Department of Geomatics Engineering

Precise Relative Positioning of Multiple Moving Platforms Using GPS Carrier Phase Observables

(URL: http://www.geomatics.ucalgary.ca/GradTheses.html)

By

Ning Luo

January 2001

UNIVERSITY OF
CALGARY

ABSTRACT

Precise relative positioning of multiple moving platforms using GPS carrier phase observables has numerous applications. The essential point for this research is the fast and reliable OTF carrier phase ambiguity resolution. Algorithms for single baseline resolution cannot provide optimal performance for this application because it does not make use of the redundancy available in the configuration of multiple moving platforms.

In this thesis, a novel method called MultiKin is proposed for OTF ambiguity resolution for multiple moving platforms. First, MultiKin applies Delaunay triangulation to select necessary baselines and to build an optimal structure of ambiguity constraints. Second, it improves the reliability of the OTF ambiguity resolution of single-baselines by optimizing the ambiguity monitoring algorithm. Finally, the resolved ambiguities from each baseline are processed using the multiple triangular constraints, which can speed up fixing ambiguity and detecting wrong fixes.

To fully evaluate the performance of MultiKin, a sophisticated GPS software simulator is developed. Its significance lies in a GPS error simulator. New GPS error models are built based on the investigation of the existing models. The spatial correlation and temporal variation of errors are highlighted in the new models; thus, these models are proper for error simulation in both single-point and differential GPS systems. In addition, all the models have adjustable parameters that allow users to generate a wide range of testing conditions.

The results of extensive simulation tests and field tests with MultiKin indicate that MultiKin is effective in speeding up ambiguity resolution. The time required to fix ambiguities can be reduced by up to 67% over the single-baseline method time. Also, MultiKin increases the limit on the distance by two to three kilometres over which ambiguity resolution can be performed. An increased magnitude of the GPS errors and weaker satellite visibility can degrade the efficiency improvement of MultiKin, but it can fix more baselines than the single-baseline method even under those critical conditions. Besides improving efficiency, MultiKin can also provide higher reliability in ambiguity

resolution. Its time to detect wrong fixes is reduced by up to 29% over the single-baseline method. Consequently, MultiKin increases the confidence that positioning is precise.

ACKNOWLEGEMENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AFM	Ambiguity Function Method
AR	Auto-Regressive
bps	Bits per second
CDMA	Code Division Multiple Access
CDPD	Cellular Digital Packet Data
CSMA	Carrier Sense Multiple Access
CSMA/CD	Carrier Sense Multiple Access / Collision Detection
DC	Doppler Change
DD	Double Difference
DGPS	Differential Global Positioning System
DLL	Delay Lock Loop
DOP	Dilution of Precision
FARA	Fast Ambiguity Resolution Algorithm
FASF	Fast Ambiguity Search Filter
FDMA	Frequency Division Multiple Access
GIMs	Global Ionosphere Maps
GPS	Global Positioning System
HDOP	Horizontal Dilution of Precision
IA	Ionospheric Acceleration
LAMBDA	Least squares AMBiguity Decorrelation Adjustment
LL	Local Level frame
LSB	Longest Solvable Baseline
LT	Local Time
NUB	Number of Unfixed Baselines
OTF	On-The-Fly
pdf	Probability density function
PDOP	Position Dilution of Precision
ppm	Parts per million
PRN	Pseudo Random Noise
PSD	Power Spectral Density
RMS	Root Mean Squares
RF	Radio Frequency
SA	Selective Availability
SF	Scale Factor
SPHA	SPherical HArmonics

SPS	Standard Positioning Service
TAF	Time to Ambiguity Fixed
TDMA	Time Division Multiple Access
TDW	Time to Detect Wrong fixes
TEC	Total Electron Content
TECU	Total Electron Unit 10^{16}el/m^2
TTAF	Time to True Ambiguity Fixed
UHF	Ultra High Frequency
UT	Universal Time
VDOP	Vertical Dilution of Precision
VHF	Very High Frequency
VIA	Vertical Ionospheric Acceleration
VIV	Vertical Ionospheric Velocity
VTEC	Vertical Total Electron Content
WAAS	Wide-Area Augmentation System
WL	Widelane

1 INTRODUCTION

1.1 BACKGROUND

The Global Positioning System (GPS) is a RF satellite-based navigation system that was developed by the United States Department of Defense as a replacement for the Transit Navy Navigation Satellite System. GPS was designed to enhance the coverage and accuracy of Transit and the first GPS operational satellite was launched on Feb 22, 1978. The system became fully operational in 1994 (Parkinson, 1996).

A Standard Positioning Service (SPS) is provided by GPS for civilian users. Prior to May 2000, The Selective Availability (SA) was on and the horizontal 2 DRMS positioning accuracy was 100 m and the vertical (95% level) accuracy was150 m. Although SA has been switched off since May 1, 2000, the ionospheric error can still limit the accuracy of the SPS to a few tens of metres when the solar activity is very high. To improve relative positioning accuracy for civilian users, the differential GPS (DGPS) method is usually used.

The purpose of DGPS is to transmit the spatial and temporal correlated corrections from a fixed or mobile reference station to adjust the measurements of a rover station, which can be either fixed or mobile. The DGPS method is often used to reduce errors in the code measurements, and the resulting accuracy is of the order of one to a few metres. The differential method can also be used for carrier phase positioning. When the GPS carrier phase of a rover station is compared to the carrier phase at a reference site, the equivalent range measurement accuracy is within a few percent (typically a few centimetres) of the carrier wavelength. Thus, the positioning accuracy can be improved to the centimetre level if other errors cancel out.

This accurate positioning technique has numerous applications, such as precise surveying, deformation monitoring, attitude determination of a vehicle, precision landing of aircraft, integrated navigation systems, atmospheric parameter estimation and solar activity monitoring. The endless demand from users for higher accuracy and reliability

drives the continuous improvement of these techniques. High-accuracy positioning using GPS carrier phase observables is a GPS research topic.

1.2 STATEMENT OF PROBLEM

Differential carrier phase is a relative positioning technique, where only the inter-platform position vector between a reference and a rover station can be directly derived from the carrier phase observables. The accurate absolute position of a rover station still relies on the accurately known coordinates of the reference station. Nowadays, in a lot of applications, relative positioning rather than absolute positioning is the main concern. For most non-professional GPS users, the absolute coordinates of an object, namely the latitude, longitude, and altitude in the World Geodetic System 1984 (WGS84) cannot give them a simple understanding of the object's location. However, if its location is given in a relative way, such as 500 metres north and 1000 metres west of an established reference, the user can more easily make a connection with the location of the object.

The present research is related to the technique of precise (centimetre to sub-decimetre level) positioning of multiple moving platforms relative to each other or to some fixed reference stations. This technique can simultaneously process GPS measurements from multiple moving platforms and can achieve a fast and precise estimation of the relative positions of these platforms. This research has numerous applications; for instance, fleet management, traffic control, deformation monitoring of large moving vehicles, numerous military applications, etc. Another example, illustrated in Figure 1.1, is GPS 'radar'. Different from the usual radar, the GPS radar technique does not obtain positions of surrounding objects from sounding and echo detection, but from the wireless data link. The surrounding object must be equipped with both a GPS receiver and data link. After receiving position data of other stations, one station can clearly derive the spatial distribution of surrounding objects. Therefore, GPS radar can give the same results as a usual radar, but at a lower cost. The GPS radar technique is currently being developed for the safe approach and landing of aircraft, collision avoidance of vehicles, automatic driving, etc (Heppe, 2000).

The research of positioning multiple moving platforms has the following characteristics:

[1] In this application, the absolute positions of the objects are not important but rather their relative positions, so that the configuration of the reference station with precisely known coordinates is not mandatory;

[2] High relative positioning accuracy and reliability are required;

[3] There are multiple platforms in the configuration, which implies that the multiplicity of platforms may improve the effectiveness of relative positioning.
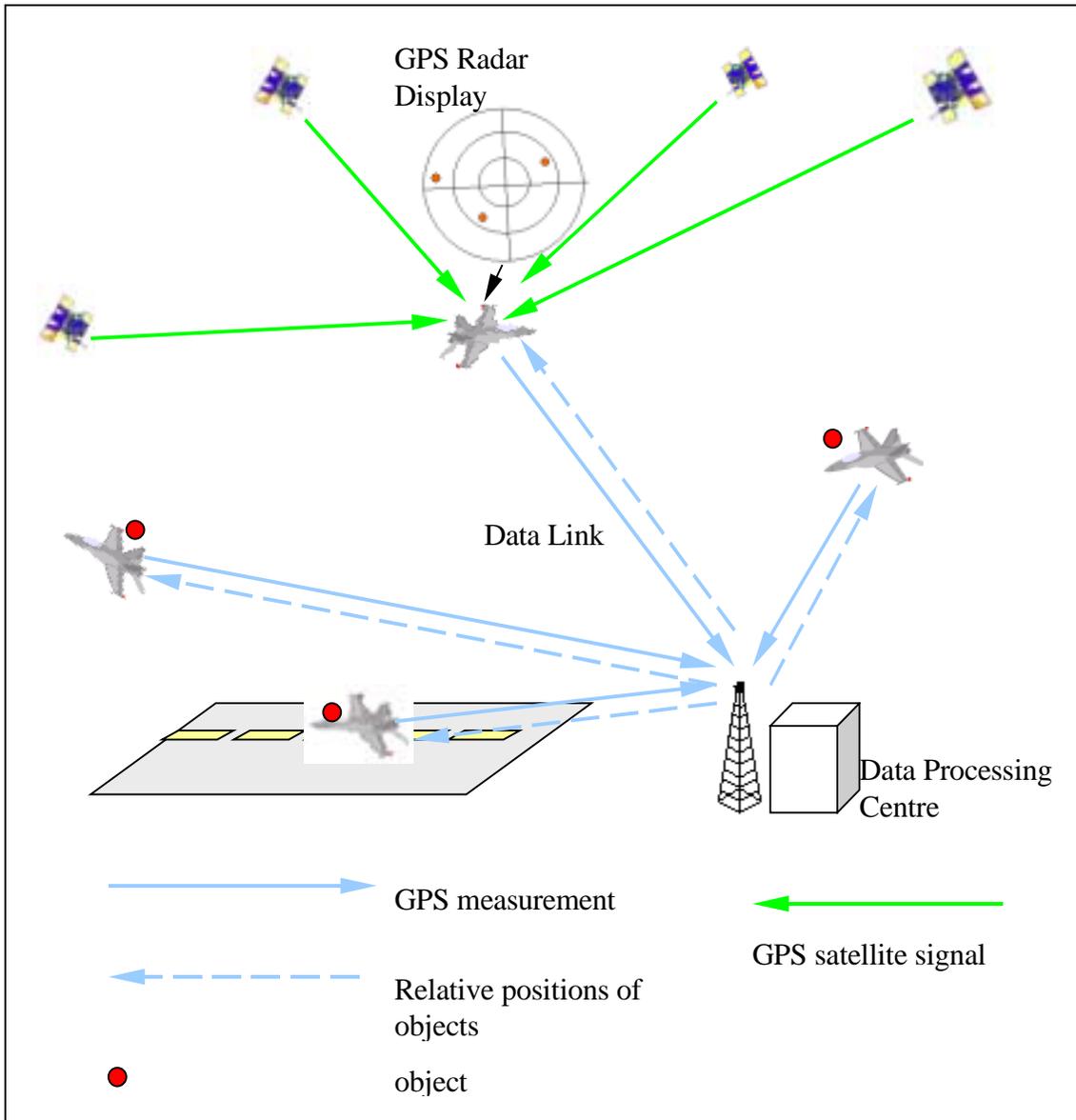
Figure 1.1 GPS radar for aeronautical application

For the first point, most of the existing methods of relative positioning assume that the precise position of a reference station is given *a priori*. Thus, the accuracy of the relative positioning only depends on the measurement errors. However, in this research, such a precondition is not given. Therefore, those previous approaches cannot be directly used for this research. Modifications are required to process the kinematic data simultaneously. The impact of these modifications on the effectiveness of relative positioning is to be investigated in this thesis.

To achieve high positioning accuracy, the double-differenced GPS carrier phase method is usually adopted. When the inter-platform distances are short, e.g., less than 10 km, double differencing can largely reduce spatially correlated errors in the carrier phase measurements. Satellite and receiver clock errors are cancelled, regardless of the inter-platform distance. When the remaining errors are small, centimetre-level accuracy relative positions can be obtained with fixed integer ambiguities.

An integer ambiguity is an unknown whole-cycle constant that occurs when integrating the beat frequency between the received GPS carrier and the replicated carrier in a GPS receiver. High accuracy relative positioning depends mostly on successful integer ambiguity resolution of the double-differenced carrier phase measurements. Only after the ambiguities are fixed to the correct integers can centimeter-level accuracy be achieved. Therefore, integer ambiguity resolution is one of the crucial problems to resolve for high accuracy relative positioning.

There are many methods that have been developed for solving On-The-Fly (OTF) ambiguity since the 1980's. OTF ambiguity resolution refers to the case when the ambiguities are resolved when at least one receiver is moving, i.e., when the receiver is in kinematic mode. The major challenges of OTF ambiguity resolution are relative error modeling, and the efficiency and reliability of the ambiguity search technique. Earlier work focused on improving ambiguity resolution on a single-baseline, such as optimal filtering and reduction of the ambiguity correlation. Later, more effort was put into using constraints for further improvement. Constraints are generally provided by the geometric information existing in the configuration of multiple platforms, or other known

information such as the height difference between two receivers. Recent results (Weisenburger, 1997; Raquet, 1998; Luo and Lachapelle, 1999) using constraints have shown significant improvement in level of success and time to integer ambiguity resolution. The research in this thesis will include investigations into the optimization of ambiguity resolution using constraints for the case of multiple moving platforms (more than two). The final objective is to develop a fast, reliable, and realizable approach for precise relative positioning of multiple moving platforms that could ultimately be implemented in real-time. The impact of the number of platforms simultaneously available is to be addressed.

To thoroughly test the effectiveness of the approach to position multiple moving platforms, a series of tests must be performed to evaluate the impact of various system parameters on this approach. It is however difficult to assess performance using full-scale field tests due to the difficulties in reproducing the wide range of differential errors that may prevail under different atmospheric conditions, and in establishing accurate reference trajectories for the mobile receivers. The use of controlled simulated scenarios is therefore preferred for this purpose. Actual field tests will still be performed however to demonstrate the effectiveness of the method under actual field conditions.

A major problem faced by the use of simulations is the validity of the differential GPS error models. Various GPS error models are in existence, however most of them are designed for single-point positioning. After differencing, the simulated errors can show quite different behaviors compared to the real situation. In these models, either the spatial and temporal correlations are not properly simulated or the achieved resolution is not high enough for testing a small GPS network. Therefore, the development of sophisticated DGPS error models forms another important part of this thesis in order to enhance the confidence of the simulation tests.

1.3   RELATED RESEARCH

1.3.1   OTF Ambiguity Resolution

OTF integer ambiguity resolution is the heart of the algorithm to position multiple moving platforms, thus, the highlights of related research are summarized as follows.

1.3.1.1 OTF Ambiguity Resolution Between Two Receivers

This is the foundation of all ambiguity resolution techniques. Generally, this sort of method includes two common procedures. The first is to define the ambiguity search range, the second is to select the best solution from all possible candidates. The ambiguity uncertainty is reduced by using optimal filtering techniques, conditional least squares and matrix transformation, etc. These techniques have greatly improved the efficiency of OTF ambiguity searching and made it applicable to real-time applications. The reliability of OTF ambiguity search is also improved by the optimal design of various distinguishing tests. Following are some representative methods developed in the past 20 years for OTF integer ambiguity resolution. A more detailed comparison of these methods is presented in Chapter 2.

[1] Ambiguity Function Method (Counselman and Gourevitch, 1981; Mader, 1990; Remondi, 1991; Al-Haifi et al, 1998)

[2] Least Squares Search (Hatch, 1990; Lachapelle et al, 1992; Landau and Euler, 1992; Borge and Forssell, 1994))

[3] Fast Ambiguity Resolution Approach (Frei and Beutler, 1990; Erikson, 1992)

[4] Fast Ambiguity Search Filter (Chen, 1993; Chen and Lachapelle, 1995)

[5] Least Squares AMBiguity Decorrelation Adjustment (Teunissen, 1994)

1.3.1.2 OTF Ambiguity Resolution with Constraints

In many applications, due to the special configuration of multiple GPS antennas, constraints can be used to increase the speed and reliability of integer ambiguity resolution. For instance, baseline constraints have been used to aid the ambiguity resolution for attitude determination using GPS (Lu, 1995), while ambiguity constraints have been used by Lachapelle et al. (1993) to enhance the effectiveness of relative positioning of aircraft. Several marine tests were conducted by Weisenburger (1997) to determine the effectiveness of different constraints and their combinations for ambiguity resolution. All the results have shown that the use of constraints remarkably improves the

ambiguity resolution process. However, most of these methods are based on fixed baseline constraints, where baselines with constant lengths are available. This implies that at least two antennas must be mounted on a rigid body, which is not the case for many GPS applications.

1.3.1.3 OTF Ambiguity Resolution Using a Static GPS Network

Research work in this area can be subdivided of two parts: one is OTF ambiguity resolution in a fixed GPS network (Sun et al, 1999); the other is the OTF ambiguity resolution for a mobile user inside a multiple GPS reference station network (Raquet, 1998).

For network ambiguity resolution, dual-frequency measurements and the precisely known coordinates of all reference stations are used to estimate the initial search space of network ambiguities. Then many inappropriate ambiguity candidates are rejected by ambiguity constraints; thus, the ambiguity search space can be reduced and the search speed can be improved. However, the method required to optimally construct ambiguity constraints in a GPS network is not discussed in Sun et al (1999).

By using the geometric constraints available in a static GPS network, the behavior of spatially correlated double-differenced errors, such as ionospheric, tropospheric and orbital errors, can be estimated through least squares collocation (Raquet, 1998). Before solving the ambiguities for a mobile user in such a network, double differenced measurements between the user and a reference station are adjusted by least squares collocation using information derived from the network. This procedure also mitigates errors in the raw measurements and hence further reduces the uncertainty in ambiguity search and speeds up the ambiguity fixing process. In other words, a longer baseline can be resolved within the same time compared to the case of a single reference station. The improvement when using a multiple reference station network to aid OTF ambiguity resolution for mobile users has proven significant (Raquet, 1998).

1.3.1.4   OTF Ambiguity Resolution for Multiple Moving Platforms

The redundancy provided by the multiplicity of moving platforms can also decrease the time to ambiguity resolution while improving reliability. Early results involving the relative positioning of two aircraft with two receivers mounted rigidly on each aircraft fuselage suggest a very significant improvement in time to resolution (Lachapelle et al, 1994). Tests performed by Luo and Lachapelle (1999) also resulted in an obvious ambiguity resolution improvement for a configuration of three moving platforms, each equipped with only one GPS receiver. Herein, a more generalized method is developed to improve OTF ambiguity resolution of multiple (more than three) moving platforms.

1.3.2   GPS Error Modeling

Many GPS error models have been developed for error simulation and reduction. Among them, the spatially correlated errors (ionosphere, troposphere and broadcast orbit) and multipath have the largest impact on DGPS performance. The modeling of these errors is introduced herein.

Generally, the ionospheric error model consists of two parts, namely a model for the vertical ionospheric delay and a mapping function. The most commonly used mapping function, which is widely used for many ionospheric models (e.g., Mannucci et al., 1993; Draganov et al., 1996; RTCA, 1998; FAA, 1997) is

$$M(E) = 1 \Big/ \sqrt{1 - \left(\frac{cos\,E}{1 + h/R_E} \right)^2} \qquad (1.1)$$

where E is the elevation of a satellite,

   H is the height of the ionosphere (single layer), and

   R_E is the radius of the earth.

The major effort of modeling ionospheric errors has focused on the vertical ionospheric delay. One of the earliest vertical ionospheric error model was developed by Klobuchar (1987), and is also called the half cosine model. This model contains four parameters to

describe the diurnal change of the vertical ionospheric delay. It is simple and can only model up to 50% of the real ionospheric delay at mid-latitude, which is not accurate enough for DGPS performance analysis.

In models based on the use of spherical harmonics (e.g., El-Arini et al, 1994; Komjathy and Langley, 1996; Schaer, 1997; Hansen et al, 1997), the ionospheric delay is represented as an expansion of spherical harmonics. The coefficients of the spherical harmonics model are derived from globally distributed GPS monitor stations. Therefore, this model is much more accurate than the Klobuchar model to describe the global distribution of the ionospheric delay, but its spatial resolution is low, due to the low density of GPS stations.

Another type of ionosphere model is the grid model (e.g., Kee, 1996; Enge and Van Dierendonck, 1996; Skone, 1999). The ionosphere layer is partitioned into $n \times m$ grids with the vertical delay at a pierce point being interpolated from surrounding grid points. Different interpolation methods are used, however they lead to similar performance (El-Arini et al, 1994). The grid model is appropriate for ionospheric error modeling in a regional network (Mannucci et al, 1997) and has been suggested for the WAAS system (FAA, 1994).

Many models have been developed for modeling tropospheric effects, such as the Saastamoinen total delay model (Saastamoinen, 1972, 1973), the Hopfield dual-quartic model (Hopfield, 1969), the Black and Eisner model (Black and Eisner, 1984). A summary of tropospheric models is given by Spilker Jr. (1996). Most of these models can very well model the dry tropospheric delay. However, the wet delay and its spatial correlation and temporal variation are rarely considered, and these can impact adversly the performance of DGPS.

The orbital error has a relatively smaller impact on DGPS than the atmospheric error does. If real-time processing is not required, precise orbits derived by IGS centres can almost eliminate the orbital error. Even for real-time applications, if the baseline is not very long (<10 km), the orbital error can be neglected. Therefore, orbital error models are

seldom used. However, the statistical properties of orbital errors derived by IGS centres are helpful in building error models for simulation (Zumberge and Bertiger, 1996).

The Department of Geomatics Engineering at the University of Calgary has developed a different approach especially for differential GPS error modeling (Raquet 1997, 1998). This approach is based on least squares collocation, and requires a static GPS reference network. By integrating the double-difference measurements from all the GPS reference stations, this approach can estimate the spatial distribution of various correlated errors within the network. A linear estimator is used and certain statistical properties are used for the differential error behavior. However, the model parameters are only valid in the GPS network. The statistical properties for these parameters have not been obtained on a global scale. In addition, the limitation of this method is that it can estimate only differential errors.

Multipath decorrelates with distance very rapidly and usually cannot be reduced by differencing; thus, it can have a large impact on DGPS performance. In addition, multipath is highly correlated to the reflecting environment surrounding a GPS station; thus, it is also difficult to model multipath in kinematic applications where the reflective properties of the environment vary dramatically as a function of location. However, the multipath generating mechanism is well understood. Multipath simulation can be performed by modeling signal reflection from a modeled environment. For instance, such models have been developed by Ray (2000) and Ryan (2000).

1.4 OBJECTIVES

The major objectives of this thesis are as follows:

[1] To design a carrier phase-based relative GPS method to position multiple moving platforms with respect to each other. The multiplicity of moving platforms are to be used to form constraints for integer ambiguity resolution, and hence to decrease the time to fix and to increase the reliability of ambiguity solution. In this research, the number of GPS platforms tested is up to 10. No reference station with precisely known coordinates is required.

[2] To develop a GPS software simulator to generate various scenarios for testing. The resolution of the spatial correlation and temporal variation of DGPS errors should be enhanced compared to the existing models. This simulator can output GPS time, range, and range-rate for both code and carrier phase measurements and includes three major parts:

- GPS constellation generator, which can simulate a standard 24-satellite GPS constellation and an enhanced 30-satellite constellation.

- Trajectory generator, which is designed to generate the trajectory and velocity of a GPS platform.

- GPS error generator, which can simulate the major error sources of the GPS measurements, such as SA, receiver noise, multipath, ionospheric, tropospheric and orbital errors. In order to properly test both the stand-alone and the differential GPS systems, the temporal and spatial variation of errors is emphasized. Furthermore, adjustable parameters, such as time, meteorological data, strength of solar activity, etc., are required, in order to simulate various testing scenarios from benign to extremely adverse conditions.

[3] To fully test the approach developed for multiple moving platforms positioning using the GPS software simulator and real field data. The impact of various system parameters on the performance of the positioning algorithm is to be investigated. Those parameters include the magnitude of differential errors, the number of moving platforms, satellite geometry and visibility, the use of different observables (L1 or widelane - WL), and the different types of ambiguity solutions (integer or float). The efficiency, reliability, robustness, and accuracy of relative positioning using constraints are compared with those of the single reference station (also called single baseline) method in order to check the improvement provided by the multiplicity of the moving platforms.

[4] To analyze data transmission and processing requirements for real-time applications. The approach developed herein is designed for real-time positioning. On the one

hand, the effectiveness of the algorithm is important; on the other hand, realistic implementation of the algorithm is also important. The double differenced approach requires data transmission and processing in real-time. The larger the number of moving platforms, the higher the system capability required (including the capability of the data link and the processor). This analysis will help to understand the necessary requirements for data processing, including the optimal structure of the data, the data link required, the functional inter-platform distance, and the platform configuration.

1.5 OUTLINE OF DISSERTATION

The remaining parts of thesis consist of the following chapters:

Chapter 2 describes the fundamentals of GPS relative positioning, which includes the selection of GPS observables, the major GPS error sources and their impact on positioning. In the last section, the principle of OTF ambiguity resolution is reviewed.

Chapter 3 first discusses the available constraints for ambiguity resolution for the case of multiple moving platforms. Then the new methodology proposed herein for relative positioning using multiple-platform constraints is described. The newly developed approach is called 'MultiKin' (Multi-Platforms Kinematic). It includes adaptive constraints construction (Delaunay triangulation), modification and optimization of the existing algorithms for OTF ambiguity resolution, and application of available constraints to improve the effectiveness of relative positioning.

Chapter 4 describes the design of major GPS error models involved in the DGPS applications, namely models for the ionospheric, tropospheric, orbital errors, multipath, and SA. The first three are spatially correlated errors; thus, the simulation of spatial and temporal correlation of errors is emphasized. Multipath is the dominant error for short-baseline resolution. Two models are discussed for multipath simulation in static and kinematic applications. A simple discussion of SA modeling is also given in this chapter for completeness.

Chapter 5 deals with the other two parts of the GPS software simulator, i.e., the constellation generator and the trajectory generator. To fully test MultiKin, the design of various simulation scenarios for testing is also described.

Chapter 6 presents the results of simulations and field tests for MultiKin. It analyzes the effects of all kinds of parameters on system performances, including the number of platforms, satellite geometry, inter-platform distance, magnitude of GPS errors, selection of observables, etc. Performance is assessed by comparing the results based on the use of constraints to the single-baseline method. The design and results of two field tests using multiple receivers are presented to confirm the conclusions drawn through the simulations.

Chapter 7 discusses operational considerations for implementation of the relative positioning of multiple moving platforms, namely, capability of data link and processor, configuration, functional distance, etc.

Chapter 8 provides conclusions and recommendations for further research.

2 GPS THEORY

2.1 GPS OBSERVABLES

Figure 2.1 shows the principle of stand-alone GPS positioning. The range from a satellite to a user's antenna, denoted as R, is measured by the propagation time of the GPS signal. The position of a satellite is calculated from the broadcast ephemeris. The position of the user's antenna is on a sphere centered at the satellite with the radius of R. When four satellites are observed, the three-dimensional position of the antenna and the receiver clock error can be estimated.
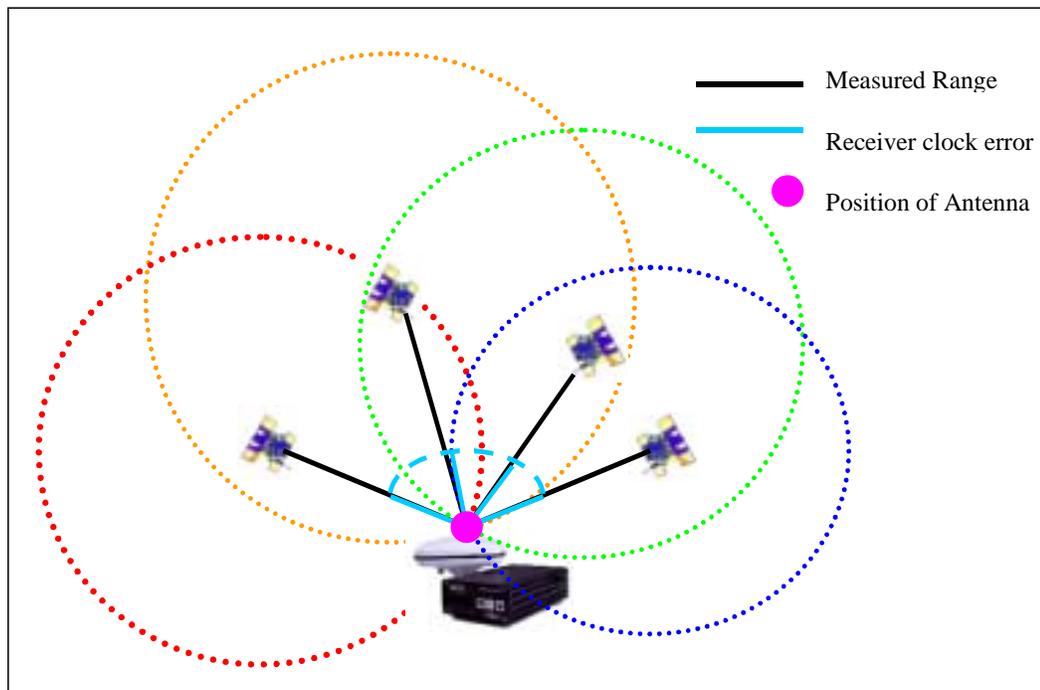
Figure 2.1 Principle of stand-alone GPS positioning

The accuracy of positions depends on the errors in the range measurements scaled by the satellite geometry. The range measurement is obtained by comparing either the PRN (pseudo random noise) code phase or the carrier phase of the received GPS signal with the replica signal generated by the GPS receiver. The observation equations of the GPS code and carrier phase measurements are:

$$P = \rho + d\rho + c(dt - dT) + d_{ion} + d_{trop} + \varepsilon_{mp} + \varepsilon_P \qquad (2.1)$$

$$\Phi = \rho + d\rho + c(dt - dT) - d_{ion} + d_{trop} + \varepsilon_{m\Phi} + \varepsilon_\Phi + \lambda N \qquad (2.2)$$

where P is the code measurement (m),

ρ is the geometric range from a satellite to a receiver (m),

$d\rho$ is the orbital error (m),

c is the speed of light (m/s),

dt is the satellite clock error (m),

dT is the receiver clock error (m),

d_{ion} is the ionospheric delay (m),

d_{trop} is the tropospheric delay (m),

ε_{mp} is the code multipath error (m),

ε_P is the receiver noise in code measurement (m),

Φ is the carrier phase measurement (m),

$\varepsilon_{m\Phi}$ is the carrier phase multipath error (m),

ε_Φ is the receiver noise in carrier phase measurement (m),

λ is the wavelength of the GPS carrier (m/cycle), and

N is the integer cycle ambiguity (cycle).

The ionospheric error in the code measurement has the same magnitude but a different sign from that in the phase measurement. Also, the carrier phase measurement differs

from the code measurement by an ambiguity term *N*, which is an unknown integer resulting from the periodic property of the GPS carrier. Generally, the range measurement is quite noisy. Even when SA is off, the accuracy of stand-alone positioning can only be a few metres. In order to achieve a higher accuracy, a reference station with precisely known coordinates is needed, and the differential method is used to measure the relative position between a reference and a rover station. The major GPS errors observed at the reference and rover stations are spatially or temporally correlated. Differential methods can reduce most correlated components when the separation of the receivers is not very large. The observation equation for the single differenced GPS carrier phase between a reference and a rover station is

$$\Delta\Phi = \Phi^i_{rover} - \Phi^i_{reference} \tag{2.3}$$

This equation can be rewritten as:

$$\Delta\Phi = \Delta\rho + \Delta d\rho + c\Delta dT - \Delta d_{ion} + \Delta d_{trop} + \Delta\varepsilon_{m\Phi} + \Delta\varepsilon_{\Phi} + \lambda\Delta N \tag{2.4}$$

After differencing, the spatially correlated errors are reduced, the satellite clock error is cancelled, and the uncorrelated errors are amplified. Usually, the uncorrelated error amplification is much smaller; thus, the differenced measurement has a much lower level of error than a single measurement.

As shown in Equation (2.4), the receiver clock error *dT* still exists and is coupled with the ambiguity term; thus, the single differenced method is seldom adopted, except in some attitude determination systems. In these systems, either a dedicated receiver is used to process the RF input of multiple antennas, or multiple receivers are driven by a common external clock (Keong, 1999). Both schemes can eliminate the receiver clock errors. However, in most other applications, the requirement of a common clock is not practical because of the large separation between a reference and a rover station. The double differenced technique is therefore adopted. It can be formulated as follows:

$$\Delta\nabla\Phi = \Delta\Phi^i - \Delta\Phi^j = \left(\Phi^i_{rover} - \Phi^i_{reference}\right) - \left(\Phi^j_{rover} - \Phi^j_{reference}\right) \tag{2.5}$$

This equation can be rewritten as:

$$\Delta\nabla\varPhi = \Delta\nabla\rho + \Delta\nabla d\rho - \Delta\nabla d_{ion} + \Delta\nabla d_{trop} + \Delta\nabla\varepsilon_{m\varPhi} + \Delta\nabla\varepsilon_{\varPhi} + \lambda\Delta\nabla N \qquad (2.6)$$

For a receiver, the different receiving channels are driven by a common internal clock, which implies that the clock errors of different channels are basically the same and can be removed by differencing between two satellites. Therefore, the double-differenced carrier phase is the most often used observable in precise relative positioning. Herein, the carrier phase ambiguity is always referred to as the double-differenced ambiguity unless indicated otherwise.

