPS, PZ-90 is a self-contained and self-consistent system within which to define a 3-D position. A comprehensive description of WGS84 is available from National Imagery and Mapping Agency (NIMA) reports and the description of PZ-90 comes mainly from the GLONASS Interface Control Document [17]. The values of the defining parameters for the PZ-90 gravity model and its ellipsoid are slightly different from those of WGS84. These differences, however, are easily accommodated. A more difficult problem arises from the differences in the ECEF coordinate frames.

The ECEF coordinate frames of WGS84 and PZ-90 differ in their formal definitions. While each locates the origin at the center of mass of the earth, the directions of the z-axes are different: WGS84 defines it as passing through the instantaneous pole of 1984; PZ-90 adopts instead the average position of the pole between the years 1900 and

1905. This description, however, is not adequate as a basis for determining a transformation between GPS and GLONASS. Actually, even if the formal definitions were identical, it would not have ensured that the coordinates of a point as determined by measurements from the two systems would be identical. The coordinate frame for each system is realized (or, implemented) by adopting the coordinates of a set of stations. A consistent set of such coordinates defines implicitly the ECEF coordinate frame (i.e., an origin, a set of directions for the Cartesian axes, and a scale factor). Therefore, even if GPS and GLONASS had adopted the same definition for the reference coordinate frame, the independent implementation of each system would have kept the two from being identical.

A transformation between PZ-90 and WGS84, the coordinate frames used by GLONASS and GPS, respectively, has been studied and implemented in combining measurements from the two systems [29]. From the research results of Lincoln Laboratory, the difference between the two coordinate frames is that the zero meridian (X-axis) of PZ-90 is East of that for WGS84. A small clockwise rotation of 0.4 seconds of arc of the Z-axis of PZ-90 brings the two coordinate frames substantially into coincidence. The residuals are reduced further, though only slightly, by a 2.5 m displacement of the origin along the Y-axis. As shown in Figure 4.1, the estimated transformation between PZ-90 (u, v, w) and WGS84 (x, y, z) is:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 2.5m \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & -1.9x10^{-6} & 0 \\ 1.9x10^{-6} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{4.3}$$

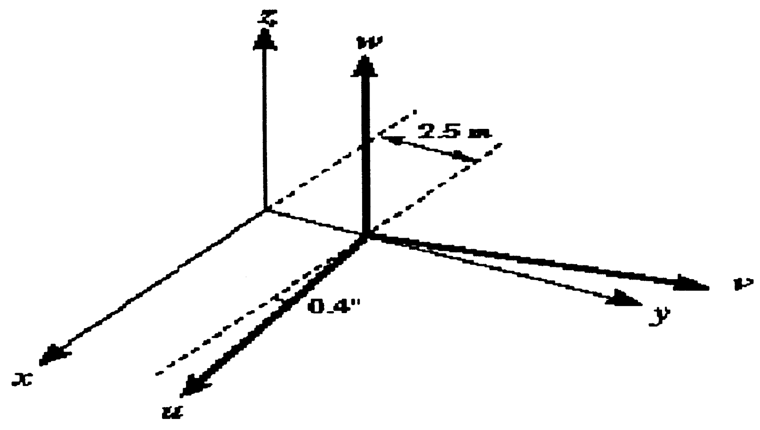


Figure 4.1 PZ-90/WGS84 Transform [29]

4.3 SATELLITE POSITION CALCULATION BY RUNGE-KUTTA METHOD

To determine the GLONASS satellite position at a given time, it is required to solve for six orbital differential equations that are published in the GLONASS ICD [17]. These equations are given below.

$$dx/dt = V_x \tag{4.4}$$

$$dy/dt = V_y \tag{4.5}$$

$$dz/dt = V_z \tag{4.6}$$

$$dV_x / dt = -\frac{\mu}{r^3}x + \frac{3}{2}C_{20}\frac{\mu(a_e^2)}{r^5}x\left[1 - \frac{5z^2}{r^2}\right] + \omega_3^2 x + 2\omega_3 V_y + \ddot{x} \tag{4.7}$$

$$dV_y / dt = -\frac{\mu}{r^3}y + \frac{3}{2}C_{20}\frac{\mu(a_e^2)}{r^5}y\left[1 - \frac{5z^2}{r^2}\right] + \omega_3^2 y - 2\omega_3 V_x + \ddot{y} \tag{4.8}$$

$$dV_z / dt = -\frac{\mu}{r^3} z + \frac{3}{2} C_{20} \frac{\mu(a_e^2)}{r^5} z \left[3 - \frac{5z^2}{r^2}\right] + \ddot{z} \tag{4.9}$$

where

$$r = \sqrt{x^2 + y^2 + z^2} \tag{4.10}$$

$\mu = 398600.44 \mathrm{km}^3/\mathrm{s}^2$	Earth's universal gravitational parameter
$a_e = 6378.136\mathrm{km}$	Earth equatorial radius
$C_{20} = -1082.63 \times 10^{-6}$	Zonal geopotential coefficient of spherical harmonic expansion
$\omega_3 = 0.7292115 \times 10^{-4} c^{-1}$	Earth's rotation rate

The values $\mu, a_e, C_{20}, \omega_3$ are given as defined in PZ90.

Figure 4.2 shows the relationship between GLONASS time and satellite reference time. The arrow signs are used to indicate the new ephemeris data broadcast times (e.g. 75600 seconds, 77400 seconds, 79200 seconds, etc.). The broadcast ephemeris data contains the exact position, velocity and acceleration of the satellite valid at a time 15 minutes after the new ephemeris data broadcast time (e.g. reference time t_b=76500 seconds, 78300 seconds, 80100 seconds, etc.). Note that the units of t_b are converted from minutes to seconds. For example, in Figure 4.3, the GLONASS satellite broadcasts a set of ephemeris data at GLONASS time 79200 seconds. This set of data contains the predicted satellite position, velocity and acceleration at time 80100 seconds and can be used as the initial conditions to calculate the satellite positions. By taking these predicted parameters as the initial conditions for a Runge-Kutta Integration Method (RKM), the satellite positions between 79200 seconds and 80100 seconds can be determined by interpolating backward from the initial conditions at 80100 seconds. The satellite positions between 80100 seconds to 81000 seconds can be interpolated forward from the initial conditions at time 80100 seconds. At GLONASS time 81000 seconds, the satellite broadcasts a new set of predicted ephemeris data for time 81900 seconds and the satellite positions between 81000 to 82800 seconds can be determined with the same procedure as explained above. Appendix A contains a detailed explanation of the RKM for the GLONASS satellite position calculation.

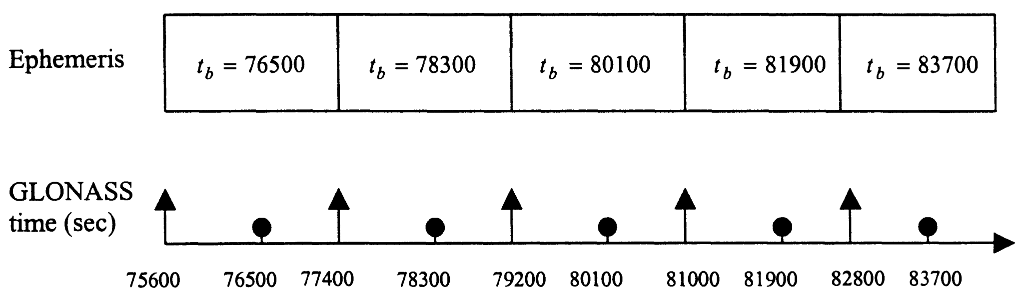


Figure 4.2 The Relationship between GLONASS Time and Satellite Reference Time

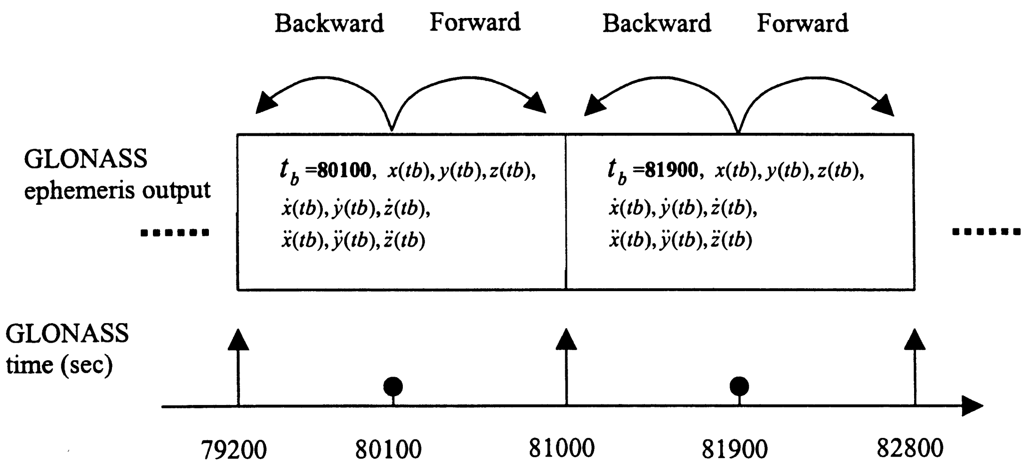


Figure 4.3 GLONASS Satellite Position Determination

4.4 GLONASS SATELLITE POSITION CALCULATION VERIFICATION

In section 4.3 and appendix A, it is shown how the GLONASS satellite position can be found at any instant of time by integrating the broadcast ephemeris data using the Runge-Kutta Method. Three different verification methods were used to determine the correctness of the satellite position calculations. The first method uses satellite azimuth and elevation angles as broadcast by a commercial GPS/GLONASS receiver. The second method uses three broadcast GLONASS satellite positions and verifies the consistency of the three positions. The third method uses the GLONASS satellite positions in combination with the pseudorange measurements to calculate the receiver position. The latter method is presented in Chapter 6.

Satellite Elevation and Azimuth Angles Verification

To verify the calculated GLONASS satellite positions, a commercial GPS/GLONASS receiver was used, the GG24 from Ashtech, Inc. This receiver does not output satellite position data, but it does output satellite azimuth and elevation angles. As shown in Figure 4.4, the azimuth angle is the clockwise angle between true North and the line of sight between the receiver antenna and a satellite. The elevation angle is the counterclockwise angle between the horizontal and the line of sight between the receiver antenna and a satellite. The calculated results and those of the receiver are shown in Figure 4.5 for a known location (the roof of Stocker Center at Ohio University). From Figure 4.5, it can be concluded that the calculated angles agree to well within one degree with the receiver-derived values.

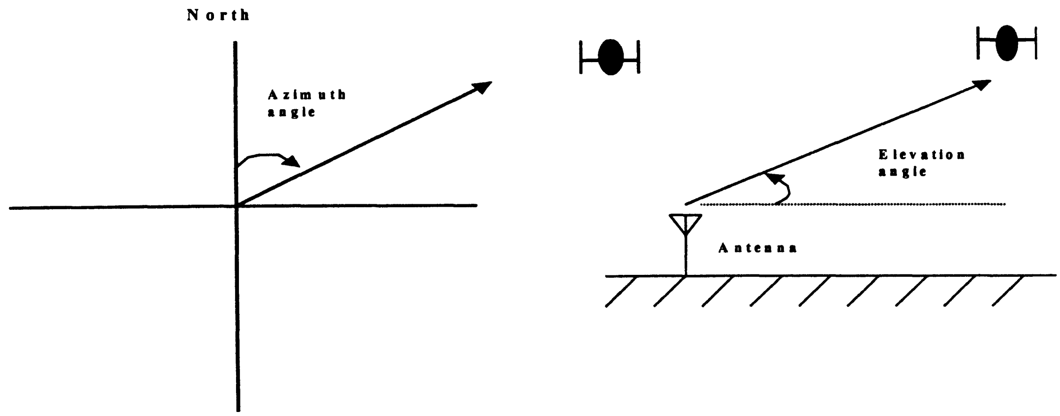
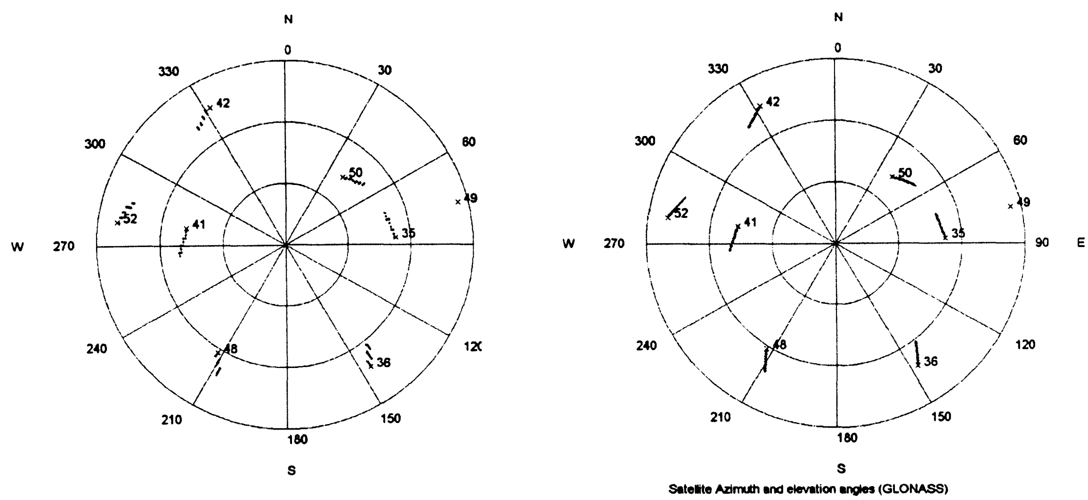


Figure 4.4 Satellite Azimuth and Elevation Angles



From GG24 receiver output

Calculated from SV positions and user position

Elevation is 90 degrees at center
'X' denotes the first point in time

Figure 4.5 Comparison of Azimuth and Elevation Angles between the Calculated and Receiver-Derived Values

Satellite Position Consistency Verification

The accuracy of the calculated orbits depends on the correctness of satellite ephemeris data and the accuracy of the Runge-Kutta method. Increasing the interpolation step size of the Runge-Kutta method reduces the time needed for the calculation but increases the satellite-position calculation error. Thus, the step size of Runge-Kutta method must be chosen carefully.

As stated in Section 4.3, a GLONASS satellite periodically refreshes its position, velocity, acceleration and reference time t_b at 30-minute intervals. To investigate the consistency of the calculated satellite orbits, the differences between some broadcast and calculated satellite position are compared. In Figure 4.6, two broadcast satellite positions are defined as reference points t_{b_1} and t_{b_2} respectively. The two points are 30 minutes apart and their ephemeris data are first broadcast at points b_{c_1} and b_{c_2}, which are 15 minutes earlier than the reference times at t_{b_1} and t_{b_2}. Point A in Figure 4.6 is integrated using the RKM forward for 15 minutes from the reference point t_{b_1}. Point B is integrated backward for 15 minutes from point t_{b_2}. Ideally points A and B would be identical, however, based on RKM step size errors and errors from the broadcast ephemeris data, the difference between points A and B is expected to be at the meter-level. Table 4.1 shows the results of the calculations; the three-dimensional position difference for this particular case was 2.03 meters. The RKM step size was varied until there was no change in the calculated positions. The final step size was 0.01. Therefore, it can be concluded that the position difference is due to ephemeris errors. Two additional hours of data were collected and the position differences are shown in Table 4.2. The 3-D satellite position differences varied from 1.47 m to 2.26 m.

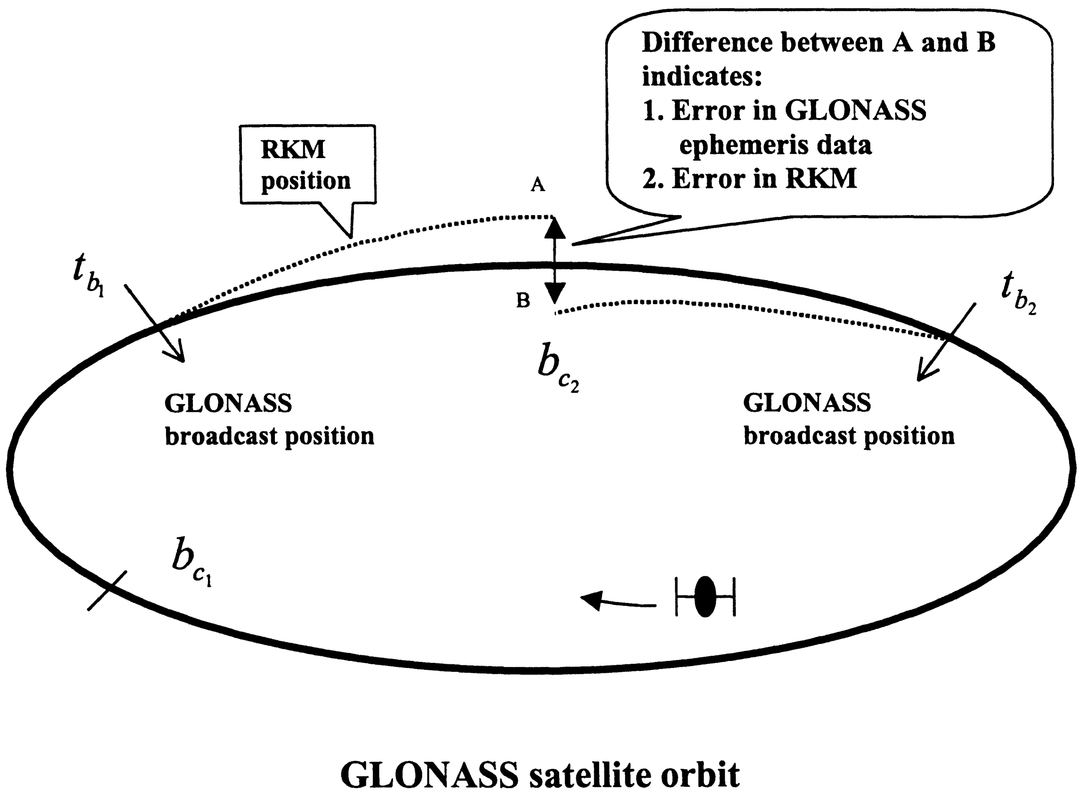


Figure 4.6 GLONASS Satellite Position Consistency Verification

Table 4.1 Satellite Position Difference between Forward and Backward Integration

t_{b_1} =80100 (sec)			t_{b_2} =81900 (sec)		
Forward t_{b_1} by 900 seconds : t=81000			Backward t_{b_2} by 900 seconds: t=81000		
GLONASS SV position (Point A)			GLONASS SV position (point B)		
X1	Y1	Z1	X2	Y2	Z2
-1.410615 899 x 10e7 (m)	-1.863421 470 x 10e7 (m)	1.017923 277 x 10e7 (m)	-1.410615 873 x 10e7 (m)	-1.863421 285 x 10e7 (m)	1.017923 358 x 10e7 (m)
dx = abs(X1-X2) = 0.264 (m)		dy = abs(Y1-Y2) = 1.848 (m)		dz = abs(Z1-Z2) = 0.810 (m)	
3-D position difference = $\sqrt{dx^2 + dy^2 + dz^2} = 2.03$ (m)					

Table 4.2 Satellite Position Differences for a Two-Hour Time Period

Satellite broadcast positions

GLN Time (seconds)	X (km)	Y (km)	Z (km)
83700	-17475.962402	-7402.074218	17035.398925
85500	-12804.069335	-7712.556152	20655.490722
900	-7548.042480	-8865.418457	22676.602050
2700	-2250.672363	-10872.844726	22941.289062
4500	2564.484863	-13554.272949	21428.229492

RKM calculated positions

GLN Time (seconds)	X (km)	Y (km)	Z (km)
83700	-17475.963856	-7402.075387	17035.400202
85500	-12804.069808	-7712.557182	20655.492524
900	-7548.044117	-8865.418970	22676.601597
2700	-2250.674156	-10872.844490	22941.289447
4500	2564.483955	-13554.272941	21428.230647

GLN Time (seconds)	Position difference (m)
83700	2.26
85500	2.12
900	1.77
2700	1.84
4500	1.46

Satellite position differences between broadcast positions and calculated positions (GLN=GLONASS)

5. CALCULATION OF THE USER POSITION

5.1 LEAST SQUARES METHOD FOR POSITION CALCULATION

There are several techniques available that could be used to calculate the user position, but the Least Squares Method (LSM) is usually adopted. Least squares works by minimizing a specific quadratic form, essentially by making the sum of the squares of the weighted residuals as small as possible. The linearized equation that relates GLONASS or GPS pseudorange measurements to the unknown user position and clock offset is:

$$\vec{v} = H\vec{x} \tag{5.1}$$

where

$\vec{v}$: vector of pseudorange measurement residuals (the vector offset of the measured and calculated pseudorange measurements).