For carrier phase positioning, the high accuracy depends on ambiguity fixing. For the L1 frequency, the wavelength is only 19 cm; centimetre-level accuracy positions can be achieved provided that the ambiguities can be correctly fixed. However, successful fixing mainly depends on the error magnitudes (in cycles) in the carrier phase observations. If the separation between a reference and a rover station is large, the residual error after differencing can result in difficulty fixing ambiguities. In this case, the combination of dual-frequency carrier phase observables can be used instead of the single-frequency observables:

$$\varPhi_{i,j} = i\varPhi_1 + j\varPhi_2 \qquad (2.7)$$

where $\varPhi_1$ is the observed carrier phase of L1 in cycles, and

$\varPhi_2$ is the observed carrier phase of L2 in cycles.

The widelane observable is often used to solve the ambiguities of long baselines. This is because the widelane observable not only has a relatively longer wavelength, but also reduces the atmospheric errors (in cycles) by differencing between L1 and L2. Five different errors in observations are shown in Equation (2.4). They can be categorized into three types:

[1] ε_{Ind}, Independent, e.g., receiver noise and multipath,

[2] ε_{FD}, Frequency-dependent, e.g., ionospheric error, and

[3] ε_{FC}, Frequency-consistent, e.g., tropospheric and orbital error.

If ε_Φ is the total carrier phase error in cycles, then the errors in L1 and L2 can be described as:

$$\varepsilon_{\Phi,1} = \varepsilon_{Ind,1} + \varepsilon_{FD,1} + \varepsilon_{FC,1} \qquad cycles \tag{2.8}$$

$$\varepsilon_{\Phi,2} = \varepsilon_{Ind,2} + \varepsilon_{FD,2} + \varepsilon_{FC,2} \qquad cycles \tag{2.9}$$

Given the properties of these errors, they can be derived as follows:

$$\varepsilon_{FD,1} f_1 = \varepsilon_{FD,2} f_2 \tag{2.10}$$

$$\varepsilon_{FC,1} / f_1 = \varepsilon_{FC,2} / f_2 \tag{2.11}$$

where f_1=1575.42 MHz,

$\qquad f_2$=1227.60 MHz.

Thus, the error (in cycles) of a dual-frequency combination can be written as:

$$\varepsilon_{\Phi,ij} = i\varepsilon_{\Phi,1} + j\varepsilon_{\Phi,2} = \left(i\varepsilon_{Ind,1} + j\varepsilon_{Ind,2}\right) + \varepsilon_{FD,1}\left(i + j\, f_1/f_2\right) + \varepsilon_{FC,1}\left(i + j\, f_2/f_1\right) \tag{2.12}$$

$$= \left(i\varepsilon_{Ind,1} + j\varepsilon_{Ind,2}\right) + \frac{60i + 77j}{60}\varepsilon_{FD,1} + \frac{77i + 60j}{77}\varepsilon_{FC,1} \quad cycles$$

Assuming that $\varepsilon_{Ind,1}$ and $\varepsilon_{Ind,2}$ have the same variance and are independent of each other, according to the covariance law, the variances in cycles^2 of the L1 and dual-frequency combined observables can be derived as:

$$\sigma_{\Phi,1}^2 = \sigma_{Ind}^2 + \sigma_{FD,1}^2 + \sigma_{FC,1}^2 \qquad cycles^2 \tag{2.13}$$

$$\sigma^2_{\Phi,ij} = \left(i^2 + j^2\right)\sigma^2_{Ind} + \left(\frac{60i + 77j}{60}\right)^2 \sigma^2_{FD,1} + \left(\frac{77i + 60j}{77}\right)^2 \sigma^2_{FC,1} \quad cycles^2$$

The variance in metres is $\lambda^2_{ij}\sigma^2_{\Phi,ij}$,

where λ_1 and λ_2 are the wavelengths of L1 and L2 carriers respectively,

$$\lambda_{ij} = \frac{\lambda_1\lambda_2}{i\lambda_2 + j\lambda_1}$$ is the resulting wavelength of the dual-frequency observable.

Table 2.1 gives the most commonly used dual-frequency combined observables in GPS, where coefficients of variances in cycles^2 are $\left(i^2 + j^2 \quad \left(\frac{60i + 77j}{60}\right)^2 \quad \left(\frac{77i + 60j}{77}\right)^2\right)$, coefficients of variances in metres^2 is $\frac{\lambda^2_{ij}}{\lambda^2_1}\left(i^2 + j^2 \quad \left(\frac{60i + 77j}{60}\right)^2 \quad \left(\frac{77i + 60j}{77}\right)^2\right)$.

Table 2.1 GPS carrier phase combinations

Observable	i	j	λ_{ij} (metre)	Coefficients of Variances (cycles^2)	Coefficients of Variances (metres^2)
L1	1	0	0.190	$(1 \quad 1 \quad 1)$	$(1 \quad 1 \quad 1)$
Widelane	1	-1	0.862	$(2 \quad 0.08 \quad 0.049)$	$(41.16 \quad 1.65 \quad 1)$
Narrowlane	1	1	0.107	$(2 \quad 5.21 \quad 3.17)$	$(0.634 \quad 1.65 \quad 1)$
Ionosphere Free	$\dfrac{f_1^2}{f_1^2 - f_2^2}$	$\dfrac{-f_1 f_2}{f_1^2 - f_2^2}$	0.484	$(10.42 \quad 0 \quad 1)$	$(67.6 \quad 0 \quad 6.49)$

The variances in cycles^2 represent the difficulty in fixing ambiguity. It is obvious that the widelane observable can greatly reduce the atmospheric impact on ambiguity resolution because the variances (in cycles^2) of the ionospheric and tropospheric errors are scaled by

0.08 and 0.049 respectively. Thus, widelaning is proper for resolution of a long baseline provided that the effect of independent errors is relatively small. However, the positioning accuracy derived by the widelane observable is poorer than that derived by the L1 observable because the variance in metres^2 is amplified after widelaning.

For very short baselines, the impact of the spatially correlated errors can be neglected so the use of narrowlane can lead to high-accuracy positions. This can be verified from the smaller coefficient (0.634) of independent errors (metres^2) for the narrowlane as opposed to the L1 (1.0).

To attract more civilian users to use the GPS system for navigation and positioning, the US government announced not only to make the C/A code on L2 available, but also to add a third civilian frequency, L5 (1176.45 MHz), on GPS. When implemented, the third frequency will greatly improve the reliability and robustness of the system, because more combinations of the three frequencies can be formed, which can have both longer wavelengths and reduced atmospheric errors (Han and Rizos, 1999).

2.2 GPS ERROR SOURCES

Equation (2.4) shows seven different GPS error sources which can be categorized into three different types according to their spatial and temporal correlations:

[1] Only satellite correlated : satellite clock error, SA (until May 2000),

[2] Spatially correlated: satellite orbital error, ionospheric and tropospheric error,

[3] Independent error: receiver noise, multipath error.

The first type of errors can be totally removed by single differencing between two receivers. The second type of error can be reduced by single differencing; however, the residual errors increase as the separation increases. The third type of error cannot be removed by any differencing techniques and they are independent of baseline lengths.

2.2.1 Orbital Error

Orbital error results from the uncertainties in the broadcast ephemeris. These uncertainties are due to the accuracy limitations associated with the predicted nature of the broadcast ephemeris. Tests have shown that the orbital error is generally a few metres; but sometimes, it can reach tens of metres due to the problem of orbit prediction. Orbital error can be greatly reduced by differencing between receivers. A more effective way to handle orbital error is to use post-processed precise orbits. Precise orbits are derived from an extensive reference network and the accuracy can be as high as a few centimetres (JPL website). However, precise orbits are not available for real-time applications.

2.2.2 Ionospheric Error

Ionospheric error is caused by the presence of free electrons when GPS signals pass through the upper layer of the atmosphere. The effect on range may vary from 150 metres (at midday, during periods of maximum sunspot activity, with the satellite near the horizon of the observer) to less than five metres, (at night, during periods of minimum sunspot activity, with the satellite at the zenith) (Wells et al. 1987). For GPS carrier frequencies, the ionospheric delay is dispersive. This fact can be used to advantage, since a special linear combination of the dual-frequency GPS observations can be formed to eliminate most of the ionospheric effect. Ionospheric correction coefficients from the broadcast ephemeris can only remove 50% of the ionospheric delay at mid-latitudes (Wells et al. 1987). Recent research on precise prediction of the ionospheric delay using a wide-area GPS network, such as WAAS, or using a regional network (Raquet, 1998), has shown some good results for correcting the ionospheric delay using interpolation or least squares collocation.

2.2.3 Tropospheric Error

Tropospheric delay is caused by the refraction of the GPS signal in the troposphere. The delay contains two parts. The larger part is caused by the dry atmosphere component, which is stable and predictable. The delay resulting from the water vapor is smaller, but varies greatly. At GPS frequencies, the troposphere is non-dispersive. The tropospheric

delay is strongly correlated over a short distance between the reference and the rover stations when the height difference of the two stations is small. However, when the separation or height difference is large, local atmospheric conditions will be different and the correlation becomes weaker. Surface meteorological data is not accurate to adequately represent atmospheric conditions along the signal path (Spilker Jr., 1996). To get more accurate estimations, the water vapour content of the atmosphere along the propagation path can be measured with water vapour radiometers (Resch, 1984)). However, the instruments are very elaborate and expensive. Now, GPS networks are also used to predict the relative tropospheric wet delay (Zhang, 1999).

2.2.4 Multipath

Multipath occurs when reflected signals, in addition to the direct signal, reach the antenna. It depends highly on the properties of the reflector, the antenna gain pattern, and the type of correlator used in a receiver. Multipath interferes with the correlator in a GPS receiver to precisely determine the time instant of signal reception. It affects both pseudorange and carrier phase measurements. The code multipath is generally much larger than the carrier phase multipath. It can reach up to one-half of a chip length of the PRN code, assuming an environment in which the multipath signal strength never exceeds that of the direct signal (Goldhirsh and Vogel, 1989). By contrast, the carrier phase multipath is always less than one-quarter of the carrier wavelength (Georgiadou and Kleusberg, 1988). Typically, for static observations, multipath is non-Gaussian in nature and shows sinusoidal oscillations with periods of a few minutes due to the change of satellite geometry. In kinematic applications, multipath behaves more randomly because the movement of the vehicle changes the reflecting geometry in a relatively random way. For most precise positioning applications, multipath is one of the major error sources, because it decorrelates very fast over distance and cannot be reduced by differencing or modeling.

2.2.5 Receiver Noise

Typical pseudorange measurement resolution is approximately 0.1 to one metre on the C/A code (Lachapelle, 1997). The NovAtel Narrow-Correlator™ receivers measure the

pseudorange with a 10 cm noise level. The carrier phase can be measured with millimetre or sub-millimetre precision. Receiver noise is mainly caused by thermal noise, dynamic stress and oscillator stability in the tracking loop. It has the least effect on carrier phase positioning.

2.2.6   Selective Availability

SA was the intentional degradation of the GPS signal in order to deny full position and velocity accuracy to unauthorized users (van Grass and Braasch, 1996). Two different methods could be used to deny the GPS accuracy: manipulation of the navigation message orbit data (referred to as the ε-error) or manipulation of the satellite clock frequency (referred to as δ-error). The real SA only contained δ-errors. SA used to be part of the standard positioning service and was turned off in May 2000. It was the largest error source for stand-alone positioning. Even for differential GPS, it could cause some problems. For instance, the impact of δ-errors depended on the latency of differential corrections.

2.3   OTF AMBIGUITY RESOLUTION

As mentioned in the Chapter 1, GPS carrier phase positioning has a higher accuracy than code positioning, assuming the integer ambiguity is correctly fixed. OTF ambiguity resolution is a method used to solve the carrier phase ambiguity in kinematic applications. It differs from the static ambiguity resolution in two ways:

[1] In kinematic applications, errors of measurement cannot be reduced by time averaging because the movement of platforms can significantly change the testing environment.

[2] In kinematic applications, the position and velocity of the object is required for every epoch, so the batch processing cannot be adopted if real-time processing is required.

Since less information is available and larger errors occur, OTF ambiguity resolution is more difficult in kinematic than in static mode. Here are some major factors affecting the OTF ambiguity resolution (Lachapelle, 1997):

[1] Selection of observables

[2] Inter-receiver distance

[3] Number and geometry of satellites

[4] Magnitude of GPS errors

[5] Ambiguity search method

[6] Performance required, etc.

The study of OTF ambiguity resolution started in the early 1980s, with several methods having been developed since then. Basically, they have the same strategies to fix ambiguities, namely, float ambiguity resolution, integer ambiguity searching, and the use of a distinguishing test.

The float ambiguity and its variance are used to define the initial search point, and the search range of the integer candidates. Usually, the float solution is derived from the pseudorange measurement, which is further smoothed by the carrier phase using least squares adjustment or Kalman filtering. The recent research of GPS network adjustment (Raquet, 1998) presents significant improvement in error reduction for long-baseline resolution. It has shown that the double differenced errors can be reduced by up to 50% at distances around 200 km when using the network adjustment approach. Since the model parameters are built from real data, the network adjustment shows a brighter prospect in error reduction than the classical Kalman filtering based on the empirical error models.

Table 2.2 gives a summary of some representative ambiguity search methods. References for each method can be found in section 1.3.1. In several ambiguity search methods, the covariance matrix after Kalman filtering is not directly used to define the ambiguity search space. This is because only three of the double difference ambiguities are independent, whereas the number of observations is generally more than that. Consequently, the variance-covariance matrix is not diagonal due to the strong correlation among ambiguities. This correlation is a nuisance for ambiguity resolution because it largely elongates the search space. Therefore, the full search method is usually

24

time-consuming and has a low efficiency. To reduce the correlation among ambiguities, the matrix transformation and conditional least squares approaches are often used, such as LAMBDA and FASF. A detailed discussion of LAMBDA and FASF method is given in Chapter 3.

Table 2.2 Comparison of ambiguity search methods

Method	Evaluation Function	Correlation Reduction
Ambiguity Function Method (AFM)	$AFM(x,y,z) = \frac{1}{(M-1)} \sum_{j=1}^{M-1} cos\theta_j$ $AFM(x,y,z) = \frac{1}{(M-1)} \sum_{j=1}^{M-1} \left[cos(\theta_{L1})_i + cos(\theta_{L2}*S)_i \right]$ M is the number of visible satellites θ is the measurement residual (x,y,z) is the position for trial s is the weighting factor applied to L2 residuals	No reduction Full search
Least Squares Search	$r^T C_\ell^{-1} r$ r is the residual vector computed using the potential integer ambiguities of primary satellites and corresponding integer ambiguities of secondary satellites C_ℓ is the covariance matrix of observations	Ambiguity search only conducted for four primary satellites
Fast Ambiguity Resolution Algorithm (FARA)	$r^T C_\ell^{-1} r$ r is the residual vector C_ℓ is the covariance matrix of observations	Reject ambiguity pair using student test
Fast Ambiguity Search Filter (FASF)	$\Omega' = (N_{float} - N_{int})^T C_N^{-1} (N_{float} - N_{int})$ C_N is the conditional covariance matrix of the float ambiguity set N_{float}, N_{int} is the integer ambiguity solution	Reduced search space by conditional least squares
Least squares AMBiguity Decorrelation Adjustment (LAMBDA)	$\Omega' = (z_{float} - z_{int})^T C_z^{-1} (z_{float} - z_{int})$ C_z is the conditional covariance matrix of transfomed float ambiguity set z_{float}, z_{int} is the transformed integer ambiguity solution	$z = ZN$ $C_Z = ZC_N Z^T$ Z is the transformation matrix which leads to diagonalization of the covariance matrix C_z

When measurements from multiple GPS receivers are available, the resulting constraints from the geometry of receivers can also be used as a way to reduce the space in ambiguity search. Generally, a Kalman filter will be used to process the constraints:

$$\begin{cases} X^+ = X^- \\ f(X) = 0 \Rightarrow HX = w \end{cases} \qquad (2.15)$$

where X is the vector of parameters,

$\quad$ $f(X) = 0$ is the constraint,

$\quad$ H is the linearized design matrix of the constraints, and

$\quad$ w is the misclosure.

The first equation is the dynamic model. The second is the available constraints used as the observation model, which is linearized in practice. The parameter X is updated by this observation model. Weisenburger (1997) gave the derivation of the design matrix H for multiple constraints.

After ambiguity searching, a distinguishing test is conducted to check the distinctiveness of the best integer candidate. If the test is passed, it means the best candidate is good enough to be the true. Otherwise, more observations are needed and the search procedure will continue. Generally, the χ^2 test (Wei and Schwarz, 1995) or the ratio test (Landau and Euler, 1992; Chen and Lachapelle, 1995; Wei and Schwarz, 1995; Han and Rizos, 1996) are selected. However, they do not generally give a satisfactory solution. This is because most of the current tests are based on the assumption that observations are free of blunders and biases, and observation errors are Gaussian in nature, which is rarely true for GPS.

Due to the errors in GPS measurements, the integer ambiguities can be incorrectly fixed. This can result in a serious degradation of the positioning accuracy; therefore, the monitoring of the integer solution is necessary for carrier phase positioning. Usually, a residual test is performed to check whether the residuals exceed a confidence range. This

test is based on the same assumption made for the distinguishing test; thus, it is not robust when blunders are present.

Much effort has been put into research for improving the performance of single-baseline ambiguity resolution. However, achieving further improvement is extremely difficult because of limited information (measurements from only two GPS receivers are available). Therefore, using constraints provided by multiple platforms to aid ambiguity resolution seems to be a valid alternative. The proper use of these constraints has proven beneficial for ambiguity resolution. In Chapter 3, an OTF ambiguity resolution method using constraints from multiple moving platforms will be discussed in detail.

3 MULTIKIN METHOD

MultiKin is a method developed for relative positioning, which can process the data from multiple GPS platforms at the same time. It has two characteristics. First, it does not need reference stations with precisely known coordinates. Second, it can make use of the constraints provided by the multiplicity of receivers to improve the OTF ambiguity resolution and hence to improve the efficiency and reliability of the relative positioning of each baseline.

The procedure of MultiKin consists of three steps:

[1] Constraints construction

[2] Individual baseline resolution

[3] Enhancement of ambiguity fixing using constraints.

It has been shown in previous research that constraints are very helpful for ambiguity resolution. The more the constraints are used, the greater the improvement. Herein, a discussion of different constraints is given first. According to their properties, the applicable constraints for this application are determined before presenting the detailed methodology of MultiKin.

3.1 AVAILABLE CONSTRAINTS

[1] The fixed baseline constraint

A fixed baseline means that the inter-platform distance (baseline length) is precisely known. This can be used to aid in fixing ambiguities between the moving rover pair. Generally, the fixed baseline length is obtained from an external source other than GPS, such as a tape measurement. This constraint is often used in attitude determination systems (Lachapelle et al, 1993; Lu, 1995), where the antenna-array has been fixed on the body frame of the measured object.

[2] Attitude constraint

When the integer ambiguities of a moving baseline are fixed, the fixed baseline can help fix the integer ambiguities between a reference station and the moving rover pair. Because the attitude of the "moving baseline" can be precisely estimated, the solved attitude can then be used as a constraint for the ambiguity resolution between the reference and rover stations.

[3] Approximate coordinate constraint

The most often used coordinate constraint is a height constraint. It is extremely useful in marine applications where the height is well known. Remondi (1992) discussed the use of height constraints to directly aid in rejecting possible ambiguity sets.

None of the above constraints is proper for ambiguity resolution in a configuration of multiple moving platforms for two reasons. First, there is no fixed baseline in this application, because all platforms are mobile and each has only one antenna. If the fixed baseline constraint cannot be used, neither can the attitude constraint. Second, this research is not specific to marine applications, so the coordinate constraints are not valid here either. The only effective constraint for positioning multiple moving platforms is the ambiguity constraint.

[4] Ambiguity constraint

The concept of the ambiguity constraint is that the sum of the double-differenced ambiguities in a closed polygon is zero, as shown in Figure 3.1. This theory was first proved and applied by Lachapelle et al (1993, 1994). Since then, it has been widely used to aid ambiguity resolution:

$$\Delta\nabla N_{1,2} + \Delta\nabla N_{2,3} + \cdots + \Delta\nabla N_{m-1,m} + \Delta\nabla N_{m,1} = 0 \tag{3.1}$$

where, $\Delta\nabla N_{i,j} = \left(N_i^A - N_j^A\right) - \left(N_i^B - N_j^B\right)$ is the double differenced integer ambiguities,

i,j are the indices of the GPS platforms (nodes of polygon),

A is the common satellite observed by all platforms, and

B is the common base satellite.
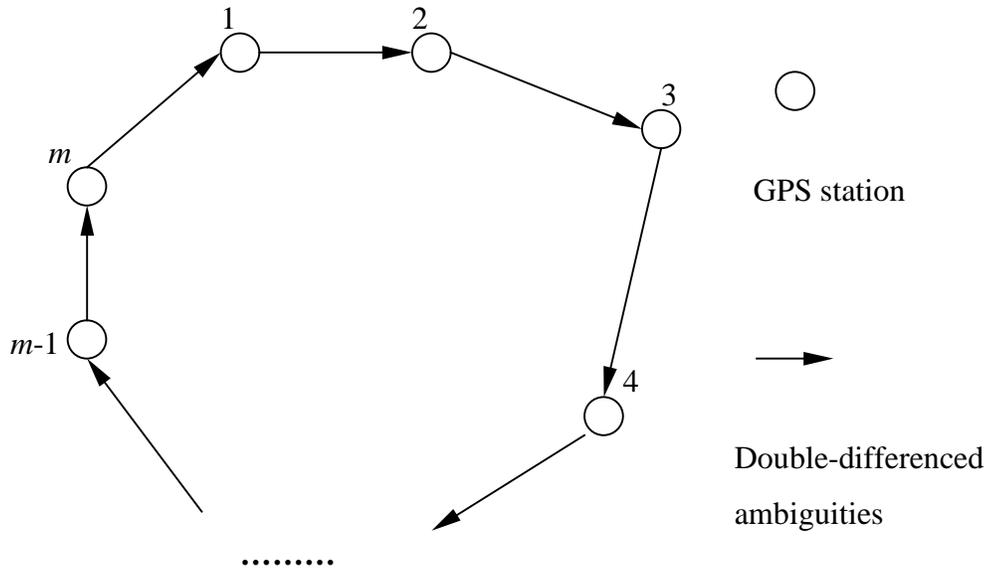
Figure 3.1 Concept of closed (polygonal) ambiguity constraints

As a special case of polygonal ambiguity constraints, the triangular constraint has been proven to be the best ambiguity constraint. Here are two prominent advantages of triangular constraints over polygonal constraints:

Higher efficiency: Once the ambiguities of any two baselines are fixed, ambiguities of the third baseline can be fixed immediately, whereas for a polygonal constraint, ℓ-1 baselines must be fixed before the constraint can help to fix the ℓth baseline.

Higher reliability: If the ambiguities of a baseline are incorrectly fixed, they will affect only the other two baselines in the same triangle. However, for polygonal constraints, all other ℓ-1 baselines in the same polygon are affected by the incorrectly fixed baseline. Therefore, the multi-triangular constraint can better localize the error in ambiguity resolution.

Because of the higher effectiveness of the triangular constraint, the multiple-triangular constraints are used in MultiKin.

3.2 DELAUNAY TRIANGULATION FOR CONSTRUCTION OF CONSTRAINTS

When there are multiple moving platforms in the application, The choices of constructing triangular constraints are also multiple. In this section, the discussion of criteria and methodology for optimally constructing constraints is presented.

3.2.1 Optimal Criteria for Baseline Selection

The selection of baselines determines the effectiveness of an algorithm to construct ambiguity constraints. An optimal approach to select the baselines and construct constraint triangles must meet all the following requirements:

[1] Reasonable computational burden

[2] Effective use of constraints

[3] Selection of the shortest baselines

In a GPS configuration containing ℓ moving platforms, the numbers of baselines and triangles are $\ell(\ell-1)/2$ and $\ell(\ell-1)(\ell-2)/6$ respectively. When there are only three platforms in the configuration, the selection of baselines for ambiguity constraints is unique. However, when the number of platforms is more than three, the numbers of optional baselines and triangles increase dramatically. If all the baselines and possible triangular constraints are used to aid ambiguity resolution, the extremely heavy computational burden can result in difficulties with real-time processing. For instance, selecting all the moving baselines and triangles under a configuration of 50 GPS platforms leads to simultaneous processing of 1225 moving baselines and 19600 triangular constraints. This requires a very high-speed processor and a very large amount of memory.

The effectiveness of the constructed constraints is also very important. Figure 3.2 gives two schemes to construct constraints. It is obvious that Scheme A needs less computation than Scheme B, but is less effective. This is because the constraints used in Scheme A are

independent of each other, i.e., one constraint is not helpful for ambiguity resolution of a baseline in another triangle. Therefore, optimal triangular constraints should interrelate with each other.
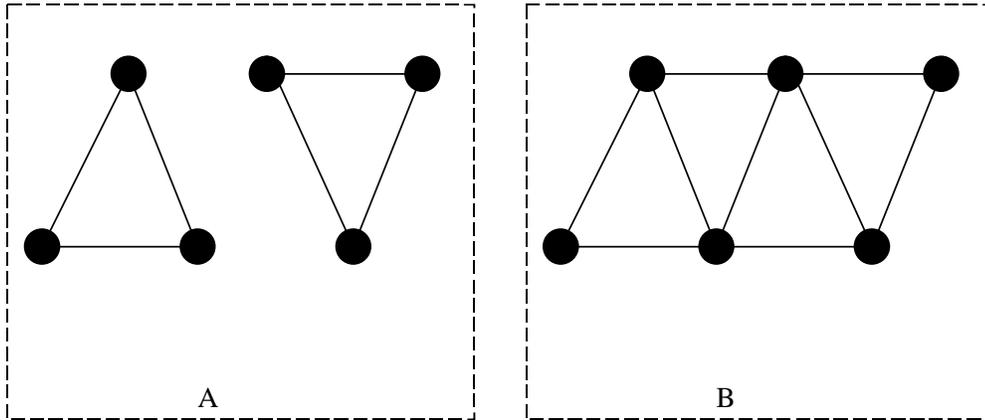
Figure 3.2 Effectiveness comparison of different triangular constraints

Because the resolution of a baseline is mainly dependent on the magnitude of differential GPS errors, the optimal triangulation should avoid using baselines which could induce large measurement errors. In Figure 3.3, both Schemes A and B have the same computational burden and interrelation of constraints; however, Scheme A is statistically better than Scheme B because it contains relatively shorter baselines. Shorter baselines cannot guarantee smaller errors in the observations, since the uncorrelated errors, such as multipath and receiver noise, are independent on the length of a baseline. However, statistically, short baselines can be treated as a good indication of small differential errors when the spatially correlated errors are dominant; thus, Scheme A should always be selected over Scheme B.

As a result of considering all the above requirements, an approach called Delaunay triangulation is selected for the optimal construction of ambiguity constraints. Here are some properties of Delaunay triangulation:

[1] For 2D Delaunay triangulation, the numbers of selected baselines and triangles increases linearly with the number of platforms;

[2] Delaunay triangles interrelate with each other. The number of Delaunay triangles is equal to the number of the independent triangular constraints;

[3] Delaunay triangulation can select the shortest baselines in most of cases.
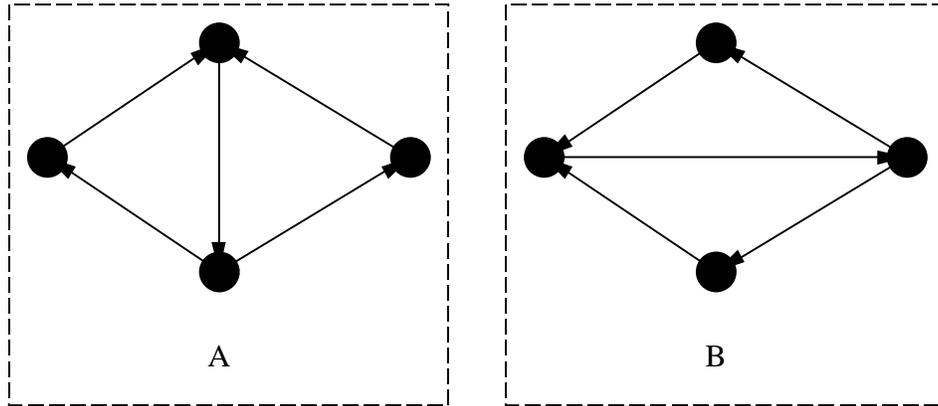
Figure 3.3 Baseline selection (reduced differential errors)

From these properties, it can be concluded that Delaunay triangulation satisfies all the criteria for optimally constructing constraints. In the following section, experimental results will be presented to compare the effectiveness of the Delaunay triangulation and the full selection method.

3.2.2   Definition of Delaunay Triangulation

The definition of Delaunay Triangulation is based on the Voronoi diagram through the principle of duality (Preparata and Shamos, 1985).

Definition of Voronoi graph: Let $P = \{p_1,\quad p_2,\quad \cdots\quad p_k\}$ be a finite set of points in the n-dimensional space R^n, and their position vectors $x_i \neq x_j \forall i \neq j$. The region given by

$$V(p_i) = \left\{ x \middle| \|x - x_i\| \leq \|x - x_j\| \forall j \neq i \right\}$$

(3.2)

is called Voronoi region (Voronoi box) associated with p_i and

$$V(P) = \bigcup_{i=1}^{k} V(p_i)$$

is the Voronoi diagram of *P*.

Definition of Delaunay edge: Let *P* be a finite set of points in a sub-domain Ω^n of the *n*-dimensional space R^n. Two points p_i and p_j are connected by a Delaunay edge *e* if and only if there exists a location $x \in \Omega^n$ which is equally close to p_i and p_j and closer to p_i, p_j than to any other $p_k \in P$. The location *x* is the centre of an *n*-dimensional sphere which passes through the points p_i, p_j and which contains no other points p_k of *P*.

Definition of Delaunay triangle: Let *P* be a finite set of points in a sub-domain Ω^n of the *n*-dimensional space R^n. Three non-collinear points p_i, p_j and p_k form a Delaunay triangle *t* if and only if there exists a location $x \in \Omega^n$ which is equally close to p_i, p_j and p_k and closer to p_i, p_j, p_k than to any other $p_m \in P$. The location *x* is the centre of an *n*-dimensional sphere which passes through the points p_i, p_j, p_k and which contains no other points p_m of *P*.

In the case of 2D triangulation, Delaunay triangulation is known to minimize the largest circumcircle, and to maximize the minimum angle of all triangles. In 3D triangulation, the Delaunay triangulation is only known to minimize the largest minimum-containment sphere (Bern and Eppstein, 1992; Rajan 1991). An important difference between 2D and 3D triangulation is the number of triangles/tetrahedra as a function of the number of points *n*. While the number of triangles in 2D triangulation grows with $O(n)$, the number of Delaunay tetrahedra in a tetrahedralization can grow with $O(n^2)$.

3.2.3   Methodology

This section presents a brief review of Delaunay triangulation algorithms for a given point set *P* without constraining boundaries. There are many algorithms used for Delaunay triangulation, most of which focus on improving the efficiency of "nearest neighbor" search. For a less optimal method, the number of required flip operations grows with $O(n^2)$ where *n* is the number of points. However, for an optimal algorithm,

the growth rate is only $O(n\log n)$. A detailed comparison of Delaunay triangulation algorithms can be found in (Su and Drysdale 1995). The four major methods used for Delaunay triangulation are

[1] Divide-and-Conquer (Preparata and Shamos, 1985)

[2] Sweepline (Fortune, 1987)

[3] Incremental Construction (Lawson, 1977)

[4] Incremental Search (Merriam 1993 and Hitschfeld, 1993)

In this research the number of platforms in the configuration is not large (<100), so the efficiency of Delaunay triangulation is not a major concern. An algorithm proposed by Avis and Bhattcharya (1983) is used herein. The three major steps of this algorithm are as follows:

[1] Set any platform (point) p_i as the origin.

[2] Find a small subset of other points to define a polyhedron P_J which contains the Voronoi polyhedron of p_i:

$$P_J = \bigcap_{j \in J} H(p_i, p_j)$$ (3.4)

where $J \subseteq \{1,2,\cdots \ell\} - \{i\}$. $H(p_i, p_j)$ is the closed half-space bounded by the perpendicular bisector of points p_i and p_j that contains p_i.

[3] Eliminate the redundancy of $H(p_i, p_j)$ with respect to P_J by solving the linear program:

$$z^* = max(p_j - p_i)x$$ (3.5)

 subject to $(p_k - p_i)x \leq [(p_k + p_i)/2](p_k + p_i)$

where $k \in J$.

The application studied herein is only related to a very small GPS network with an area of less than 20×20 km². If the height difference of the platforms is much less than the horizontal separation, this 3D Delaunay triangulation problem can degenerate to the 2D Delaunay triangulation. Correspondingly, the search algorithm can become more efficient and the upper bound of the baseline number can also be reduced. For 2D Delaunay triangulation, the maxium number of baselines is 3ℓ-6 (ℓ is the number of the stations), whereas, for 3D triangulation it can be up to C_ℓ^2. However, in both cases, the number of baselines is usually much less than the upper bound and the computational burden of the triangulation can be reduced. Herein, the 2D Delaunay triangulation is actually applied in the simulation tests presented in Chapter 6.