H: design matrix containing the linear relationships between the pseudorange measurements and the user position and clock offset.

$\vec{x}$: user position and clock offset correction vector.

To estimate the receiver position and clock offset with respect to GPS or GLONASS system time, the user state vector is given by

$$\vec{x} = [\Delta x \;\; \Delta y \;\; \Delta z \;\; \Delta t_r \;] \tag{5.2}$$

where

Δx: Position deviation in the X direction

Δy: Position deviation in the Y direction

Δz: Position deviation in the Z direction

Δt_r: Receiver clock offset with respect to GPS or GLONASS system time.

The design matrix, H is given by:

$$H = \begin{bmatrix} \alpha_x^{sv1} & \alpha_y^{sv1} & \alpha_z^{sv1} & 1 \\ \alpha_x^{sv2} & \alpha_y^{sv2} & \alpha_z^{sv2} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_x^{svn} & \alpha_y^{svn} & \alpha_z^{svn} & 1 \end{bmatrix} \tag{5.3}$$

The term $\alpha_{x,y,z}^{svN}$ represents the direction cosines from the receiver to satellite N in the X, Y and Z directions. The pseudorange residual vector is given by:

$$\vec{v} = \begin{bmatrix} PR_{c1} - PR_{m1} \\ PR_{c2} - PR_{m2} \\ \vdots \\ PR_{cn} - PR_{mn} \end{bmatrix} \tag{5.4}$$

Where PR_{cn} is the n^{th} calculated pseudorange measurement based on the satellite position and the estimated receiver position, PR_{mn} is the n^{th} measured pseudorange.

The solution for minimizing the residual of the user position is given by [2]:

$$\vec{x} = (H^T H)^{-1} H^T \vec{v} \tag{5.5}$$

By using the Newton-Raphson method [12], the new residual vector x is iteratively generated and added to the estimated position and clock vector. The updated value will be used as a new estimate for the next iteration. The iteration will continue until the norm of the residual vector x converges to a desired value. For this thesis, the convergence value used was 0.01m. Appendix B contains an example MATLAB algorithm for the Newton-Raphson method. If the measurements are not of equal quality, a weighted solution can be implemented. This is discussed in the next section. To relate pseudorange errors to position and clock errors, several geometry factors are introduced which are also referred to as dilution of precision (DOP) parameters. The DOP parameters are obtained by taking the covariance of both sides of Equation (5.5)

$$COV(\vec{x}) = (H^T H)^{-1} COV(\vec{v}) \tag{5.6}$$

Next, $(H^T H)^{-1}$ is written as a full matrix:

$$(H^T H)^{-1} = \begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} \\ D_{21} & D_{22} & D_{23} & D_{24} \\ D_{31} & D_{32} & D_{33} & D_{34} \\ D_{41} & D_{42} & D_{43} & D_{44} \end{bmatrix} \tag{5.7}$$

The DOP parameters are expressed in term of the elements of $(H^T H)^{-1}$ as follows:

$$PDOP = \sqrt{D_{11} + D_{22} + D_{33}} \tag{5.8}$$
$$VDOP = \sqrt{D_{33}} \tag{5.9}$$
$$HDOP = \sqrt{D_{11} + D_{22}} \tag{5.10}$$

where P stands for position, V for vertical and H for horizontal.

5.2 WEIGHTED LEAST SQUARES METHOD (WLSM)

In order to improve the user position accuracy, if the measurements are not of equal quality, a weight matrix, W, is introduced into equation (5.5).

$$\vec{x} = (H^T \; \mathrm{W} \; H)^{-1} \; H^T \; \mathrm{W} \; \vec{v} \tag{5.11}$$

Since the low-elevation angle satellites usually contain higher noise levels because the signals travel through more atmosphere than higher elevation satellites, the lower elevation satellites should be weighted less than the higher elevation satellites. In addition, GLONASS and GPS satellites measurements also exhibit different error sources; specifically, the GPS measurements are degraded by SA. By properly estimating the noise levels for the different satellites (GPS/GLONASS), a more accurate user position can be obtained than in the case of an equally weighted solution.

Assuming that the pseudorange measurements are independent and have a Gaussian distribution, the weight matrix for N GPS and R GLONASS satellites is given by

$$W = \begin{bmatrix} 1/\sigma^2_{gps1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/\sigma^2_{gps2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & : & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\sigma^2_{gpsN} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/\sigma^2_{\mathrm{gln1}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & : & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/\sigma^2_{\mathrm{glnR}} \end{bmatrix} \tag{5.12}$$

where

σ_{gpsn} is the standard deviation for the n^{th} GPS satellite and σ_{glnr} is the standard deviation for the r^{th} GLONASS satellite.

5.3 COMBINED GPS/GLONASS POSITION SOLUTION

Both GLONASS and GPS have their own time reference and coordinate system. Before the measurements from the two systems can be combined, a connection must be established between the two time scales and the coordinate frames. The time scale issue can be resolved without an external means by "sacrificing" one measurement to estimate the instantaneous bias between the two time scales. GLONASS and GPS employ different geocentric Cartesian coordinate frames to express the positions of satellites and, therefore, the position of user. As documented in Section 4.2, the difference between the coordinates of points on earth in WGS84 and PZ-90 is less than 15 m. The two coordinate frames are brought substantially into coincidence by a small rotation (0.4") of the z-axis of either system and a 2.5-meter translation along the y-axis.

The procedure of combining GLONASS and GPS to calculate the user position consists of the following six steps. These steps are necessary for the processing of data from the Ashtech GG24 GPS/GLONASS receiver.

(I) The GG24 provides only GPS time for the measurements. Therefore, the first step is to transfer GPS time to GLONASS time see Equation (4.2) for the GLONASS satellite position calculation, i.e. GLONASST=GPST+3 hours-12secs. (Leap second = 12 seconds after 1 July 1997).

(II) GLONASS satellite positions are calculated using the received GLONASS ephemeris data, the GLONASS time of the measurement and the Runge-Kutta Method, as detailed in Section 4.3.

(III) Convert the GLONASS satellite positions from the PZ-90 coordinate system to the WGS84 coordinate system [29]. The approximate transformation between PZ-90 and WGS 84 is given by Equation (4.3).

(IV) GPS satellite positions are calculated in accordance with the GPS Interface Control Document [1]. The details on these calculations can be found in the dissertation by Diggle [40].

(V) Combine GLONASS and GPS measurements for the user position calculation. Since the receiver clock offset for GLONASS and GPS is different, measurements from at least five satellites are required to calculate the user position. The user state vector is given by

$$\vec{x} = \begin{bmatrix} \Delta x & \Delta y & \Delta z & \Delta t_{gps} & \Delta t_{\mathrm{gln}} \end{bmatrix} \tag{5.13}$$

where

Δx: Position deviation in the X direction

Δy: Position deviation in the Y direction

Δz: Position deviation in the Z direction

Δt_{gps} : Receiver clock offset with respect to GPS time

$\Delta t_{g\ln}$: Receiver clock offset with respect to GLONASS time

The design matrix, H is given by

$$H = \begin{bmatrix} \alpha_x^{gps1} & \alpha_y^{gps1} & \alpha_z^{gps1} & 1 & 0 \\ \alpha_x^{gps2} & \alpha_y^{gps2} & \alpha_z^{gps2} & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_x^{gpsN} & \alpha_y^{gpsN} & \alpha_z^{gpsN} & 1 & 0 \\ \alpha_x^{\mathrm{gln1}} & \alpha_y^{\mathrm{gln1}} & \alpha_z^{\mathrm{gln1}} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_x^{\mathrm{glnR}} & \alpha_y^{\mathrm{glnR}} & \alpha_z^{\mathrm{glnR}} & 0 & 1 \end{bmatrix} \tag{5.14}$$

The elements of H represent the direction cosines from the receiver to the n^{th} GPS satellite ($1 \le n \le N$) or r^{th} GLONASS satellite ($1 \le r \le R$) in the X, Y and Z directions, see also Section 5.4. The pseudorange residual vector, $\vec{v}$, is given by:

$$\vec{v} = \begin{bmatrix} PR_{gpsc\,1} - PR_{gpsm\,1} \\ PR_{gpsc\,2} - PR_{gpsm\,2} \\ \vdots \\ PR_{gpscN} - PR_{gpsmN} \\ PR_{\mathrm{glnc1}} - PR_{\mathrm{glnm1}} \\ \vdots \\ PR_{\mathrm{glncR}} - PR_{\mathrm{glnmR}} \end{bmatrix} \tag{5.15}$$

Where PR_{gpscn} and PR_{gpsmn} are the calculated and measured pseudorange measurements for the n^{th} GPS satellite. PR_{glncr} and PR_{glnmr} are the calculated and measured pseudorange measurements for the r^{th} GLONASS satellite.

(VI) Iterate the solution until the state residual is less than the convergence threshold, as described in Section 5.1 and Appendis B.

The next chapter provides static and dynamic positioning results for GPS, GLONASS and combined GPS/GLONASS.

6. APPLICATIONS AND RESULTS

One major advantage of a combined GLONASS and GPS system is the availability of more satellites compared to using either GPS or GLONASS by itself. The resulting redundancy in pseudorange measurements allows for real-time detection and identification of faulty signals by *receiver autonomous integrity monitoring* (RAIM). Also, GLONASS or GPS alone may not provide enough coverage for a position solution when vehicles travel under severe shadowing conditions, such as those present in mountains and urban areas. Section 6.1 compares the static positioning accuracies of GPS, GLONASS and combined GPS/GLONASS. In Section 6.2, the comparison focuses on vehicle navigation and Section 6.3 provides flight test results.

6.1 STATIC GPS, GLONASS AND COMBINED GPS/GLONASS POSITIONING

To examine the static accuracy of the calculated user position, the GG24 receiver antenna was placed at a known location on the roof of the Electrical Engineering building (Stocker Center). Figure 6.1 shows the equipment block diagram. All data collection and processing software was written in the C programming language and implemented on a *QNX™* operating system [41]. The GG24 receiver collected GLONASS and GPS data for twenty minutes at a rate of once per second. Data were collected for a total of 6 GPS and 6 GLONASS satellites. The collected data were used to calculate user positions with and without weighting of the measurements. Since the antenna position is known, the positioning errors can be calculated. Figure 6.2 shows

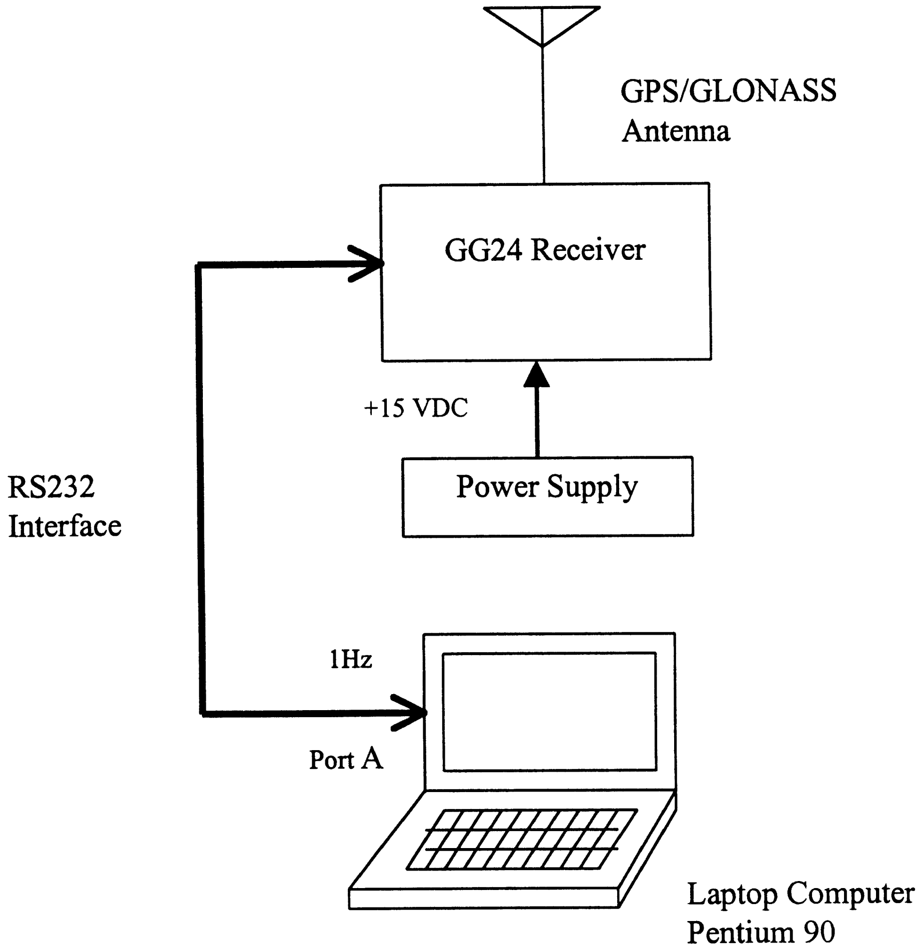


Figure 6.1 System Setup for GPS/GLONASS Data Collection

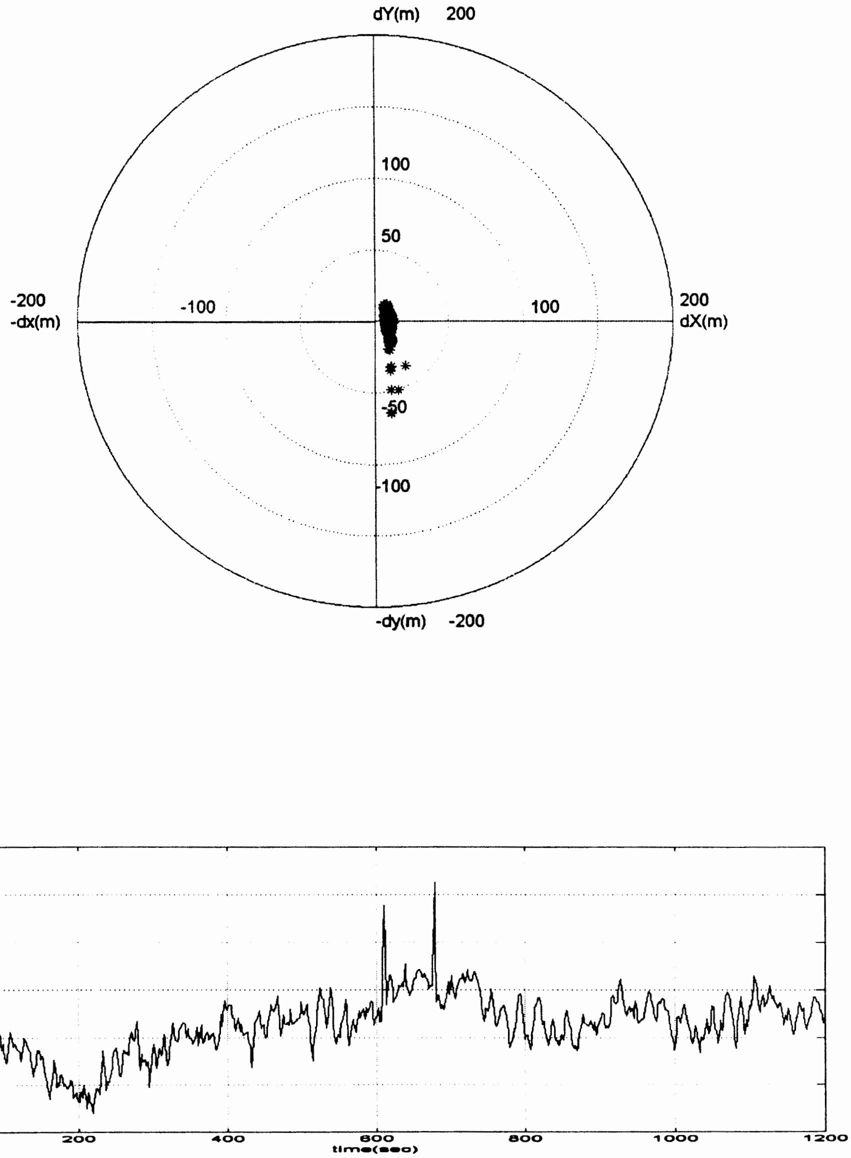


Figure 6.2 GLONASS Positioning Errors (Equal Weights)

the GLONASS position errors without weighting. The GPS position errors are shown in Figure 6.3. Figure 6.4 shows the combined GPS/GLONASS positioning errors. As expected, the GPS positioning errors are larger than those for GLONASS due to Selective Availability. Table 6.1 compares the standard deviations of the 3-D positioning errors for the above three cases.

Table 6.1 Static Positioning Errors (Equal Weights)

3-D Position Error	GLONASS	GPS	GPS+GLONASS
Standard Deviation(m)	8.88	49.51	19.06

Figures 6.5 through 6.7 compare the non-weighted (Equal Weights) and weighted positioning errors. The weighted result uses a weight matrix described in Section 5.2 with the pseudoranges weighted according to the inverse of their estimated variances. The estimate variances were obtained from the post-processing of the GG-24 receiver measurement data. The weighted solutions improve the positioning accuracy somewhat.

Table 6.2 Comparison of Static Positioning Errors (Weighted and Non-Weighted)

3-D Position Error	Non-Weighted			Weighted		
	GLONASS	GPS	GPS+ GLONASS	GLONASS	GPS	GPS+ GLONASS
Standard Deviation (m)	8.88	49.51	19.06	8.62	47.15	17.10

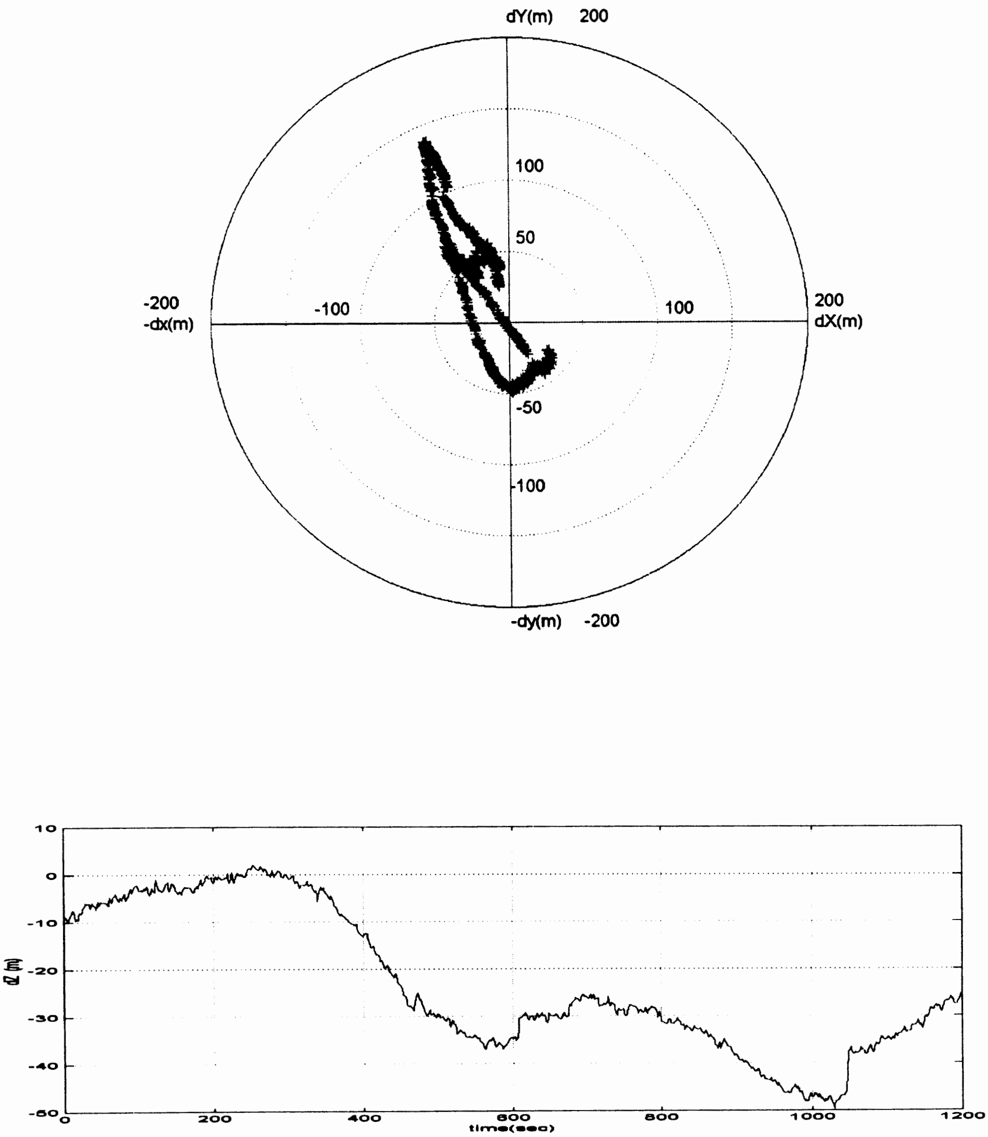


Figure 6.3 GPS Positioning Errors (Equal Weights)

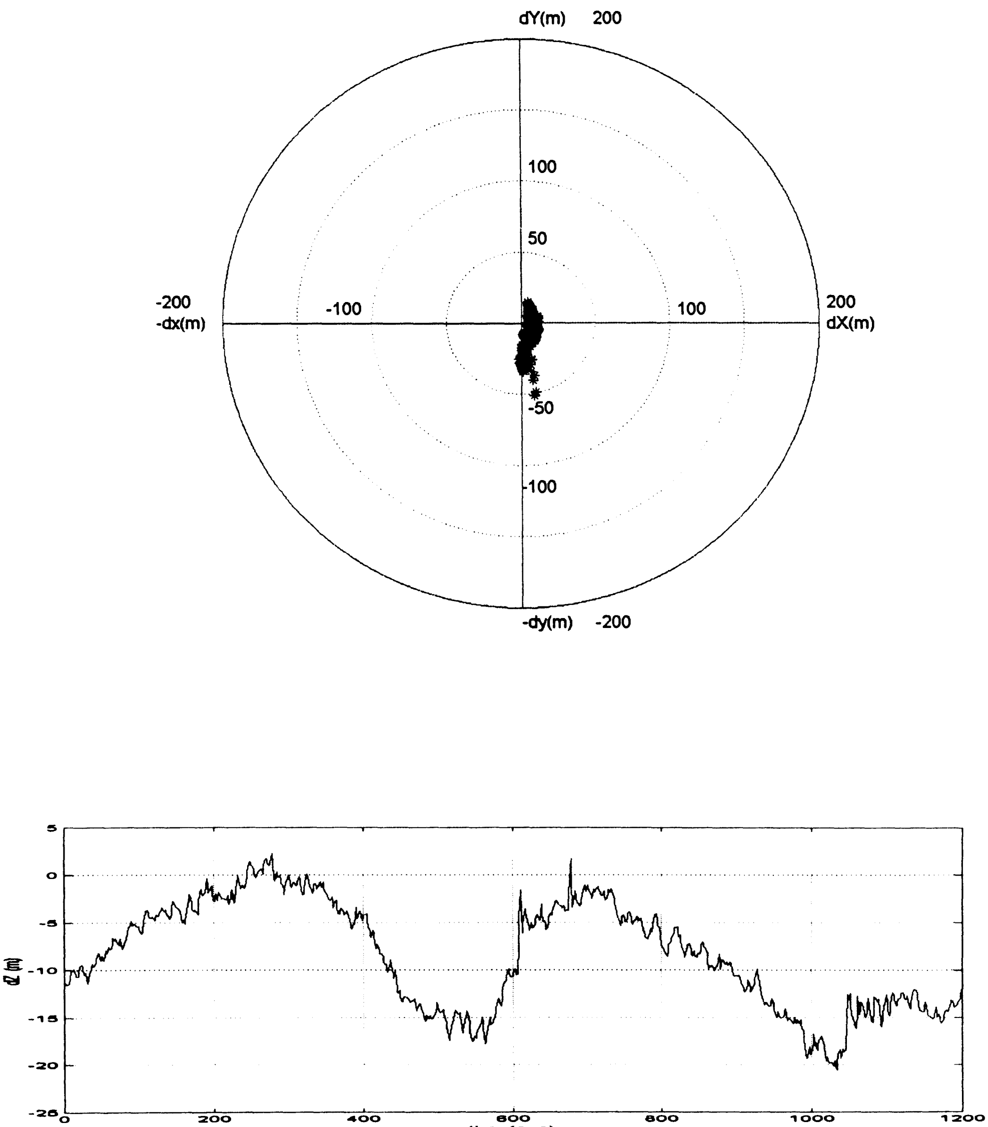


Figure 6.4 Combined GPS/GLONASS Positioning Errors (Equal Weights)

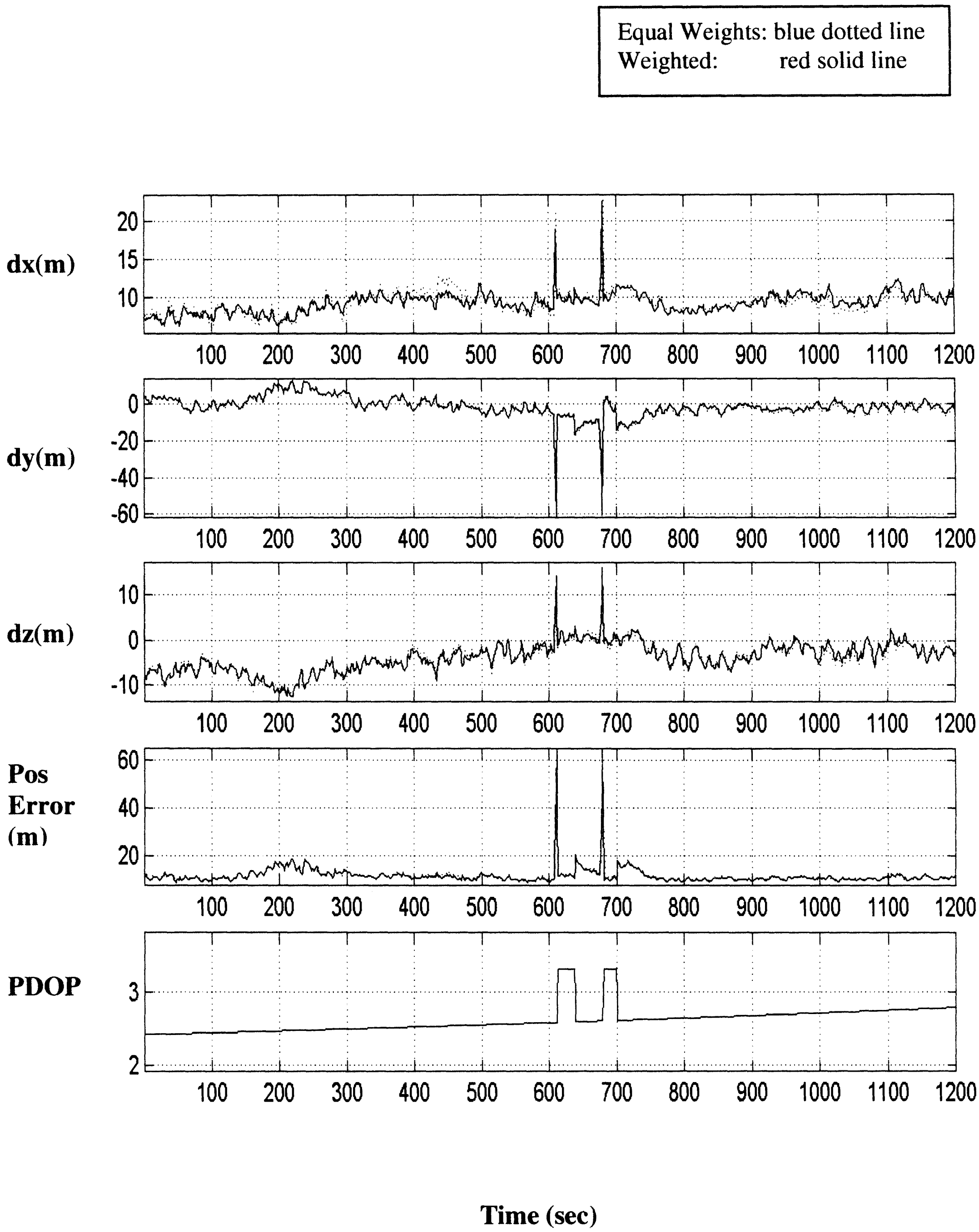


Figure 6.5 GLONASS Positioning Errors (Equal Weights and Weighted)

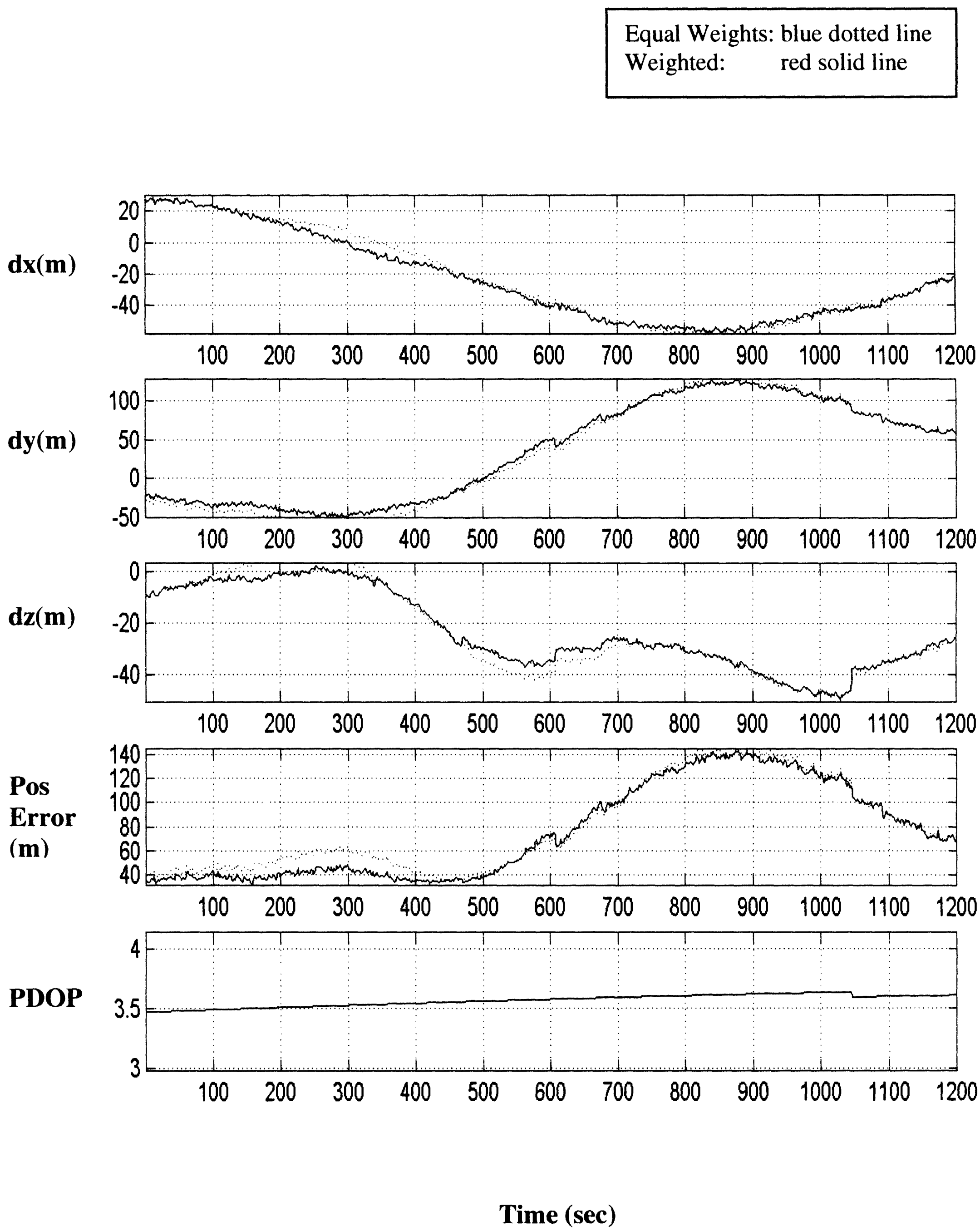


Figure 6.6 GPS Positioning Errors (Equal Weights and Weighted)

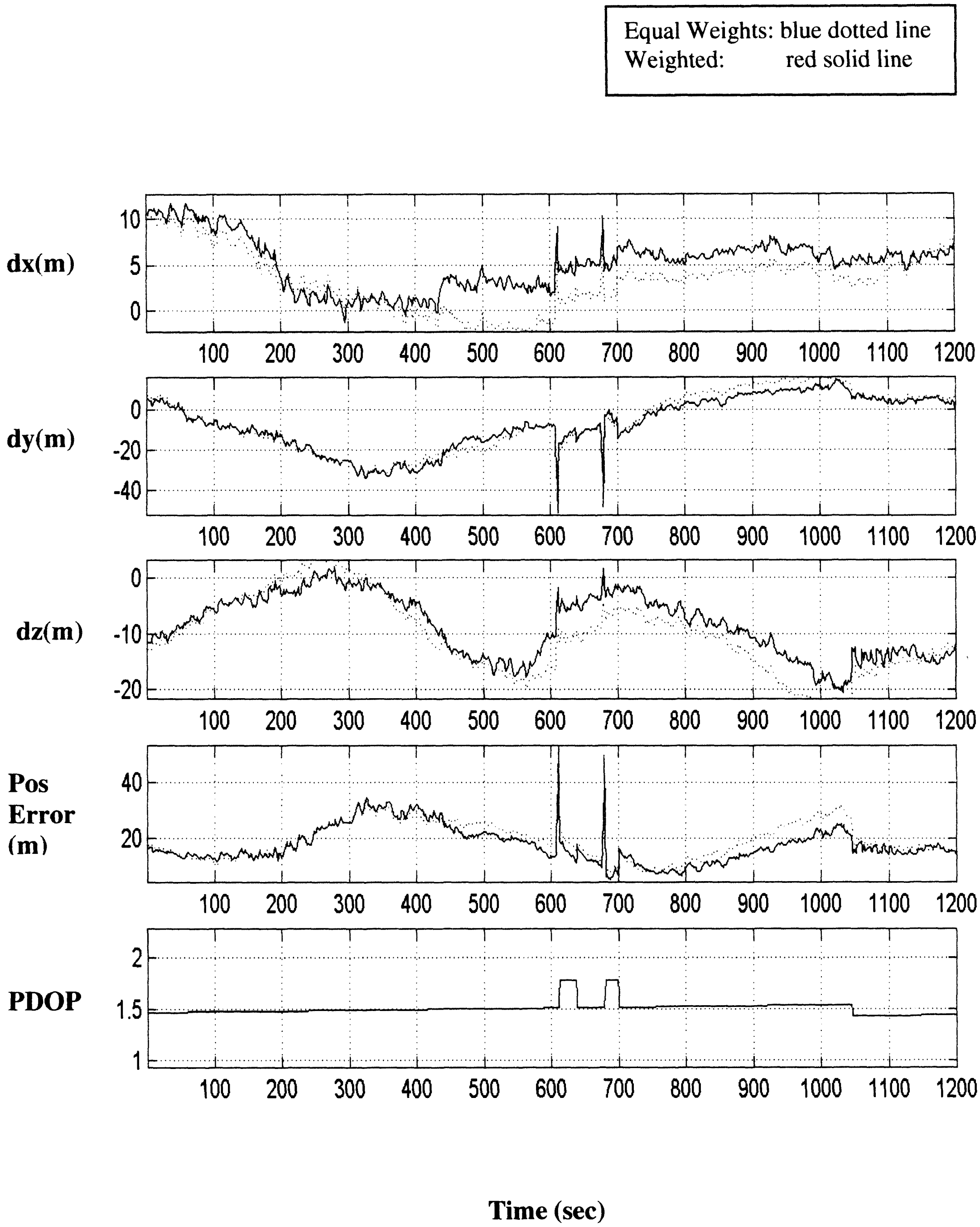
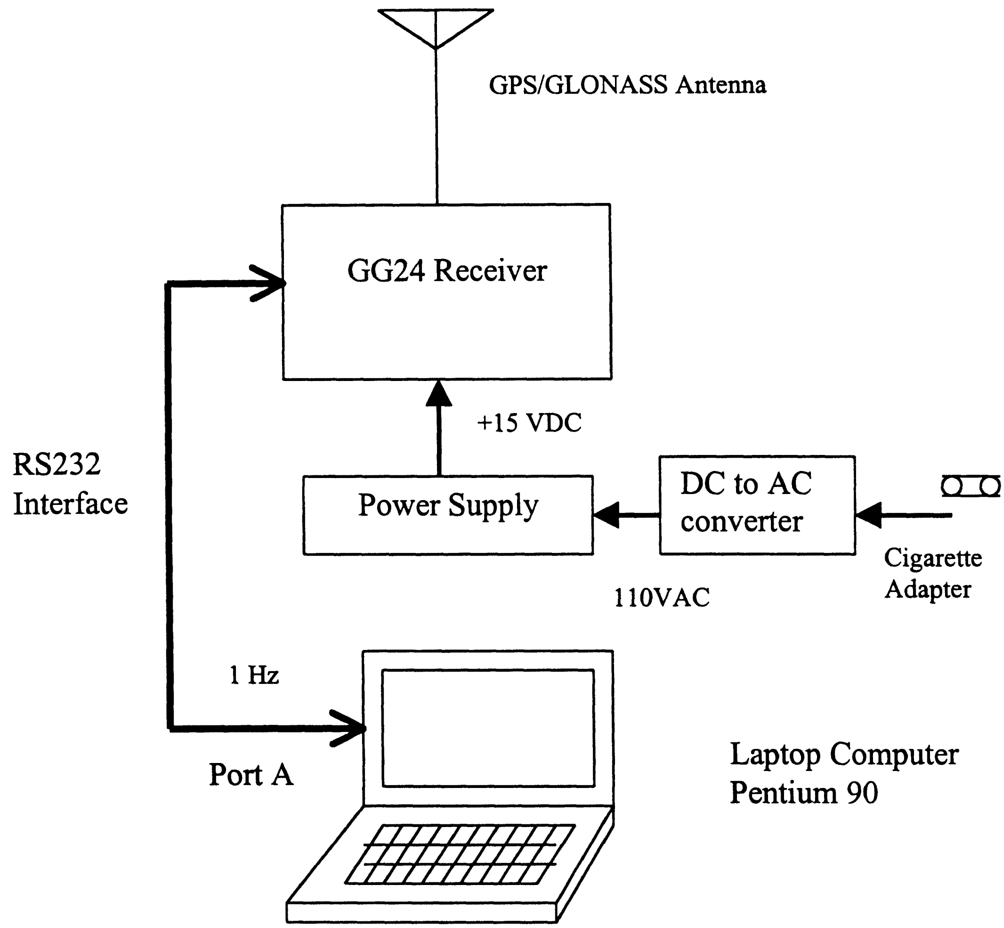


Figure 6.7 Combined GPS/GLONASS Positioning Errors
(Equal Weights and Weighted)

6.2 VEHICLE TRACKING RESULTS

The aim of the GPS/GLONASS vehicle positioning test was to determine the advantage of combining GPS and GLONASS in an application where satellite availability is a problem. In urban areas satellite masking by nearby buildings, trees and other vehicles is an issue. Often only satellites with high elevation angles can be observed. When there are fewer than 4 satellites, for GLONASS or GPS, 3-D positioning is not possible without other aids. It should be noted that the purpose of the work described in this paper was not to produce a complete vehicle navigation system since this would require integration with other sensors. The purpose is to compare the percentage of time that 3-D positioning is possible with, and without, combining GLONASS and GPS. This would determine the potential for GPS/GLONASS positioning as a component of a vehicle navigation system.