3.2.4 Special Considerations for Implementation

It is also worth noting that in this application all the platforms are mobile, so the baseline lengths change over time. This geometry change can eventually cause the triangulation to change, if the three criteria defined in section 3.2.1 are continuously applied. For instance, in Figure 3.4, Platforms 1 and 3 are static, while Platforms 2 and 4 are moving south and north, respectively. The initial best triangulation is shown in A. After some time, with the distance increasing between 2 and 4, the best triangulation changes to B. If the triangulation can be performed in real-time, the shortest baselines will always be selected, but at the expense of increased complexity. When the dynamics of the platform are not very high, frequent re-triangulation is not necessary. Therefore, a simplified approach can be used for triangulation.
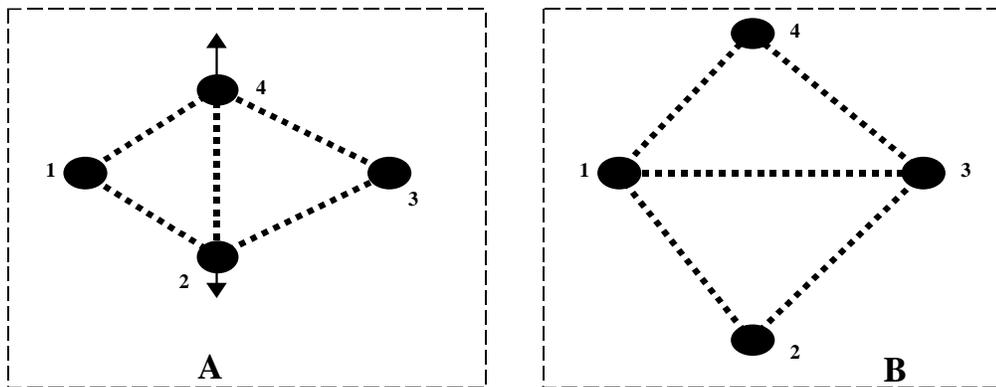
Figure 3.4 Triangulation change in a kinematic network

In the simplified approach, two preconditions are necessary to keep the optimality of triangulation:

[1] GPS platforms have low velocity.

[2] Differential GPS errors decorrelate slowly as a function of receiver separations.

Based on these two assumptions, it can be derived that the low update rate of re-triangulation does not cause an obvious increase of differential errors for a baseline. Consequently, the efficiency, reliability and accuracy of fixing a moving baseline will not be degraded. A re-triangulation period of five minutes is suggested, based on the assumption of 40 km/h velocity (maximum) of vehicles.

3.2.5   Performance of Delaunay Triangulation

Figure 3.5 shows an example of two-dimensional Delaunay triangulation. According to the properties mentioned in section 3.2.2, Delaunay triangulation minimizes the largest circumcircle, and maximizes the minimum angle of all triangles. However, it cannot guarantee that the shortest baselines are always selected. Figure 3.6 shows an example where Delaunay triangulation fails to select the shortest baseline.

In this case, three of the nodes are almost collinear and another node is far away from these three nodes. The criterion of selecting the shortest baseline (Scheme B) leads to a very sharp angle in the triangle, while using Delaunay triangulation (Scheme A), a longer baseline is selected to maximize the minimum angle. This geometry is called the least preferable geometry for MultiKin in the following discussion. Although Delaunay triangulation cannot select the shortest baseline under the least preferable geometry, it does not degrade the efficiency of ambiguity constraints. Monte Carlo tests prove that if all the nodes are averagely distributed in the testing area, the probability of generating this least preferable geometry is small enough to be ignored (less than 1%). In addition, the three baselines that are determined by the criterion of selecting the shortest baselines are almost linearly correlated, which reduces the effectiveness of the ambiguity constraints. Therefore, Delaunay triangulation can still be treated as the optimal method to construct constraints.
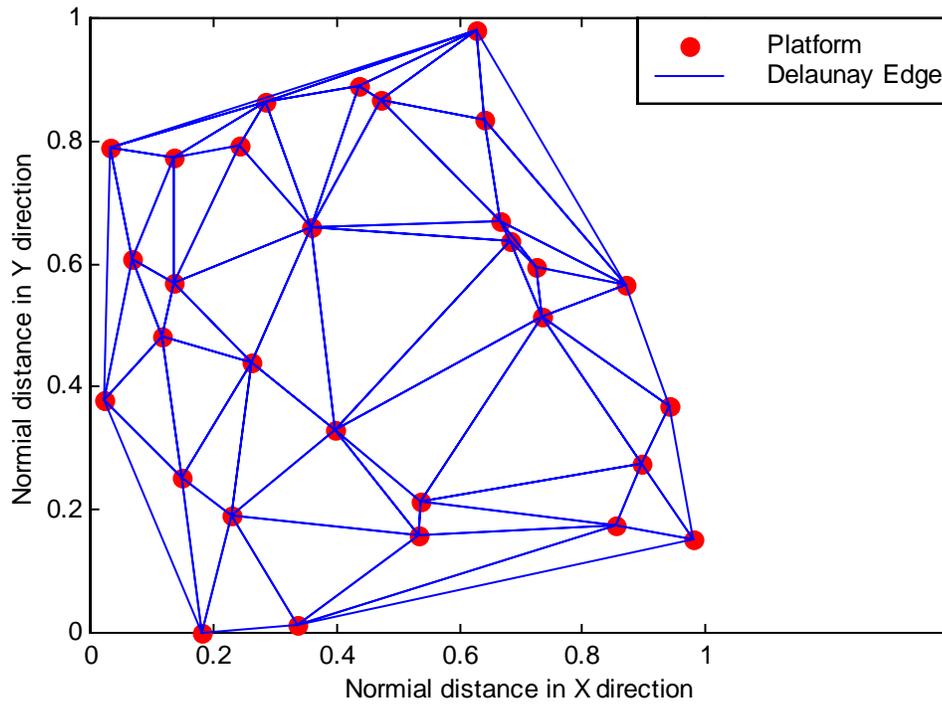
Figure 3.5 Example of 2D Delaunay Triangulation (30 nodes)
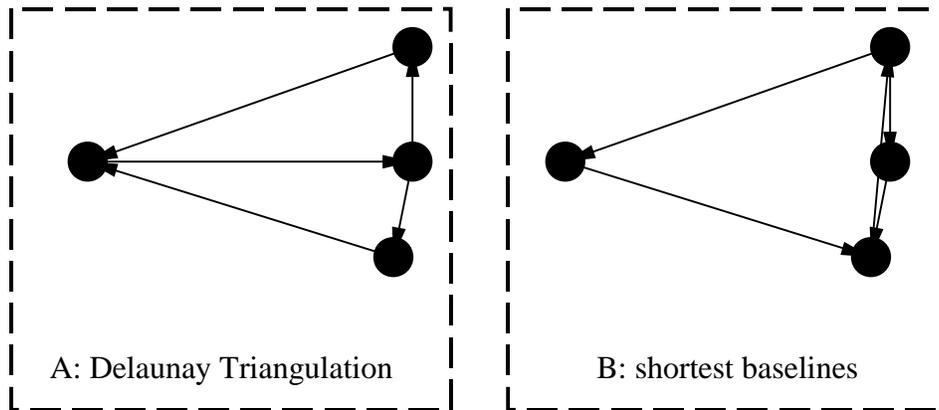
Figure 3.6 Comparison of Delaunay triangulation and shortest-baseline triangulation in the case of the least preferable geometry

Table 3.1 shows the results of Delaunay triangulation of random distributed platforms. Each scenario contains 10000 triangulations. When Delaunay triangulation is used, the

number of baselines to be solved increases very slowly with the increased number of platforms. However, the use of full selection can lead to the numbers of baselines and triangles growing with $O(n^2)$ and $O(n^3)$, which is unacceptable for real-time processing when a configuration consists of more than 50 platforms.

Table 3.1 Computational burden of triangulation (Delaunay Triangulation vs. Full selection)

Number of Platforms	5	10	25	50	100
Mean Number of Baselines (Delaunay Triangulation)	7.75	21.03	63.70	136.85	285.10
Mean Number of Triangles (Delaunay Triangulation)	3.75	12.03	39.70	87.85	186.10
Number of Baselines (full selection)	10	45	300	1225	4950
Number of Triangles (full selection)	10	120	2300	19600	161700

A simulation test was also conducted to compare the effectiveness of the full selection method and the Delaunay triangulation. Six platforms are configured at the vertices and the centre of an equilateral pentagon with the edge length of 1.5 km, see Figure 3.7. The tests were performed assuming a full GPS constellation and average differential errors (see Chapter 4). When using Delaunay triangulation (Scheme A), 10 baselines and five triangular constraints are selected. For the full selection method (Scheme B), 15 baselines and 20 triangular constraints are selected. The mean time to correctly fix ambiguities using Delaunay triangulation is 23.7 s. When using the full selection method, it is 23.4 s. The efficiency improvement brought by the full selection method is 1.3%. Since the full selection method can only introduce a very small improvement in ambiguity resolution while largely increasing the computational burden, the full selection method is not an optimal solution for constraints construction.
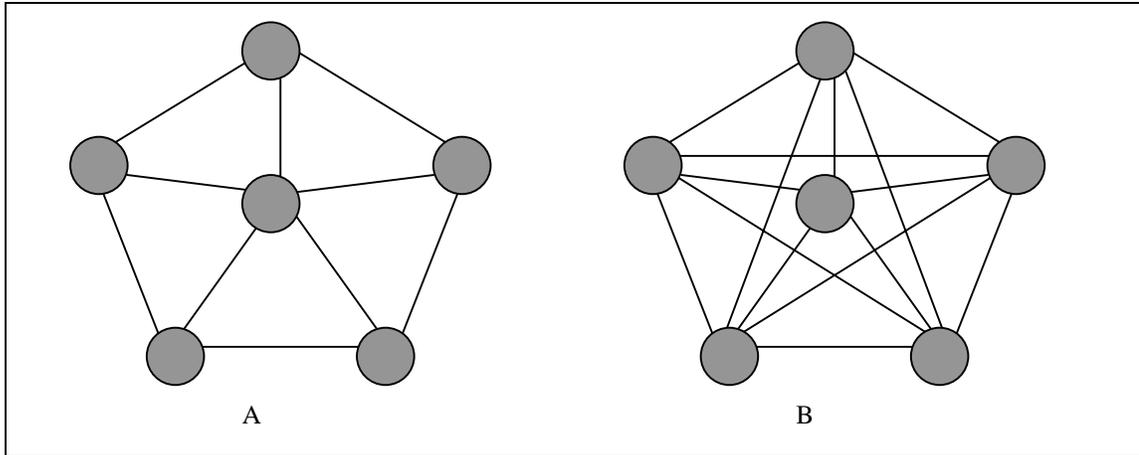
Figure 3.7 Baseline selection by Delaunay triangulation and full selection

3.3   OTF AMBIGUITY RESOLUTION FOR A SINGLE BASELINE

3.3.1   Introduction of Processing Software FLYKIN™

The ambiguity set and the position vector for each baseline are solved using FLYKIN™, a GPS software suite developed by the Department of Geomatics Engineering at the University of Calgary (FLYKIN web). It can process double differenced pseudorange and carrier phase data in either static or kinematic relative positioning mode. It employs Kalman filtering to reduce the observation noise and initially calculate precise float solutions of the carrier phase ambiguities. Successful integer resolution depends on the magnitude of the differential errors and applied observables. The use of dual frequency data in widelaning mode has resulted in successful ambiguity resolution over distances in excess of tens of kilometres under good satellite geometry and quiet ionosphere activity. Single frequency data can result in better accuracy in positioning, but the time to resolution is substantially longer and the effective distance is much shorter (less than 10 to 15 km).

As most positioning software using GPS carrier phase, the heart of FLYKIN™ is its OTF ambiguity resolution algorithm. This consists of three main steps:

[1] ambiguity initialization (float solution)

[2] ambiguity search

[3] distinguishing test

In the first step, a *6+N*-state Kalman filter is employed, where *N* is the number of double differenced ambiguities. This filter can estimate the three-dimensional position, velocity and *N* float ambiguities. It applies a first-order dynamic model to predict the position of a rover station, as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{+} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{-} + \begin{bmatrix} V_X \\ V_Y \\ V_Z \end{bmatrix}^{-} \Delta t \tag{3.6}$$

It also includes two observation equations to update the parameters. One consists of the pseudorange measurements from C/A code; the other consists of the carrier phase measurements. This approach is advantageous to many other approaches using only GPS carrier phase to update the Kalman filter. Although pseudoranges are much noisier than carrier phase measurements, proper design of the observation variances can lead to a faster convergence of the float ambiguities. It has been shown by experiment that updating a Kalman filter with both pseudorange and carrier phase measurements can reduce the ambiguity fixing time by 10%~20% as opposed to pure carrier phase updating.

Float ambiguities together with the corresponding covariances output by the Kalman filter are applied to initialize the ambiguity search space. To improve the efficiency of the search, the least squares ambiguity decorrelation adjustment, LAMBDA (Teunissen, 1994) and the fast ambiguity search filter, FASF (Chen and Lachapelle, 1995) are used to optimize the structure of the search space.

Because only three double differenced ambiguities are independent (Lachapelle 1992), whereas in practice the number of observations is often more than three, strong correlations exist between ambiguities, which can lead to an extremely elongated ambiguity search space. In the LAMBDA method, a matrix transform is used to diagonalize the covariance matrix of ambiguities and to reduce the variances of transformed ambiguities:

$$z = ZN \tag{3.7}$$

$$C_z = ZC_N Z^T$$

where z is the transformed ambiguity vector,

 N is the original ambiguity vector,

 Z is the transform matrix,

 C_N is the original covariance matrix of the ambiguity vector N,

 C_z is the covariance matrix of the transformed ambiguity vector.

However, there are three conditions which the Z-transformation must conform to (Teunissen, 1994). The first is to keep the integer property of the transformed ambiguities. The second condition is that the transformation must preserve the volume of the multi-dimensional confidence ellipsoid. Third, the product of all ambiguity variances must be reduced, otherwise the number of possible ambiguity combinations will not be reduced. The Gauss transformation can satisfy the latter two requirements; however, it is not an integer transformation. To retain the integer values of the ambiguities, the float numbers in the Gauss transformation matrix are rounded to the nearest integer values. Thus, the transformation can only generate a diagonal-dominant instead of a purely diagonal covariance matrix. To maximize decorrelation, rearranging the order of the ambiguities may be necessary.

The remaining correlations in C_z imply that the variances of the transformed ambiguities cannot be minimized. Fortunately, FASF is a method which can make use of the remaining correlation between ambiguities. It applies the conditional least squares principle to further reduce the variance of ambiguities, i.e., if an ambiguity is assumed correctly fixed, the variances of other ambiguities can be reduced due to their correlations with the fixed ambiguity. Equation (3.7) shows the relationship between the conditional and non-conditional variances of parameters.

$$C_{\tilde{x}} = C_{\hat{x}} - c_n c_n^T /(C_{\hat{x}})_{n,n} \tag{3.8}$$

where $\tilde{x}$ are the estimated parameters assuming the nth parameter (ambiguity) has been fixed,

$\hat{x}$ are the original parameters,

c_n is the nth column of the covariance matrix $C_{\hat{x}}$,

$(C_{\hat{x}})_{n,n}$ is the variance of nth parameter (ambiguity).

It has been proved by the theorem in conditional least squares that the variances of $\tilde{x}$ are less than the variances of $\hat{x}$. Thus, the search space for other ambiguities can be reduced.

The distinguishing test is performed to isolate the correct integer ambiguity set from the other candidates. A ratio test defined below is used in FLYKIN™ for this purpose:

$$F = \frac{\Omega(\tilde{N}^2)}{\Omega(\tilde{N}^1)} > F_T \tag{3.9}$$

where $\tilde{N}^1$ is the best integer ambiguity candidate,

$\tilde{N}^2$ is the second best integer ambiguity candidate,

$\Omega(\tilde{N}) = (\tilde{N} - N_{float})^T C_{float/\tilde{N}}^{-1} (\tilde{N} - N_{float})$, and

$C_{float/\tilde{N}}$ is the conditional covariance matrix for the float ambiguities.

This ratio test has the same form as a Fisher test, however, the necessary assumptions of Gaussian distribution and independent variables made for the Fisher test are not satisfied in most GPS applications. Thus, the relationship between the probability of wrong fixing and threshold F_T is not yet clear. Current threshold is set to 4.0 according to empirical results.

3.3.2 Modification of FLYKIN™

3.3.2.1 Addition of Stand-alone Positioning Module

To accurately solve the relative position vector between two platforms, the approximate position of one platform should be known. This can be explained from the observation equation of double differenced carrier phase measurements, where the coordinates of the reference station (or approximate coordinates of the rover station) are used to compute the linearized design matrix and misclosure of the observation equation.

In a configuration which only contains multiple moving platforms, the precise coordinates of the moving reference station cannot be known *a priori*. Therefore, a stand-alone positioning module is added to FLYKIN™ to supply the approximate coordinates of the moving reference station, see Figure 3.8.
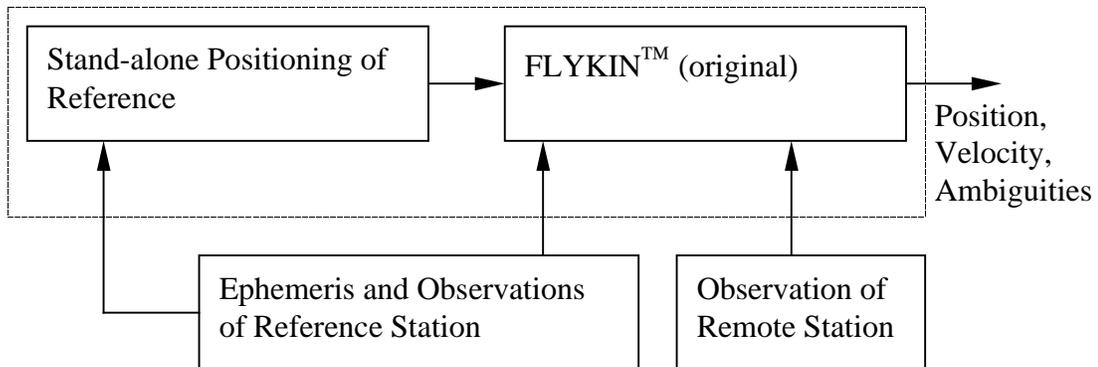
Figure 3.8 Relative positioning of a moving baseline by FLYKIN™

The iteration algorithm used for stand-alone positioning is as follows.

[1] Linearize observation equation

$$\bar{P} = \bar{\rho}_0 + A\bar{X} \tag{3.10}$$

where $\bar{P}$ is the pseudorange measurement vector from C/A code,

$\bar{X} = \begin{bmatrix} \Delta x & \Delta y & \Delta z & \Delta T \end{bmatrix}^T$ is the correction vector for position and receiver clock,

$\vec{\rho}_0$ is the distance vector from satellites to an approximate reference point $\begin{bmatrix} x_0 & y_0 & z_0 \end{bmatrix}$, $\vec{\rho}_0$ can be written as:

$$\vec{\rho}_0 = \begin{bmatrix} \sqrt{\left(x_s^1 - x_0\right)^2 + \left(y_s^1 - y_0\right)^2 + \left(z_s^1 - z_0\right)^2} \\ \sqrt{\left(x_s^2 - x_0\right)^2 + \left(y_s^2 - y_0\right)^2 + \left(z_s^2 - z_0\right)^2} \\ \vdots \\ \sqrt{\left(x_s^n - x_0\right)^2 + \left(y_s^n - y_0\right)^2 + \left(z_s^n - z_0\right)^2} \end{bmatrix} \qquad (3.11)$$

where $\begin{pmatrix} x_s^i & y_s^i & z_s^i \end{pmatrix}$ is the position of the ith satellite.

[2] Calculate the correction using least squares

$$\bar{X} = \left[A^T C_P^{-1} A \right]^{-1} A^T C_P^{-1} \left(\bar{P} - \bar{\rho}_0 \right) \qquad (3.12)$$

Generally, the covariance matrix is a unit matrix because errors of each measurement are assumed to have the same variance and be independent of each other.

[3] Update the position of the reference station

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \Rightarrow \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \qquad (3.13)$$

The updated position is used to construct a new linear observation in Equation (3.10). The iteration will not stop until the position correction is less than a certain value, e.g., 1.0 millimetre.

The accuracy of stand-alone positioning with C/A code was very poor due to the presence of SA until May 2000. The horizontal position error of the reference station could be up to 100 m (95%), while the vertical could be up to 150 m (95%). Even if SA has been turned off, the second largest error, ionospheric delay can still result in a large positioning error when the solar activity is very strong. This error can also cause serious accuracy degradation in relative positioning.

3.3.2.2 Accuracy Degradation of Relative Position

When converting the relative position from an Earth-fixed frame to a local-level frame, the transformation can produce an extra error in relative positioning. In Figure 3.9, O_1 is the true reference station's position and O_2 is the computed position of O_1 computed by the stand-alone positioning module. They define two local-level frames with different orientations, so the projection of the relative positioning vector (the Earth frame) on a different local level frame is different.
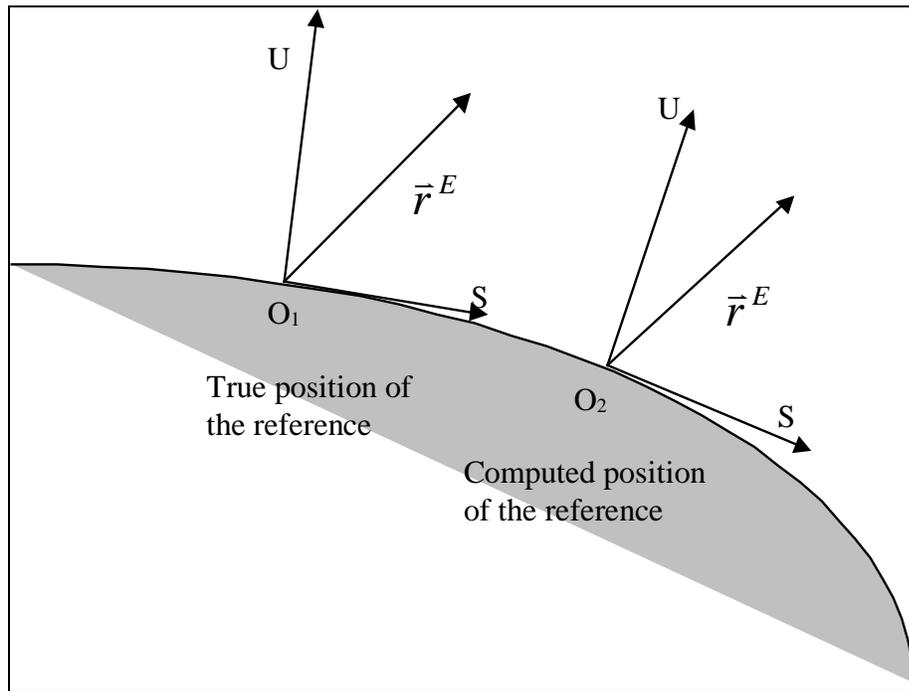
Figure 3.9 Error in defining the local level frame caused by a reference error

The following equation is applied to convert the relative position vector from the Earth-fixed frame to a local level frame.

$$\vec{r}^{\,L} = R_E^L \vec{r}^{\,E} \tag{3.14}$$

where $\vec{r}^{\,L}$ is the relative position vector in local level frame,

$\vec{r}^{\,E}$ is the relative position vector in Earth frame, and

R_E^L is the rotational matrix to convert coordinates from Earth frame to local level frame:

$$R_E^L = \begin{bmatrix} -\sin\lambda & \cos\lambda & 0 \\ -\sin\phi\cos\lambda & -\sin\phi\sin\lambda & \cos\phi \\ \cos\phi\cos\lambda & \cos\phi\sin\lambda & \sin\phi \end{bmatrix} \qquad (3.15)$$

where ϕ and λ are the latitude and longitude of the reference station respectively.

The error of the relative position vector in the local level frame can be described as:

$$\Delta\vec{r}^L = R_E^L\Delta\vec{r}^E + \Delta R_E^L\vec{r}^E \qquad (3.16)$$

$\Delta\vec{r}^E$ is the relative position error in the Earth frame, which is caused by the double differenced errors and the reference error. It can be referred to Tang (1996) that a 10 m reference error can approximately cause 1 ppm baseline error in relative positioning.

The second term in Equation (3.16) results from the error in the rotational matrix, which is actually caused by the reference error. Ignoring the second-order error terms, the extra relative positioning error $\Delta R_E^L\vec{r}^E$ caused by a reference position error can be approximated as:

$$\Delta\vec{r}_{extra}^L = \Delta R_E^L\vec{r}^E \approx (\Delta\phi R_1 + \Delta\lambda R_2)\vec{r}^E \qquad (3.17)$$

$$R_1 = \begin{bmatrix} 0 & 0 & 0 \\ -\cos\phi\cos\lambda & -\cos\phi\sin\lambda & -\sin\phi \\ -\sin\phi\cos\lambda & -\sin\phi\sin\lambda & \cos\phi \end{bmatrix}$$

$$R_2 = \begin{bmatrix} -\cos\lambda & -\sin\lambda & 0 \\ \sin\phi\sin\lambda & -\sin\phi\cos\lambda & 0 \\ -\cos\phi\sin\lambda & \cos\phi\cos\lambda & 0 \end{bmatrix}$$

where $\Delta\phi$ and $\Delta\lambda$ are the latitude and longitude errors of the reference.

From Equation (3.17), it can be seen that the extra relative position error $\Delta \vec{r}_{extra}^{L}$ is proportional to the baseline length and the magnitude of the reference error. In the worst case, a 100 m horizontal positioning error of the reference can cause the 3D relative position accuracy (RMS) to degrade at a rate of 20 mm/km (20 ppm). It should be noted that in the following discussion, the accuracy degradation in relative positioning is always referred to as $\Delta \vec{r}_{extra}^{L}$, not $\Delta \vec{r}^{E}$.

Figure 3.10 shows the relationship between the degradation of the relative positioning accuracy and the absolute position error of the reference station for a 1.5 km baseline.
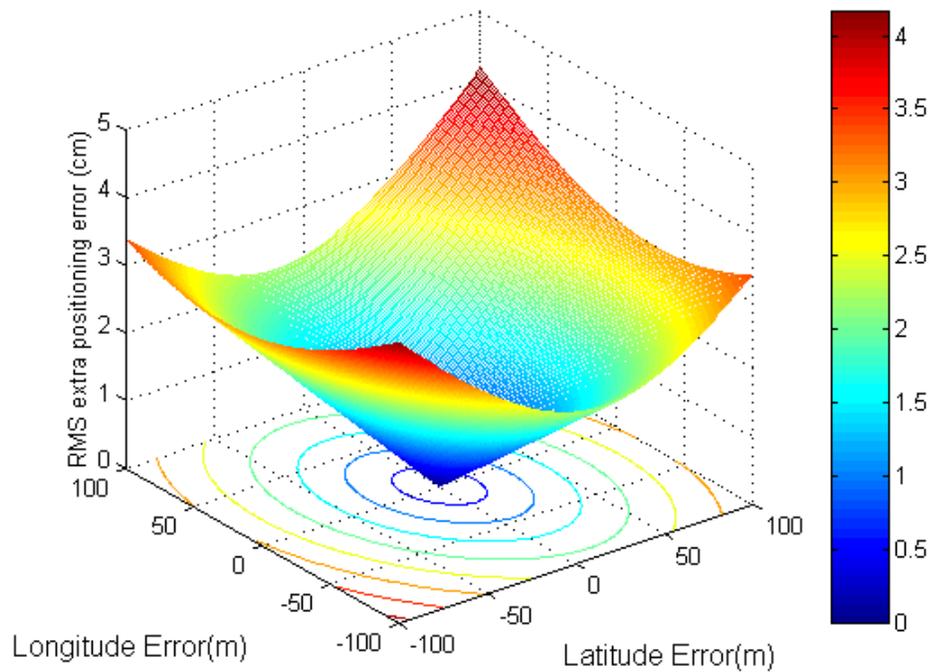
Figure 3.10 Degradation of relative positioning accuracy due to the reference error – 1.5 km baseline

The error degradation rate is calculated at a mid-latitude ($\phi \approx 51°$) point. When calculating at a higher latitude point, the degradation worsens. Figure 3.11 shows the degradation rate of relative positioning accuracy with respect to the latitude. At high latitude (80°), the degradation rate can be as large as 60 ppm. For a 10 km baseline, a 100

m reference error can result in a 60 cm extra error in relative positioning, which is unacceptable for precise positioning. Therefore, the reference error should be reduced as much as possible to mitigate accuracy degradation. One way to reduce the reference error is to use DGPS to obtain range corrections from a static reference station, which can keep the accuracy of a moving reference at the metre level.
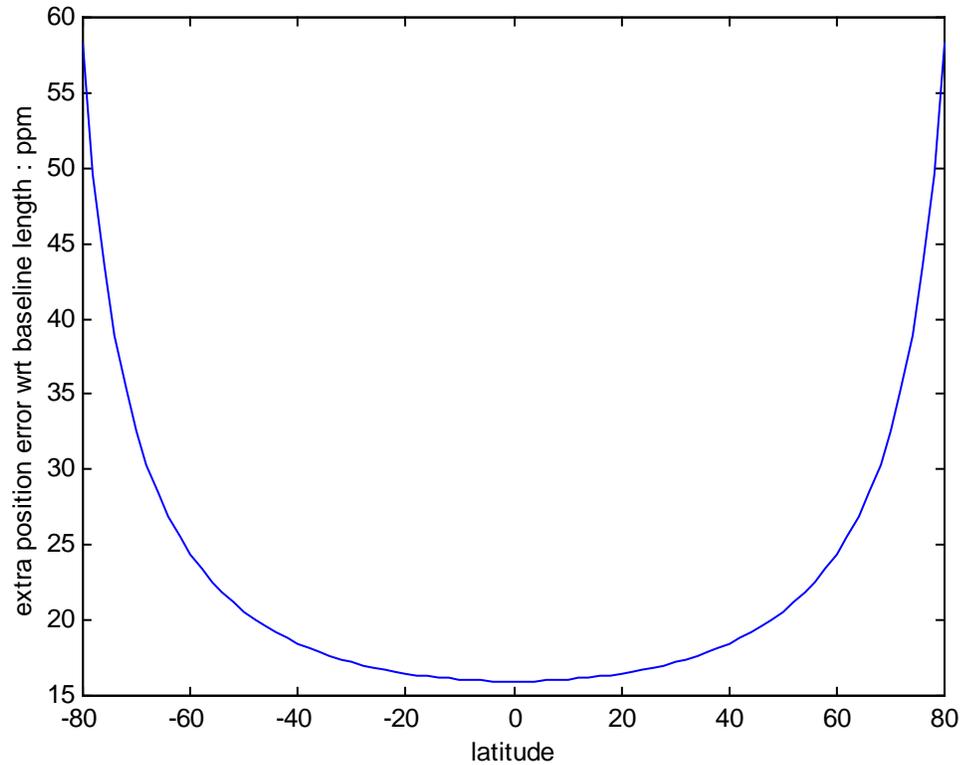
Figure 3.11 Degradation of relative positioning accuracy over latitude

3.3.2.3 *Modification of the Residual Test in FLYKIN™*

In FLYKIN™, a residual test is performed after the ambiguities are fixed. This test is used to monitor whether the integer ambiguities are fixed correctly. Because large errors, especially errors with a bias nature, can cause wrong fixes, the positioning accuracy can seriously be degraded. The previous tests show clearly that only the correct integer solution can provide centimetre level accuracy. Therefore, the correctness of the ambiguity set is crucial to high accuracy positioning. One of the commonly used methods

to check the correctness of the solution is the residual test. Figure 3.12 shows the functional diagram of the residual test implemented in FLYKIN™.
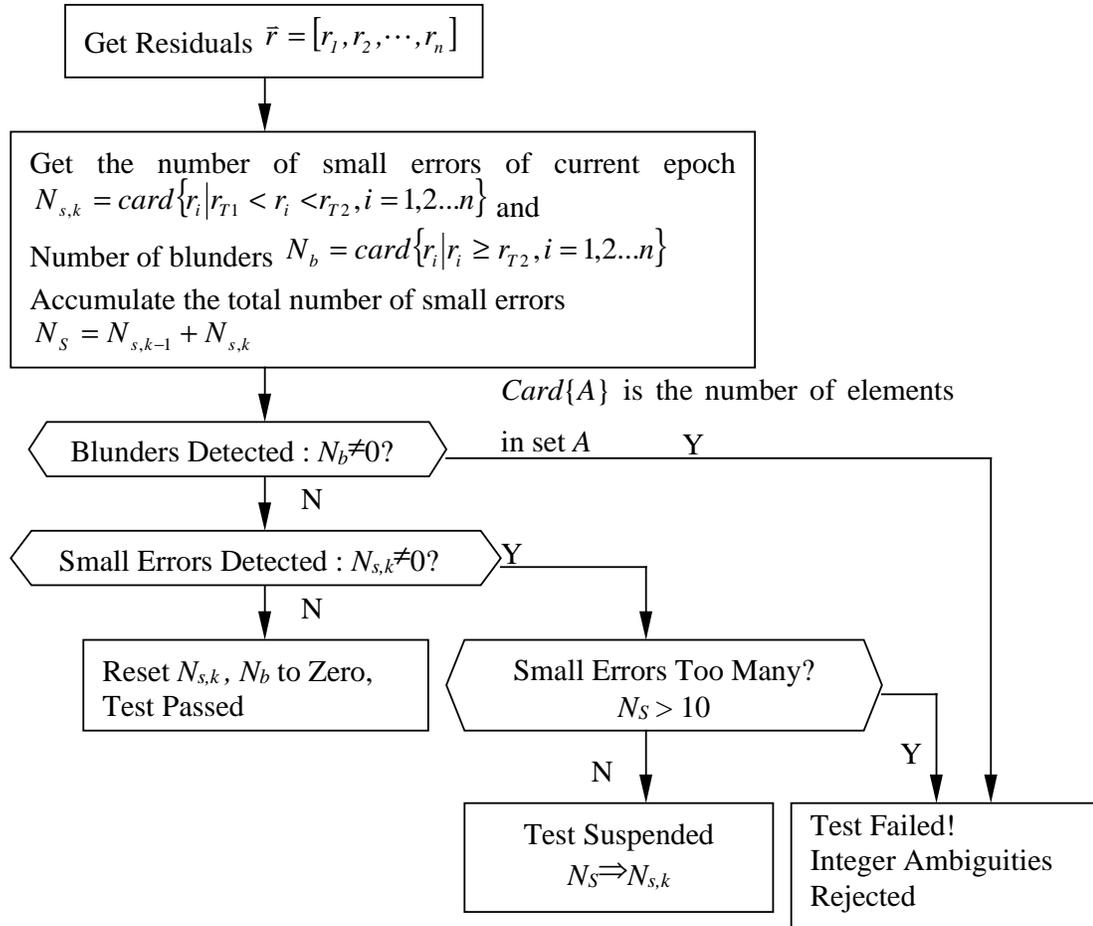
Figure 3.12 Diagram of residual test in FLYKIN™

It should be noted that the setup of two parameters is very important to the effectiveness of the residual test, namely, the two residual thresholds r_{T1} and r_{T2}. They are used to identify the small errors and blunders in estimation. In FLYKIN™, the empirical values are given as (3.5 cm, 5.0 cm) for a "short" baseline (<1000 m) and (5.0 cm, 7.0 cm) for a "long" baseline (>1000 m). However, results of simulation tests indicated that this residual test is not effective enough for detecting wrong fixes.