As shown in Figure 6.8, a GG24 receiver was placed on the front seat of the vehicle and powered by the car battery via an inverter. The receiver was connected to a Pentium 90 laptop computer which was used to collect both GLONASS and GPS data at 1 second intervals. The antenna which receives both GLONASS and GPS signals was mounted on the roof of the vehicle. To simulate the urban area environment, the vehicle was driven around the North Green of Ohio University's campus for a period of time of about 30 minutes. In this area, some of the satellite signals are shielded by nearby buildings, high trees and other vehicles. By comparing the campus map of Figure 6.9 and the tracking paths of the vehicle in Figures 6.10 through 6.12, it is clear



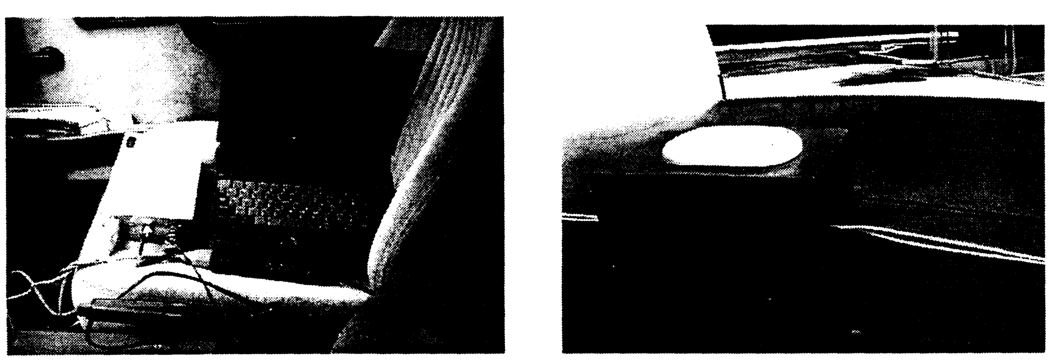


Figure 6.8 System Setup for Vehicle Tracking

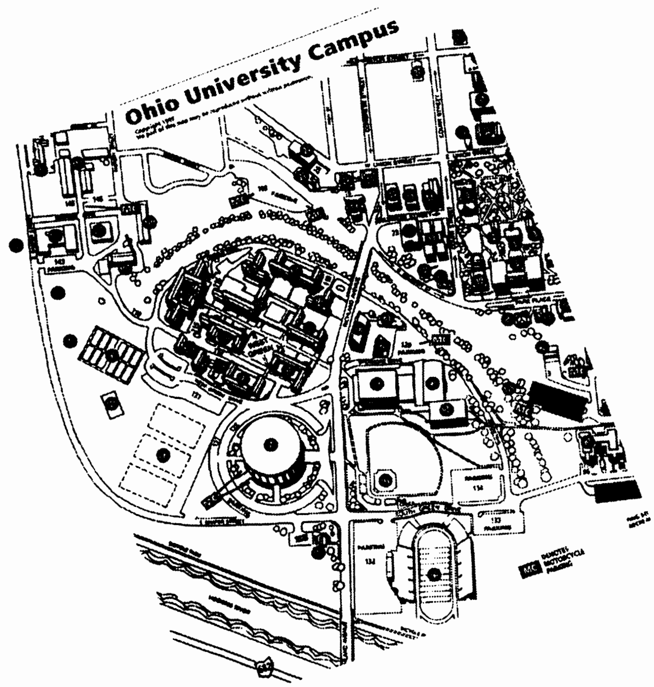


Figure 6.9 Map of Ohio University's North Green

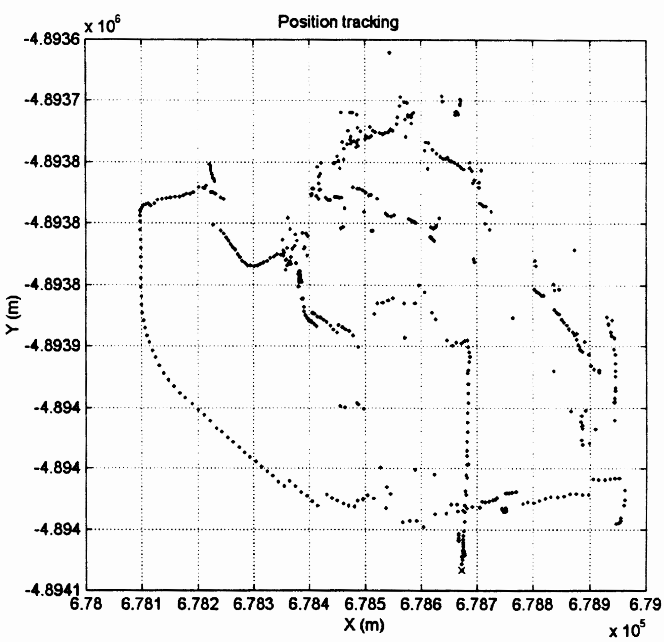


Figure 6.10 Tracking Path of the North Green (GLONASS+GPS)

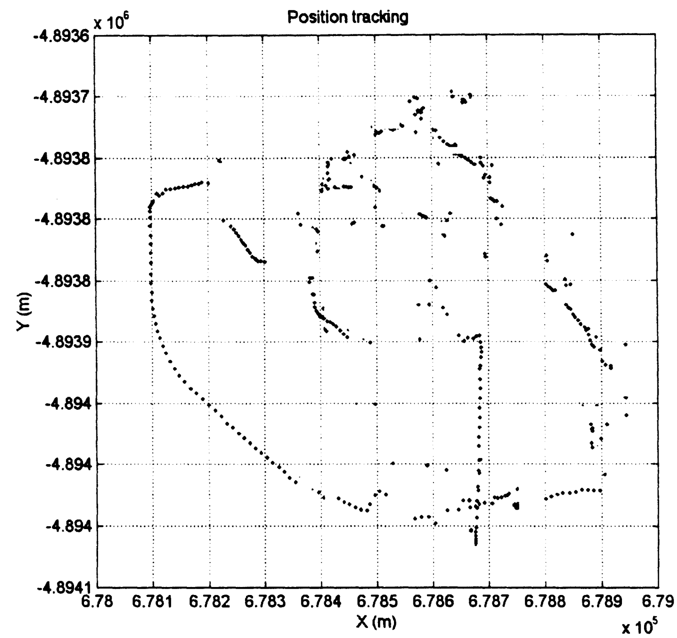


Figure 6.11 Tracking Path of the North Green (GLONASS)

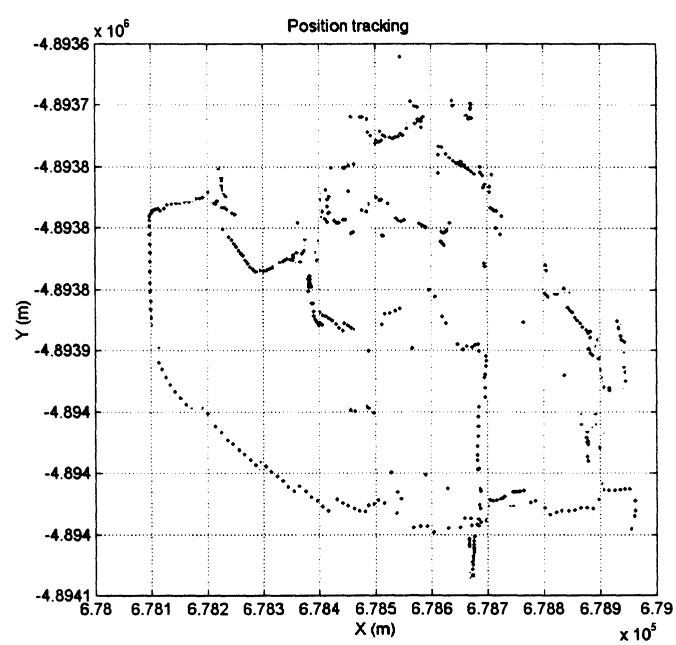


Figure 6.12 Tracking Path of the North Green (GPS)

that the position solutions were not available every second. Note that the satellite positions are given in ECEF-coordinates, which means that they cannot be compared directly with the map in Figure 6.9. However, the ECEF vehicle track in X and Y coordinates is similar in shape to the vehicle track in East and North coordinates. Some calculated positions that are very far away from the path are due to poor satellite geometry. Only those satellites that were at high-elevation angles were not blocked and could be used to determine the vehicle's position. The percentages of time that a 3D-position solution was determined for GLONASS alone, GPS alone and GLONASS/GPS in urban areas are shown in Figure 6.13. Clearly, the tracking path of the combined system as shown in Figure 6.10 can provide more 3D coverage (82.6%) than the two systems alone. At the time of this writing, since the number of operational GLONASS satellites is less than 16, GPS with 24 operational satellites can provide better 3-D coverage (45.3%) than GLONASS (38.1%) as shown in Table 6.3.

Table 6.3 Availability of a 3-D Position in Urban and Countryside Areas

	GLONASS (%)	GPS (%)	GPS+GLONASS (%)
Urban Areas	38.1	45.3	82.6
Countryside Areas	80.8	86.3	90.9

To simulate the countryside area situation, the vehicle was driven for 90 minutes on roads around Athens, Ohio to record data in a countryside environment. The route taken during the test is shown in Figure 6.14. As shown in Figures 6.15 through 6.17, GLONASS alone, GPS alone and the combined system all receive enough satellites to provide vehicle tracking with reasonable availability. Table 6.3 and Figure 6.18 also indicate that the improvement in 3-D position availability in countryside areas is small but still significant at 12.5 % for GLONASS and 5.3% for GPS.

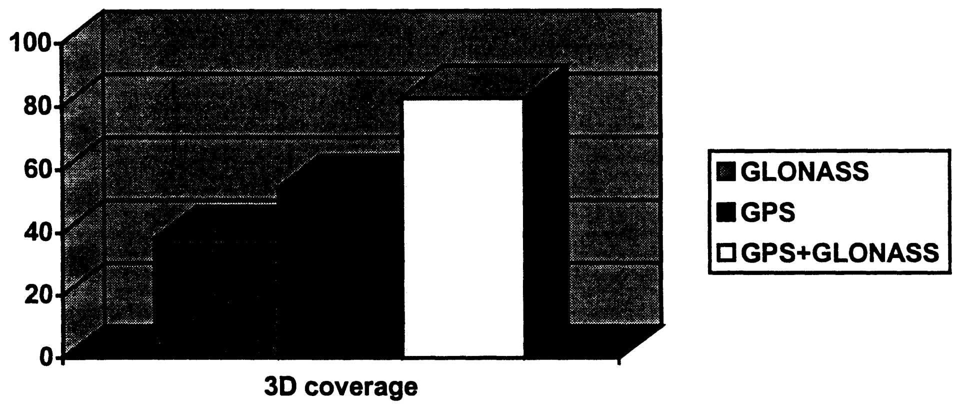


Figure 6.13 Availability of a 3-D Position in Urban Areas

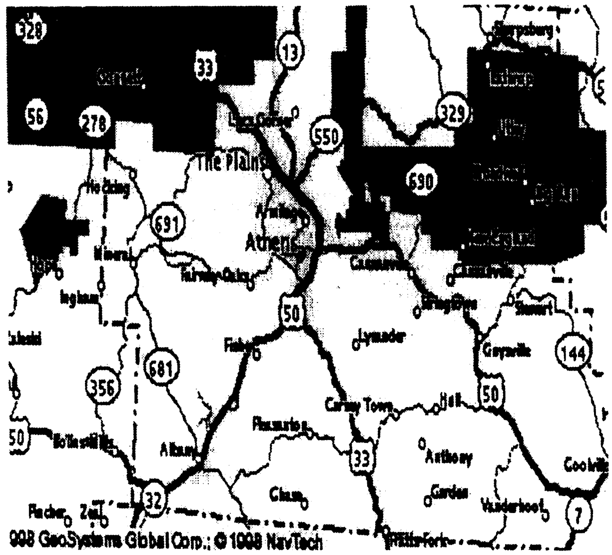


Figure 6.14 Map of the Countryside Around Athens, OH

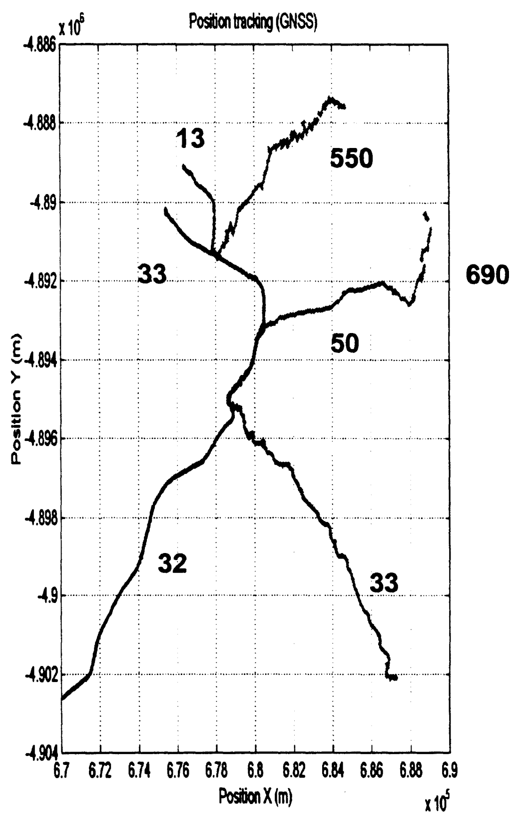


Figure 6.15 Vehicle tracking in Countryside Area (GLONASS+GPS)

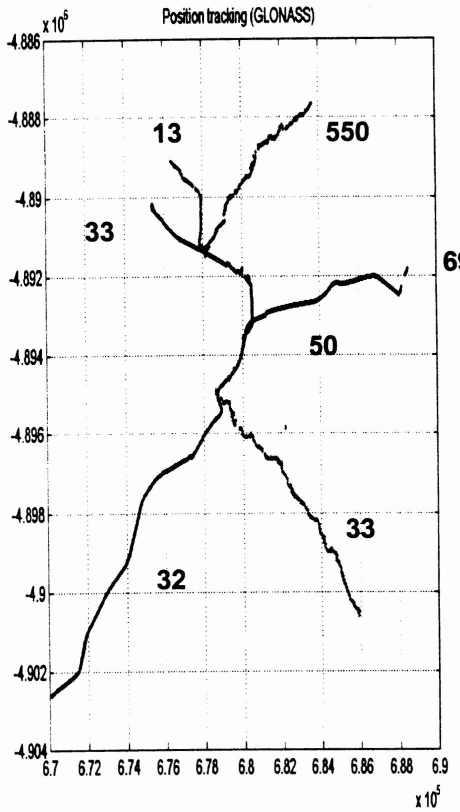


Figure 6.16 Vehicle Tracking in Countryside Area (GLONASS)

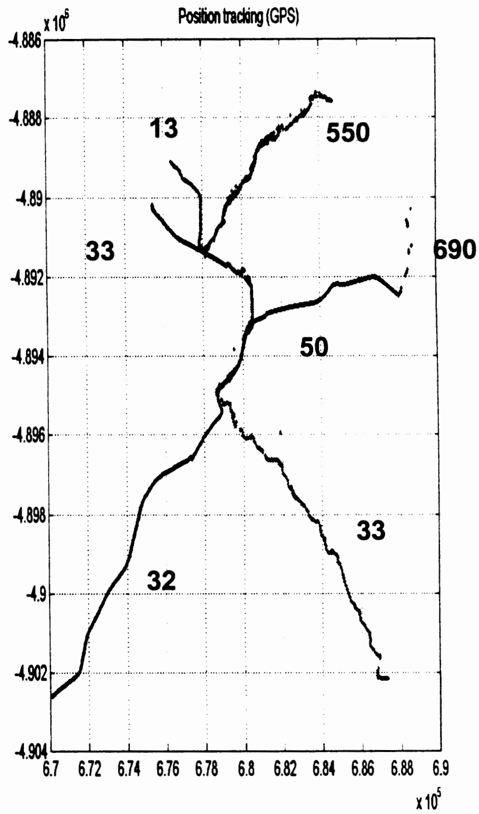


Figure 6.17 Vehicle Tracking in Countryside Area (GPS)

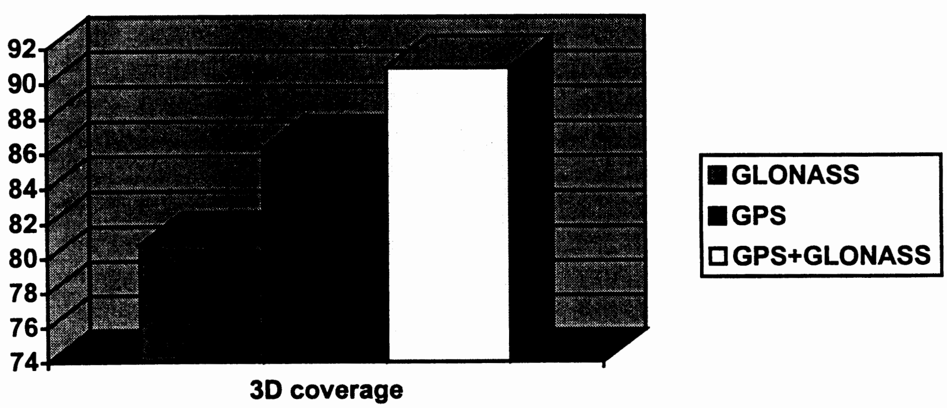


Figure 6.18 Availability of a 3-D Position in Countryside Areas

6.3 FLIGHT TESTS RESULTS

In this application, the flight test was performed using the GG24 GPS/GLONASS receiver and an Ashtech dual-frequency Z12 reference system. Both GPS and GPS/GLONASS antennas were mounted on top of Ohio University's DC-3 airplane. Figure 6.19 shows the block diagram of the equipment set-up. The test flight was performed at the Ohio University Albany airport on 28 May 1998. The aircraft took off and climbed to approximately 2500 feet and circled the Ohio University campus. Data were collected for about 2 hours. The flight path calculated from the GG24 data is then compared with the reference from the Z12 after post-processing. The Z-12 path is considered to be the true flight path as its positioning accuracy is on the order of 0.1 meters.

The flight path of the DC-3 calculated using GLONASS, GPS and combined GPS/GLONASS are similar as shown in Figure 6.20. Without the shielding due to buildings, trees and vehicles, both receivers continuously track the incoming signals and the minimum number of tracked satellites remained above 3 for GLONASS alone, GPS alone and combined GPS/GLONASS. To investigate the accuracy of the position for all combinations, the differences between the calculated positions and the reference positions are shown in Figure 6.21. The accuracies are consistent with the static results of Section 6.1. Table 6.4 shows the standard deviations of the flight test positioning accuracies for GPS, GLONASS and combined GPS/GLONASS.

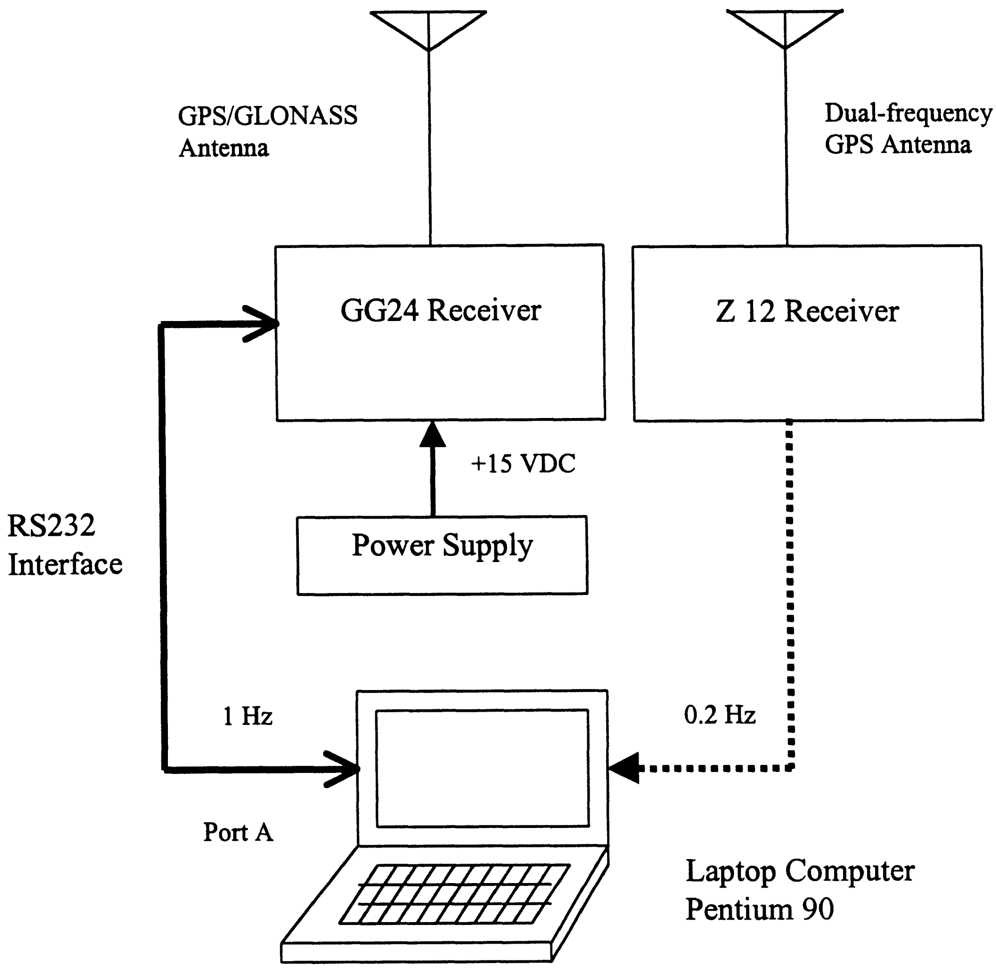
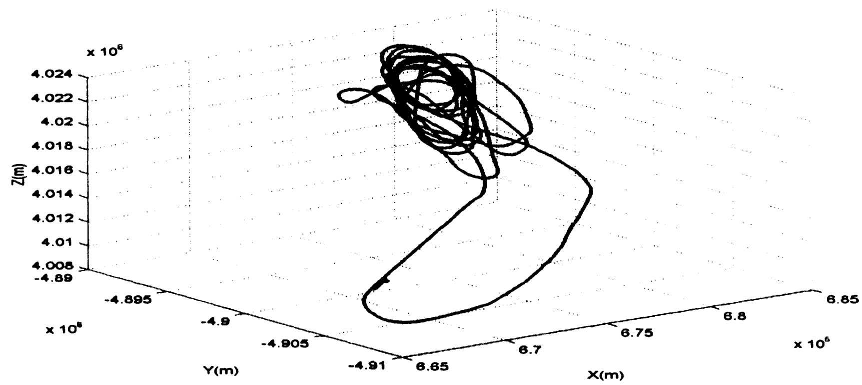
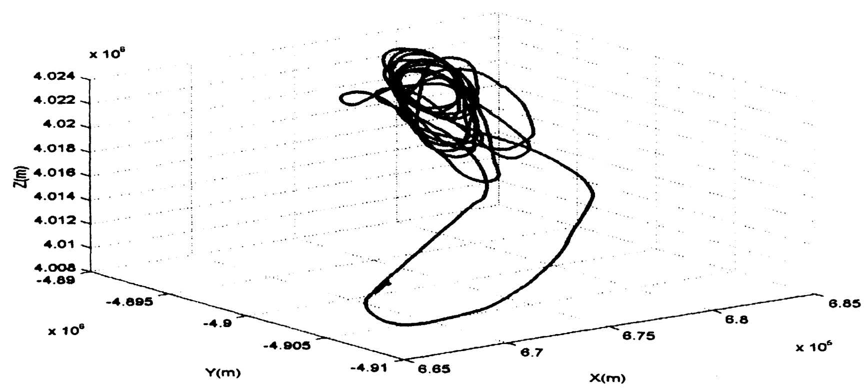


Figure 6.19 System Setup for Flight Testing

Flight path (GLONASS)



Flight path (GPS)



Flight path (GPS/GLONASS)

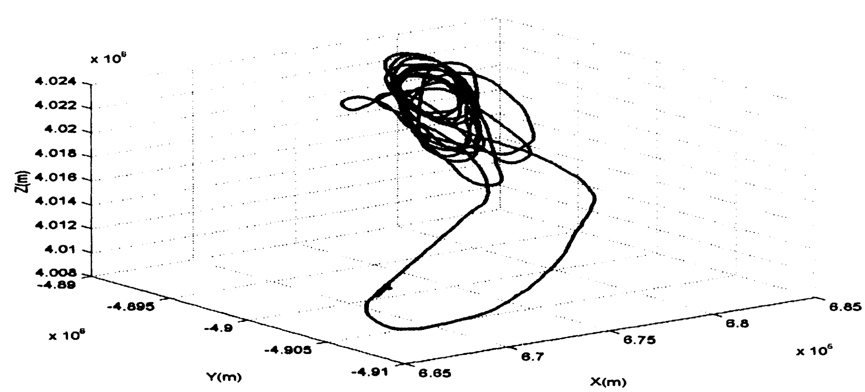


Figure 6.20 Flight Paths of GLONASS, GPS and Combined GPS/GLONASS (WGS84 Coordinate System)

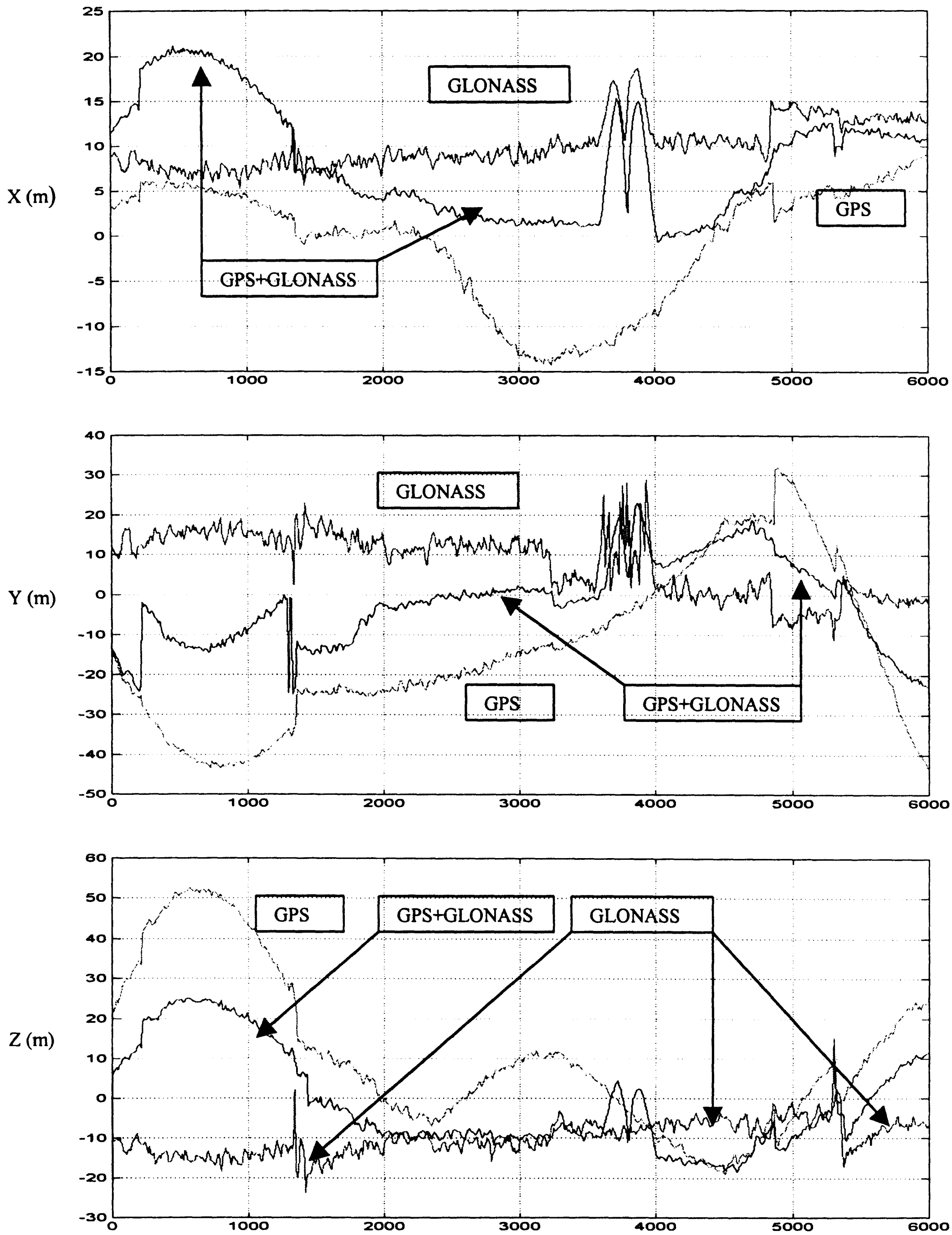


Figure 6.21 GPS, GLONASS and Combined GPS/GLONASS Flight Test Position Errors

The performance of the combined GPS/GLONASS can be assessed in terms of the number of satellites tracked and the PDOP of the estimated solution. These data are shown in Figures 6.22 and 6.23. It can be seen in Figure 6.22 that the average number of GLONASS and GPS satellites tracked during the flight was 13, with a maximum of 14 tracked on several occasions. The number of GLONASS satellites fell as low as 4. Thus if only GLONASS were being used, then no integrity monitoring could be performed, and the possibility of an invalid solution would exist. The minimum of 11 combined satellites shows the advantage of a combined GLONASS/GPS system for aviation applications.

In Figure 6.23, it can be observed that the PDOP for the combined GPS/GLONASS solution remains below 1.75.

Table 6.4 Flight Test Positioning Accuracies

	GLONASS	GPS	GPS+GLONASS
Standard Deviation (m)	16.67	23.27	14.29

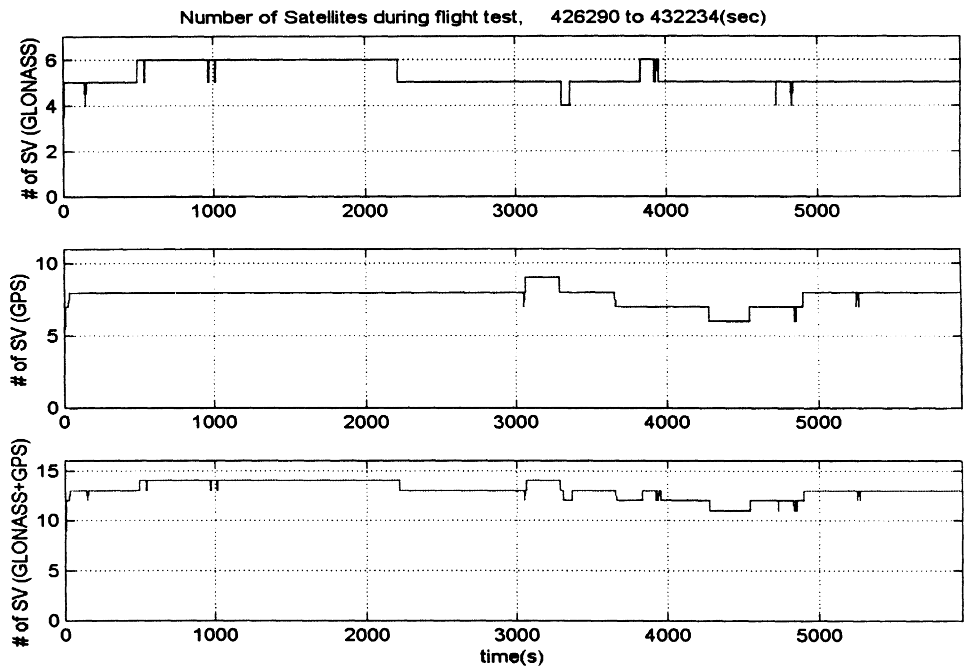


Figure 6.22 Number of Satellites during the Flight Test

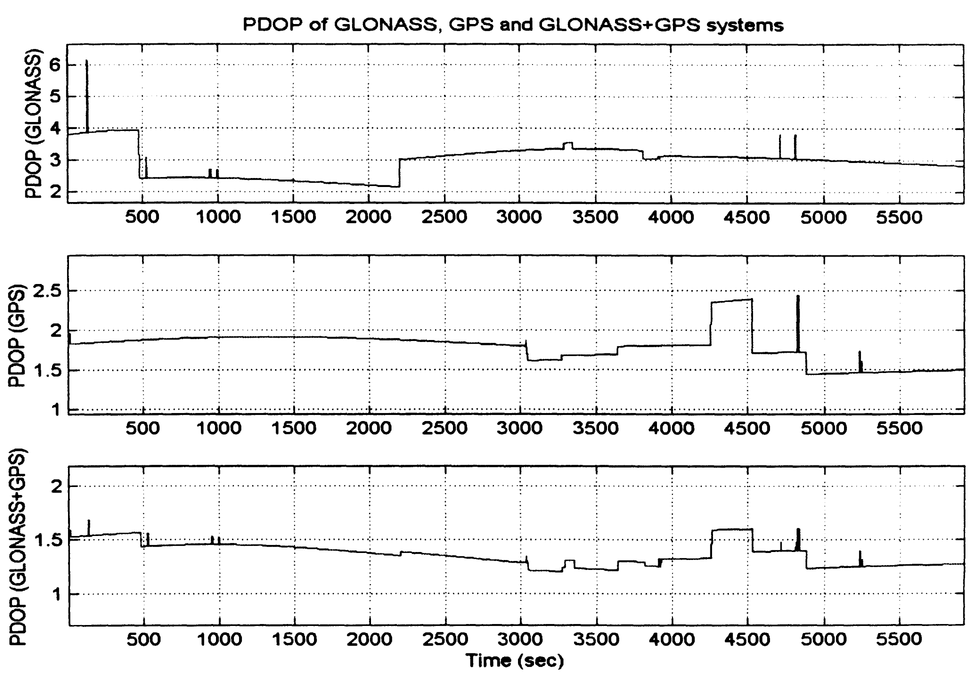


Figure 6.23 PDOP during the Flight Test

7. CONCLUSIONS AND RECOMMENDATIONS

Based on the research presented in this thesis, the following conclusions are made.

- GLONASS satellite positions can be successfully calculated by implementing the satellite position equations in the GLONASS ICD and by using the Runge-Kutta integration Method.

- At the time of this writing, GLONASS alone can provide a more accurate position solution than GPS alone or a combined GPS/GLONASS system.

- The availability of a combined GPS/GLONASS solution is significantly higher than each of the systems alone. It was found that this is especially significant for positioning in urban areas.

- The combined system used during the flight trial demonstrated the advantage of using a combined GLONASS/GPS navigation system. At no point in the flight did the number of satellites drop below 11 or the PDOP rise above 1.75. The availability of a large number of satellites at all times improves the availability of receiver autonomous integrity monitoring.

Recommendations for further research include the following.

- Establish a standard method for the calculation of GLONASS satellite positions.

- Investigate the integration of GPS/GLONASS with other navigation sensors for vehicle positioning applications.

- Evaluate the use of GLONASS in a differential mode of operation.

REFERENCES

1. ICD-GPS-200 (GPS Interface Control Document), Rockwell International Corp., ICD GPS 200, Rev. B, 1987.
2. Leick, A., GPS Satellite Surveying (Second Edition), John Wiley & Sons, New York, 1994.
3. Ashtech GG24 GPS/GLONASS OEM board Receiver Reference Manual, Revision B, Ashtech Inc., Sunnyvale, CA, 1997.
4. Braasch, M. and Van Graas, F., "Guidance Accuracy Considerations for Real Time GPS Interferometry", Proceedings of the 4th International Meeting of the ION Satellite Division, Albuquerque, NM, September 1991, pp. 122-142.
5. Riley, S., Daly, P., "Architecture and results from the Leeds University 20 Channel GNSS Receiver", Proceedings of the ION National Technical meeting, Anaheim, CA, January 1995, pp. 1001-1012.
6. Daly, P., "GLONASS Approaches Full Operational Capability", Proceedings of the 8th International Technical Meeting of the ION Satellite Division, Palm Springs, CA, September 1995, pp. 1021-1230.
7. Daly, P., *Progress Toward the Operational Phase of GLONASS*, NAVIGATION, Journal of The Institute of Navigation, Vol. 38, No. 1, 1991, pp. 37-51.
8. Misra, P., Burke, B., Pratt, M. and Hall, T., "Comparison of GPS and GPS+GLONASS Positioning Performance", Proceedings of ION GPS-97, Kansas City, MO, September 1997, pp. 107-212.
9. Wells, D., Guide to GPS Positioning, Canadian GPS Associates, 1987.

10. Eastwood, R. A., *An Integrated GPS/GLONASS Receiver*, NAVIGATION, Journal of The Institute of Navigation, Vol. 37, No.2, 1990, pp. 141-151.

11. Kaplan, Elliott D. (Editor), Understanding GPS Principles and Applications, Boston/London, Artech House, 1996.

12. Kreyszig, Erwin, Advanced Engineering Mathematics, Seventh edition, John Wiley & Sons, 1993.

13. Balendra, A., Kim, S., and Beser, J., "Fully Integrated GLONASS Dual Frequency P Code and GPS/GLONASS Signal Frequency C/A code Receiver. Features and Performance", Proceedings of ION GPS 94, Salt Lake City, UT, September 1994, pp. 21-50.

14. Cooper, George R. and McGillem, Clare D., Probabilistic Methods of Signal and System Analysis (Second Edition), Chicago, Holt, Rinehart and Winston, 1971.

15. Cook, Gerald L., *GLONASS Performance, 1995-1997, and GPS-GLONASS Interoperability Issues*, NAVIGATION, Journal of The Institute of Navigation, Vol. 44, No. 3, 1997, pp. 291-300.

16. *GLONASS History*, Coordinational Scientific Information Center Internet page, December 1997, http://www.rssi.ru/SFCSIC/5.txt.

17. GLONASS Interface Control Document (Second Revision), Navtech Seminars & GPS Supply, Inc. 1995.

18. Kleusberg, Alfred, *Comparing GPS and GLONASS*, GPS World, November/December 1990, pp. 52-54.

19. GPS Standard Positioning Service Signal Specification (Second Edition), GPS Navstar, 1995.

20. Ivanov, N.E. and Salischev, V., *GLONASS and GPS: Prospects for a Partnership*, GPS World, Vol. 2, No. 4, April 1991, pp. 36-40.

21. Leick, Alfred, Li, Jinye, Beser, Jacques and Mader, Gerald, "Processing GLONASS Carrier Phase Observations Theory and First Experience", Proceedings of ION GPS-95, Palm Springs, CA, September 1995, pp. 322-350.

22. Lowe, D., Walsh, D., Capaccio, S., Daly, P., Richards, G. and Sharkey, S., "Real Time Differential Positioning of Aircraft using GPS and GLONASS", Proceedings ION 52nd Annual Meeting, Cambridge, Massachusetts, June 1996, pp. 236-249.

23. Lebedev M., et al., "The GLONASS System Status and Prospects for Development", Proceedings of ION GPS-95, Palm Springs, CA, September 1995, pp. 959-969.

24. Misra, P., Muchnik, R., Burke, B., and Hall, T., "GLONASS Performance: Measurement Data Quality and System Upkeep", Proceedings of 9th International Technical Meeting of the ION Satellite Division, Kansas City, MO, September 1996, pp. 261-269.