A simulation test of 4320 five km baselines was performed to evaluate the residual test in FLYKIN™. The simulated baselines were conducted assuming a full constellation and

differential errors considered average except for the ionosphere when a RMS differential error of about 5 ppm was assumed (The differential error models are described in Chapter 4). L1-only observables were used in data processing. In total, 267 wrong fixes occurred in those tests. Figure 3.13 shows the histogram of the time to detect wrong fixes using the residual test. The mean time to detect wrong fixes was 427.6 s. The last bar in Figure 3.13 corresponds to the frequency of those wrong fixes which could not be detected within 20 minutes. The undetectable rate is more than 8%.
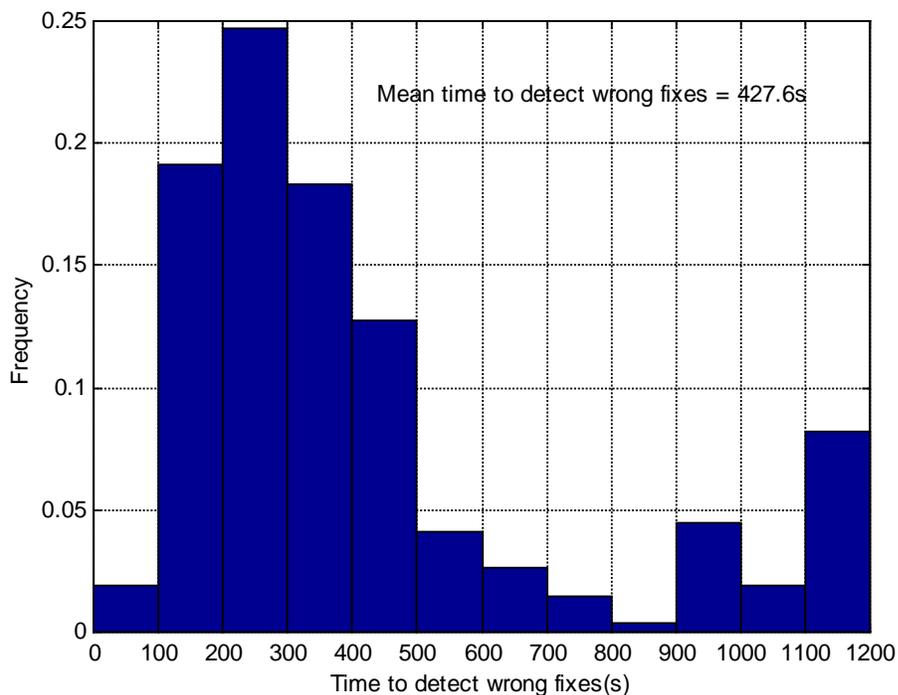
Figure 3.13 Time to detect wrong fixes using residual test (5000 m baseline)

Figure 3.14 shows the resulting histogram from a group of analogous simulation tests with a baseline length of 10 km. In total, 463 wrong fixes occur. In these tests, the mean time to detect wrong fixes decreases and only 6% of the wrong fixes are undetectable. This is because the residual error becomes large when the differential error is increased. However, the improved performance of detecting wrong fixes is obtained at the expense of increasing the probability of false alarm, i.e., correct fixes are detected as wrong fixes and float ambiguities are used instead. False alarms result in the instability of the fix solution and degrade the positioning accuracy.
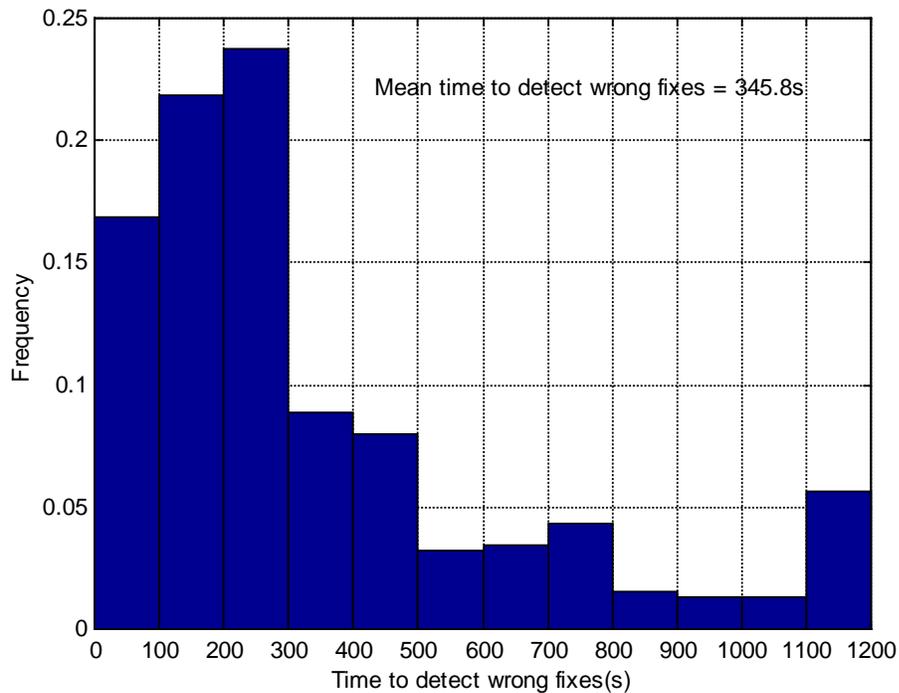
Figure 3.14 Time to detect wrong fixes using residual test (10 km baseline)

To fully evaluate the residual test, another simulation test was designed to check the false alarm probability. In total, 360 baselines were tested. Each baseline was tested for one hour and the ambiguities were initialized to the correct integers. When the baseline length is five kilometres, no false alarm is generated, while for the 10 km baseline, fifteen false alarms (4.2%) occur due to the failure of the residual test.

Good performance of the residual test depends on the compatibility of the residual threshold and the magnitude of the measurement errors. Only when these two values are matched can the best compromise be obtained between the probability of missing detection and the probability of false alarms. Therefore, an optimal residual test must have adaptive thresholds. However, it is very difficult to get *a priori* information about the magnitudes of errors. Although this information can be approximately represented by the variance-covariance matrix from a Kalman filter, the estimation is only valid when the errors are Gaussian in nature and zero-mean, which is rarely true for GPS. When adaptive thresholds cannot be applied, the residual test cannot work well.

To get a better compromise between detecting wrong fixes and minimizing false alarms, a ratio test is conducted instead of the residual test. This ratio test uses the same concept as the one used in the distinguishing test. The following condition is used to detect wrong fixes:

$$\frac{\Omega_{int}}{\Omega_{best}} > Threshold \tag{3.18}$$

where $\Omega_X = \left(N_{float|X} - N_X\right)^T C_{float|X}^{-1} \left(N_{float|X} - N_X\right) \quad X \in \{best, int\},$

$N_{float|X}$ is the conditional float solution of ambiguities,

$C_{float|X}$ is the covariance matrix of conditional float ambiguities,

N_{best} is the current best candidate of integer ambiguities,

N_{int} is the current fixed integer ambiguities, and

Threshold=4.0

When the current fixed integer solution is distinctly poorer than the current best integer candidate, i.e., the condition in Equation (3.18) is satisfied, then the current integer ambiguities are treated as incorrect. The ratio test has an obvious advantage over the residual test. It does not need any information about the absolute magnitude of the measurement error because the threshold of ratio reflects the relative difference between the two integer solutions.

The experimental results (Figure 3.15 and Figure 3.16) show that the ratio test is much more effective than the residual test. In the five km baseline case, the ratio test can detect wrong fixes much faster than the residual test (118.4 s vs. 427.6 s) and all wrong fixes can be detected within 15 minutes. In the 10 km baseline case, the ratio test takes a slightly longer time to detect wrong fixes (371.4 s vs. 345.8 s), but only 0.43% wrong fixes cannot be detected within 20 minutes. In addition, the result of the false alarm test

indicates that no false alarm is generated by the ratio test. Therefore, the correct integer ambiguity can be reliably fixed.

Mean time to detect wrong fixes = 118.4s
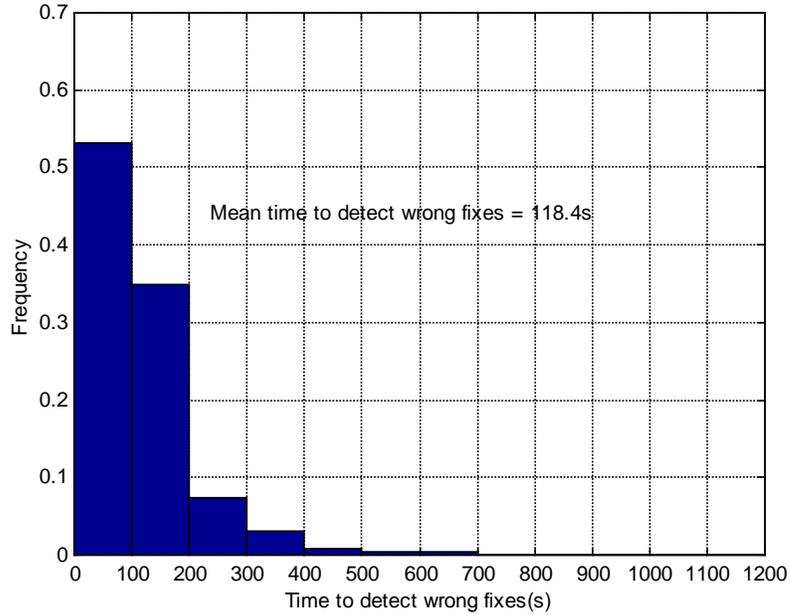
Figure 3.15 Time to detect wrong fixes (5 km baseline, ratio test)

Mean time to detect wrong fixes = 371.4s
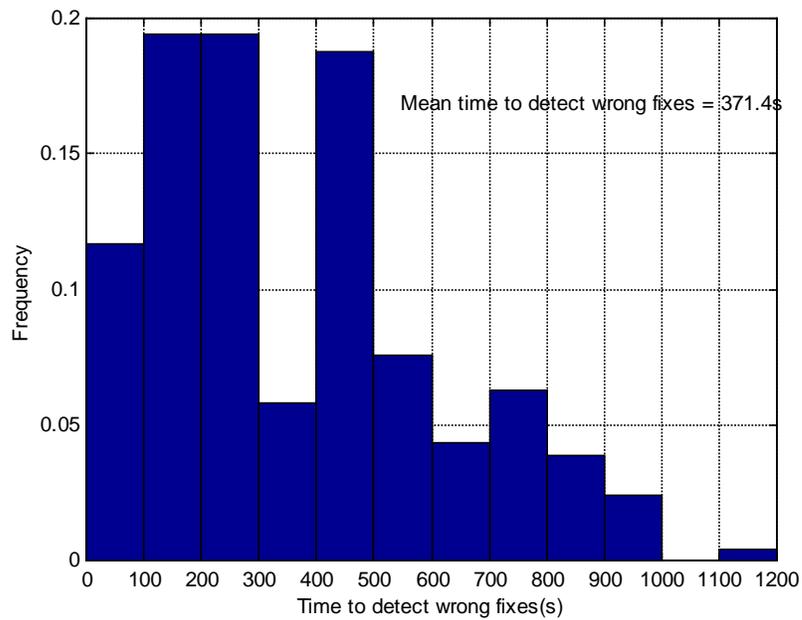
Figure 3.16 Time to detect wrong fixes (10 km baseline, ratio test)

It is worth mentioning that the value of the threshold in the ratio test is the same as that in the distinguishing test, which implies that the new integer ambiguities are fixed at the

moment when the wrong fixes are detected. However, for the residual test, it can still take a long time to fix ambiguities after detecting wrong fixes.

Figure 3.17 shows the time comparison of the ratio test and residual test to acquire the correct integer ambiguity when initial ambiguities are incorrectly fixed. The ratio test can save three minutes to correct the wrong fixes on average. In addition, the percentage of unfixed ambiguities within 20 minutes decreases from 8.64% to 0.43% after using the ratio test. Thus, in MultiKin, the residual test for single-baseline ambiguity monitoring is replaced by the ratio test.
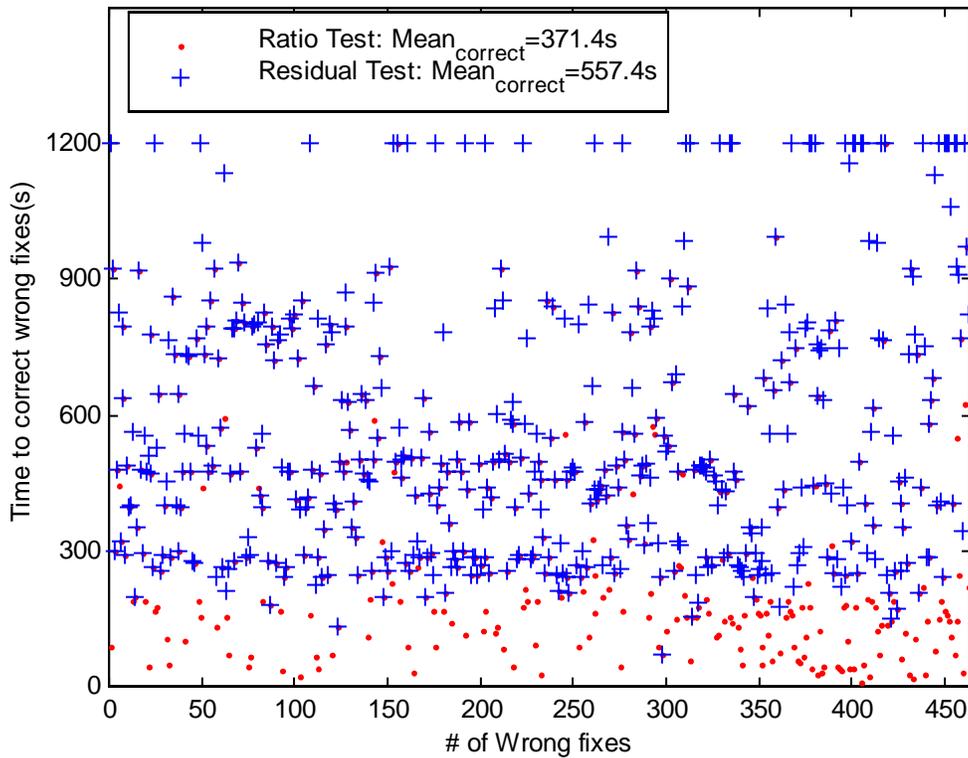
Figure 3.17 Comparison of time to correct wrong fixes (ratio test vs. residual test) in case of the 10 km baselines

To use the ratio test, N_{best}, the current best candidate of the integer ambiguities is indispensable. This requires another modification to the original version of FLYKIN™. When using the ratio test, the ambiguity searching procedure is always necessary, while

in the original version of FLYKIN™, the ambiguity searching is only performed when ambiguities are not fixed or after wrong fixes are detected. Figure 3.18 compares the functional diagrams of FLYKIN™ before and after using the ratio test.
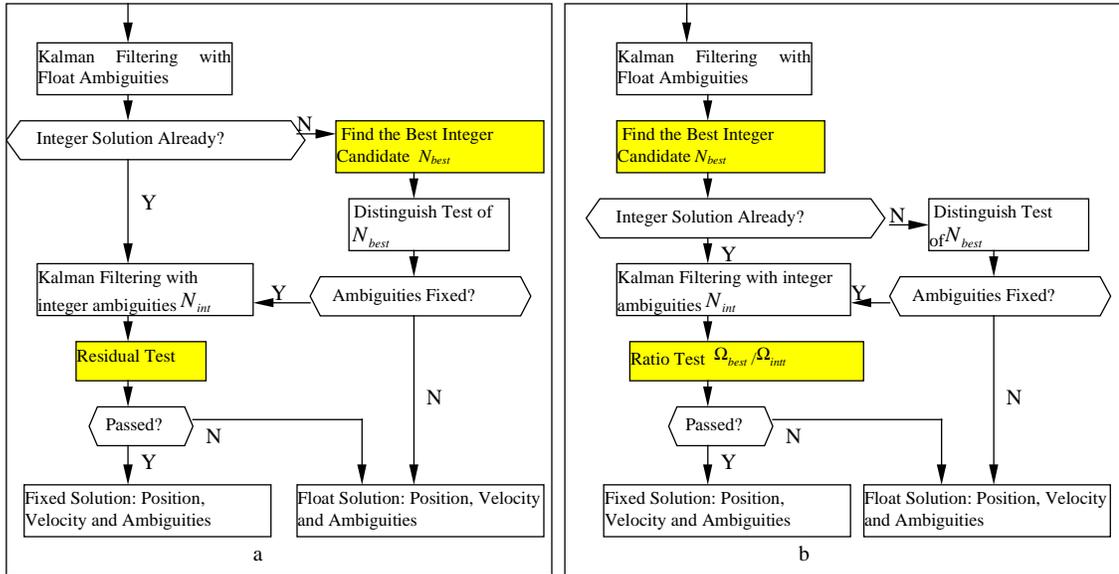
Figure 3.18 Modification of FLYKIN™ from residual test to ratio test

3.4   AMBIGUITY DETERMINATION/MONITORING USING CONSTRAINTS

The application of ambiguity constraints of integer ambiguities was first suggested by Lachapelle et al (1993). Luo and Lachapelle (1999) have studied its application in the case of three moving platforms. Herein, the approach is generalized to a configuration of more than three platforms. Figure 3.19 shows the procedure of ambiguity determination and monitoring using multiple triangular constraints in MultiKin. The detailed discussion of this procedure is presented in the following section.
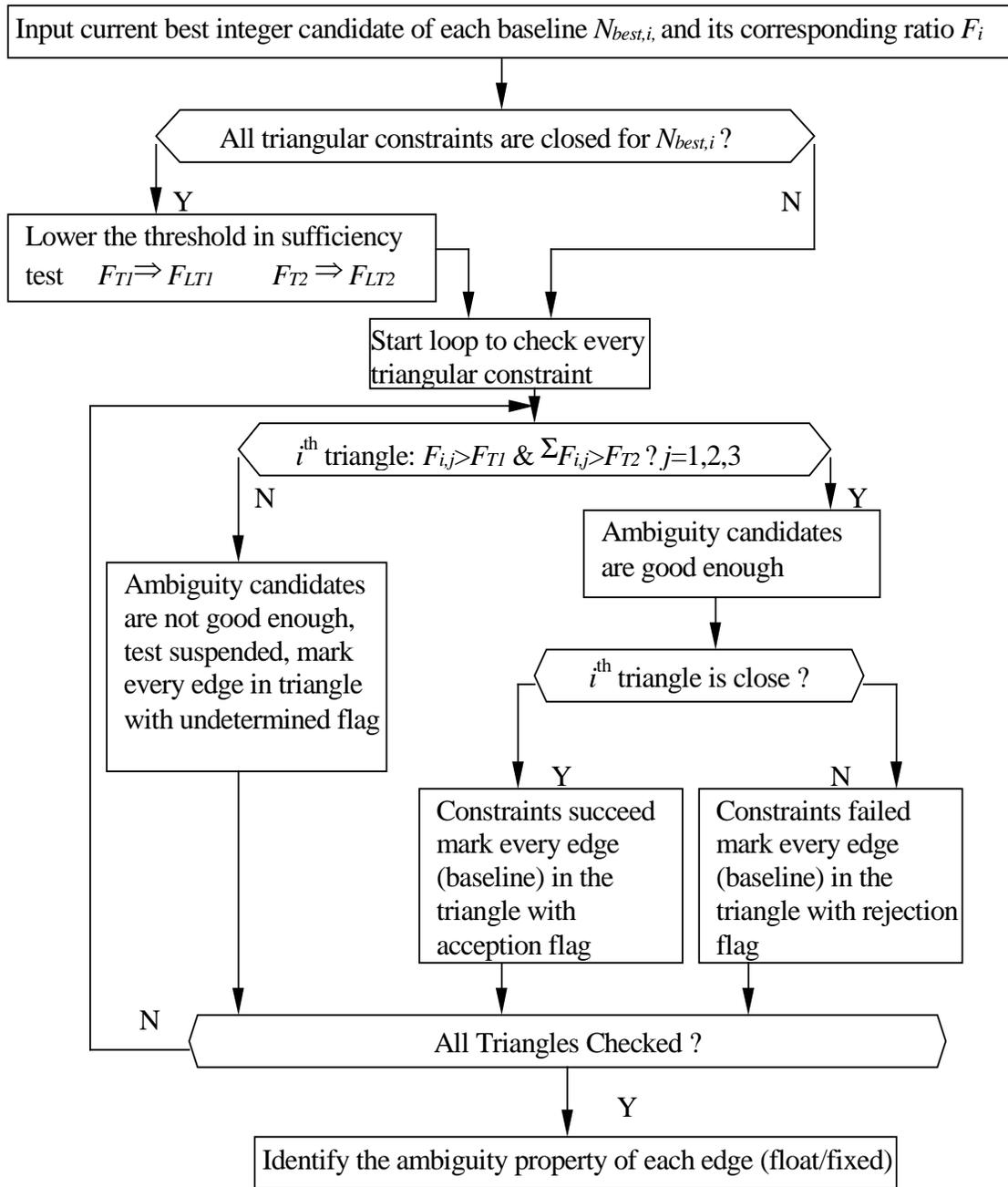
Figure 3.19 Procedure of ambiguity determination and monitoring using multiple triangular constraints

3.4.1 Synchronization of Observations

In order to use the ambiguity constraints, all the moving platforms must have common observations and the same selection of base satellite. In MultiKin, this is called

synchronization of observations. Generally, the visibility of satellites in a small area (20 km × 20 km) is constant; however, in practice, due to the complexity of the observation environment, the blockage of satellite signal varies at each station. Thus, the number of common satellites also varies from baseline to baseline.

Figure 3.20 shows an example. The bold number is the visible satellite number for a moving platform, the italic number is the visible satellite number for both platforms of a baseline, and the number of the regular font is the satellite number common to a triangle. Although the total number of visible satellites is nine, there are only five common satellites applicable for constraints. The number of common satellites in a triangle affects the effectiveness of the constraint. Generally, the greater the number of commonly observed satellites, the higher the efficiency and reliability in resolving ambiguities. The impact of satellite visibility and geometry on ambiguity resolution will be further discussed in Chapter 6.
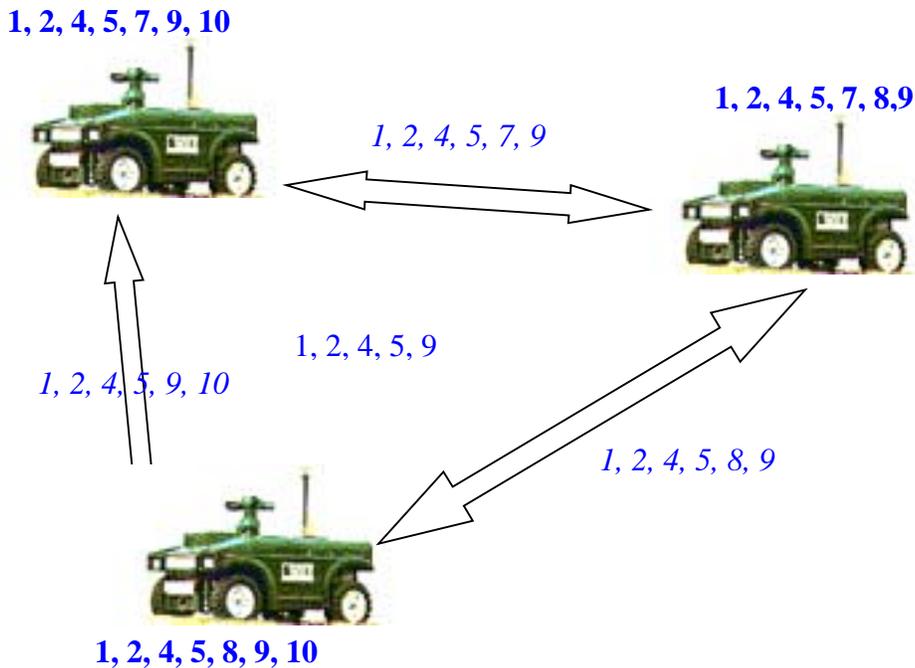
Figure 3.20 Synchronization of constrained observations

3.4.2　Sufficiency Test

In the module for multi-platform ambiguity determination/monitoring, two tests are conducted to control the effectiveness of fixing ambiguity and detecting wrong fixes. The first is the necessity test, namely, the closure of the triangular ambiguities. The second is the sufficiency test, which is used to assure the reliability of the solutions. In the second test, two criteria are used to define the sufficiency of a triangular constraint:

$$\begin{cases} F_1 > F_{T1} \\ F_2 > F_{T1} \\ F_3 > F_{T1} \end{cases} \tag{3.19}$$

$$F_1 + F_2 + F_3 > F_{T2} \tag{3.20}$$

where F_i is the ratio defined in Equation (3.9). The threshold F_{T1} is set up to 2.0 according to empirical results. It is much lower than that used originally in FLYKIN™ (F_{T1}=4.0) because more information is integrated for ambiguity determination.

The first criterion is used for individual sufficiency check. The most likely integer ambiguity $N_{best,i}$ solved for each baseline is good enough to be a candidate for multi-platform ambiguity determination if the first condition is satisfied. The second criterion is to check whether the combination of all $N_{best,i}$ has a high confidence level. It should be noted that these two criteria are not equivalent.

There are several ways to define criteria of sufficiency. For example, the product of ratios was used by Luo and Lachapelle (1999) to evaluate the sufficiency of integer ambiguity sets in a triangle. However, a large number of simulation tests show that the sum of ratios is more effective for ambiguities fixing and monitoring, because its threshold setup is less sensitive to the changes of scenarios (Luo, 2000).

To further improve effectiveness, a criterion for adaptive threshold adjustment is added in this algorithm. According to the theory of probability and statistics (Kendall and Stuart, 1968), conditional probability is less than non-conditional probability, i.e.,

$P(X) > P(X|H)$. Provided that all the triangular constraints are closed, the conditional probability of wrong fixes can be largely reduced when compared to the original probability of error. In this case, if the threshold can be properly lowered, more rapid ambiguity resolution can be obtained without increasing the error probability (see Figure 3.21). However, this is extremely difficult to realize since the analytical expression of the multi-dimensional conditional *pdf* (probability density function) of ratio $f(F_i|H)$ is unknown.
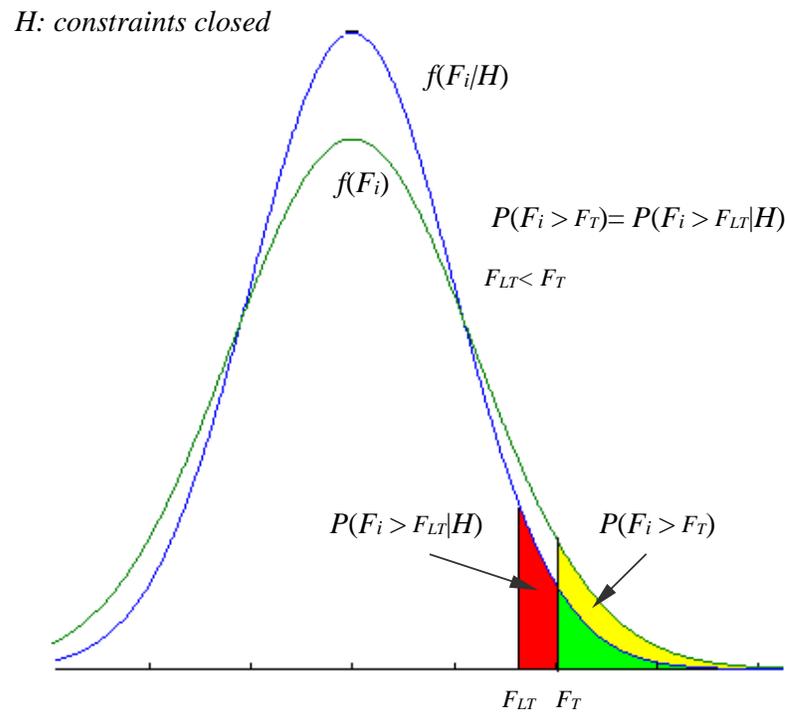
Figure 3.21 Comparison of conditional and non-conditional probability

The setup of the adaptive threshold completely depends on empirical results. Based on several simulation tests, the following equation is used to define the conditional threshold for the case when all closed triangular constraints exist:

$$F_{LT} = \sqrt{\frac{N_S}{N_B}} F_T \tag{3.21}$$

where *Ns* is the number of the total moving platforms, and

N_B is the number of the total selected baselines (Delaunay edges).

A simulation test is designed to verify the method of threshold adjustment, where 4320 five km baselines are fixed under normal atmospheric and multipath conditions. Figure 3.22 shows the effect of adjusting the threshold, where a 2.3% improvement in the fixing speed can be observed. Meanwhile, the adjustment also causes the probability of wrong fixes to slightly increase from 0.1028 to 0.1035. Since the degradation of reliability is relatively small (0.7%) with respect to the improvement in efficiency, Equation (3.21) can be treated as an effective adjustment of the threshold.
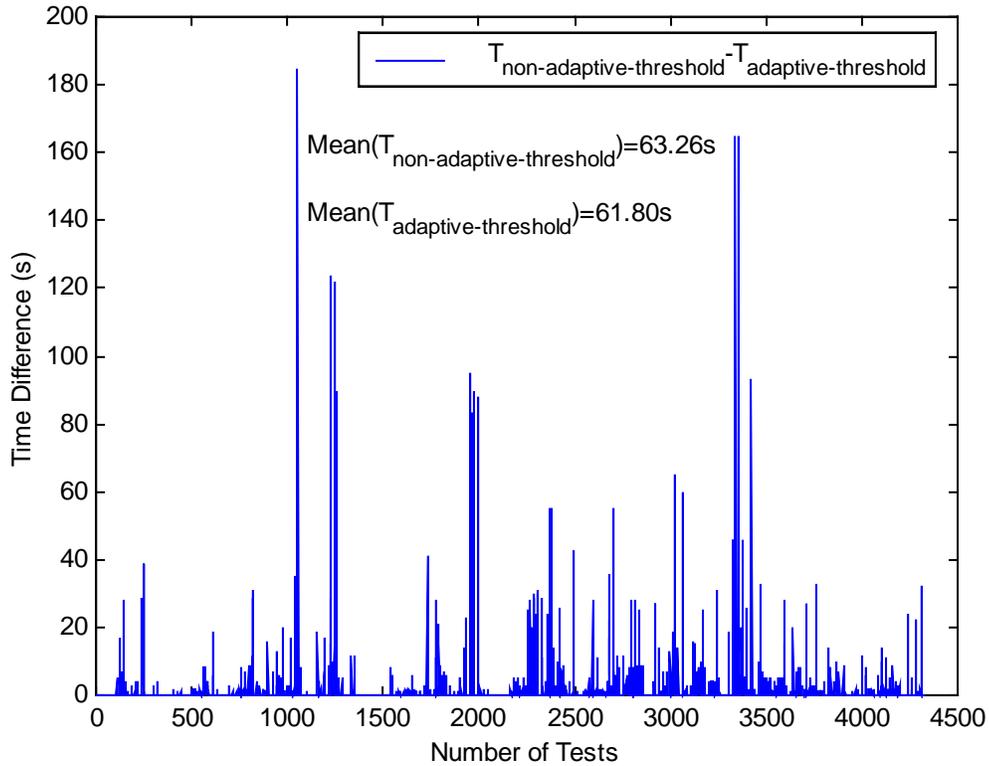
Figure 3.22 Difference of Time to fix ambiguity (fixed threshold vs. adaptive threshold)

3.4.3 Integrated Ambiguity Determination

If the sufficiency test fails, the current combination of the best integer ambiguity sets is not reliable enough to be used for constraints, and more observations are needed. Consequently, the ambiguity determination module of MultiKin will keep the original ambiguity solution from FLYKIN™. If both of the above two criteria are satisfied, i.e., the sufficiency test is passed, ambiguity constraints will be used to verify the correctness of the combination of the integer ambiguity sets. If all the triangle constraints are satisfied, then the combination will be treated as the true integer ambiguity solution, i.e., ambiguity sets for all baselines are assumed correct. Otherwise, the ambiguity set of each baseline will be determined by the ambiguity identification module as described below.

When multiple triangular constraints are used, one selected baseline (Delaunay edge) can be shared by two Delaunay triangles. Thus, it is possible that the integer ambiguity set of a baseline is rejected by one triangular constraint but accepted by the other. Whether to reject or accept the integer solution is determined by the ambiguity identification module. According to the properties of Delaunay triangulation, i.e., no overlap triangles, one baseline can belong to at most two triangles. The algorithm defined below is then used to decide the state of the final solution.

[1] Get the determination flags (d_j) of the ambiguities at each baseline in each triangle.

$$d_j = \begin{cases} 1 & Accepted \\ 0 & Undetermined \\ -1 & Rejected \end{cases} \quad j = 1,2...m-1 \tag{3.22}$$

where m is the number of common visible satellites, and m-1 is the number of common double differenced observations in a triangle.

[2] Get the total determination flag (Td_j) of the ambiguities of each baseline.

$$Td_j = \sum_{k=1}^{K} d_{j,k} \tag{3.23}$$

where K is the total number of triangles that a baseline belongs to. K=1 or 2.

[3] Calculate the final determination flags (Fd_j) of ambiguities of a baseline.

$$Fd_j = \begin{cases} 1 & Td_j \geq 1 \\ 0 & Td_j = 0 \\ -1 & Td_j \leq -1 \end{cases}$$

(3.24)

where Fd_j shows whether the jth ambiguity of a baseline is accepted(1), undetermined(0) or rejected (-1) by the triangular constraint(s).

[4] Calculate the ambiguity determination flags of a baseline (Bd).

$$Bd = \sum_{j=1}^{m-1} Fd_j$$

(3.25)

[5] Determine whether ambiguities of a baseline are fixed.

$$ambiguities\ of\ a\ baseline \begin{cases} fixed\ ambiguities & Bd \geq 3 \\ original\ FLYKIN\ solution & -2 \leq Bd \leq 2 \\ float\ ambiguities & Bd \leq -3 \end{cases}$$

(3.26)

If at least three double differenced ambiguities are accepted after applying constraints, the best integer ambiguity candidate of this baseline is selected as the fixed solution. If at least three ambiguities are rejected by constraints, the float ambiguities will be used in positioning. If the ambiguities are undetermined by using constraints, the solution of FLYKIN™, which can be either fixed or float, will be kept.

The threshold of Bd, which determines the effectiveness of the constraints, must be properly set up. If the threshold is too high, constraints will not help to speed up the ambiguity fixing. That is because the number of common visible satellites is usually not large, due to the difference of masking of satellites at each platform. However, the threshold cannot be too low either, because it will then reduce the reliability of the fixed solution. Herein, the threshold is configured as three, which is the least requirement of satellite availability to realize positioning. More detailed results will be presented in Chapter 6.

The above method is also used for ambiguity monitoring. Even with constraints, the ambiguities can still be incorrectly fixed; thus, it is necessary to detect the wrong fixes as soon as possible. Tests of sufficiency and closure of constraints are continuously performed during positioning. If N_{best} passes all the tests but it is not equal to the integer solution, then a wrong fix is detected. It is found that the constrained method is much more effective in detecting wrongly fixed ambiguities than is the single-baseline method. This is because the constrained method integrates all the observations from the ℓ moving platforms. Once a wrong fix is detected, it will affect error detection of other baselines via the ambiguity constraints, while the single-baseline method only uses the information collected at each baseline.

3.4.4 Summary of MultiKin Procedure

The overall procedure used in MultiKin is summarized in Figure 3.23. First, m baselines connecting ℓ moving platforms are selected by Delaunay triangulation to construct n triangles for applying ambiguity constraints. Second, those m ambiguity search modules try to fix ambiguity sets for each baseline individually. Each module outputs the float ambiguity set $\hat{N}_i$, the best integer ambiguity set $\tilde{N}_i$, and its corresponding ratio F_i. F_i is defined by Equation (3.9). If the integer ambiguity set is successfully fixed to $\overline{N}_i$, it will be also output. In the third step, the algorithm for multiple-platform ambiguity determination described in section 3.4.3 is used to check whether the combination of the best integer candidates $\tilde{N}_i$ can be the integer solution or whether $\overline{N}_i$ is wrongly fixed.

Observations input of ℓ platforms

Baseline (m) selection and triangulation (n)

$\hat{N}_i$: Float ambiguity set

$\tilde{N}_i$: The best integer candidate

$\overline{N}_i$: Integer ambiguity (if fixed)

F_i : Corresponding F-ratio of $\tilde{N}_i$

Ambiguity search for baseline _1_

Ambiguity search for baseline _2_

Ambiguity search for baseline _m_

$(\hat{N}_1, \tilde{N}_1, \overline{N}_1, F_1)$

$(\hat{N}_2, \tilde{N}_2, \overline{N}_2, F_2)$

$\vdots$

$(\hat{N}_m, \tilde{N}_m, \overline{N}_m, F_m)$

Multi-platform ambiguity determination / monitoring

Float/Integer ambiguity output

Position Velocity

Position Velocity

Position Velocity

Go back for new observations

Process End?

N

Y

STOP

Figure 3.23 Procedure of MultiKin

3.4.5   Representation of Position

Due to the lack of reference stations with precisely known coordinates, the absolute positioning accuracy of a moving platform is poor. However, the relative positioning accuracy is of the major concern in this research. Even using a moving reference with approximate coordinates, the relative positioning accuracy can still be at the centimetre or sub-decimetre level. Herein, a data structure is defined to describe the relative positions between platforms in the configuration. It contains n data records where n is the number of moving baselines. Figure 3.24 shows the format of a data record.

Start point of a baseline	End point of a baseline	Latitude increment $\Delta\phi$	Longitude increment $\Delta\lambda$	Height increment Δh	distance

Figure 3.24 Data record of relative position

Since Delaunay triangulation is used, only some of the baselines in the configuration are selected. There may not be a direct baseline link between any two moving platforms. For instance, in Figure 3.25, the relative position between Platforms 3 and 4 cannot be directly found from the data record, whereas, it can be indirectly obtained by adding the position vectors $\vec{R}_{42}$ and $\vec{R}_{23}$. Herein, the vector sum $\vec{R}_{43}$ is defined as the relative position vector of the "virtual baseline" between Platforms 4 and 3. This solution however causes another problem. As shown in Figure 3.25, either the vector sum of $\vec{R}_{42}$ and $\vec{R}_{23}$ or $\vec{R}_{41}$ and $\vec{R}_{13}$ can be used to represent the relative position vector $\vec{R}_{43}$. Which option is optimal?

Generally, the longer baselines induce larger errors because, on the one hand, longer baselines contain larger spatially correlated errors, and on the other hand, the extra error in the relative positions caused by the reference errors is proportional to the baseline length, see Equation (3.17). Consequently, it is expected that the best relative position is the vector sum of the shortest baselines. Now the question of selecting a group of short baselines can be equivalent to the "Shortest Path Problem" for a non-directional graph. In graph theory, the solution to this problem leads to the shortest path from one designated

point to another. The most commonly used algorithm is Dijkstra's algorithm (Dijkstra, 1959). The structure of the graph, i.e., the connectivity and the path lengths, must be known before applying this algorithm, which can be obtained from the data records as defined in Figure 3.24.
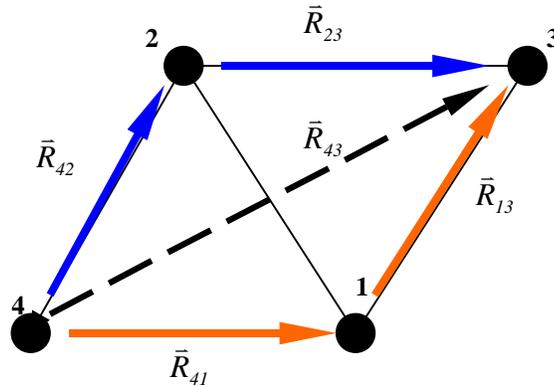
Figure 3.25 Representing of relative position

Before introducing the procedure of Dijkstra's algorithm, a definition of terms is given below:

Length(i) is the length of current path from the start node to the current node *i*.

Prev(i) is the previous node along the current shortest path from start node to the current node *i*.

N is the set of all nodes.

l(i,j) is the length of the baseline (Delaunay edge).

The algorithm contains two parts: initialization and the search loop.

Initialization:

[1] Set $Length(start\_node) = 0$, and $Length(i) = +\infty$ for all other nodes *i* in *N*.

[2] Set *j*=0.

[3] Let *Prev(i)* be undefined for each node *i* in *N*; all nodes are unmarked.

Main Loop:

Until the end node is marked with a * do the following:

[1] Set *j=j+1*.

[2] Among all unmarked nodes, select a node *i* for which *length(i)* is minimum.

[3] Mark node *i* with a *, which means the shortest path from the start node to current node *i* has been found.

[4] For each baseline leaving from node *i*, compare *Length(j)* with $Length(i)+l(i,j)$ when a Delaunay edge exists between *i* and *j*. If the latter length is shorter, then set $Length(j)=Length(i)+l(i,j)$, and set *Prev(j):=i*.

Figure 3.26 shows the results of a simulation test to find the shortest path between Nodes 1 and 10, where 25 nodes are randomly generated with a 2D average distribution. Each line is a Delaunay edge, namely the selected baselines. The bold line is the shortest path from Node 1 to Node 10.
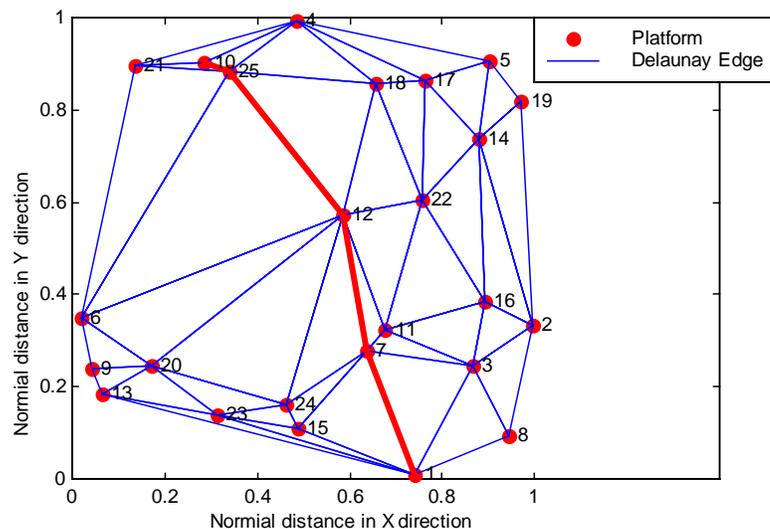
Figure 3.26 The shortest path selected by Dijkstra algorithm

When the shortest path is found, the vector sum of the relative position along the path can be treated as the relative position between two designated points. Using the shortest path algorithm, a moving reference platform can know the relative positions of all other platforms. However, these positions are represented by the increment of the latitude, longitude and altitude. If relative position in local level frame (East-North-Up) is preferred, the approximate absolute coordinates of the moving reference must be used for coordinate transformation.

4 GPS CARRIER PHASE ERROR MODELING FOR SOFTWARE SIMULATIONS

To test the impact of various system parameters and the effectiveness of the algorithms developed in Chapter 3, a full-scale test should be performed. Field tests often have difficulties in providing wide range, controllable and repeatable test conditions; therefore, the use of simulated scenarios is preferable.

To avoid using an expensive and complicated hardware simulator, a software GPS simulator has been built for simulation tests. A fully developed GPS simulator must simulate the GPS constellation and GPS error sources, as well as the trajectory and dynamic of a user platform. Since GPS errors have a major impact on the performance of the tested system, detailed methods for simulating five major GPS errors are discussed. These errors are ionospheric, tropospheric, orbital error, multipath, and SA.

The validity of GPS error models is crucial to the confidence of simulation tests. Clearly, a simulator must generate GPS errors with similar statistical properties to real GPS errors. Various GPS error models are in existence but most of them are designed for single point positioning. The focus of this chapter is the design of sophisticated DGPS error models. Therefore, the simulation of temporal and spatial correlation of various GPS errors is emphasized.

The stochastic characteristics of these errors are studied through theoretical analysis and statistical tests with data collected under various conditions. All models include adaptable parameters in order to represent a full range of testing conditions. Parameters such as meteorological data, time, ionospheric activity, and user's location have been taken into account to ensure the full range of possibilities. Each error model is validated through statistical testing of the reproduced GPS errors.

4.1 SIMULATION OF IONOSPHERIC ERROR

There are many methods for ionosphere delay modeling. In this research, the objective is to develop an ionospheric model which is valid for DGPS simulation tests. This involves

not only simulating the absolute ionosphere delay along the observation line of sight, but also modeling the residual ionosphere delay after differencing. Herein, a combined spherical harmonics and grid model is developed, which has proven to be effective in simulation.

4.1.1 Description of Ionosphere Model

The modeling of ionospheric delay involves three steps:

[1] Generate a global profile of Total Electron Content (TEC) distribution.

[2] Generate a global grid-network with additional random TEC.

[3] Compute the vertical TEC at any pierce point.

4.1.1.1 Generate a Global Profile of TEC

The main objective of this research is to study the characteristics of ionospheric delay in differential mode. This corresponds to the difference in TEC distribution. However, it is better to model the absolute value of TEC to a certain level of accuracy. Therefore, the developed model can also be used for testing stand-alone positioning systems or the DGPS systems with long baselines. Among the global ionosphere models in existence, the spherical harmonics (SPHA) model is considered one of the best. In this model, the TEC is developed into a series of spherical harmonics in a Sun-fixed reference frame:

$$E(\beta, s) = SF \sum_{n=0}^{n_{max}} \sum_{m=0}^{n} \tilde{P}_{nm}(\sin\beta)(a_{nm}\cos ms + b_{nm}\sin ms) \qquad (4.1)$$

where E is the vertical TEC value,

β is the geocentric latitude of the pierce point of the ionosphere,

s is the Sun-fixed longitude of the ionospheric pierce point — or sub-ionospheric point, see Figure 4.1. It is also the difference between the Earth-fixed longitude of the pierce point and the longitude of the Sun,

n_{max} is the maximum degree of the spherical harmonics expansion,

$\tilde{P}_{nm} = \Lambda_{nm} P_{nm}$ are the normalized associated Legendre functions of degree n and order m based on the normalization factor Λ_{nm} and the classical Legendre functions P_{nm},

a_{nm} b_{nm} are the coefficients of the spherical harmonics, and

SF is the scale factor to control the level of TEC value in simulation.
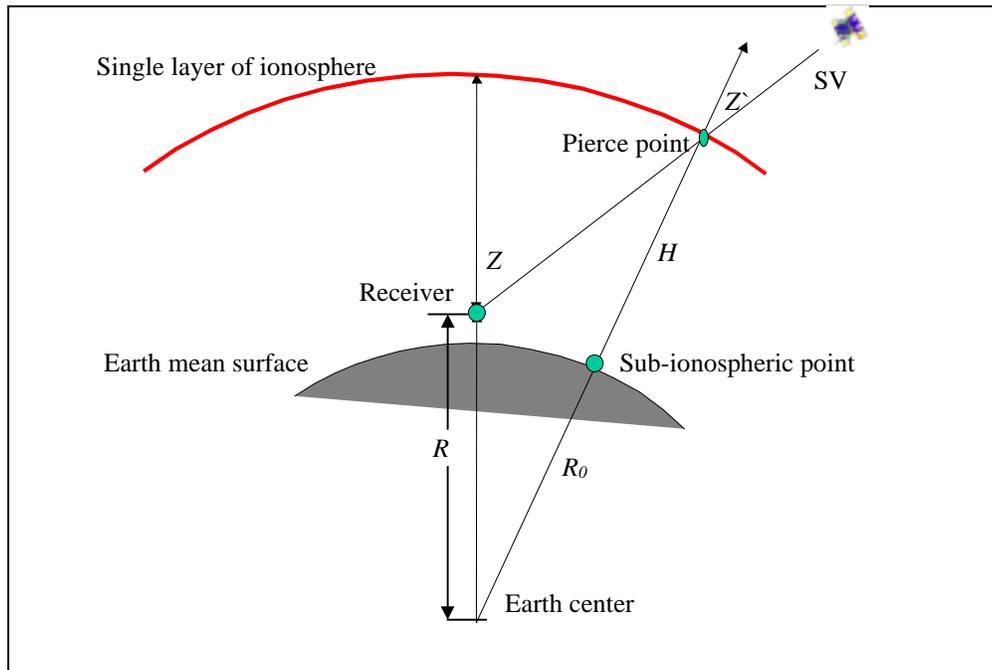
Figure 4.1 Description of the single-layer ionosphere model

The SPHA model is based on the assumption that the ionosphere is a thin single layer. This means that all free electrons are distributed in a spherical shell of infinitesimal thickness. Figure 4.1 shows the outline of this model. The conversion from vertical TEC value E into the slant TEC value E_s is realized by multiplying E with a mapping function:

$$E_s = m(\,z\,)E = \frac{1}{\cos z'}E \qquad \text{with} \qquad \sin z' = \frac{R}{R_0 + H}\sin z \qquad (4.2)$$

where $m(z) = \dfrac{1}{\cos z'}$ is the single-layer mapping function,

$z,\ z'$ are the geocentric zenith distances of a satellite at the height of the GPS receiver and the single layer of the ionosphere, respectively,

R is the radius with respect to the receiver considered,

R_0 is the mean radius of the Earth, approximated as 6371 km, and

H is the height of the single-layer of ionosphere above the Earth mean surface, which is set to 450 km.

To use the SPHA model, the coefficients a_{nm} b_{nm} must be known. In this model, Global Ionosphere Maps (GIMs) files are used. These files contain the coefficients of spherical harmonics and other ionospheric parameters (Schaer, 1997). These GIMs can be obtained from the Centre for Orbit Determination in Europe (CODE), one of the Analysis Centres of International GPS Service (IGS). Since January 1, 1996, GIMs are routinely presented as an additional product at CODE. Every day a set of TEC coefficients are determined. These coefficients give approximate estimation of the distribution of the vertical TEC on a global scale by analyzing the so-called geometry-free linear combination of GPS carrier phase data collected at 84 globally distributed stations.

GIMs are now available on a daily basis. Each contains 12 two-hour sections, where each section contains 149 coefficients. To get the coefficients for our model, 100 GIMs are used. These GIMs are sampled from January 1996 to February 2000, with two samples each month. The coefficients used in our model are the average of these samples. The output vertical TEC computed by Equation (4.1) can be scaled to a different magnitude according to the strength of solar activity required by the scenarios.

4.1.1.2 *Generate Globally Distributed Grids Based on SPHA Model*

Presently, the GIMs can give a spherical harmonics expansion with a maximum degree of twelve and a maximum order of eight. Such resolution is good enough for describing the

profile of the global TEC distribution but not sufficient for error simulation of DGPS in a small region.

To meet the requirement of DGPS simulation in a small area, the resolution of the TEC distribution of the ionosphere should be greatly increased. An approach to achieve this objective is to add a high-resolution TEC increment grid-network to the profile of the global TEC distribution. The TEC increment can be treated as an unmodeled part of the ionospheric delay by the SPHA model. The following procedure is to generate a global TEC distribution with an enhanced resolution.

[1] Partition the ionosphere single-layer into a network with averagely distributed $n \times m$ grids in the sun-fixed frame.

The vertical TEC at certain grid point (β_i, s_i), $VTEC_{SPHA}(\beta_i, s_j)$, can be calculated using Equation (4.1). It should be noticed that although the density of the grids in the network can be very high by increasing m and n, the actual resolution of the TEC distribution has not yet been increased, since the maximum degree and order of the model are not changed.

[2] Generate the TEC increment $v_{i,j}$ for each grid point.

$v_{i,j}$ is a random variable representing the unmodeled part of the ionospheric delay. The statistical characteristics of $v_{i,j}$ will be discussed later.

[3] Build the high-resolution grid network by adding the profile of TEC distribution and the TEC increment. The vertical TEC at a grid point can be computed as follows:

$$VTEC(\beta_i, s_j) = VTEC_{SPHA}(\beta_i, s_j) + v_{i,j} \tag{4.3}$$

By introducing the TEC increment, the grid-network can represent the TEC distribution of the ionosphere with an enhanced resolution of $360°/m \times 180°/n$. Considering the necessary requirements of DGPS simulations, the resolution of the grid network is 1.5°×1.5°.

In this approach, the TEC increment $v_{i,j}$ is a very important component because its statistical properties are coherent with the ionosphere spatial decorrelation, which can affect the performance of DGPS to a large extent. According to the experimental results derived by Stanford University (Christie et al, 1999), the exponential distribution is preferable. This is described by the distribution given below:

$$f(x) = 1/(\sqrt{2}\sigma)exp(-\sqrt{2}|x-m|/\sigma)$$ (4.4)

The exponential distribution is a heavy-tailed distribution, that is, the exponential random generator has a larger probability to generate the variable with very large value than the commonly used normal random generator, see Figure 4.2. Therefore, $v_{i,j}$ produced by the exponential random generator can cause very large spatial decorrelation in some area.
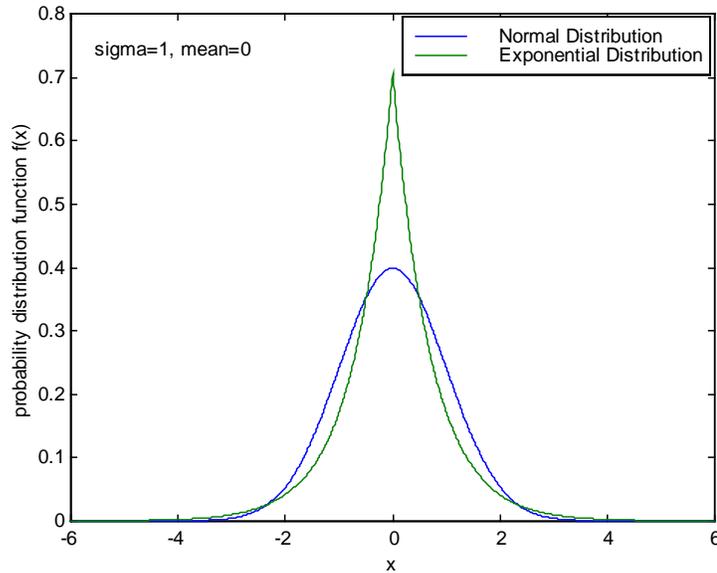
Figure 4.2 Comparison of normal distribution and exponential distribution

In general, $v_{i,j}$ should be zero-mean. The global average of TEC is not affected by the inducing of $v_{i,j}$. The selection of the variance of $v_{i,j}$ is dependent upon the required spatial decorrelation (equivalent to the TEC gradient) in the scenarios of the simulation. For example, assuming $v_{i,j}$ to be position independent, then the additional south-north

TEC gradient Δg and its variance caused by $v_{i,j}$ within a grid can be expressed as follows:

$$\Delta g = \frac{v_{i,j} - v_{i+1,j}}{\Delta\phi} \tag{4.5}$$

$$\sigma_{\Delta g}^2 = \frac{2\sigma_v^2}{\Delta\phi^2} \tag{4.6}$$

where $v_{i,j}, v_{i+1,j}$ are the TEC increment at two adjacent grid points on the same meridian,

$\Delta\phi$ is the latitude resolution of the grid.

According to the required spatial decorrelation rate $\sigma_{\Delta g}$, the variance of $v_{i,j}$ can be computed by Equation (4.6).

4.1.1.3 Compute the Vertical TEC at the Pierce Point

After building the grid network of the ionosphere with enhanced spatial resolution, the vertical TEC value at the pierce point can be computed in two ways. First, the SPHA model can be expanded to the higher degree and order. Since the TEC values at the gird points are known, the higher order coefficients a_{nm} and b_{nm} ($n>12$, $m>8$) can be derived using the orthogonality of the spherical harmonics:

$$a_{nm} = \frac{1}{SF} \int_0^{2\pi} \int_0^{\pi} E(\beta,s)\tilde{P}_{nm}(\sin\beta)\cos\beta\,\cos(ms)d\beta ds \tag{4.7}$$

$$b_{nm} = \frac{1}{SF} \int_0^{2\pi} \int_0^{\pi} E(\beta,s)\tilde{P}_{nm}(\sin\beta)\cos\beta\,\sin(ms)d\beta ds$$

However, this method can cause severe computational difficulty when the degree and the order of the model are high. For example, a SPHA model with resolution of 1.5°×1.5° has more than 20,000 coefficients. This means that to compute the vertical TEC value at a pierce point, 10,000 associated Legendre polynomials should be calculated. Furthermore, the SPHA model has a tendency to underestimate or overestimate TEC values in regions

neighboring steep localized gradients (Mannucci et al, 1997). Therefore, the method of planar interpolation is adopted instead.

In the first two steps, a high-density grid network of TEC distribution has been established. The TEC value at any point within the network can be computed using interpolation. Figure 4.3 gives the conception of the four-point grid-based algorithm to estimate the vertical TEC at the pierce point where P is the pierce point and G_i is the surrounding grid point.
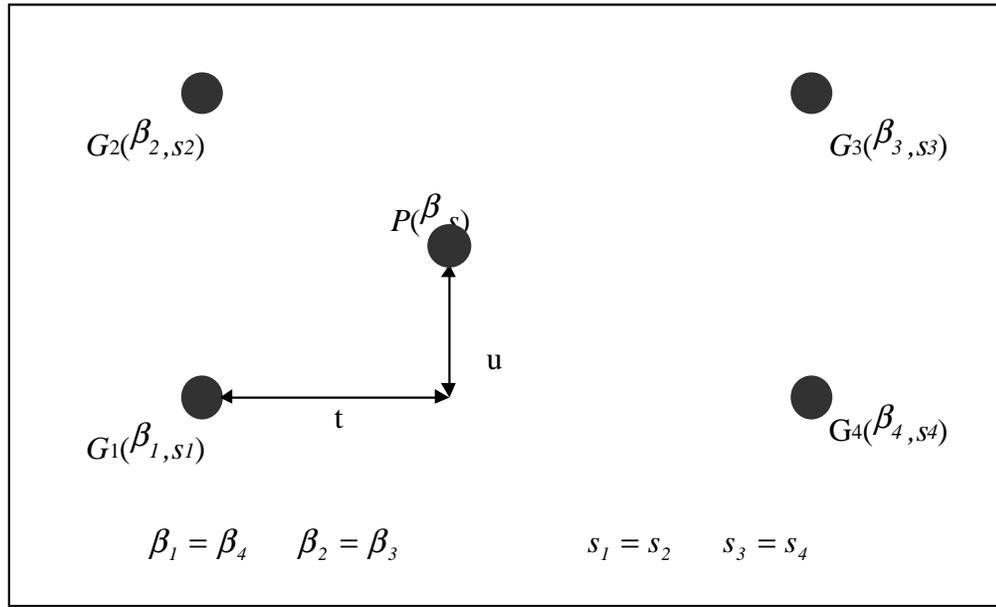
Figure 4.3 Compute TEC in a grid network

The vertical TEC at P can be computed using following equation:

$$VTEC_P = \sum_{i=1}^{4} w_i VTEC_i \qquad (4.8)$$

where $w_1 = w(t,u)$ $\qquad\qquad$ $w_2 = w(t,1-u)$

$w_3 = w(1-t,1-u)$ $\qquad\qquad$ $w_4 = w(1-t,u)$

$w(t,u) = (1-t)(1-u)$

$$u = (\beta - \beta_1)/(\beta_2 - \beta_1) \qquad t = (s - s_1)/(s_4 - s_1)$$

To get the geomagnetic latitude β and longitude s, the geographical latitude and longitude of the pierce point and the Sun must be calculated first, and then they can be converted to the geomagnetic frame using a rotational transformation. The rotation angles are relevant to the position of the north geomagnetic pole, which is set to (79.45° N, 71.71° W). The choice of the spatial weighting function $W(\bullet)$ is arbitrary. The algorithms are based on inverse distance weighted averaging, bilinear interpolation, or multi-quadratic weighting functions. In the chosen model, the bilinear interpolation weighting function is adopted according to the recent specifications for users of WAAS (FAA, 1997, RTCA, 1998).

From the above modeling procedures, it can be seen that the characteristics of both the spherical harmonic model and the grid model are combined. Thus, the above model is referred to as the combined model in the following discussion.

4.1.2   Modeling of Scintillation

The simulation of ionosphere scintillation is also included in the software simulator; thus, some extreme ionospheric conditions can be generated according to the testing requirement, but the scintillation is not applied in the simulation tests of MultiKin.

Irregularities in the Earth's ionosphere produce both diffraction and refraction, which can cause short-term signal strength fading and rapid variation of signal phase. These effects are called scintillation, which can result in difficulties in signal detection and tracking. Scintillation is often correlated with location and time. Generally, equatorial regions experience stronger scintillation, but over shorter periods than polar caps. In addition, the frequency and the strength of the scintillation vary with season and time (Klobuchar, 1996). For simplification and the convenience of users to design interested scenarios, these variations are not simulated in our model. Therefore, scintillation can occur at any place and any time. Unlike other scintillation models (Kumar and Munjal, 1998) built in signal domain, the combined ionosphere model is built in range domain. Therefore, amplitude scintillation cannot be simulated, only simulation of phase scintillation is implemented in the GPS software simulator.

4.1.2.1 Methodology for Modeling Phase Scintillaion

Phase scintillation is caused by a fast and irregular change of the TEC distribution. In a normal situation, the Doppler change, i.e., the ionosphere acceleration caused by the temporal variation of the TEC distribution, is very small. For example, the maximum value of the vertical ionosphere acceleration generated by the combined model in a solar maximum is about 0.0057 Hz/s (L1). For a common GPS receiver with a bandwidth (carrier phase tracking loop) of 15 Hz (L1), this small change in Doppler will not cause any problem in phase tracking. However, during times of severe phase scintillation, the phase will not change in a consistent and rapid manner to yield greater ionospheric Doppler shift. The phase of the incoming signal will have a large random fluctuation superimposed upon the changes associated with normal rate of change in TEC (Klobuchar, 1996). The irregular change of TEC, denoted as $\Delta VTEC$, is simulated by white noise with a Gaussian distribution. The vertical TEC at the pierce point under a condition of scintillation can be treated as the sum of the normal TEC and the irregular change:

$$VTEC_{Scint\,illation}(t) = VTEC_{Combined}(t) + \Delta VTEC(t) \tag{4.9}$$

where $VTEC_{combined}(t)$ is the vetical TEC at the pierce point calculated by the combined model under normal conditions, and

$VTEC_{Scintillation}(t)$ is the total vertical TEC under scintillation conditions.

Generally, the mean value of ΔVTEC is set to zero. Therefore, the following discussion will focus on the selection of the variance of $\Delta VTEC(t)$. This selection depends highly upon the probability of a loss of lock in the phase tracking loop, as well as the magnitudes of additional errors in the measurements of range and Doppler.

Assume that $\Delta VTEC(t_1)$, $\Delta VTEC(t_2)$ and $\Delta VTEC(t_3)$ exhibit the fast irregular change in TEC observed at three adjacent epochs, where $t_2 - t_1 = t_3 - t_2 = \Delta t$. A second order polynomial can be used to fit this change:

$$\Delta VTEC(t) = a_2(t - t_2)^2 + a_1(t - t_2) + a_0 \tag{4.10}$$

Denote $\Delta VTEC(t_i)$ as ΔVT_i. The coefficients of the fitting polynomial can be represented as follows:

$$\begin{bmatrix} a_2 \\ a_1 \\ a_0 \end{bmatrix} = \frac{1}{2\Delta t^2} \begin{bmatrix} 1 & -2 & 1 \\ -\Delta t & 0 & \Delta t \\ 0 & 2\Delta t^2 & 0 \end{bmatrix} \begin{bmatrix} \Delta VT_1 \\ \Delta VT_2 \\ \Delta VT_3 \end{bmatrix} \tag{4.11}$$

Ignoring the very small Doppler caused by the normal variation of the ionosphere, the vertical ionosphere velocity $VIV(t)$ only results from the fast change components of the TEC, i.e., $\Delta VTEC$. The change in Doppler measurement caused by the ionosphere can be computed by scaling $VIV(t)$ with the mapping function:

$$VIV(t_2) = \frac{1.34 \times 10^9}{f} \left. \frac{d\Delta VTEC}{dt} \right|_{t=t_2} (Hz) \tag{4.12}$$

$$VIV(t_2) = \frac{1.34 \times 10^9}{f} a_1 = \frac{1.34 \times 10^9}{f} \frac{1}{2\Delta t}(\Delta VT_3 - \Delta VT_1)$$

$$\Delta D_{Iono} = VIV(t)m(Z) \tag{4.13}$$

where $\Delta VTEC$ has the unit of TECU (1 TEC Unit = 10^{16} el/m^2) . The vertical ionosphere acceleration, $VIA(t)$, can be described as the second order derivative of the vertical TEC. Multiplying it by the mapping function will give the slant ionosphere acceleration $IA(t)$:

$$VIA(t_2) = \frac{1.34 \times 10^9}{f} \left. \frac{d^2VTEC}{dt^2} \right|_{t=t_2} = \frac{1.34 \times 10^9}{f} 2a_2 \ Hz/s \tag{4.14}$$

$$VIA(t_2) = \frac{1.34 \times 10^9}{f} \frac{1}{\Delta t^2}(\Delta VT_1 - 2\Delta VT_2 + \Delta VT_3) \ Hz/s$$

$$IA(t) = VIA(t)m(Z) \tag{4.15}$$

Assuming that ΔVT_i is white noise with zero mean and has a Gaussian distribution, the relationship between the variance of ionosphere acceleration and the variance of ΔVT can be written as:

$$\sigma_{VIA} = \frac{1.34 \times 10^9}{f} \frac{\sqrt{6}}{\Delta t^2} \sigma_{\Delta VT} \tag{4.16}$$

The above Equation (4.16) clearly shows the relationship between the variance of random TEC and the variance of vertical ionosphere acceleration. The remaining problem is how to decide the value of σ_{VIA}. The ionospheric acceleration can cause the phase loop to lose lock, thus, in the next section, the relationship between the probability of losing lock and σ_{VIA} will be discussed.