25. Misra, P., et al., "Integrated Use of GPS and GLONASS in Civil Aviation Navigation I: Coverage and Data Models", Proceedings of the ION National Technical meeting, Colorado Springs, CO, September 1990, pp. 1021-1045.

26. Misra, P., et al., *GLONASS Data Analysis: Interim Results*, NAVIGATION, Journal of The Institute of Navigation, Vol.39, No. 1, 1992, pp. 93-109.

27. Misra, P., et al., "GLONASS Performance in 1995: A Review", Proceedings of ION GPS-96, Kansas City, MO, 17-21 September 1996, pp. 145-152.

28. Misra, P., et al., *Integrated Use of GPS and GLONASS in Civil Aviation*, Lincoln Laboratory Journal, Vol. 6, No. 2, 1993, pp. 219-320.

29. Misra, P., et al., "Integrated Use of GPS and GLONASS: Transformation between WGS84 and PZ-90", Proceedings of ION GPS-96, Kansas City, MO, September 1996, pp. 345-402.

30. Zarzhan, P. (Editor-in-Chief), Global Positioning System: Theory and Applications, Vol. 1,2. American Institute of Aeronautics and Astronautics, 1996.

31. Raby, P., Daly, P., "Surveying with GLONASS: Calibration, Error Sources and Results", Proceedings of DSNS94, London, April 18-22, 1994, pp. 100-124.

32. Raby, P., Daly, P., "Integrated GPS/GLONASS Navigation: Algorithms and Results", CAA-Institute of Satellite Navigation, University of Leeds.

33. Raby, P., Daly, P., "Using the GLONASS system for Geodetic Survey", Proceedings of the ION National Technical meeting, Salt Lake City, Utah, September 1993, pp. 456-471.

34. Spilker, J.J., Digital Communications by Satellite, Prentice Hall, 1977.

35. Townsend, B.R. and Fenton, P.C., "A Practical Approach to the Reduction of Pseudorange Multipath errors in L1 GPS Receiver", Proceedings of the 7th International Meeting of the ION Satellite Division, Salt Lake City, September 1994, pp. 1024-1059.

36. Van Graas, F., et al., "FAA/Ohio University/UPS Autoland Flight Test Results", Proceedings of the ION National Technical meeting, Anaheim, January 1995, pp. 21-40.

37. Walsh, D., Riley, S., Cooper, J., Daly, P., "Precise Positioning Using GPS/GLONASS Carrier Phase and Code Phase Observables", CAA-ISN, Leeds, September 12-15 1995, pp. 499-506.

38. Walsh, D., Riley, S., Daly, P., "Differential Vehical Positioning using GPS and GLONASS in Urban Areas", CAA-Institute of Satellite Navigation, University of Leeds, 1997.

39. Gouzhva,Yuri, Koudryavtsev, Igor, Korniyenko, Vladimir, Pushkina, Irina *GLONASS Receivers:An Outline*, GPS World, January 1994, pp.30-36.

40. Diggle, D.W., "An Investigation Into The Use Of Satellite-Based Positioning Systems For Flight Reference/Autoland Operations", Ph.D. Dissertation, Department of Electrical and Computer Engineering, Ohio University, Athens, Ohio, 1994.

41. <u>ONX 4 Operating System User's Guide</u>, ONX Software Systems Ltd. Ontario Canada, 1993

42. *GLONASS Satellite Constellation*, MIT Lincoln Lab Internet page, May 1998, http://satnav.atc.ll.mit.edu/glonass/mages/deploy.gif

APPENDIX A

A.1 RKM EQUATIONS FOR GLONASS SATELLITE POSITION CALCULATION

The motivation for using the RKM is that six differential equations and initial conditions are given. The RKM allows for the solution of such a problem. For example, consider the following differential equation

$$\frac{dy}{dt} = f(t, y) \tag{A.1}$$

Where *f(t,y)* is some function of *t* and *y*. Assume also that we are given the initial condition $y(t_0) = y_0$, which specifies a known value of *f* at some $t = t_0$. In the case of GLONASS, there are six differential equations, so *f* can be interpreted as a velocity equation or an acceleration equation. Note that the satellite accelerations are assumed to be constant during a 30-minute time interval. Further, the initial conditions are the initial position, velocity and acceleration of the GLONASS satellite at reference time t_b.

The GLONASS ICD suggests the use of a fourth-order Runge-Kutta method [17]. To implement this, the Runge-Kutta formula uses a weighted average of values of the derivative *f(t,y)* taken at different points in the interval $t_n \le t \le t_{n+1}$. The value of y at time n+1 is related to the value of y at time n according to the following equation.

$$y_{n+1} = y_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{A.2}$$

where

$$k_1 = hf(t_n, y_n) \tag{A.3}$$

$$k_2 = hf(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}) \tag{A.4}$$

$$k3 = hf(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}) \tag{A.5}$$

$$k4 = hf(t_n + h, y_n + k_3) \tag{A.6}$$

The weighted sum $\frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$ in Equation (A.2) can be thought of as an average slope of the function, and h is the step size. The process of applying the RKM to the GLONASS satellite position calculation proceeds as follows. First, the data broadcast by the GLONASS satellites is used for the initial conditions of the RKM. These data are the reference time t_b, the initial satellite position (x, y, z) and the initial satellite velocity ($\dot{x}, \dot{y}, \dot{z}$).

$$t_0 = t_b \tag{A.7}$$

$$\omega_{10} = x \tag{A.8}$$

$$\omega_{20} = y \tag{A.9}$$

$$\omega_{30} = z \tag{A.10}$$

$$\omega_{40} = \dot{x} \tag{A.11}$$

$$\omega_{50} = \dot{y} \tag{A.12}$$

$$\omega_{60} = \dot{z} \tag{A.13}$$

Next, for each of the six differential GLONASS equations, the k-parameters are obtained by applying equation (A.3) through (A.6) to equation (4.4) through (4.9). Specifically, Equation (4.4) results in the following k-parameters.

$$k_{11} = hf_1(t_0, \omega_{10}, \omega_{20}, \omega_{30}, \omega_{40}, \omega_{50}, \omega_{60})$$

$$= h(\omega_{40}) \tag{A.14}$$

$$k_{21} = hf_1(t_0 + \frac{1}{2}h, \omega_{10} + \frac{1}{2}k_{11}, \omega_{20} + \frac{1}{2}k_{12}, \omega_{30} + \frac{1}{2}k_{13}, \omega_{40} +$$

$$\frac{1}{2}k_{14}, \omega_{50} + \frac{1}{2}k_{15}, \omega_{60} + \frac{1}{2}k_{16}) = h(\omega_{40} + \frac{1}{2}k_{14}) \tag{A.15}$$

$$k_{31} = hf_1(t_0 + \frac{1}{2}h, \omega_{10} + \frac{1}{2}k_{21}, \omega_{20} + \frac{1}{2}k_{22}, \omega_{30} + \frac{1}{2}k_{23}, \omega_{40} +$$

$$\frac{1}{2}k_{24}, \omega_{50} + \frac{1}{2}k_{25}, \omega_{60} + \frac{1}{2}k_{26}) = h(\omega_{40} + \frac{1}{2}k_{24}) \tag{A.16}$$

$$k_{41} = hf_1(t_0 + h, \omega_{10} + k_{31}, \omega_{20} + k_{32}, \omega_{30} + k_{33}, \omega_{40} +$$

$$k_{34}, \omega_{50} + k_{35}, \omega_{60} + k_{36}) = h(\omega_{40} + k_{34}) \tag{A.17}$$

where the first subscript of k refers to one of the four integration parameters and the second subscript of k refers to one of the six differential GLONASS equations. Similarly, the remaining 20 k-parameters are obtained as follows.

$$k_{12} = h(\omega_{50}) \tag{A.18}$$

$$k_{22} = h(\omega_{50} + \frac{1}{2}k_{15}) \tag{A.19}$$

$$k_{32} = h(\omega_{50} + \frac{1}{2}k_{25}) \tag{A.20}$$

$$k_{42} = h(\omega_{50} + k_{35}) \tag{A.21}$$

$$k_{13} = h(\omega_{60}) \tag{A.22}$$

$$k_{23} = h(\omega_{60} + \frac{1}{2}k_{16}) \tag{A.23}$$

$$k_{33} = h(\omega_{60} + \frac{1}{2}k_{26}) \tag{A.24}$$

$$k_{43} = h(\omega_{60} + k_{36}) \tag{A.25}$$

$$k_{14} = h(\frac{-\mu}{\gamma^3}\omega_{10} + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}\omega_{10}\left(1 - \frac{5}{\gamma^2}\omega_{30}^2\right) + \omega_3^2\omega_{10} + 2\omega_3\omega_{50} + \ddot{x}) \tag{A.26}$$

$$k_{24} = h\{\ \frac{-\mu}{\gamma^3}(\omega_{10} + \frac{1}{2}k_{11}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{10} + \frac{1}{2}k_{11})$$
$$\left(1 - \frac{5}{\gamma^2}(\omega_{30} + \frac{1}{2}k_{13})^2\right) + \omega_3^2(\omega_{10} + \frac{1}{2}k_{11}) + 2\omega_3(\omega_{50} + \frac{1}{2}k_{15}) + \ddot{x}\} \tag{A.27}$$

$$k_{34} = h\{\ \frac{-\mu}{\gamma^3}(\omega_{10} + \frac{1}{2}k_{21}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{10} + \frac{1}{2}k_{21})$$
$$\left(1 - \frac{5}{\gamma^2}(\omega_{30} + \frac{1}{2}k_{23})^2\right) + \omega_3^2(\omega_{10} + \frac{1}{2}k_{21}) + 2\omega_3(\omega_{50} + \frac{1}{2}k_{25}) + \ddot{x}\} \tag{A.28}$$

$$k_{44} = h\{\ \frac{-\mu}{\gamma^3}(\omega_{10} + k_{31}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{10} + k_{31})$$
$$\left(1 - \frac{5}{\gamma^2}(\omega_{30} + k_{33})^2\right) + \omega_3^2(\omega_{10} + k_{31}) + 2\omega_3(\omega_{50} + k_{35}) + \ddot{x}\} \tag{A.29}$$

$$k_{15} = h(\frac{-\mu}{\gamma^3}\omega_{20} + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}\omega_{20}\left(1 - \frac{5}{\gamma^2}\omega_{30}^2\right) + \omega_3^2\omega_{20} - 2\omega_3\omega_{40} + \ddot{y}) \tag{A.30}$$

$$k_{25} = h\{\ \frac{-\mu}{\gamma^3}(\omega_{20} + \frac{1}{2}k_{12}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{20} + \frac{1}{2}k_{12})$$
$$\left(1 - \frac{5}{\gamma^2}(\omega_{30} + \frac{1}{2}k_{13})^2\right) + \omega_3^2(\omega_{20} + \frac{1}{2}k_{12}) - 2\omega_3(\omega_{40} + \frac{1}{2}k_{14}) + \ddot{y}\} \tag{A.31}$$

$$k_{35} = h\{\ \frac{-\mu}{\gamma^3}(\omega_{20} + \frac{1}{2}k_{22}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{20} + \frac{1}{2}k_{22})$$
$$\left(1 - \frac{5}{\gamma^2}(\omega_{30} + \frac{1}{2}k_{23})^2\right) + \omega_3^2(\omega_{20} + \frac{1}{2}k_{22}) - 2\omega_3(\omega_{40} + \frac{1}{2}k_{24}) + \ddot{y}\} \tag{A.32}$$

$$k_{45} = h\{ \frac{-\mu}{\gamma^3}(\omega_{20} + k_{32}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{20} + k_{32})$$
$$\left(1 - \frac{5}{\gamma^2}(\omega_{30} + k_{33})^2\right) + \omega_3^2(\omega_{20} + k_{32}) - 2\omega_3(\omega_{40} + k_{34}) + \ddot{y}\} \quad \text{(A.33)}$$

$$k_{16} = h(\frac{-\mu}{\gamma^3}\omega_{30} + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}\omega_{30}\left(3 - \frac{5}{\gamma^2}\omega_{30}^2\right) + \ddot{z}) \quad \text{(A.34)}$$

$$k_{26} = h\{ \frac{-\mu}{\gamma^3}(\omega_{30} + \frac{1}{2}k_{13}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{30} + \frac{1}{2}k_{13})$$
$$\left(3 - \frac{5}{\gamma^2}(\omega_{30} + \frac{1}{2}k_{13})^2\right) + \ddot{z}\} \quad \text{(A.35)}$$

$$k_{36} = h\{ \frac{-\mu}{\gamma^3}(\omega_{30} + \frac{1}{2}k_{23}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{30} + \frac{1}{2}k_{23})$$
$$\left(3 - \frac{5}{\gamma^2}(\omega_{30} + \frac{1}{2}k_{23})^2\right) + \ddot{z}\} \quad \text{(A.36)}$$

$$k_{46} = h\{ \frac{-\mu}{\gamma^3}(\omega_{30} + k_{33}) + \frac{3}{2}C_{20}\frac{\mu a_e^2}{\gamma^5}(\omega_{30} + k_{33})$$
$$\left(3 - \frac{5}{\gamma^2}(\omega_{30} + k_{33})^2\right) + \ddot{z}\} \quad \text{(A.37)}$$

Finally, the satellite positions and velocities at the next time-step are obtained using Equation (A. 2).

$$\omega_{11} = \omega_{10} + \frac{1}{6}\left(k_{11} + 2k_{21} + 2k_{31} + k_{41}\right) \quad \text{(A.38)}$$

$$\omega_{21} = \omega_{20} + \frac{1}{6}\left(k_{12} + 2k_{22} + 2k_{32} + k_{42}\right) \quad \text{(A.39)}$$

$$\omega_{31} = \omega_{30} + \frac{1}{6}\left(k_{13} + 2k_{23} + 2k_{33} + k_{43}\right) \quad \text{(A.40)}$$

$$\omega_{41} = \omega_{40} + \frac{1}{6}\left(k_{14} + 2k_{24} + 2k_{34} + k_{44}\right) \quad \text{(A.41)}$$

$$\omega_{51} = \omega_{50} + \frac{1}{6}\left(k_{15} + 2k_{25} + 2k_{35} + k_{45}\right) \tag{A.42}$$

$$\omega_{61} = \omega_{60} + \frac{1}{6}\left(k_{16} + 2k_{26} + 2k_{36} + k_{46}\right) \tag{A.43}$$

APPENDIX B

B.1 POSITION ITERATION USING THE NEWTON RAPHSON METHOD

```
function usrpos = pr2pos(svinfo,usest,no)
%
%   USRPOS: Computation of the 3-dimensional user-position from
%   satellite data (pseudo range + satellite position) received
%   from four or moresatellites, asuming there exists a clock
%   bias between the GPS system time and the user time.
%
%   Algorithm description:  The user position is iteratively
%   computed using linearized expansions.  The iterations use
%   the current user-state-estimate and current satellite data.
%   The iteration ends when the desired accuracy (epsi)
%   with respect to the user position is reached or when
%   the number of iterations exceeds itmax.
%
%   Input: svinfo(1,j) = X  coordinate for satellite j
%          svinfo(2,j) = Y  coordinate for satellite j
%          svinfo(3,j) = Z  coordinate for satellite j
%          svinfo(4,j) = Measured Range for satellite j
%          usest(1:4) = Estimate of user X, Y, Z, B
%          no = number of satellites to be used
%
%   Output:usrpos(1:3) = User X, Y, Z  coordinates
%          usrpos(4)   = User Clock Offset (B)
%

itmax = 50;          % Maximum number of iterations
itflg = 0;           % No iterations done, yet
epsi = 0.01;         % Desired accuracy in solution
duab = 2. * epsi;    % Set initial error larger than epsi

while duab > epsi & itflg < itmax,
  itflg = itflg + 1;
  for j = 1:no,
    temp = usest(1:3) - svinfo(1:3,j)';
    estr = norm(temp) + usest(4);
    dr(j) = svinfo(4,j) - estr;
    denum  = svinfo(4,j) - usest(4);
    h(j,1:3) = temp / denum;
    h(j,4) = 1.0;
  end
  du = inv(h' * h) * h' * dr';
  usest = usest + du';
  duab = norm(du);
end
usrpos = usest;
```

APPENDIX C

C.1 FLOWCHARTS (QNX C)