4.1.2.2 Selection of Variance of VIA

Assume that the tracking loop bandwidth of a common GPS receiver is B Hz. Subtracting the phase acceleration caused by the relative motion between a satellite and a receiver, which is less than DC_{max} Hz/s, the phase tracking loop can at least tolerate ($B - DC_{max}$) Hz/s phase acceleration caused by ionosphere phase scintillation. The tolerence threshold is denoted as IA_T as follows. Define the probability of a loss of phase lock as α:

$$\alpha = P\left(|IA + DC_{sv}| > B Hz/s \right) = \iint\limits_{|IA+DC_{sv}|>B} f(IA) f(DC_{sv}) dIA dDC_{sv} \tag{4.17}$$

where DC_{sv} is the Doppler change caused by the relative motion between the receiver and the satellite,

   $f(IA)$ is the *pdf* of ionospheric acceleration,

   $f(DC_{sv})$ is the *pdf* of the DC_{sv}, and

   B is the bandwidth of the carrier phase tracking loop.

The difficulty in computing this probability α is obvious, because the *pdfs* of IA and DC_{sv} are very complicated. They are all functions of the satellite's elevation. Moreover,

$f(DC_{sv})$ contains also the dynamics of the user. So for simplification, we just redefine α as the probability when *IA* is larger than the tolerance threshold:

$$\alpha = P\left(IA > IA_T, or, IA < -IA_T\right) = \int_{-\infty}^{-IA_T} f(IA)dIA + \int_{IA_T}^{\infty} f(IA)dIA = 2\int_{IA_T}^{\infty} f(IA)dIA \qquad (4.18)$$

Compared wtih the real probability of loss of lock, the new definition will actually result in a smaller probability of losing lock. This is shown in Figure 4.4 by comparing the range of the integral in Equations (4.17) and (4.18).
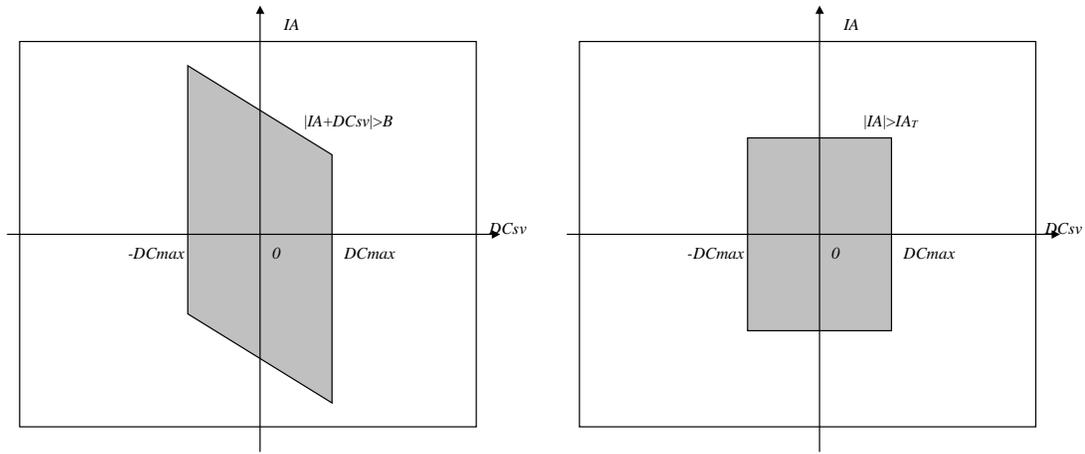
Figure 4.4 Comparison of integral ranges in case of different definitions of α

Since *IA* is obtained by scaling *VIA* with the mapping function $m(Z)$, α can be described as the double integral of the *pdf* of the zenith distance *Z* of a satellite and the vertical ionosphere acceleration *VIA* at a pierce point:

$$\alpha = 2\int_0^{\pi/2} f(Z)\int_{\frac{IA_T}{m(Z)}}^{\infty} f(VIA)dVIAdZ \qquad (4.19)$$

where
$$\begin{cases} f(Z) = \dfrac{2}{\pi} & Z \in \left[0, \dfrac{\pi}{2}\right] \\ f(VIA) = \dfrac{1}{\sqrt{2\pi}\sigma_{VIA}} exp\left[-\dfrac{VIA^2}{2\sigma_{VIA}^2}\right] & VIA \in (-\infty, \infty) \end{cases}$$

82

Define $\Phi(x) = \int_x^\infty f(\beta)d\beta$, where $f(\beta)$ is the *pdf* of the normalized Gaussian distribution with zero mean and variance of one. α can be expressed as

$$\alpha = \frac{4}{\pi}\int_0^{\pi/2}\Phi\left(\frac{IA_T}{m(Z)\sigma_{VIA}}\right)dZ \qquad (4.20)$$

Because the mapping function is not linear (see Figure 4.5) and $\Phi(x)$ does not have an analytic expression, it is very difficult to give an explicit expression for the relationship between α and IA_T/σ_{VIA}. To solve this problem, the numerical computation method is applied.
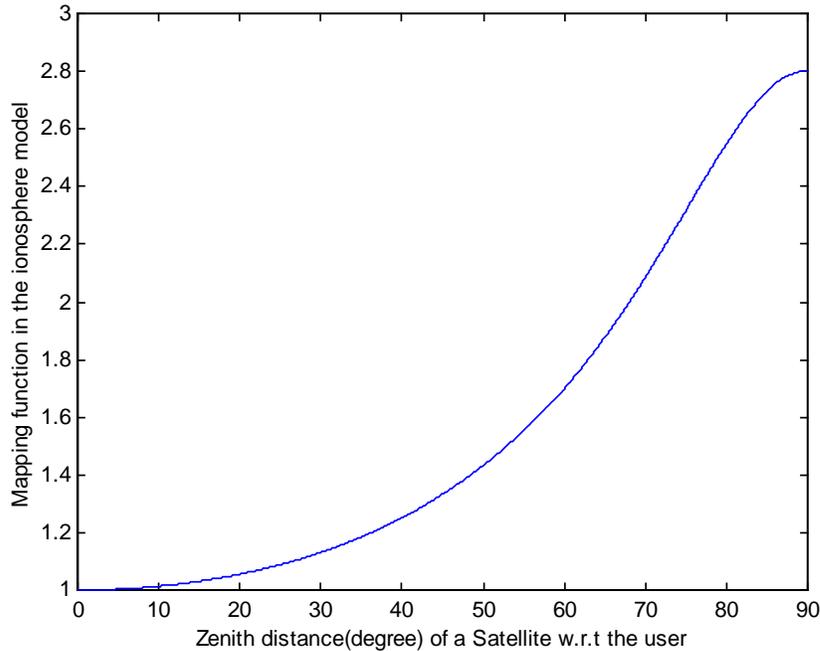
Figure 4.5 Mapping function of ionospheric delay model

Figure 4.6 shows the result of the computation. In the simulation software, several tables are used to look up the proper IA_T/σ_{VIA} for a preset α. It should be noticed that when the cutoff angle of the observation changes, the selection of IA_T/σ_{VIA} also changes. The reason is that most of the large ionosphere accelerations occur at the low elevation angle due to the amplification by the mapping function. However, when the satellite elevation

is lower than the cutoff angle, observations of the satellite will not be generated; thus, the large ionosphere acceleration will actually have no impact on simulation.
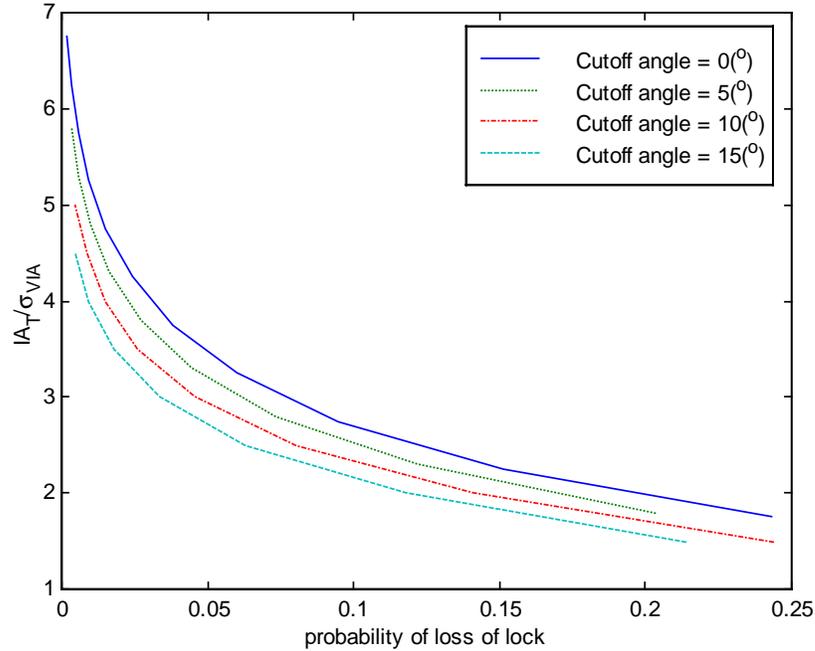
Figure 4.6 Selection of IA_T / σ_{VIA} **in case of different cutoff angles**

Figure 4.7 gives the procedure of scintillation simulation.

As a summary, this example shows how to select the variance of ΔVT for a required scenario:

Scenario: Δt=0.25s, α=0.1, f=1575.42MHz , cutoff angle= 10°, VIA_T=14Hz.

Solution: $\sigma_{\Delta VT} = 0.188TECU$

TEC change rate = $\sigma_{\Delta VT} / \Delta t = 0.75TECU/s$

This solution means that if the scintillation generates a fast change of TEC with an average rate of 0.75 $TECU/s$, it can cause the phase tracking loop to lose lock (*IA* larger than the threshold) with 10% probability. In the model, the loss of the lock in the tracking

loop is simulated by preventing the observation of the specified satellite from being output.

Initialization:
- Set the probability of loss of tracking $-\alpha$
- Set the threshold of ionosphere acceleration IA_T
- Set the cutoff angle of the observation
- Get IA_T/σ_{VIA} by looking up the table
- Compute variance of the fast TEC change $\sigma_{\Delta VT}$

Generate three random variables $\Delta VTEC_{1,2,3}$ to get the fitting polynomial of the TEC change

Compute the ionosphere acceleration along the line of sight $-IA$

$IA>IA_T$?

Y   N

Pause measurement output at this epoch

Compute the total vertical TEC

Compute the slant delay of the ionosphere

Compute the Doppler increment caused by the ionosphere scintillation,

N

Simulation ends?

Y

STOP

Figure 4.7 Summary of the procedure to generate ionosphere error under the scintillation scenarios

4.1.3 Evaluation of the Combined Ionosphere Model

4.1.3.1 TEC Distribution on the Ionosphere Shell

Figure 4.8 shows the comparison of the TEC distribution generated by the standard SPHA model and the combined model.

a	b
c	d
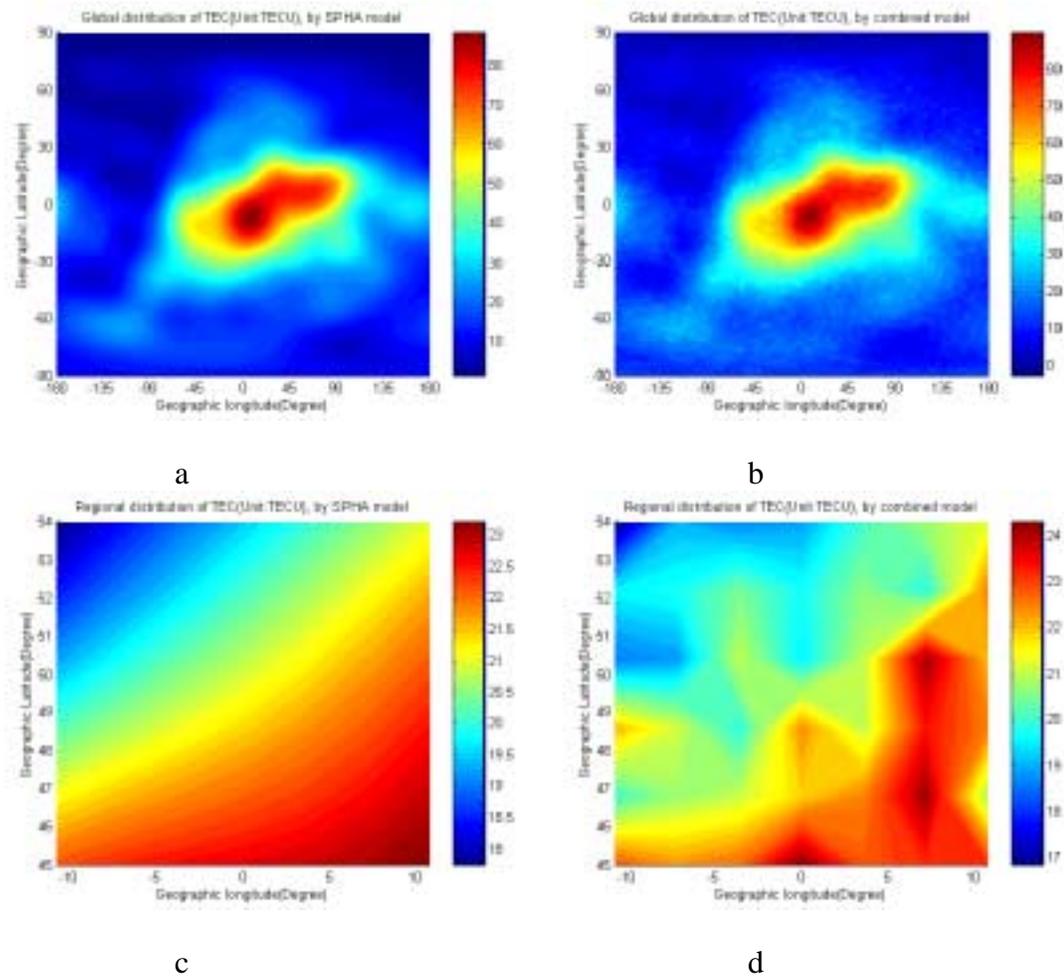
Figure 4.8 Comparison of the global and regional distribution of TEC (SPHA vs. Combined). Time:12:00h(UT), $\sigma_v = 1TECU$

Figure 4.8a and b exhibit the global TEC distribution on the ionosphere shell. The higher TEC is distributed at the equatorial area. The maximum value of TEC is 85.1 TECU. While in the polar area, TEC is smaller. The minimum value is 0.35 TECU. For a detailed comparison of the standard SPHA model and the combined model, Figure 4.8c and d show the TEC distribution in a regional area. Based on the fact that the combined

model shows more irregularities in the distribution of TEC than the standard SPHA model, it can be derived that the combined model has higher frequency components in TEC distribution. This is because the irregularities correspond to the high-frequency components in the spatial distribution of TEC. These components imply that the combined model has a higher spatial resolution than the standard SPHA model.

The comparison in Figure 4.8 also shows that although the global average of TEC is not affected by the introduction of $v_{i,j}$, the regional TEC value varies more largely in the combined model (16.8 *TECU*~24.4 *TECU*) than in the standard model (17.7 *TECU*~23.3 *TECU*). This larger variation implies that larger spatial decorrelations can be generated by the combined model than by the standard SPHA model.

4.1.3.2 Ionosphere Gradient

Ionosphere gradient is used to describe the spatial decorrelation rate of ionosphere delay. Generally, large ionosphere gradient means ionosphere delay decorrelates very quickly with the increase of the spatial separation. This is a nuisance in DGPS applications. The absolute value of the ionosphere gradient at any pierce point can be expressed as the norm of the East and North gradients:

$$\left|\frac{\Delta VTEC}{\Delta r}\right| = \sqrt{\left|\frac{\partial}{\partial E}VTEC\right|^2 + \left|\frac{\partial}{\partial N}VTEC\right|^2} \tag{4.21}$$

$$\frac{\partial}{\partial E}VTEC = \frac{\partial}{\partial t}\frac{dt}{d\lambda}\frac{d\lambda}{dE} = \left[(V_4 - V_1)(1-u) + (V_3 - V_2)u\right]/R_e\Delta\lambda\cos\phi$$

$$\frac{\partial}{\partial N}VTEC = \frac{\partial}{\partial u}\frac{du}{d\phi}\frac{d\phi}{dN} = \left[(V_2 - V_1)(1-t) + (V_3 - V_4)t\right]/R_e\Delta\phi$$

where R_e is the average radius of the earth,

   $\Delta\phi$ and $\Delta\lambda$ are the resolutions of the grids,

   E and N denote the direction of east and north respectively,

V_i is the vertical TEC values at the four grid points around the pierce point, and

t and u are the normalized distances from the pierce point to a reference grid point (see Figure 4.3).

Figure 4.9 shows the global and regional distributions of the TEC gradient produced by the standard SPHA model and the combined model. It is worth mentioning that in Figure 4.9, the distribution of TEC gradient is not shown in the polar area because of the computational singularity near the geomagnetic poles, see Equation (4.21). However, this problem does not affect the reliability of the combined model since the gradient of TEC is not used in the simulation process.

a

b

c

d
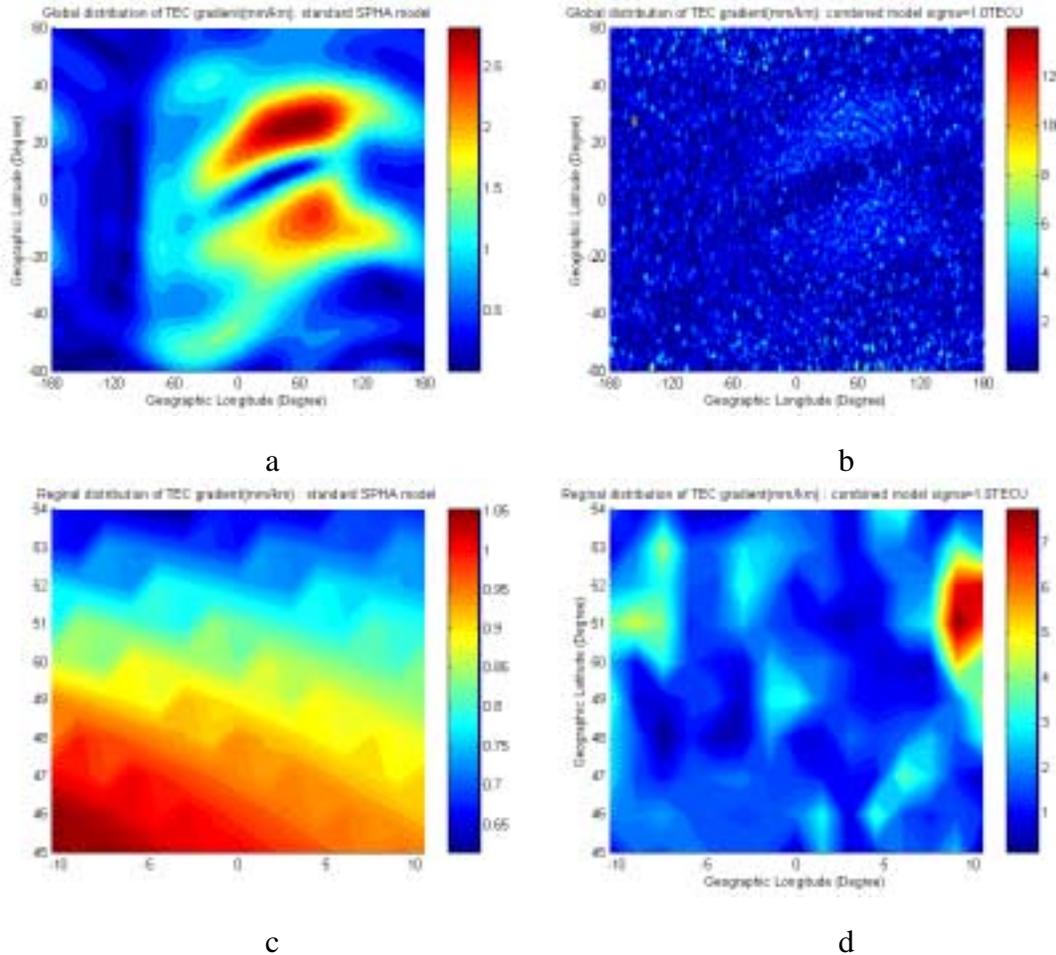
Figure 4.9 Comparison of the global and regional distribution of TEC gradient (SPHA vs. Combined). Time:12:00h(UT), $\sigma_v = 1TECU$

From the results, it can be seen that a relatively larger spatial decorrelation rate can be observed in the combined model. In Figure 4.9b, some areas show the gradient to be as large as 12 mm/km. This can result in very large residual ionosphere errors in DGPS even for short baselines. However, for the standard SPHA model, the largest TEC gradient is only 2.7 mm/km because of the lower spatial resolution.

Figure 4.10 shows the statistical *pdf* of the global TEC gradient. It can be found that the combined model can generate large differential ionospheric errors (gradient) with a higher probability than the standard SPHA model can. Figure 4.11 shows the relationship between the variance of $v_{i,j}$ and the increased global ionosphere gradient. This relationship can help users to reasonably select the variance of $v_{i,j}$ to obtain the required spatial decorrelation for DGPS simulations.

From the testing results, it is also found that when using the standard SPHA model to simulate large differential ionospheric errors, the global TEC average has to be increased correspondingly. However, when using the combined model, the magnitude of differential errors can be separately controlled by only adjusting the variance of the random TEC, which will not change the global TEC average. Therefore, the combined model presents more flexibility in error simulations.
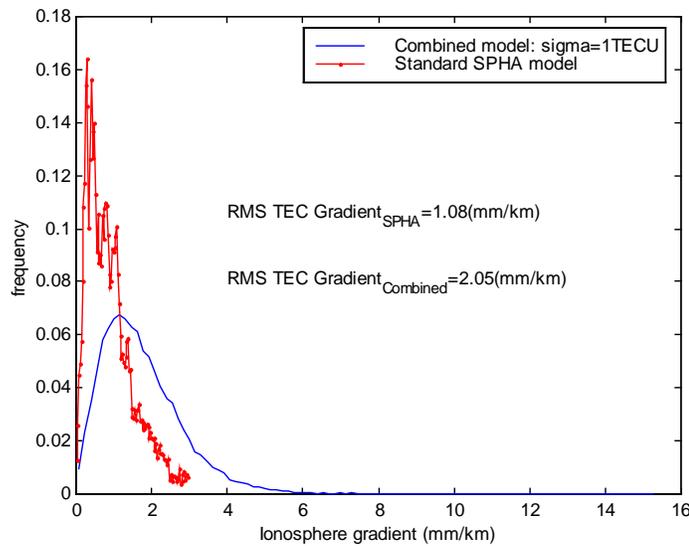
Figure 4.10 *pdf* of the TEC gradient (standard SPHA vs. combined model)
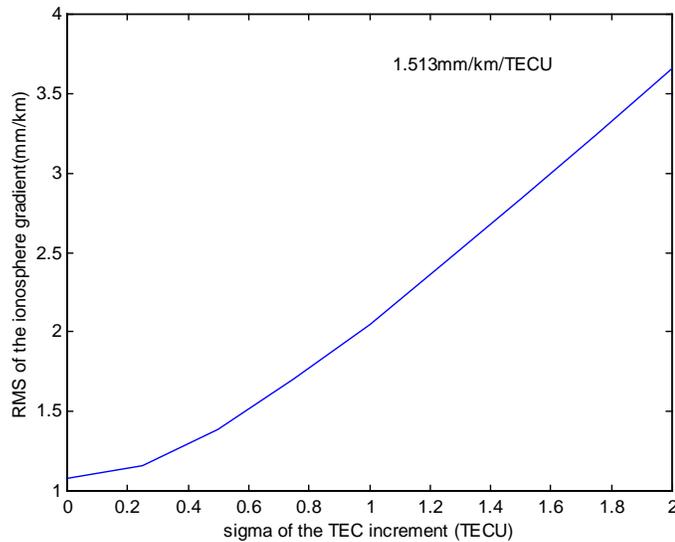
Figure 4.11 Relationship between the vertical ionospheric gradient and selection of the variance of the TEC increment $v_{i,j}$

4.1.3.3 Ionosphere Velocity

Ionosphere velocity represents the change of TEC distribution with time. The TEC at each grid point in the sun-fixed frame is actually constant in the combined model. However, due to the Earth rotation, the grid point is apparently moving with respect to the observer on the Earth. This causes the user-observed TEC to change. Figure 4.12 shows the diurnal change of observed TEC at mid-latitude. It can be seen than the combined model presents a much faster temporal variation of TEC than the standard SPHA model.

The time variation of the observed TEC can also result in a Doppler shift in the GPS measurements. In the standard SPHA model, the shift is very small and its change is very slow, while the combined model can generate a much larger Doppler shift than the standard SPHA mode, see Figure 4.13, Figure 4.14, and Figure 4.15. However, the Doppler shift is still small enough (<1.0 mm/s = 0.0053Hz) to be ignored when scintillation is not applied. It should be noticed that the ionosphere velocity has an equivalent unit of Hz. One Hz Doppler shift in the L1carrier phase is equal to the ionospheric velocity of 190.34 mm/s.
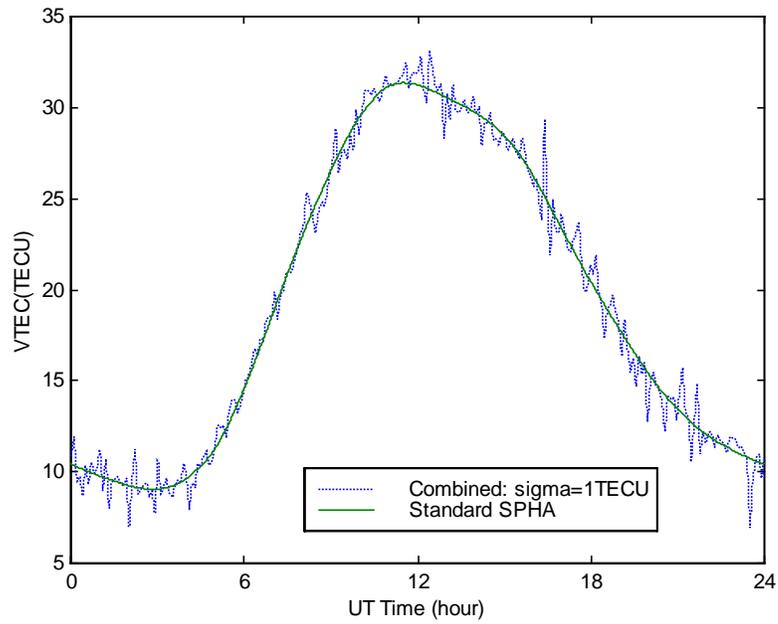
Figure 4.12 Comparison of Daily change of TEC observed at 45°N, 0°E, SPHA vs. the combined model
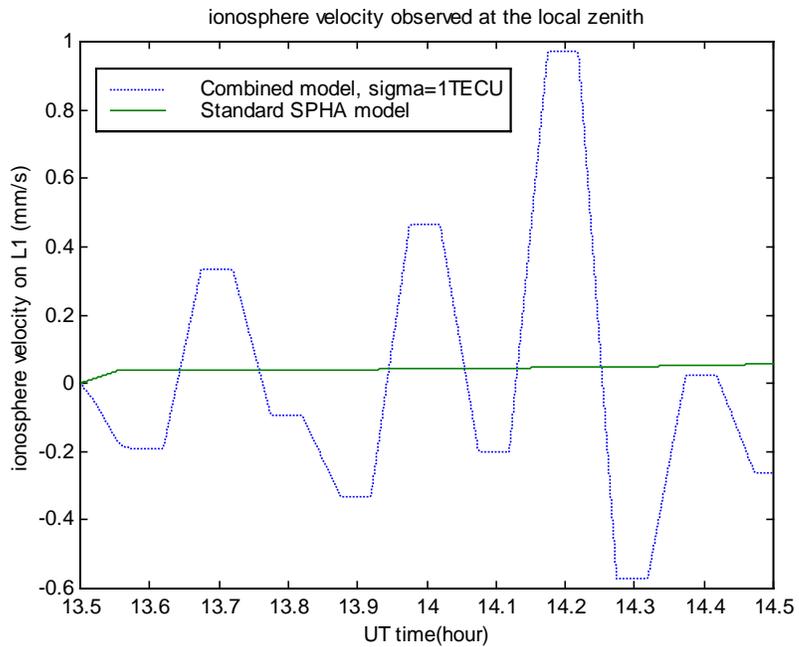
Figure 4.13 Comparison of Doppler shift caused by TEC variation, observed at 45°N, 0°E, SPHA vs. the combined model
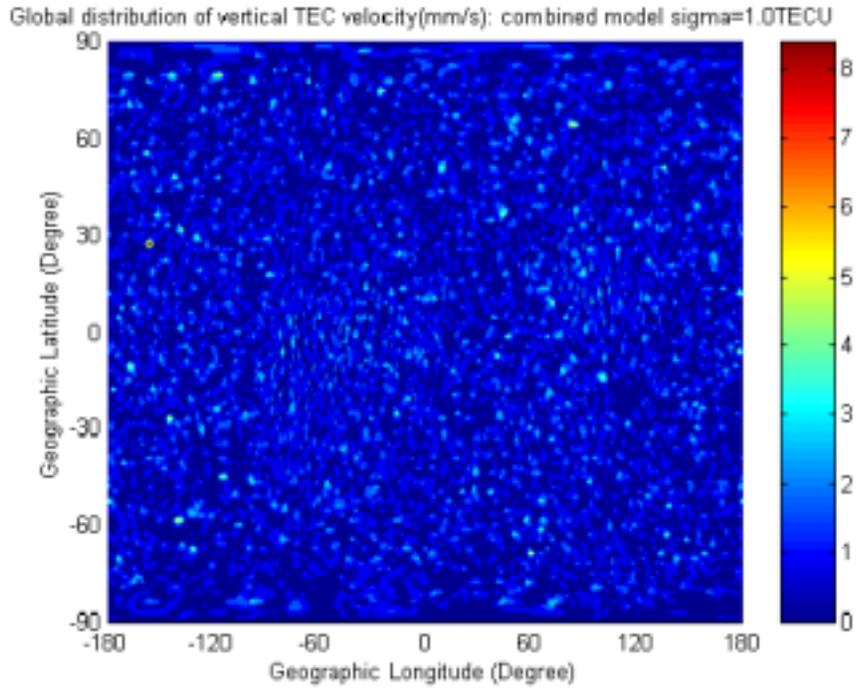
Figure 4.14 Global distribution of vertical ionospheric velocity, UT=12:00pm
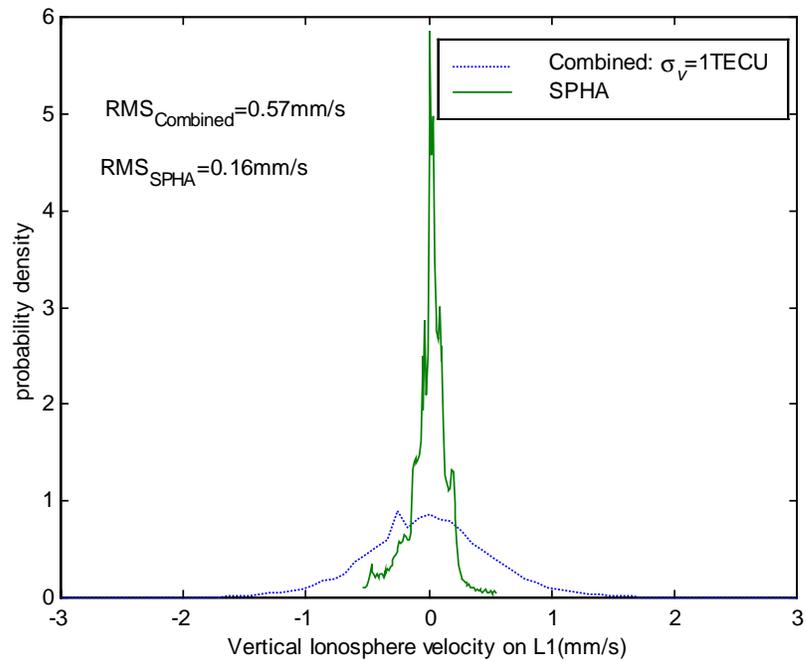
Figure 4.15 Comparison of pdf of the vertical ionospheric velocity, SPHA vs. the combined model