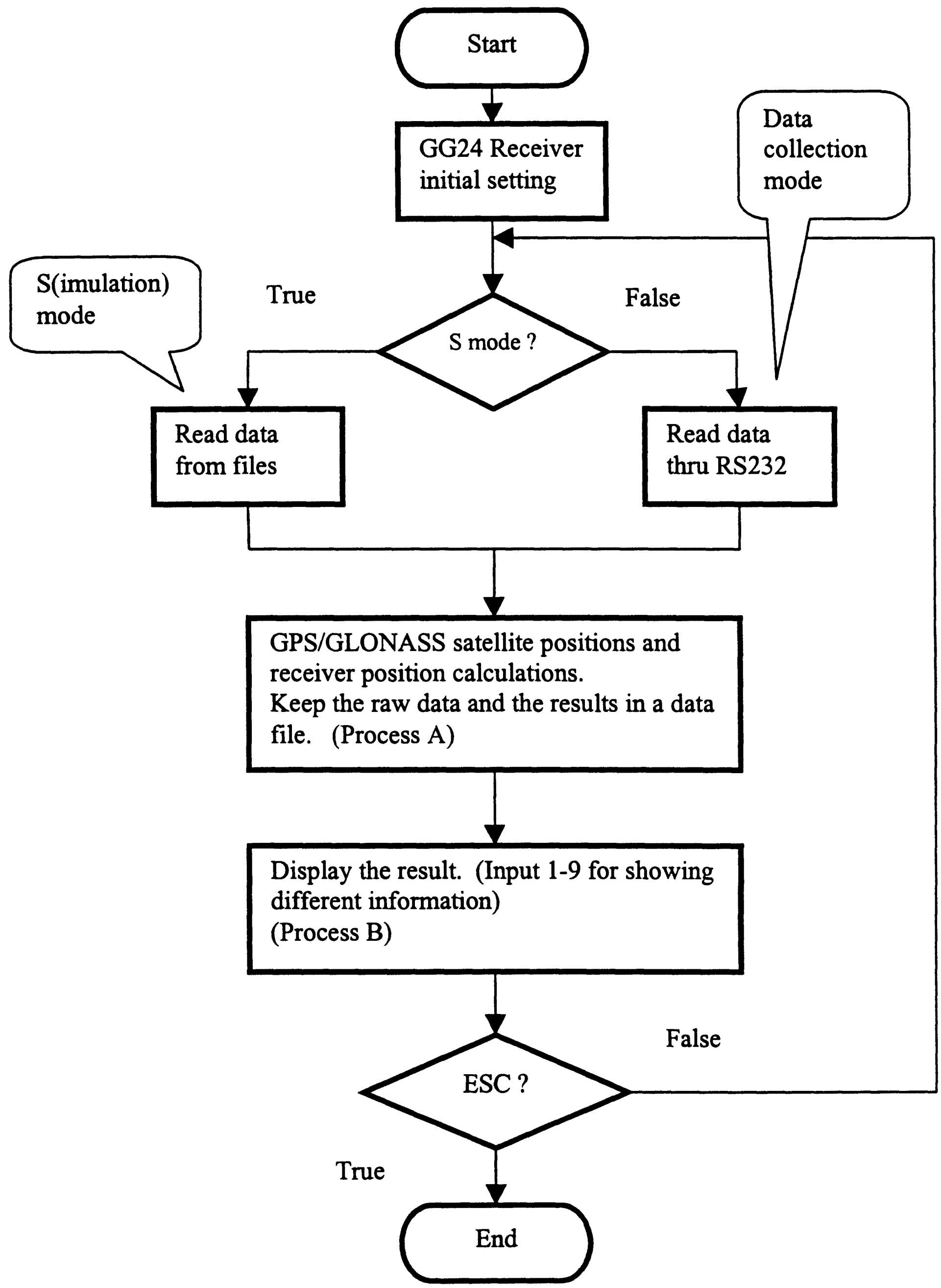


Figure C.1 Main Program Flowchart

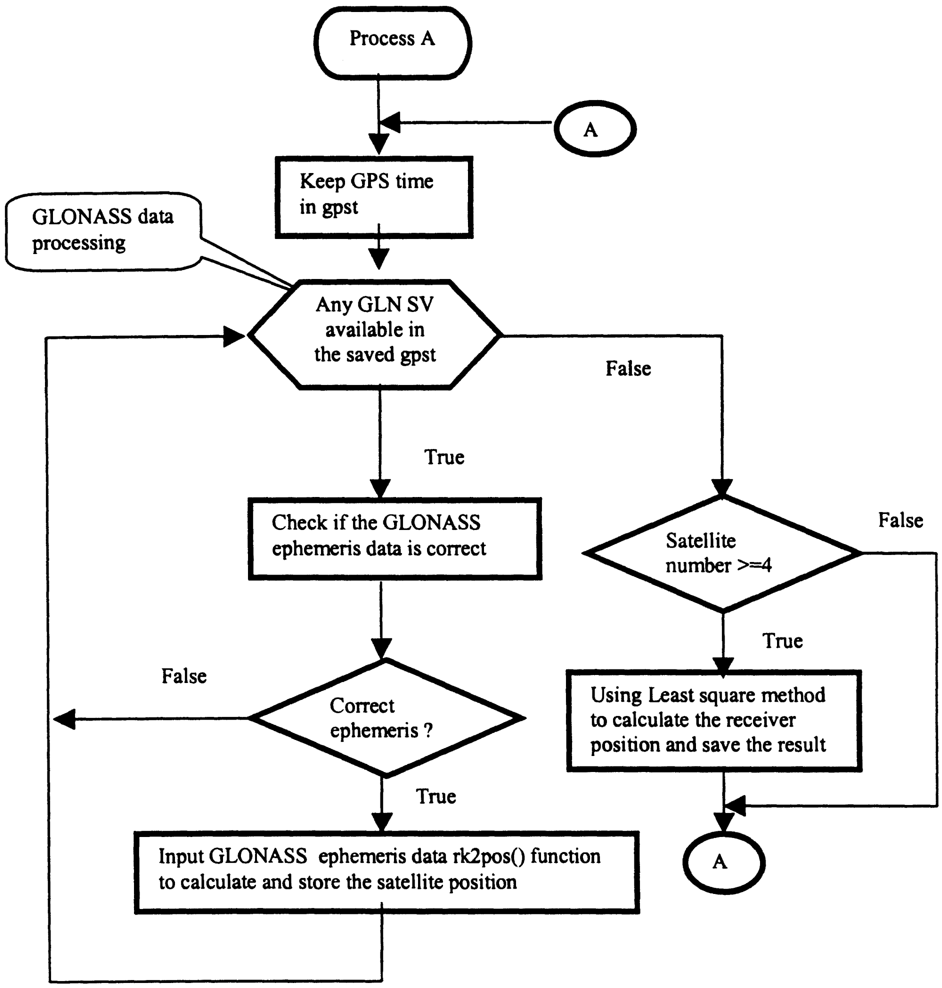


Figure C.2 GLONASS satellite Positions and User Position Calculations Flowchart

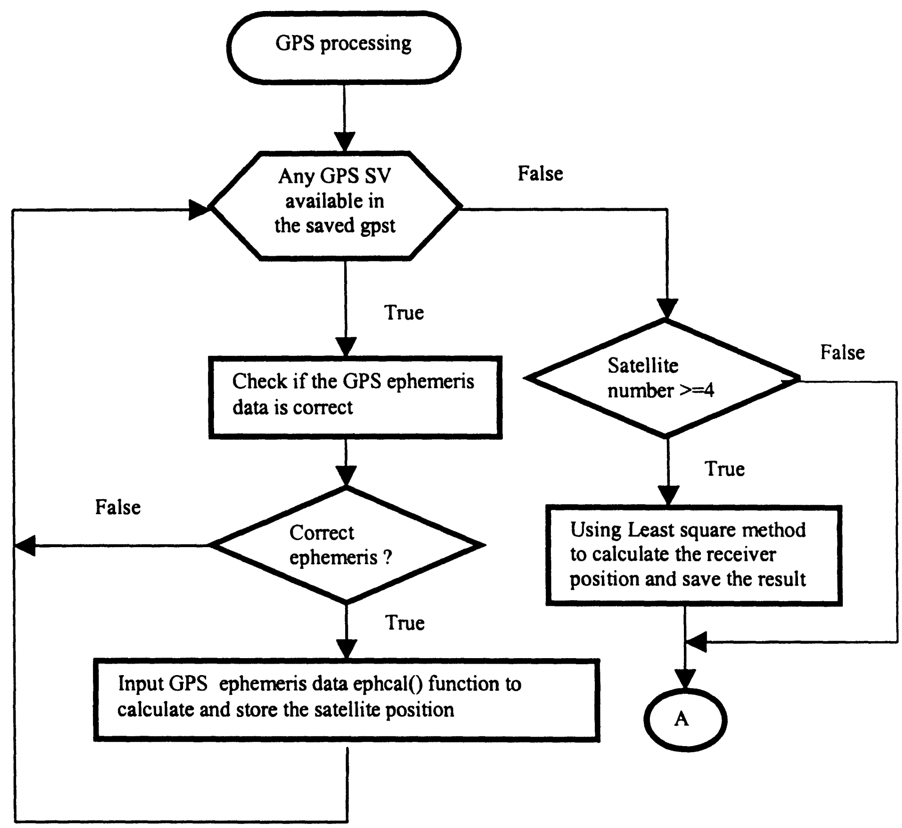


Figure C.3 GPS Satellite Positions Calculation Flowchart

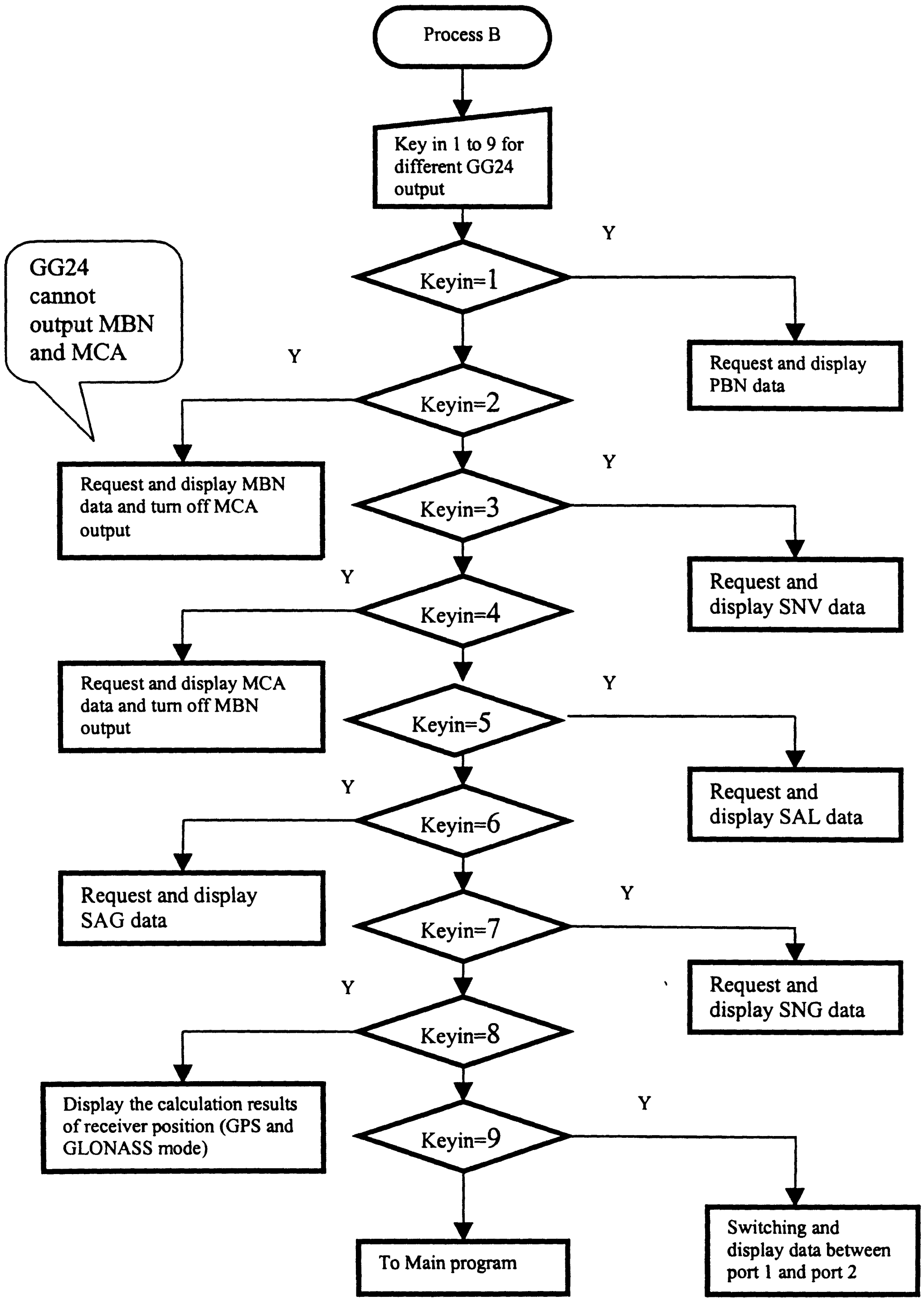


Figure C.4 Monitor Mode Flowchart

C.2 SOFTWARE LISTINGS (MATLAB)

```
%=================================================
% go.m
% main program for calculating SV/user positions
%
% weighted: 1,weighted matrix added; 0,non-weighted
% by CCH 4/20/1998
%=================================================

clear;clc;

%tic; % start stopwatch
light=2.99792458*10^8; % the speed of light
t0=clock;
from=10;to=-1; % for plots display           % input
srctmp=zeros(1,57);
var_in=zeros(1,56);
weighted=1;                                        % input
fprintf('Procedure loading data, processing ...\n');
gln_file='e:\thesis\ch6\data\m3271403.ch1';  % input
load (gln_file);
glnuser=m3271403;                                 % input
gps_file='e:\thesis\ch6\data\g3271403.ch1';  % input
load (gps_file);
gpsuser=g3271403;                                 % input
fprintf('Procedure loading data, done!\n');
t1=etime(clock,t0);
fprintf('Consuming time: %f (secs)\n\n',t1);

% load raw data from rawgln.c
% =================] Adjusting GLONASS data [===================
gate=1; % 1,able 0,disable
if gate~=0 & exist('glnuser')
   fprintf('Procedure GLONASS SVfilter, processing ...\n');
   SVin=(glnuser);
    prn_set=[35 36 41 42 50 52];
    keep_del=1; % 1,keep; 0,delete
    gps_gns=0;  % 1,GPS; 0,GLONASS
    mode=0;     % 1,this function on
                % 0,off( SVout= SVin )
                 % 2,skip this function (SVout=[])
   glnSVout=SVfilter(SVin,prn_set,keep_del,gps_gns,mode);
   fprintf('Procedure GLONASS SVfilter, done!\n');
   t1=etime(clock,t0);
   fprintf('Consuming time: %f (secs)\n\n',t1);

   %------------------------------------------------------------------------
   fprintf('Procedure GLONASS rawgln2pos, processing ...\n');
   mode=1; % 1,GG24 mix mode;  0:GG24 GLONASS only mode
   adjust=0; %1,shift adjust (in rawgln2posx.m) on; 0,off
   glnSVpos=rawgln2posf(glnSVout,adjust,mode); % convert GLONASS ephem.  to SV pos
                                        % "glnSVpos" formated as "gpsSVpos" below
   fprintf('Procedure GLONASS rawgln2pos, pdone!\n');
   t1=etime(clock,t0);
   fprintf('Consuming time: %f (secs)\n\n',t1);

    %------------------------------------------------------------------------
end
% =================] Adjusting GPS data [===================
gate=1; % 1,able 0,disable
if gate~=0 & exist('gpsuser');
   fprintf('Procedure GPS SVfilter, processing ...\n');
    SVin=gpsuser;
    prn_set=[2 4 7 9 14 15 19 27];
    keep_del=1; % 1,keep; 0,delete
    gps_gns=1;  % 1,GPS; 0,GLONASS
```

```
    mode=0;      % 0,off( SVout= SVin )
                 % 1,this function on
                 % 2,skip this function (SVout=[])
   gpsSVpos=SVfilter(SVin,prn_set,keep_del,gps_gns,mode);

   fprintf('Procedure GPS SVfilter, done!\n');
   t1=etime(clock,t0);
    fprintf('Consuming time: %f (secs)\n\n',t1);
end
%  format (GPS): [ID gpst prn pseudo(m) doppler SVx SVy SVz ck_off
%                      PBNx PBNy PBNz]

% ================] Combine GPS/GLONASS data [===================
% Combine glnSVout $ gpsSVout according to the gpst
gate=1; % 1,able 0,disable
if gate~=0
   fprintf('Procedure Combine, processing ...\n');
    if exist('glnSVpos') & exist('gpsSVpos')
        mode=1; % 0,disable; 1,enable
    cmbSVpos=Combine(glnSVpos,gpsSVpos,mode);
   end
   fprintf('Procedure Combine, done!\n');
   t1=etime(clock,t0);
    fprintf('Consuming time: %f (secs)\n\n',t1);
end

%---------------------------------------------------------------------------
mode=1; %enable this function
% gpsSVpos: using GPS SV only for user position calculation

if mode~=0 & exist('gpsSVpos')
   fprintf('Procedure GPS GNSSrcvrpos, processing ...\n');
   [gSVpos_azmele, gpsRcrPos]=GNSSrcvrpos(gpsSVpos,var_in,mode,weighted);
   fprintf('Procedure GPS GNSSrcvrpos, done!\n');
   t1=etime(clock,t0);
   fprintf('Consuming time: %f (secs)\n\n',t1);
end

%---------------------------------------------------------------------------
mode=1; %enable this function
% glnSVpos: using GLONASS SV only for user position calculation
if mode~=0 & exist('glnSVpos')
  fprintf('Procedure GLONASS GNSSrcvrpos, processing ...\n');
  [mSVpos_azmele, glnRcrPos]=GNSSrcvrpos(glnSVpos,var_in,mode,weighted);
  fprintf('Procedure GLONASS GNSSrcvrpos, done!\n');
  t1=etime(clock,t0);
  fprintf('Consuming time: %f (secs)\n\n',t1);
end

%---------------------------------------------------------------------------
mode=1; %enable this function
% cmbSVpos: using GPS and GLONASS SV for user position calculation
if mode~=0 & exist('cmbSVpos')
  fprintf('Procedure GNSS GNSSrcvrpos, processing ...\n');
  [cSVpos_azmele, cmbRcrPos]=GNSSrcvrpos(cmbSVpos,var_in,mode,weighted);
  fprintf('Procedure GNSS GNSSrcvrpos, done!\n');
  t1=etime(clock,t0);
  fprintf('Consuming time: %f (secs)\n\n',t1);
end

% output:
%          mode=1, RcrPos=[gpspos gpscmp]   11
%                  RcrPos=[glnpos glncmp]   11
%                  RcrPos=[gnsspos gnsscmp] 11
%          mode=0, disable this function
%
% format:(gpspos, glnpos, gnsspos)
```

```
%          [gpst  X  Y  Z  ck_offset   #_of_SV(:glnk,gpsk,gpnk) pdop] 7
%
%  (gpscmp, glncmp, gnsscmp):
%          [gpst PBNx PBNy PBNz] 4
%
%  (glnRcrPos, gpsRcrPos, cmbRcrPos ): (mode=1,2,3)
%     [gpst RcvrX RcvrY RcvrZ grcr_ck_off mrcr_ck_off glnk pdop gpst PBNx PBNy PBNz] 12

% ==================] Plot the results [===================
fprintf('Procedure results ploting, processing ...\n');
on_off=0; % 1,able 0,disable
if exist('cmbRcrPos') & on_off==1
   plot1(cmbRcrPos,from,to);
    title('GG24 position (GNSS)');
end
on_off=0; % 1,able 0,disable
if exist('cmbRcrPos') & on_off==1
   plot2(cmbRcrPos,from,to);
    title('GG24 position (GNSS) error');
end

% -----------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('gpsRcrPos') & on_off==1
   plot1(gpsRcrPos,from,to);
    title('GG24 position (GPS)');
end

on_off=0; % 1,able 0,disable
if exist('gpsRcrPos') & on_off==1
   plot2(gpsRcrPos,from,to);
    title('GG24 position (GPS) error')
end

% -----------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('glnRcrPos') & on_off==1
   plot1(glnRcrPos,from,to);
    title('GG24 position (GLONASS)');
end

on_off=0; % 1,able 0,disable
if exist('glnRcrPos') & on_off==1
   plot2(glnRcrPos,from,to);
    title('GG24 position (GLONASS) error');
end
% -----------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('gSVpos_azmele') & on_off==1
   plot4(gSVpos_azmele);
    text(-70,-120,'Satellite Azimuth and elevation angles (GPS)');
end
% -----------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('mSVpos_azmele') & on_off==1
   plot4(mSVpos_azmele);
    text(-80,-120,'Satellite Azimuth and elevation angles (GLONASS)');
end
% -----------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('cSVpos_azmele') & on_off==1
   plot4(cSVpos_azmele);
    text(-70,-120,'Satellite Azimuth and elevation angles (GNSS)');
end
% -----------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('gpsRcrPos') & on_off==1
```

```
   plot5(gpsRcrPos,from,to);
    text(-75,-250,'GG24 position error (GPS only)');
end
% ------------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('glnRcrPos') & on_off==1
   plot5(glnRcrPos,from,to);
    text(-90,-250,'GG24 position error (GLONASS only)');
end
% ------------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('cmbRcrPos') & on_off==1
   plot5(cmbRcrPos,from,to);
    text(-70,-250,'GG24 position error (GNSS)');
end
% ------------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('cmbRcrPos') & on_off==1
   plot6(cmbRcrPos,from,to);
    title('Position tracking (GNSS)');
end
% ------------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('gpsRcrPos') & on_off==1
   plot6(gpsRcrPos,from,to);
    title('Position tracking (GPS)');
end
% ------------------------------------------------------------------
on_off=0; % 1,able 0,disable
if exist('glnRcrPos') & on_off==1
   plot6(glnRcrPos,from,to);
    title('Position tracking (GLONASS)');
end
% ------------------------------------------------------------------

fprintf('Procedure results ploting, done\n');
%runtime=toc; % stop stopwatch
t1=etime(clock,t0);
fprintf('Consuming time: %f (secs)\n',t1);
% ========================] End of the main program [============================
```

```
% ==========================================================================
% GNSSrcvrpos.m
% Read a file generated from rawgln2pos.m
% [SVpos_azmele, RcrPos]=GNSSrcvrpos(SVpos,var_in,mode,weighted)
% SVpos: output of rawgln2pos.m
% format: (SVpos)
% ID gpst prn pseudo(m) acc.doppler  SVx  SVy  SVz  ck_offset pbnx pbny pbnz
%  1   2   3     4           5        6    7    8        9      10   11   12
%
% srctmp: keep sv data used in rcvr position cal.
%
% mode: 1,using GPS SV only for user position calculation
%        2,using GLONASS SV only for user position calculation
%        3,using GPS and GLONASS SV for user position calculation
%
% var_in: variance of each SV measurement (1x56)
% weighted: 1,weighted matrix added; 0,non-weighted
%
% output:
%          mode=1, RcrPos=[gpspos gpscmp]   12
%                  RcrPos=[glnpos glncmp]   12
%                  RcrPos=[gnsspos gnsscmp] 12
%          mode=0, disable this function
%
% format:(gpspos, glnpos, gnsspos)
%          [gpst  X  Y  Z  g_ck_offset m_ck_offset #_of_SV(:glnk,gpsk,gpnk) pdop] 8
%
%  (gpscmp, glncmp, gnsscmp):
%          [gpst PBNx PBNy PBNz] 4
%
%  (RcrPos): (mode=1,2,3)
%     [glnpos glncmp]
%
% 4/15/1998   by CCH
% ==========================================================================

function [SVpos_azmele, RcrPos]=GNSSrcvrpos(SVpos,var_in,mode,weighted)
light=2.99792458*10^8; % the speed of light
SVpos_azmele=[];
r1=[0 2.5 0]';
r2=[1 -1.9*10^-6 0
      1.9*10^-6 1 0
      0 0 1];
initpos=[678444.36 -4893794.642 4020485.636 0 0]; % GHU Antenna pos.

if mode~=0 % GLONASS only mode
    [m,n]=size(SVpos);
    %m=100; % testing only (first m data)
    t_start=SVpos(1,2); t_end=SVpos(m,2);
    SVpos(m+1,2)=-1;   % end of the data file
    k=1;
    glnk=0; % # of SV used to cal.  pos.
    glnr=1; % # of calculated user position (GLONASS)

   gln_initpos=initpos; % raddom init.  user pos.  [1x5]
    i=1; % counter: 1 to m (the length of SVpos)

    while i<m
       gpst=SVpos(i,2); % save gps time
       existGPS=0;
         existGLN=0;
    while SVpos(i,2)==gpst
        glnk=glnk+1;
          glndat(glnk,:)=SVpos(i,:);
          if glndat(glnk,3)<=32 % GPS SV
             existGPS=1;
          end
```

```
    if glndat(glnk,3)>32 % GLONASS SV
       existGLN=1;
    end

    i=i+1;
    if i>m
      break;
    end
 end

 if glnk>=4+(existGPS*existGLN) % at least 5 SV for GPS+GLONASS combination
    glnpos(glnr,:)=rcvrpos(glndat,glnk,gln_initpos,var_in,0); % non-weighted
    % output format of function rcvrpos.m (glnpos):
    %       [gpst Rcvr_est(1:5) nSV pdop] 8
    % "glndat" input format:(for GPS or GLONASS), dat[nSV][12]
    % [ID gpst prn pseudo(m) acc.doppler  SVx  SVy  SVz  ck_offset PBx PBy PBz]

    if weighted~=0
       for pk=1:glnk
          prn=glndat(pk,3);
              SVxyz=glndat(pk,6:8);
          RCRxyz=glnpos(glnr,2:4);
              dist(pk)=sqrt(sum((SVxyz-RCRxyz).^2));
              pseudo(pk)=glndat(pk,4); %(raw+dt)*VLIGHT in QNX/c
          sv_ck_offset(pk)=glndat(pk,9); % sv dt in QNX/c
          g_rcr_ck_offset=glnpos(glnr,5); % ck_offset for gps
          m_rcr_ck_offset=glnpos(glnr,6); % ck_offset for glonass
          dis_err(pk)=dist(pk)-pseudo(pk);
          if prn>32 % GLONASS
             rcr_ck_offset=m_rcr_ck_offset;
          else   % GPS
             rcr_ck_offset=g_rcr_ck_offset;
          end
          %ck(pk)=rcr_ck_offset;
          dis_err_comp(pk)=dis_err(pk)+rcr_ck_offset;
          if prn<=32 % or prn<=32
             var_in(prn)=sqrt(dis_err_comp(pk)^2); % GPS
          else
             var_in(prn)=10; % GLONASS (forced)
          end
       end
       glnpos(glnr,:)=rcvrpos(glndat,glnk,gln_initpos,var_in,weighted); % weighted
       end
   gln_initpos=glnpos(glnr,2:6); % init.  rcvr pos(t+1)=rcvrpos(t)
   glncmp(glnr,:)=[gpst glndat(1,10:12)];
   % new Rcvr X,Y,Z,ck_offset as init.  pos
   glnr=glnr+1;
  elseif exist('glnpos')
     pdop=-1;
   glnpos(glnr,:)=[gpst gln_initpos glnk pdop];
   if glnk > 0
           glncmp(glnr,:)=[gpst glndat(1,10:12)];
   else
        glncmp(glnr,:)=[gpst gln_initpos(1:3)];
   end
   glnr=glnr+1;
  % SV < 4, keep init.  Rcvr X, Y, Z, ck_offset
end

  if glnk>0
     if exist('glnpos')