4.1.3.4 Ionospheric Properties under Scintillation

In case of ionosphere scintillation, the statistical properties of ionosphere can change dramatically. The comparison of scenarios with and without scintillation are shown from Figure 4.16 to Figure 4.19. First, it can be observed that the magnitude of the TEC does not change much (the variance is 0.188TECU) but has a high-frequency variation, see Figure 4.16. This fast changing rate results in a relatively large Doppler shift (the variance is 0.45 Hz) and a Doppler change (the variance is 6.26 Hz/s). The large Doppler shift increases the error in the estimation of the user's velocity, and the large Doppler change can cause the phase tracking loop to lose lock, especially for a low elevation satellite. In Figure 4.19, Doppler changes of about 60 Hz can be observed for low elevation satellites. This has largely exceeds the bandwidth of the carrier-phase tracking loop in most GPS receivers.
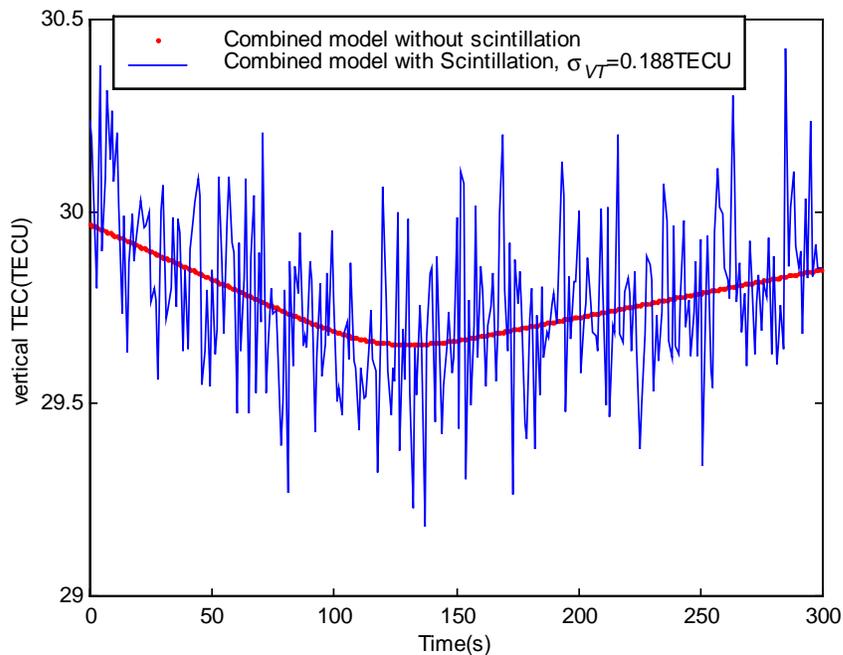
Figure 4.16 Temporal variation of vertical TEC (with/without scintillation)
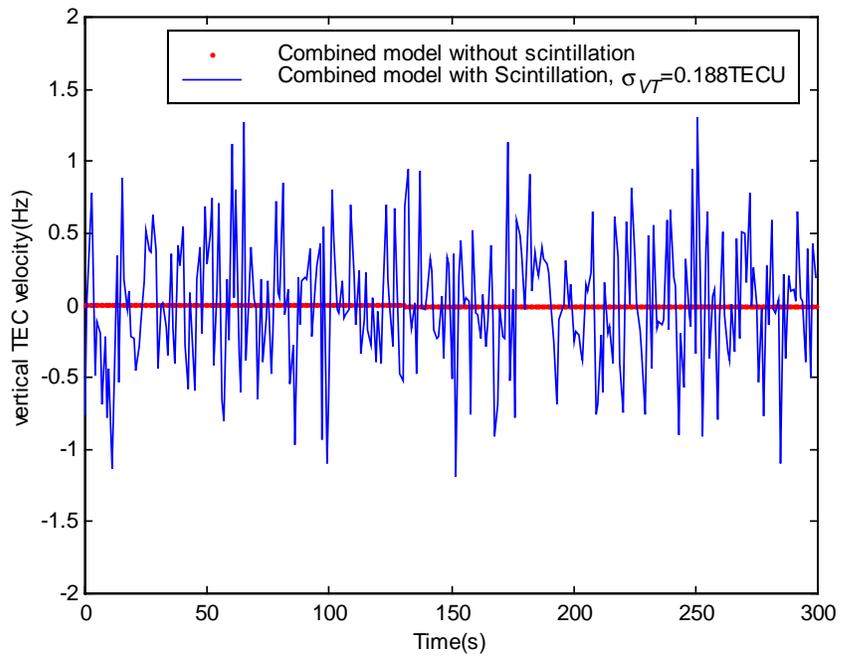
Figure 4.17 Temporal variation of vertical TEC velocity (with/without scintillation)
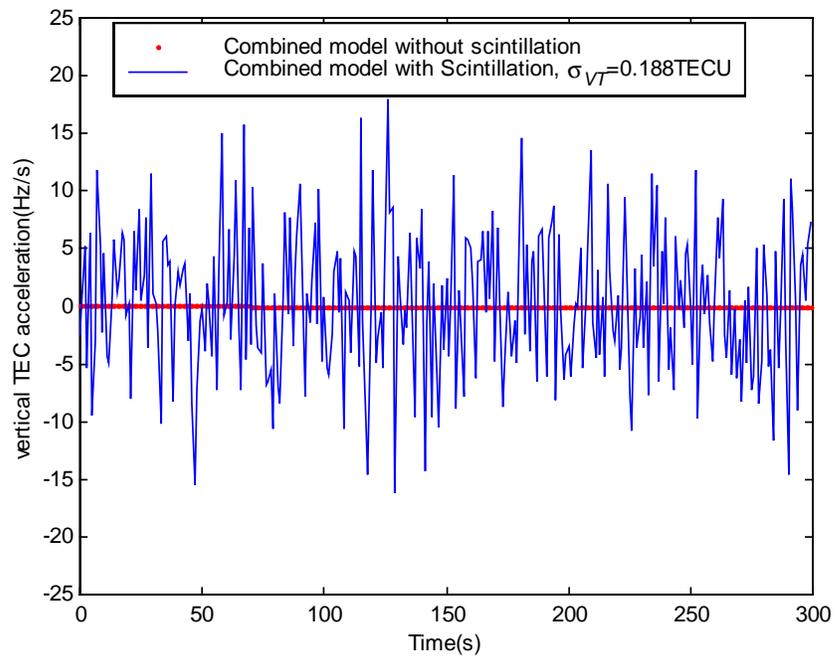
Figure 4.18 Temporal variation of vertical TEC acceleration (with/without scintillation)
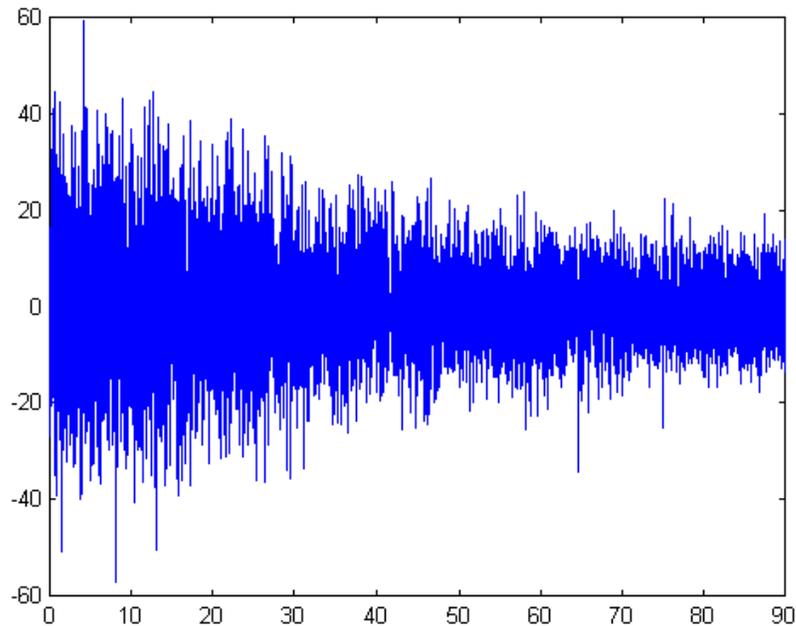
Figure 4.19 Relationship between the ionospheric acceleration and satellite elevation

4.1.3.5 *Summary of the Ionospheric Error Modeling*

The ionosphere model in this GPS software simulator is the combination of the SPHA model and the grid model, which combines the advantages of both. First, it can simulate the global profile of the TEC distribution well, hence it is globally optimized. Second, the grid algorithm is used, which not only simplifies the computation in simulation, but also improves the spatial resolution of the model to a higher level (1.5°×1.5°). Therefore, this model is valid for the simulation of DGPS applications in a small area.

The ionosphere phase scintillation model is also built to meet the requirement of simulating some extreme conditions of the ionosphere. The strategy of stopping output when ionosphere acceleration exceeds the carrier tracking bandwidth of a receiver can simulate the loss of lock in the phase-lock-loop.

The model parameters can easily be preset according to the requirement of users. Therefore, ionosphere conditions under different levels of solar activity can be simulated. In addition, for this DGPS simulator, the spatial decorrelation rate of the ionosphere can

be independently adjusted by changing the variance of TEC increment without changing the global average of the TEC.

Figure 4.20 shows an example of double differenced ionospheric errors generated by the combined model. When the ionosphere is quiet, the RMS of the differential errors is about 1 ppm. While in the case of strong ionospheric activity, large differential errors with RMS of 10 ppm can be observed.
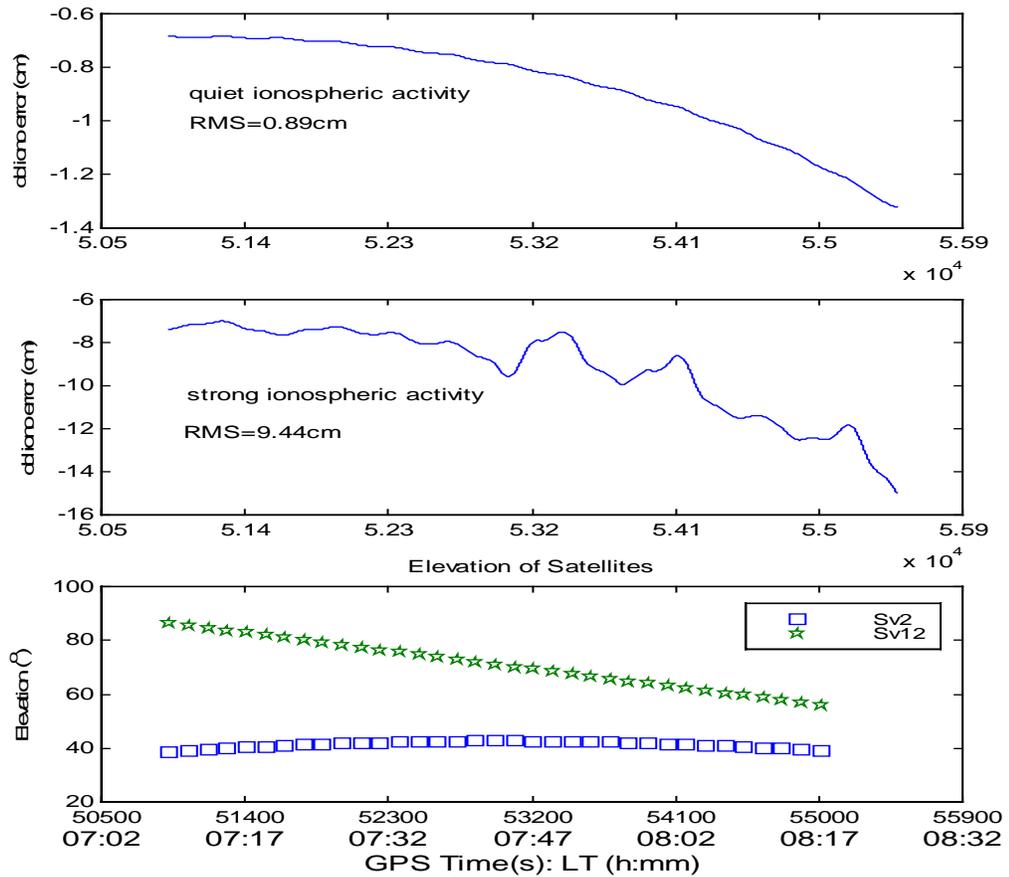
Figure 4.20 Example of double differenced (Sv 2 and Sv 12) ionospheric errors for a 10 km baseline (generated by the combined ionospheric model)

4.2   SIMULATION OF ORBITAL ERROR

Orbital (broadcast) error is one of the major error sources in GPS. DGPS greatly reduces the orbital error, however, the residual orbital errors are directly correlated with the separation of the GPS receivers. The residual DGPS orbital error depends highly on the

geometry between the reference and remote stations. The following equation is used to estimate the residual orbital error:

$$\Delta E \leq \frac{-\left|\overline{\Delta R^T}\right| \cdot \left|\overline{\Delta r_{r,u}}\right|}{R} \qquad (4.22)$$

where ΔE is the range correction error,

$\Delta r_{r,u}$ is the baseline vector between two receivers,

ΔR is the orbital error vector, and

R is the distance from a satellite to a user.

As a rule of thumb, 20 m orbital error induces 1.0 ppm baseline error (Lachapelle, 1997).

4.2.1   Data Source and Extraction of Orbital Error

To analyze the statistical characteristics of the orbital error, we must first separate it from the other GPS errors. The orbital error can be computed by subtracting the satellite's position, computed using the broadcast ephemeris, from an accurate reference orbit. In this research, the precise orbit derived by JPL, one of the data analysis centres of IGS, is selected as the reference. According to the estimation of JPL, its precise orbit (final) has an accuracy of 5.0 cm, namely, 2.5 mm differential error over a 1000 km baseline. Thus, it is good enough to be the reference.

To obtain the accurate statistic of the orbital error, a large amount of both broadcast and precise ephemeris data are required. The broadcast ephemeris used in our modeling are downloaded from the National Oceanic and Atmospheric Administration, (NOAA,http://www.ngs.noaa.gov/). The precise orbit files are downloaded from the JPL (http://igscb.jpl.nasa.gov/). These files are free for public use. The data used for this research contains ephemerides from March 1 to March 31, 2000 and 6820 ephemeris records were tested. Figure 4.21 gives an example of the three-dimensional orbital errors of a satellite for approximately two days. Some properties of the orbital error can be directly observed from this figure:

[1] It is not continuous because it is reset every two hours by the newly downloaded ephemeris.

[2] It changes very slowly.

[3] Long-term correlation exists in errors.
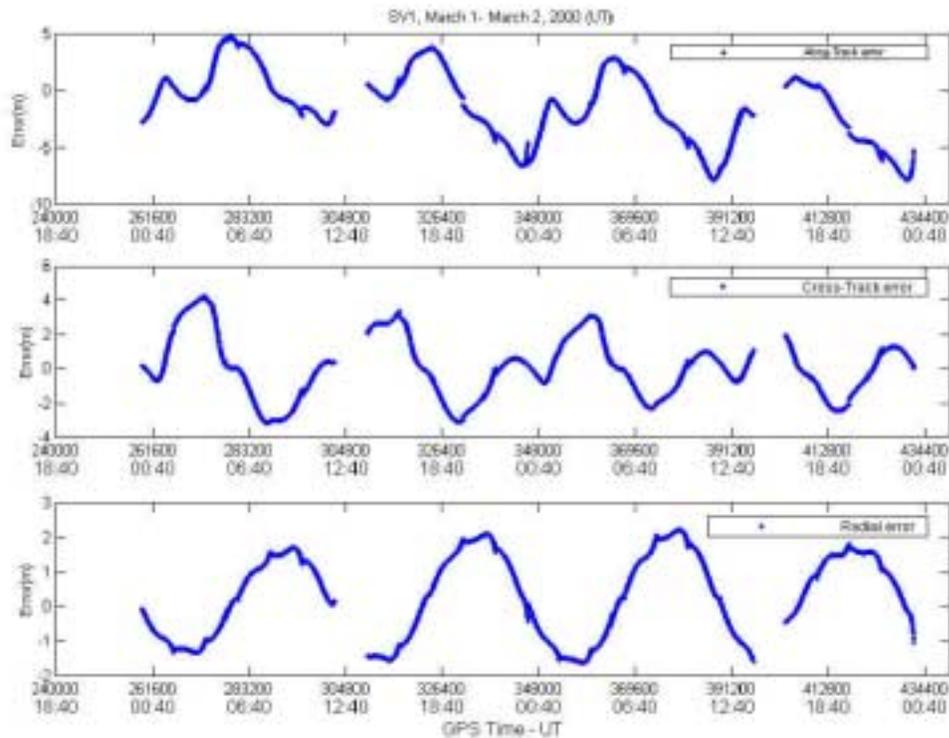
Figure 4.21 Samples of GPS orbital error

4.2.2 Parametrization of Orbital Error

After extracting the orbital error, several statistical tests can be conducted to obtain the properties of the error. To fully describe a random process, both the probability distribution and the spectrum (or the correlation function) are necessary. In the following sections, the detailed method and results of statistical tests of orbital error are presented.

4.2.2.1 Probability Distribution of Orbital Error

Generally, Gaussian distribution is the most preferable in error simulations, because it can be easily represented by its mean and variance. In addition, Gaussian random variables are easily generated in simulation. Therefore, the first test is designed to obtain the probability distribution of orbital error.

The statistical distribution of orbital error is compared with the Gaussian distribution which has the same mean and variance. The overlap area of these two distributions can be treated as an indicator of the consistency of the two distributions. Of 800,000 3D orbital error vectors tested, Figure 4.22 shows the statistical results. It can be seen that the cross-track error has the best fit distribution to a real Gaussian distribution (97.5% overlap), while the distribution of radial error deviates relatively largely (only 91.74% overlap). For the purpose of simulation, this extent of consistency is satisfactory; thus, the assumption of Gaussian distribution of orbital error can be accepted.
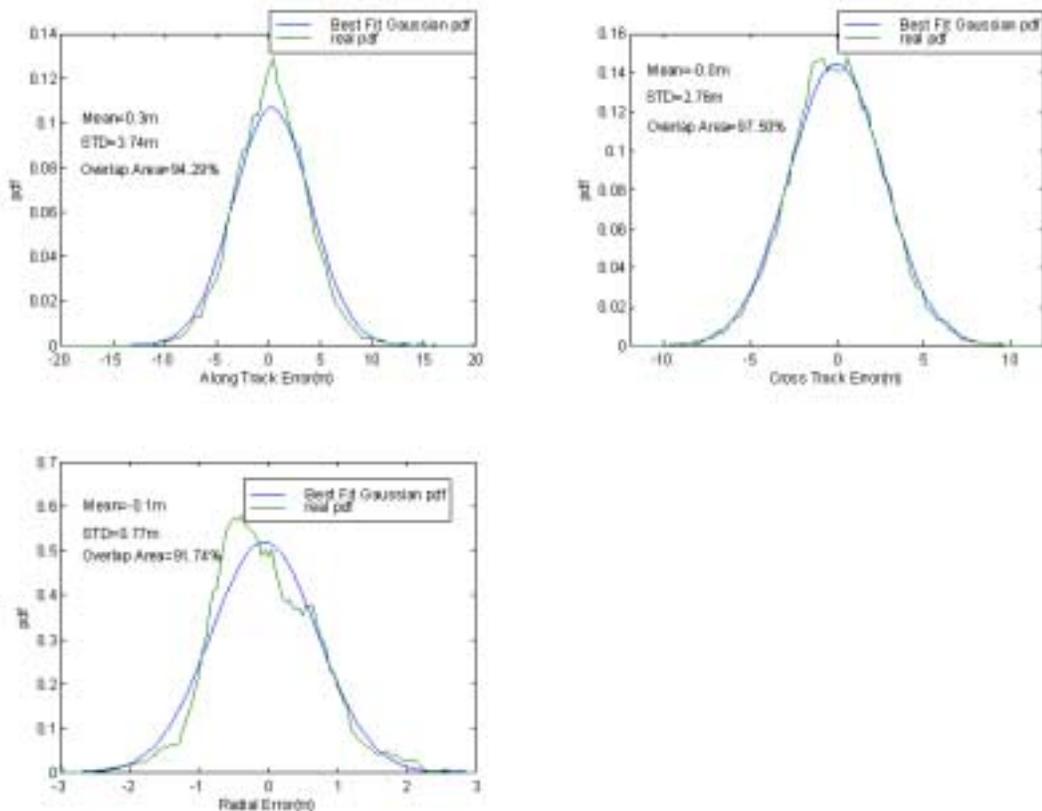
Figure 4.22 Distributions of 3D orbital errors

4.2.2.2 Analysis of Spectrum and Correlation Functions

Correlation functions are very important in describing random processes because they characterize the temporal variations of random processes. Correlation functions can be derived from either spectral or correlation analysis. Usually, due to difficulty in computation, the correlation function is rarely estimated from the time domain when the tested random sequence is very long. Instead, spectral analysis is the commonly used method. Herein, the power spectral density of the orbital error is estimated using Welch's averaged periodogram method (Welch, 1967) as shown in Figure 4.23.
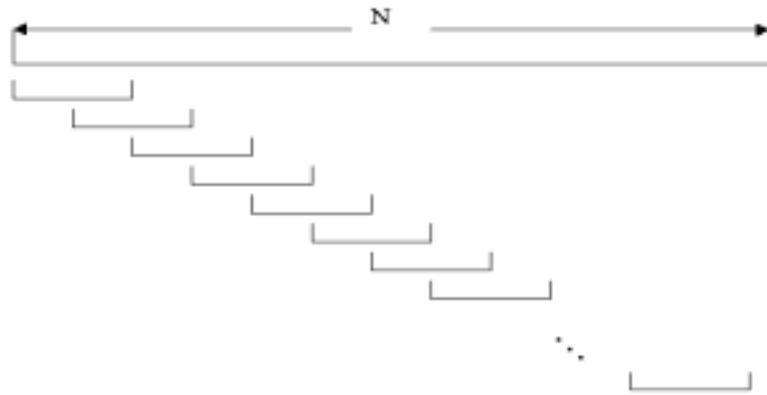
Figure 4.23 Concept of Welch's averaged periodogram method

The following procedure is used for spectrum estimation:

[1] The input signal is divided into overlapping segments, each of which is detrended, and then weighted by a Hanning window function, see Figure 4.24.

[2] The Fourier transform is performed for each segment of weighted signal to get its power spectrum $S_i(\omega)$.

$$F_i(\omega) = \int_{t_i}^{t_i+T} f_i(t)w(t-t_i)e^{-j\omega t}dt \tag{4.23}$$

$$S_i(\omega) = \frac{1}{T}F_i(\omega)F_i^*(\omega) \tag{4.24}$$

where $f_i(t)$ is one segment of signal,

$w(t)$ is the Hanning window function.

Hanning Window Function
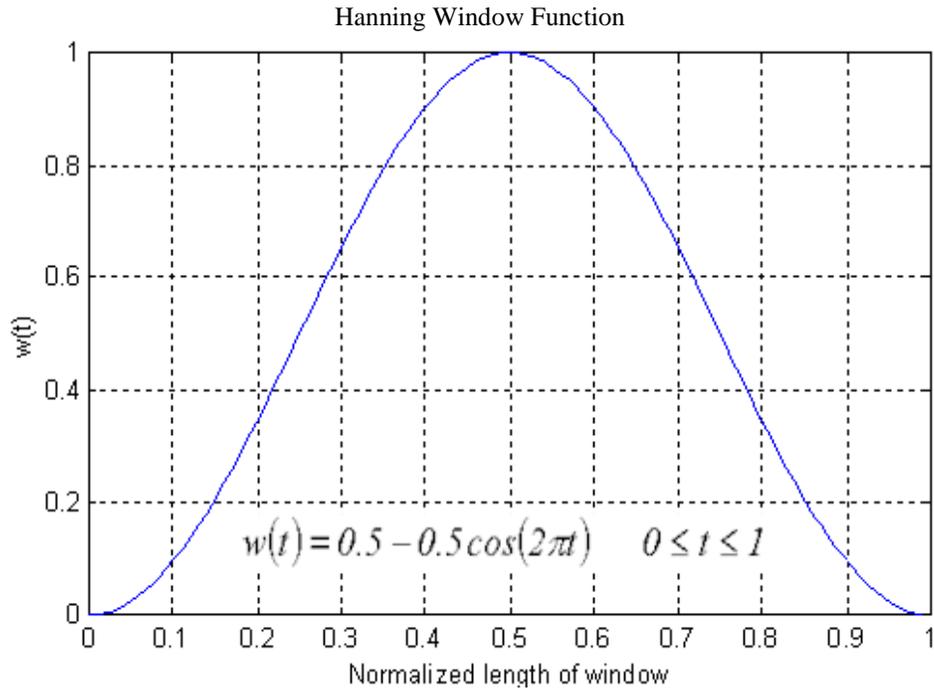
Figure 4.24 Normalized Hanning window for spectrum analysis

[3] The segmental spectrums are averaged to get the estimation of the signal spectrum $S(\omega)$:

$$S(\omega) = \frac{1}{KU} \sum_{i=1}^{K} S_i(\omega)$$ (4.25)

where $U = \frac{1}{T} \int_0^T w^2(t)\,dt$ is the normalized coefficient

K is the number of segments

[4] The correlation function $r(\tau)$ can be derived by performing the inverse Fourier Transform of $S(\omega)$:

$$r(\tau) = \frac{1}{2\pi} \int_0^{2\pi} S(\omega) e^{j\omega\tau}\,d\omega$$ (4.26)

The sampling rate is 1/180 Hz (sample/3min). In total, 818,400 samples of orbital error are used for testing. Figure 4.25 and Figure 4.26 show the estimated power spectrum and the correlation functions of the orbital errors. The RMSs of the 3D orbital errors are 3.97 m, 2.50 m and 0.73 m in along-track, cross-track and radial channels, respectively.
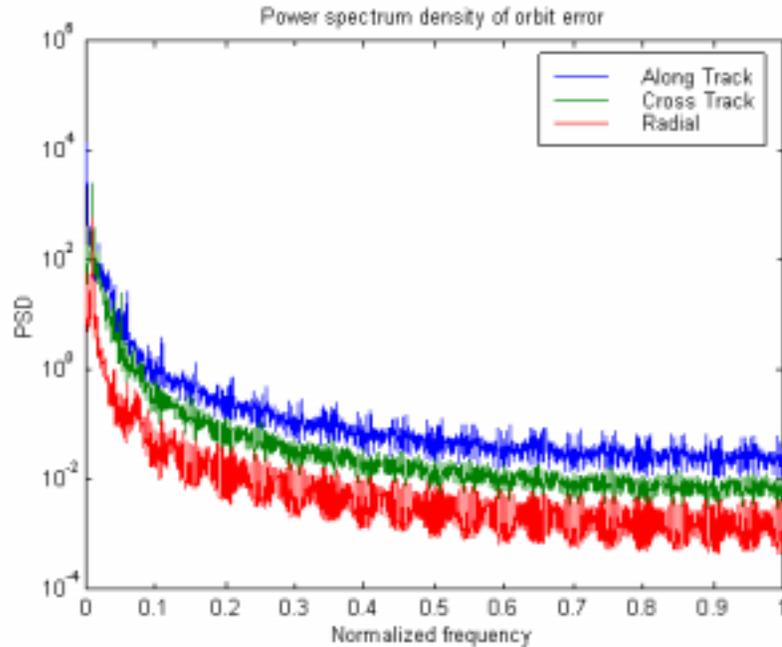
Figure 4.25 Power spectrums of 3D orbital errors
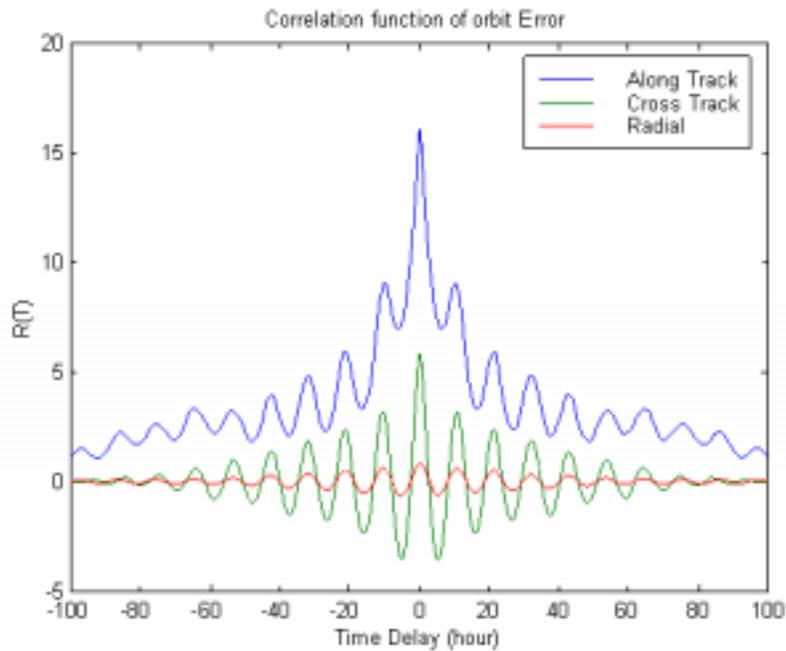
Figure 4.26 Correlation functions of 3D orbital errors

4.2.3   Error Generator

Once the statistical properties of a random process are fully estimated, a simulated process with the same properties can be generated by passing a white noise sequence through a shaping filter. The relationship between the frequency response $H(\omega)$ of the shaping filter and the spectrum of the interested signal $S(\omega)$ can be described as follows:

$$S(\omega)= H(\omega)H^*(\omega) \tag{4.27}$$

The problem of building the shaping filter in the frequency domain is that without the analytic expression of $S(\omega)$, it is impossible to derive $H(\omega)$. However, since the correlation function is known, an autoregressive (AR) model can be used to construct the shaping filter in the time domain.

Definition of AR process: The time series $u(n),u(n\text{-}1),...,u(n\text{-}M)$ represents the realization of an AR process of order M if it satisfies the difference equation:

$$u(n)+a_1u(n-1)+\cdots+a_Mu(n-M)= v(n) \tag{4.28}$$

where $v(n)$ is a white noise sequence driving the model. Since the distribution of orbital error has been shown to be approximately Gaussian, the distribution of this white noise is also Gaussian, because the Gaussian process will keep its nature after passing a linear system. The Yule-Walker equations (Haykin, 1996) can then be used to solve the coefficient a_i:

$$\begin{bmatrix} r(0) & r(1) & \cdots & r(M-1) \\ r(-1) & r(0) & \cdots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(-M+1) & r(-M+2) & \cdots & r(0) \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix}+\begin{bmatrix} r(-1) \\ r(-2) \\ \vdots \\ r(-M) \end{bmatrix}=\bar{0} \tag{4.29}$$

$$\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix}=-\begin{bmatrix} r(0) & r(1) & \cdots & r(M-1) \\ r(-1) & r(0) & \cdots & r(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(-M+1) & r(-M+2) & \cdots & r(0) \end{bmatrix}^{-1}\begin{bmatrix} r(-1) \\ r(-2) \\ \vdots \\ r(-M) \end{bmatrix}$$

where $r(k)$ is the discrete correlation function.

The variance of $v(n)$ is derived as:

$$\sigma_v^2 = \sum_{k=0}^{M} a_k r(k)$$

(4.30)

As can be seen from Figure 4.26, the orbital error has a strong long-term correlation. This implies that the order of the AR model must be very high if the sampling frequency is high. For instance, if the sampling frequency is 1/180 Hz, the order of the AR model should be larger than 2000 to represent the long-term correlation up to 100 hours. Such a high order can cause difficulty in simulation because large memory and high processing speed are required to save the coefficients and complete the filtering process. To reduce the memory requirement and to simplify the computation, a much lower sampling frequency should be used.

After analyzing the spectrum of the orbital error, it is found that more than 98% signal energy is distributed within the bandwidth of 0.05 (normalized). Ignoring the small distortion of spectral caused by an aliasing effect, the minimum sampling frequency can be set as low as 0.1, according to the Nyquist Sampling Theorem (Lathi, 1992). This corresponds to a sampling interval of one hour. As a sequence, the order of AR model can be reduced to 100. However, this solution caused another problem because the low sampling frequency of a digital filter implies a low data rate of the output data. That is, the model can only output data every hour; whereas, in simulations, the data rate can be as high as a few Hz. To densify the output, Lagrange Interpolation is used.