        orgece=glnpos(glnr-1,2:4);
     else
        orgece=gln_initpos(1:3);
     end
    for w=1:glnk
      svece=glndat(w,6:8);
```

```
         azele(w,:)=azmelev(svece, orgece);
         app(w,:)=[glndat(w,:) azele(w,:)];
           end
        SVpos_azmele=[SVpos_azmele app(1:glnk,:)'];
        % SVpos_azmele=[ID gpst prn pseudo(m) acc.doppler  SVx  SVy  SVz
        %                  ck_offset pbnx pbny pbnz azm ele]
      end

    glnk=0; % reset # of SV for LSM
    end

    if exist('glnpos')
    glnpos=glnpos(6:length(glnpos),:); % discard first 5 sec
    glncmp=glncmp(6:length(glncmp),:);
   end
   % glnpos: [gpst RcvrX RcvrY RcvrZ g_ck_off m_ck_off glnk pdop] 8
   % glncmp: [gpst PBNx PBNy PBNz] 4
   RcrPos=[glnpos glncmp];
   % output: (RcrPos), mode=1,2,3
   %     [gpst RcvrX RcvrY RcvrZ ck_offx2 glnk pdop gpst PBNx PBNy PBNz] 12
   SVpos_azmele=SVpos_azmele';
   %            (SVpos_azmele):
   %     [ID gpst prn pseudo(m) acc.doppler  SVx  SVy  SVz ck_offset pbnx pbny pbnz azm
ele]
end
```

```
% ==========================================================
% combine.m
% combSVset=combine(SVset1, SVset2, mode)
% combine SVset1 and Svset2 according to the gps time
% mode: 0,disable; 1,enable
%
%  format: (SVset1, SVset2, combSVset)
%             [ID gpst prn pseudo(m) doppler SVx SVy SVz ck_off
%              PBNx PBNy PBNz]
%
% 4/16/1998 by CCH
% ==========================================================

function combSVset=combine(SVset1, SVset2, mode)

if mode==1
    [m1,n1]=size(SVset1);
    gpst=SVset1(1,2);
    i1=1;
   k=0;
   while i1<m1
        while gpst==SVset1(i1,2) & i1<m1
        k=k+1;
        combSVset(k,:)=SVset1(i1,:);
        i1=i1+1;
    end

      [m2,n2]=size(SVset2);
        entered=0;
        mark1=-1;
        for p=1:m2
        if gpst==SVset2(p,2)
            if mark1==-1
                mark1=p; % beginning of deleation
            end
            entered=1;
            k=k+1;
            combSVset(k,:)=SVset2(p,:);
        elseif gpst~=SVset2(p,2) & entered==1
            entered=2;
            mark2=p-1; % end of deleation
           end

        if entered==2
        SVset2(mark1:mark2,:)=[];
            break;
        end
    end
    gpst=SVset1(i1,2);
   end
else % mode=0
   combSVset=[];
end
```

```
% GLONASS ephermis function
% glnfun.m

function xyz_pv=glnfun(t,xyz)

    global ddx ddy ddz;
    u=      398600.44;  % km^3/s^2
    ae=     6378.136;
    C20=   -1082.63*10^(-6);
    w3=     0.7292115*10^(-4);

    x(1)=xyz(1);    % pos
    x(2)=xyz(4);    % vol
    y(1)=xyz(2);
    y(2)=xyz(5);
    z(1)=xyz(3);
    z(2)=xyz(6);

    r=sqrt(x(1)^2+y(1)^2+z(1)^2);

    xdot(1)=x(2);
    xdot(2)=(-u/r^3)*x(1)+3/2*C20*u*ae^2/r^5*x(1)*(1-5*z(1)^2/r^2)+...
        w3^2*x(1)+2*w3*y(2)+ddx;

    ydot(1)=y(2);
    ydot(2)=(-u/r^3)*y(1)+3/2*C20*u*ae^2/r^5*y(1)*(1-5*z(1)^2/r^2)+...
        w3^2*y(1)-2*w3*x(2)+ddy;

    zdot(1)=z(2);  % position
   zdot(2)=(-u/r^3)*z(1)+3/2*C20*u*ae^2/r^5*z(1)*(3-(5*z(1)^2/r^2))+ddz;


   xyz_pv=[xdot(1) ydot(1) zdot(1) xdot(2) ydot(2) zdot(2)]';
```

```
% ======================================================================
% rawgln2posf.m
% SVpos=rawgln2pos(gnsraw,adjust,mode)
% gnsraw: datas from mxxxxxxx.chx filtered by SVfilter.m = SVout
% mode: 1,GG24 mix mode; 0,GG24 GLONASS only mode
% %1,shift adjust (in rawgln2posx.m) on; 0,off
%
% data from rawgln.c (QNX) filteded by SVfilter.m
% input GLONASS ephem.  data
%       format: (gnsraw)
%           [gpst prn raw(m) x y z dx dy dz ddx ddy ddz(km) tb
%            x y z(rcvr, pbn, m) acc_doppler sv_ck_offset tn tc gn ele azi]
%
% output SV positions & checking the distance error in Tb
% format: (glnsrc)
%         [ID gpst prn pseudo(m) doppler SVx SVy SVz ck_off PBNx PBNy PBNz]
% 4/15/1998 by CCH
% ======================================================================

function SVpos=rawgln2posf(gnsraw,adjust,mode)

    global ddx ddy ddz;
   if mode==1
      leap=12; % GG24 mix mode
   end
   if mode==0
      leap=0; % GG24 GLONASS mode
   end

   shift=-28.6/1000000;
   light=2.99792458*10^8;
   r1=[0 2.5 0]';          % PZ90 -> WGS84
   r2=[1 -1.9*10^-6 0  % when mode=0
      1.9*10^-6 1 0       % (GLONASS mode only, PZ90 coord.)
      0 0 1];

   svpos(150,7,24)=0; % 150 tl/nxyzt, t&nxyz(7), 24 sv
   srcinfo(3,24)=0; % length of ode45 output(p), tmax, tmin for 24 sv

   [m,n]=size(gnsraw);

   glnr=0;
   ID=0;

   %m=100;  % testing only

   for k=1:m
     gpst=gnsraw(k,1);
     prn=gnsraw(k,2);
     if adjust==1 % adjust
         if prn==35
         shift=1;
          end
         if prn==36
         shift=-1;
         end
         if prn==41
         shift=0.6;
         end
         if prn==52
         shift=-0.7;
         end
      end

     glnt=rem(rem(gpst,86400)+3*60*60+shift-leap,86400); % glnt
     tb=gnsraw(k,13);
     doppler=gnsraw(k,17);
```

```
ck_off=gnsraw(k,18);
tn=gnsraw(k,19);
tc=gnsraw(k,20);
gn=gnsraw(k,21);
raw=gnsraw(k,3);
pseudo=raw*light;
pnxyz=gnsraw(k,14:16);

if glnt>srcinfo(2,prn-32) | glnt<srcinfo(3,prn-32)
    xyz=gnsraw(k,4:9); % km
   ddx=gnsraw(k,10); ddy=gnsraw(k,11); ddz=gnsraw(k,12); % km
   tspan1=[tb tb-920];
   [t1,nxyz1]=ode45('glnfun',tspan1,xyz);
   nxyz1=flipud(nxyz1);
   t1=flipud(t1);
   tspan2=[tb tb+920];
   [t2,nxyz2]=ode45('glnfun',tspan2,xyz);
   m2=length(t2);
   tt=[t1' t2(2:m2)'];
   nxyzt=[nxyz1
          nxyz2(2:m2,:)];
   t_pos=[tt' nxyzt];
   [p,q]=size(t_pos);
   svpos(:,:,prn-32)=0;
   svpos(1:p,1:q,prn-32)=t_pos;
   srcinfo(1,prn-32)=p;
   srcinfo(2,prn-32)=max(tt);
   srcinfo(3,prn-32)=min(tt);
end
p=srcinfo(1,prn-32);
t1=svpos(1:p,1,prn-32);
nxyz1=svpos(1:p,2:7,prn-32);
ti=glnt-raw;
if ti>max(t1) | ti<min(t1)
   fprintf('Possible ephemeris error\n !');
else
     y=interp1(t1,nxyz1,ti,'linear');
   svxyz=y(1:3)*1000.0;% y= [x y z dx dy dz]
   glnr=glnr+1;
   glnsrc(glnr,:)=[ID gpst prn pseudo doppler svxyz ck_off pnxyz];
   % [ID gpst prn pseudo(m) doppler SVx SVy SVz ck_off PBNx PBNy PBNz]12
end
SVpos=glnsrc;
end
```

```
% ========================================================================
% pos=rcvrpos(dat,nSV,initpos,var,weighted) : finding receiver position
%
% input format:(for GPS or GLONASS), [nSV][12]
% ID gpst prn pseudo(m) acc.doppler  SVx  SVy  SVz  ck_offset PBx PBy PBz
%  1   2   3    4           5         6    7    8       9      10  11  12
% nSV: # of SV for LSM (>=4)
% initpos: init.  SV position and ck offset, ex: [0 0 0 0 0]
% var: variance of each SV measurement (1x32), (1,x)=var, x=1~32
% mode: 1, weighted; 0, non-weighted
%
% Testing: [pos,ure]=rcvrpos(testdat,4,[0 0 0 0 0])
% "pos" output format: [gpst X Y Z ck_offset nSV]
% 4/24/1998
% ========================================================================

function pos=rcvrpos(dat,nSV,initpos,var,weighted)
    err_range=1e-03;  % error range for recursive loop
   light=2.99792458*10^8; % the speed of light
    Rcvr_est=initpos; % [1x5]
   gpst=dat(1,2); % gpst
   pdop=-1;

   if (nSV>=4) % Check if the # of SV >= 5
     W=zeros(nSV,nSV);
     for k=1:nSV
       if var(dat(k,3))~=0
           W(k,k)=1/var(dat(k,3));
       else
           W(k,k)=0;
       end
         % "glndat" input format:(for GPS or GLONASS), dat[nSV][12]
         % [ID gpst prn pseudo(m) acc.doppler  SVx  SVy  SVz  ck_offset PBx PBy PBz]

         prn=dat(k,3);
       SV(k,:)=[prn dat(k,6:8)]; % SV=[prn SVx, SVy, SVz]
       PR(k)=dat(k,4); % pseudo(m)
       if prn<=32
          rcr_ck_offset=Rcvr_est(4);
       else
          rcr_ck_offset=Rcvr_est(5);
       end
       % Rcvr_est(4)=dat(1,9);
       PR_est(k)=sqrt(sum((SV(k,2:4)-Rcvr_est(1:3)).^2))+rcr_ck_offset;
         del_PR(k)=PR(k)-PR_est(k);
     end
%-------------------------------------------------------------
% The true range and estimate range from Rcvrs to satellites

       endloop=0;
      loopcnt=0;
      existGPS=0; % checking if GPS SV exist
      existGLN=0; % checking if GLONASS SV exist
       while endloop~=1 & loopcnt<10   % == the begining of the recursive loop

         loopcnt=loopcnt+1;
         for i=1:nSV       % H matrix
            for j=1:3
              H(i,j)=(Rcvr_est(j)-SV(i,j+1))/(PR_est(i)-rcr_ck_offset);
             if SV(i,1)<=32 % GPS prn
                H(i,4)=1; % GPS rcvr_ck_offset
                H(i,5)=0; % GLONASS rcvr_ck_offset
                existGPS=1;
             else
                H(i,4)=0;
                H(i,5)=1;
                existGLN=1;
```

```
            end
          end
       end

       [hm,hn]=size(H);
       if existGLN==0
          H=H(:,1:hn-1);
          Rcvr_est=Rcvr_est(1:4);
       end

      if existGPS==0
          H(:,4)=[];
          Rcvr_est(4)=[];
      end

% Solve least squares solution
        if weighted==0 % non-weighted
           [hm,hn]=size(H);
          W=zeros(hm,hm);
          for w=1:hm
             W(w,w)=1; % identify matrix
         end
       end

         dopmatrix=inv(H'*H);
       pdop=sqrt(dopmatrix(1,1)+dopmatrix(2,2)+dopmatrix(3,3));
       deltaXYZ=inv(H'*W*H)*H'*W*del_PR';

        eval_del(loopcnt)=sum(deltaXYZ.^2);
        endloop=abs(eval_del)<=err_range; % to check if the calculated
                                            % position is close enough

       if existGLN==0 % GLONASS SV not exist
          deltaXYZ=deltaXYZ';
          Rcvr_est=Rcvr_est+deltaXYZ;
          Rcvr_est=[Rcvr_est -1000];
       elseif existGPS==0 % GPS SV not exist
          deltaXYZ=deltaXYZ';
          Rcvr_est=Rcvr_est+deltaXYZ;
          Rcvr_est=[Rcvr_est(1:3) -1000 Rcvr_est(4)];
         else
          Rcvr_est=Rcvr_est+deltaXYZ';
       end

       if prn<=32
          rcr_ck_offset=Rcvr_est(4);
       else
          rcr_ck_offset=Rcvr_est(5);
       end

        for i=1:nSV
          PR_est(i)=sqrt(sum((SV(i,2:4)-Rcvr_est(1:3)).^2))+rcr_ck_offset;
           del_PR(i)=PR(i)-PR_est(i);
        end

    end         % == End of the recursive loop
      pos=[gpst Rcvr_est nSV pdop]; %8
   else
      pos=[gpst Rcvr_est nSV pdop];
    end
```

```
% ==========================================================================
% SVfilter.m
% Keeping or take off SV according to SV prn
% SVout=SVfilter(SVin,prn,keep_del,gps_gns,mode)
% SVin: input source data
%       format (GLONASS): [gpst prn raw X Y Z dx dy dz ddx ddy ddz tb
%                                    PBNx PBNy PBNz doppler ck_off tn tc gn ele azi]
%                                        (23)
%         format (GPS): [ID gpst prn pseudo(m) doppler SVx SVy SVz ck_off
%                               PBNx PBNy PBNz]
%                                        (12)
% prn: [sv#1, sv#2, ...] for keeping or take_off
% keep_del: 1,keep; 0,delete
% gps_gns: 1,SVin is GPS; 0,SVin is GLONASS data
% mode: 1,doing this function;
%         0,disable this function (SVout=SVin)
%       2,skip this function (SVout=[])
%
% 4/15/1998
% by CCH
% ==========================================================================

function SVout=SVfilter(SVin,prn,keep_del,gps_gns,mode)

if mode==1 % activate this function
   if gps_gns==1 % GPS
      if keep_del==1 % keep SV
         p=length(prn); % which SVs for delete
         q=0; % SVout, SV counter
         [m,n]=size(SVin);
         for k=1:m
            for j=1:p
                if SVin(k,3)==prn(j)
                   q=q+1;
                   SVout(q,:)=SVin(k,:);
                   break;
                end
             end
         end

        else % delete SV
         p=length(prn); % which SVs for delete
         q=0; % SVout, SV counter
            [m,n]=size(SVin);
         for k=1:m
             pass=1;
            for j=1:p
                if SVin(k,3)==prn(j)
                   pass=0;
                end
             end

             if pass~=0
               q=q+1;
               SVout(q,:)=SVin(k,:);
             end;
         end
        end  % delete SV

    %=======================================================
    else % GLONASS
       if keep_del==1 % keep SV
          p=length(prn); % which SVs for delete
          q=0; % SVout, SV counter
          [m,n]=size(SVin);
          for k=1:m
             for j=1:p
```

```
               if SVin(k,2)==prn(j)
                  q=q+1;
                  SVout(q,:)=SVin(k,:);
                  break;
               end
            end
         end

        else % delete SV
         p=length(prn); % which SVs for delete
         q=0; % SVout, SV counter
         [m,n]=size(SVin);
         for k=1:m
            pass=1;
           for j=1:p
               if SVin(k,2)==prn(j)
                  pass=0;
               end
            end

            if pass~=0
              q=q+1;
              SVout(q,:)=SVin(k,:);
            end;
         end
      end  % delete SV
   end
elseif mode==0
   SVout=SVin; % this function disabled
else % mode=2
   % skip this function
   SVout=[];
end
```