Definition of Lagrange Interpolation: Given a set of $N+1$ known samples (t_k, y_k), $k=0,1,2,…,N$, the sample value at any point (t,y) can be interpolated by an Nth order polynomial:

$$y(t) = \sum_{k=0}^{N} l_k(t) y_k$$

(4.31)

where $l_k(t) = \dfrac{(t-t_0)\cdots(t-t_{k-1})(t-t_{k+1})\cdots(t-t_N)}{(t_k-t_0)\cdots(t_k-t_{k-1})(t_k-t_{k+1})\cdots(t_k-t_N)}$

In simulation, a ninth-order Lagrange interpolator is applied. Figure 4.27 gives an example of the simulated orbital errors.
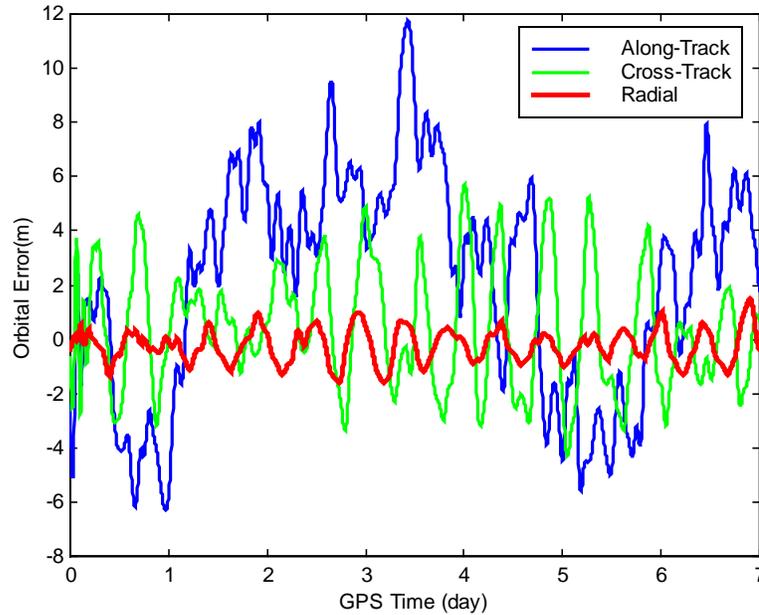
Figure 4.27 Sample of simulated orbital errors

Lowering the sampling frequency and implementing a Lagrange Interpolator can cause distortions of the spectrums or correlation functions. Only when the distortions are small enough can these two methods be used in simulation. Figure 4.28 compares the correlation functions of the real orbital errors with those of the simulated orbital errors. It shows that the correlation functions of cross-track and radial orbital error are nearly perfectly matched, while very small distortion happens at the long-term correlation of along-track error. However, this will not affect kinematic positioning at all, because long-term averaging is not applied for kinematic data processing.
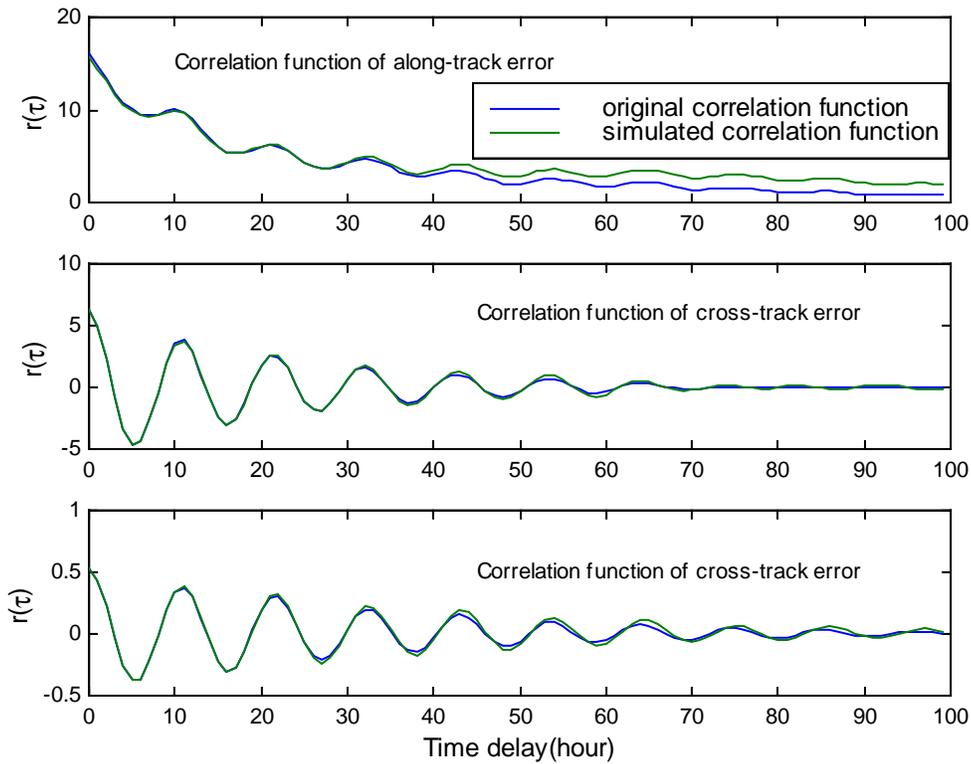
Figure 4.28 Comparison of correlation functions (statistical vs. simulated)

In summary of the orbital error modeling, Figure 4.29 shows the functional diagram of a scalar error generator (one dimension). Three scalar generators construct a vector generator which can simulate the 3D orbital errors of one satellite. Coefficients of three scalar generators are determined by correlation functions of along-track, cross-track and radial error respectively. Twenty-four vector generators are used to simulate all orbital errors of 24 GPS satellites. It should be noted that the driving noise sequences of all 72 scalar generators are independent.

In each scalar generator, the random error can be scaled to any level, which can help to study the impact of different level of orbital errors. However, in simulation tests of MultiKin, the scale factor is always set up as 1.0. A bias generator is also included in the scalar generator which is used to simulate the large orbit bias caused by the blunder in orbit predication. The magnitude of the bias can be separately controlled from channel to channel because the bias generally happens on only a few satellites.

The 3D errors output from a vector generator are transformed from the satellite-centre frame to WGS-84 to distort the real satellite position. The satellite-centre frame is defined by the motion of a satellite in an inertial frame. The orientations of its three axes point to along-track, cross-track (perpendicular to the satellite orbit plane) and Earth centre (circular orbit).
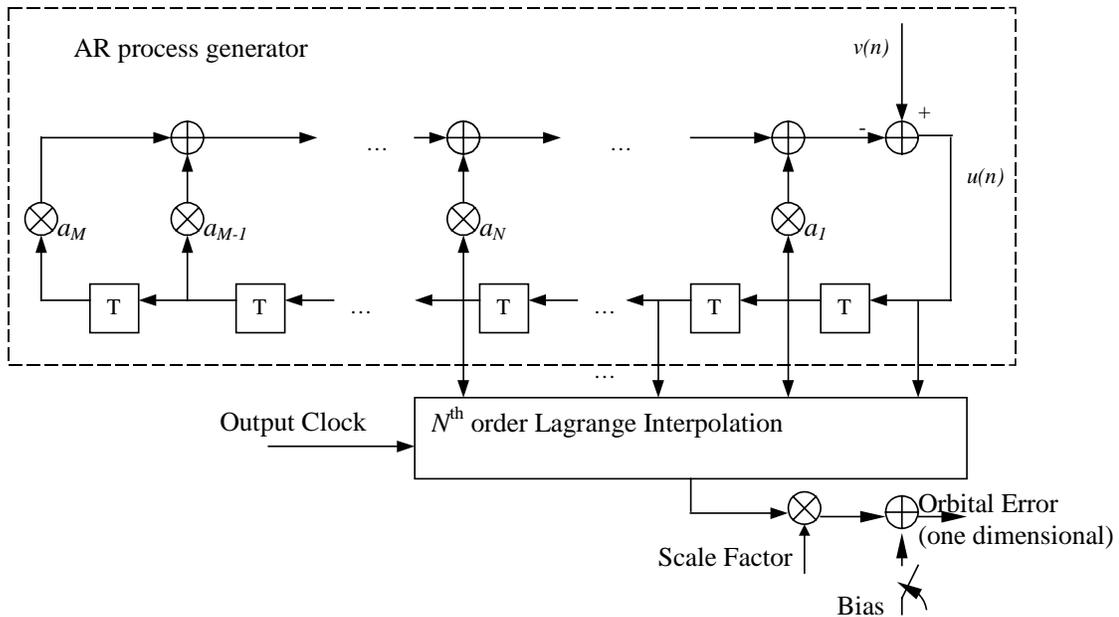
Figure 4.29 Error generator of one-dimensional orbital error

4.3   SIMULATION OF TROPOSPHERIC ERROR

There are two major delay effects of the troposphere. The first and larger effect is the dry atmosphere excess delay, typically on the order of 2.3 m. The dry effect varies slowly with local temperature and atmospheric pressure in a reasonably predictable manner. The second effect caused by the water vapor is generally smaller, 1-80 cm at zenith. Although approximately one tenth the size of the dry effect, the wet delay varies markedly, 10-20% in a few hours, and is less predictable even with surface humidity measurements (Spilker Jr., 1996).

The tropospheric error has significant effects on DGPS applications, and therefore much effort has been put into this topic and many empirical models have been developed.

Generally, the tropospheric error model consists of two parts. One is the model of the vertical tropospheric delay, such as the Saastamoinen model (Saastamoinen, 1972, 1973) and Hopfield model (Hopfield, 1969). The other part is the mapping function, such as B&E (Black and Eisner, 1984), Davis (Davis, et al, 1985), Chao (Chao, 1974), Marini (Marini, 1972) and Niell (Niell, 1993) mapping functions. Herein, a new model based on the modified Hopfield model is developed and tested.

4.3.1 Model Description

4.3.1.1 *Vertical Delay Model and Mapping Function*

Hopfield has developed a two-quartic zenith model of the refraction index, with different quartics for the dry and wet atmospheric profiles (Hopfield, 1969). Black has extended this zenith model to add the elevation angle mapping function (Black and Eisner, 1984).

The tropospheric delay is caused by the larger refractive index n ($n>1$) of atmospheric gases than that of free space ($n=1$), which causes the speed of light (group velocity) in the medium to decrease below its free space value c. The increase of propagation time caused by troposphere can be expressed as

$$\Delta t = \int_{TransmissionPath} [n(s) - 1] ds = \Delta t_d + \Delta t_w \tag{4.32}$$

where n represents both the wet and dry terms.

The refractivity is defined as $N=10^6(n-1)$. The basic two-quartic model for the refractivity versus altitude h can be expressed as:

$$N_d = N_{d_0} \left(1 - h/h_d\right)^4 \qquad h \le h_d = 43 \ km \tag{4.33}$$

$$N_w = N_{w_0} \left(1 - h/h_w\right)^4 \qquad h \le h_w = 12 \ km$$

where N_{d_0} and N_{w_0} are the dry and wet refractivities at the surface of the Earth respectively,

h_d is the height of the top shell of the "dry gas", and

h_w is the maximum height of the wet atmosphere.

The total zenith delay is then the sum of the intergrated dry and wet delays along the vertical path:

$$\Delta = 10^{-6} \int_{h_{usr}}^{h_d} N_{d_0} \left(1 - \frac{h}{h_d}\right)^4 dh + 10^{-6} \int_{h_{usr}}^{h_w} N_{w_0} \left(1 - \frac{h}{h_w}\right)^4 dh \tag{4.34}$$

where $\quad \Delta = \Delta_d + \Delta_w$

$$\Delta_d = \begin{cases} \dfrac{10^{-6}}{5} N_{d_0} \left(h_d - h_{usr}\right) & h_{usr} \le h_d \\ 0 & h_{usr} > h_d \end{cases} \tag{4.35}$$

$$\Delta_w = \begin{cases} \dfrac{10^{-6}}{5} N_{w_0} \left(h_w - h_{usr}\right) & h_{usr} \le h_w \\ 0 & h_{usr} > h_w \end{cases}$$

where $\quad h_{usr}$ is the altitude of the user antenna.

N_{d_0} and N_{w_0} can be expressed by the suface meteorologic data, namely, temperature, air pressure and humidity (Spilker Jr., 1996).

$$N_{d_0} = 77.604 \left(P_d / T\right) Z_d^{-1} \tag{4.36}$$

$$N_{w_0} = \left(e / T\, Z_w\right)\left(64.79 + 377600 / T\right)$$

where $\quad T_c$ is temperature in °Celsius,

$\qquad$ T is temperature in Kelvin,

$\qquad$ P_d is the dry air pressure in millibars,

$\qquad$ e is the partial pressure of the water vapour in millibars, and

$\qquad$ R_h is the relative humidity, which can vary from 0.0 to 1.0.

Z_d, Z_w and e can be computed as follows:

$$Z_d^{-1} = 1 + P_d \left[57.97 \times 10^{-8} \left(1 + 0.52/T \right) - 9.4611 \times 10^{-4} \, T_c / T^2 \right] \tag{4.37}$$

$$Z_w^{-1} = 1 + 1650 \left(e/T^3 \right) \left[1 - 0.01317 T_c + 1.75 \times 10^{-4} T_c^2 + 1.44 \times 10^{-6} T_c^3 \right]$$

$$e = 6.108 R_h \cdot exp \left(\frac{17.15T - 4684}{T - 38.45} \right)$$

Black and Eisner built a mapping function which can describe the slant troposphere delay as a function of both the local elevation angle of a satellite and the surface temperature:

$$M(E,T) = 1 / \sqrt{1 - \left[cos E / \left(1 + X_{dw} h_d / Re \right) \right]^2} \tag{4.38}$$

For elevation angles in the range $7° < E < 90°$ and surface temperatures in the region $-30°C < T < 40°C$, the value of $X_{dw} h_d / Re$ is in the range $0.00088 < X_{dw} h_d / Re < 0.01$. Because the temperature dependence is small enough to be ignored, the B&E mapping function can be simplified as follows:

$$M(E,T) \approx M(E) = 1.001 / \sqrt{(.001)^2 + 0.002 + sin^2 E} \tag{4.39}$$

Finally, the total slant delay of troposphere can be denoted as:

$$\Delta_s = \Delta \cdot M(E) \tag{4.40}$$

4.3.1.2 Temporal Variation of the Model

The temporal variation of the tropospheric delay is simulated in the model developed herein. This variation is realized by adjusting the meteorological data with time. The diurnal variations of the temperature and relative humidity are simulated with functions shown below:

$$T(t) = T_0 + \frac{T_v}{2} cos \left(\frac{\pi}{12} \left(t - t_{T\,max} \right) \right) \tag{4.41}$$

$$RH(t)=\begin{cases} RH_0 + \dfrac{RH_v}{2}\cos\left(\dfrac{\pi}{12}(t-t_{RH\,max})\right) & 0 \le RH(t) \le 1 \\ 0 & RH(t) < 0 \\ 1 & RH(t) > 1 \end{cases} \qquad (4.42)$$

where T_0 is the average temperature of a day,

T_v is the daily change of temperature,

$t_{T\,max}$ corresponds to the epoch with the highest temperature,

RH_0 is the average relative humidity,

RH_v is the daily change of the relative humidity, and

$t_{RH\,max}$ represents the epoch when relative humidity is maxium.

The time used in Equations (4.41) and (4.42) is local time with units of hours. These two functions are designed according to experimental results (Lachapelle, 1997). Figure 4.30 gives an example of these two functions.

$T_0=10^0 C$
$T_v=8^0 C$
$t_{Tmax}=15:00(LT)$
$RH_0=50\%$
$RH_v=20\%$
$t_{RHmax}=5:00(LT)$
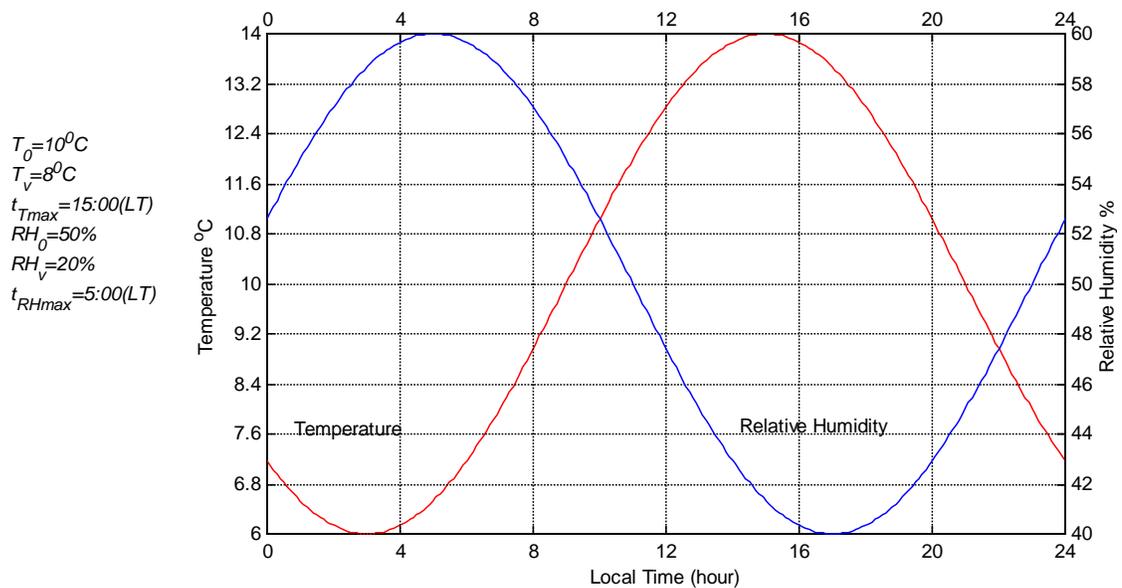
Figure 4.30 Example: Temperature and relative humidity variation with time

4.3.1.3 Spatial Variation of the Model

For DGPS simulation, the spatial correlation of the tropospheric delay must be considered. The vertical tropospheric delay is the function of temperature, relative humidity and air pressure. Regardless of the spatial correlation, these parameters are assumed constant everywhere, which can result in the constant vertical tropospheric delay in the testing area. This is far from the real situation when the testing area is large. On the other hand, if meteorological parameters at each GPS station are assumed to be random, it will also cause problem in simulation, especially in the kinematic case. For instance, two stations which are very close to each other can have very different vertical tropospheric delays due to the different setup of the meteorological parameters. This situation is obviously unrealistic. Therefore, the spatial correlation must be well designed in the model.

Both Raquet (1998) and Zhang (1999) used the least squares collocation method to describe the spatial correlation of the tropospheric delay based on the data in a regional GPS network. Their methodologies have proven very effective in predicting the spatial distribution of the tropospheric delay, but their methods can only estimate the double differenced (relative) delay, not the absolute one. To better reflect the spatial correlation of the absolute tropospheric delay in the model, the spatial distribution of temperature, relative humidity, and air pressure should be known first.

Although the global meteorological data is available, it is extremely complicated to give a full description of the spatial distribution of those data with enough density and accuracy. Therefore, the methodology is only used in some regional tropospheric error modeling, such as the UNB3 model (Collins and Langley, 1999), which uses only North American data. These data cannot represent the characteristics of meteorological data on a global scale. Thus, a stochastic model needs to be built to simulate the distribution of the required meteorological data instead of using the real one. Although the stochastic model deviates from the real situation to some degree, it provides flexibility in simulating various tropospheric effects. The following procedure is applied to produce the spatially correlated tropospheric delay.

[1] Establish the simulation area of interest. This is a rectangle consisting of trajectories of all the simulated stations. This rectangle is partitioned into $n \times m$ squares. The edge length of a square is adjustable according to the spatial decorrelation rate required in simulation. The grid size used in simulation tests of MultiKin is 100 km×100 km. See Figure 4.31.
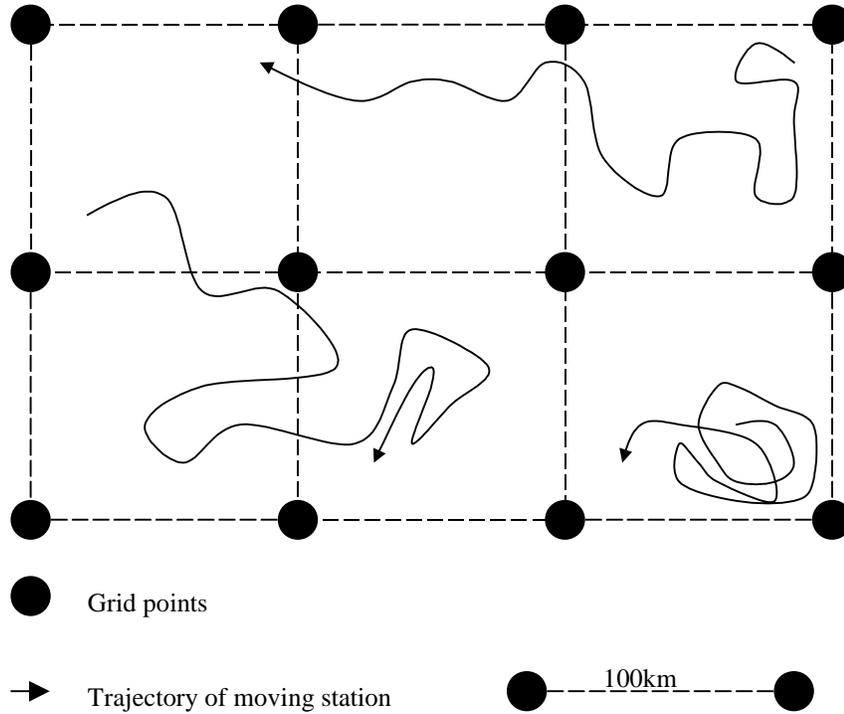
Figure 4.31 Regional tropospheric grid network

[2] Independently assign to each grid point the values of mean temperature $T_{0,i}$, temperature diurnal variation $T_{v,i}$, mean relative humidity $RH_{0,i}$, diurnal change of relative humidity $RH_{v,i}$, and air pressure Pd_i. All of these parameters are random variables satisfying the following distributions

$$T_{0,i} \in N\left(m_{T0}, \sigma_{T_0}\right)$$

$$T_{v,i} \in N\left(m_{Tv}, \sigma_{Tv}\right)$$

$$RH_{0,i} \in N\left(m_{RH0}, \sigma_{RH_0}\right)$$

$$RH_{v,i} \in N\!\left(m_{RHv}, \sigma_{RHv}\right)$$

$$Pd_i \in N\!\left(m_{Pd}, \sigma_{Pd}\right)$$

where $N(a,b)$ is the normal distribution, a is the mean value and b is the standard deviation.

[3] Compute the meteorological data of each grid point, namely, $T_i(t)$ and RH_i (t) at the given epoch t using Equations (4.41) and (4.42). $Pd_i(t)$ is assumed to be time-invariant.

[4] Interpolate the meteorological data at the GPS antenna using a four-point bilinear interpolation, which is similar to that used in ionospheric error simulation (see section 4.1.1.3). The only difference is that the unit of distance used here is kilometres, while in ionospheric error simulation, it is arc degrees.

[5] Compute the total slant delay of the troposphere using the meteorological data and the elevation of a satellite.

Although the meteorological data at each grid point is independent, the interpolation will generate the spatial correlation within the network. Thus, the resulting tropospheric delay is also spatially correlated.

4.3.2 Test of the Troposphere Model

Figure 4.32 shows the change of tropospheric delay with the meteorological data and the altitude of a user. It can be observed that the wet delay and dry delay change in opposite directions with temperature. The total vertical delay is not sensitive to temperature variation. Besides, the air pressure is often treated as constant; thus, the major element affecting the tropospheric delay is the relative humidity.

Figure 4.33 exhibits the diurnal variation of the meteorological data and related tropospheric parameters at the centre of a four-point grid network (100 km×100 km). It further shows that the relative humidity has the greatest effect on the tropospheric delay because the total vertical tropospheric delay changes in the same way as the relative

humidity. This can be seen by comparing Figure 4.33a and c. Figure 4.34 displays an example of the spatial distribution of the meteorological data and the corresponding tropospheric parameters in a four-point grid network (100 km×100 km). Simulation results show the typical values of the vertical tropospheric delay (2.4 m) and its gradient (0.5 ppm).

a

b

c

d
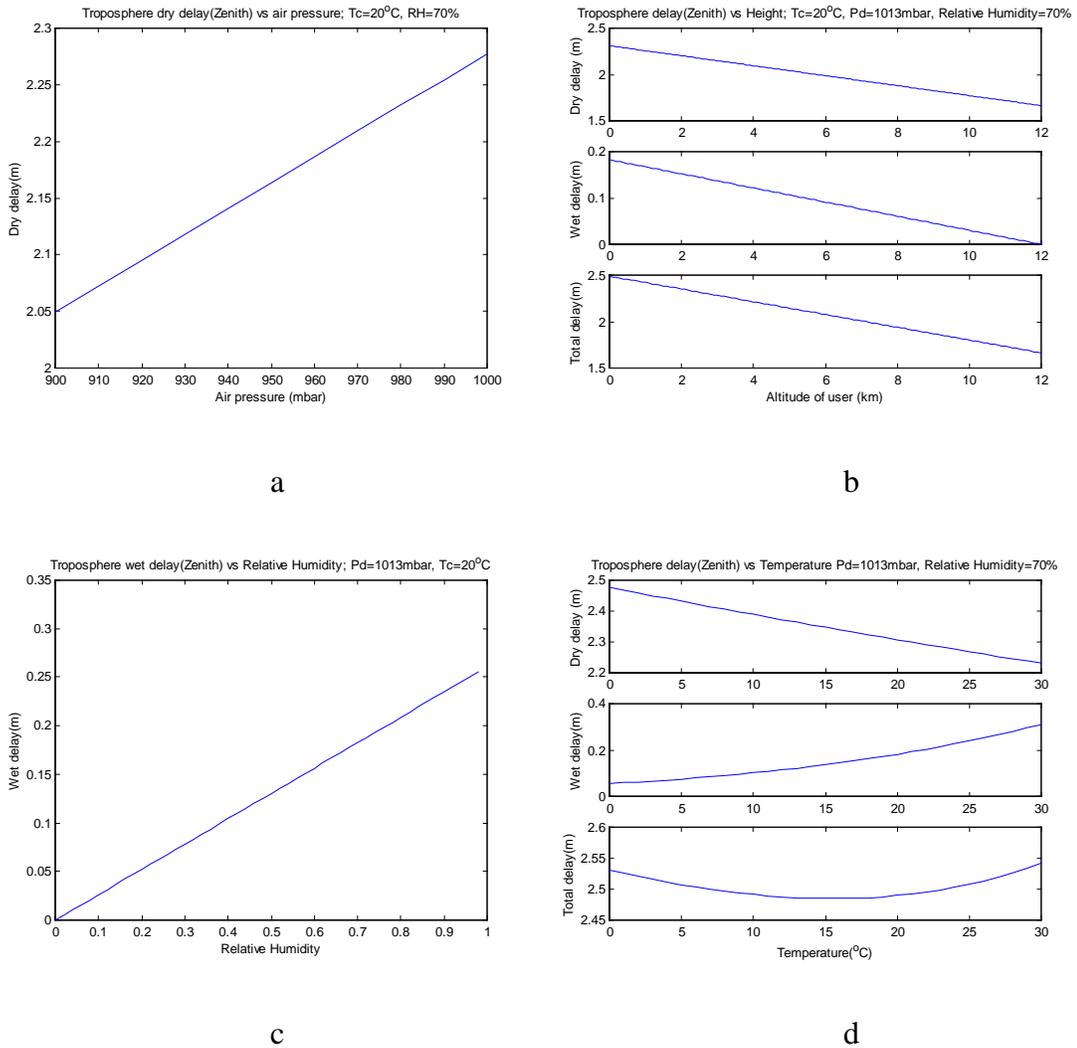
Figure 4.32 Tropospheric delay variation with meteorological data and user height
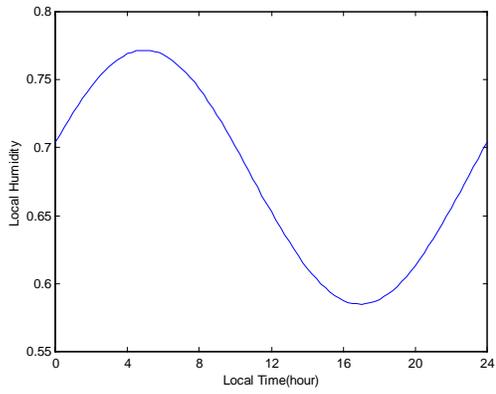
a
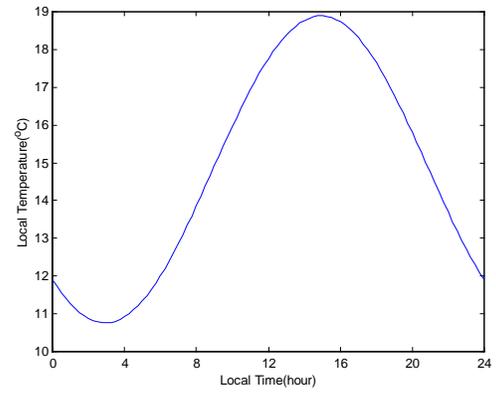
b
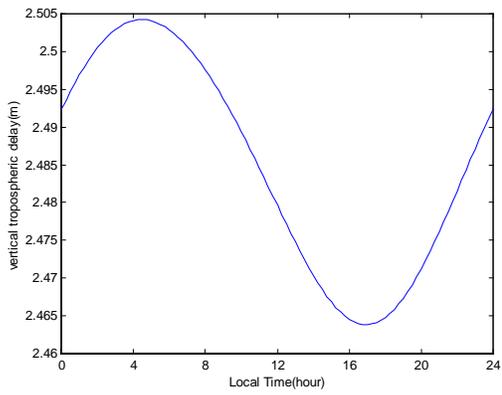
c
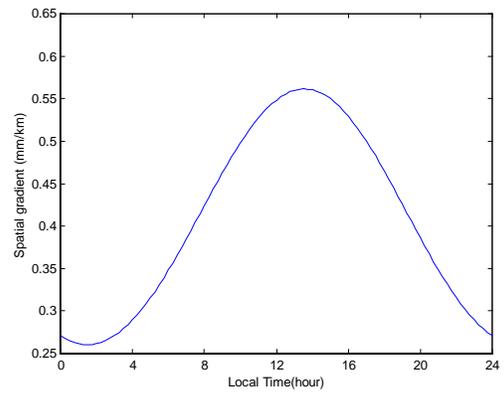
d

Figure 4.33 Temporal variations of the meteorological data and tropospheric delay in a 100 km×100 km regional network

a

b

c

d

$m_{T0} = 15°C \ \sigma_{T_s} = 2°C$,
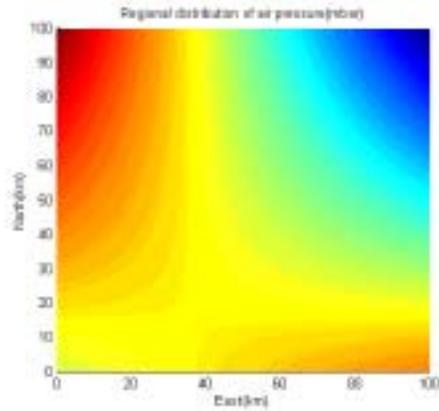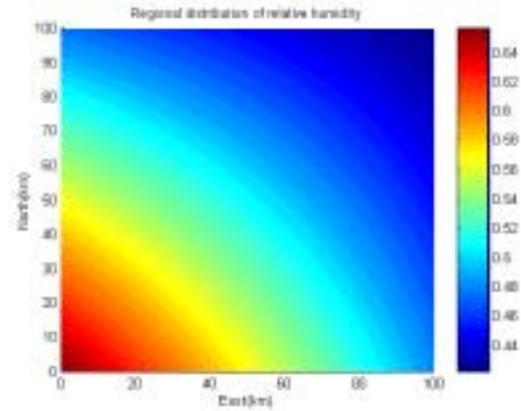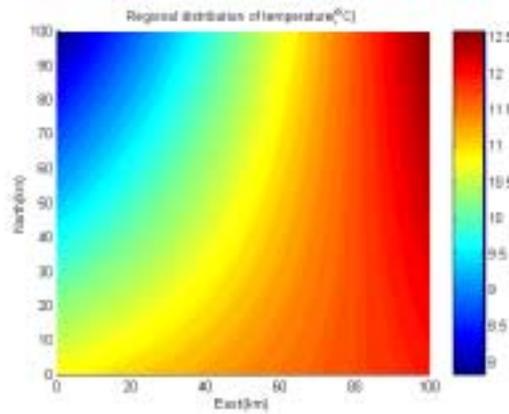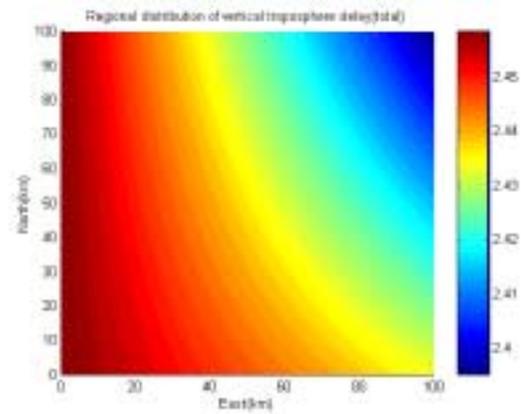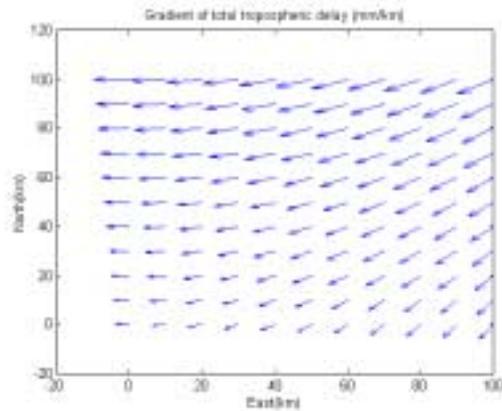
$m_{RH0} = 0.65$, $\sigma_{RH_s} = 0.05$,

$m_{Pd} = 1003mbar$

$Time = 12 : 00 pm(LT)$

$Gradient_{max} = 0.64mm/km$

$Gradient_{mean} = 0.42mm/km$

e

Figure 4.34 Distributions of meteorological data and vertical tropospheric delay in a 100 km×100 km regional network

4.4   SIMULATION OF MULTIPATH

Multipath is one of the dominant errors in GPS positioning. It is caused by the mixture of the direct signal and indirect signals from various reflectors around the antenna. The magnitude of multipath depends on the properties of the reflector, the distance between the reflector and the antenna, the gain pattern of the antenna, and the type of the tracking loop used in a receiver.

The simulation of multipath is very important to test DGPS systems. Multipath is not spatially correlated beyond a few centimetres (Ray 2000); thus, it cannot be reduced by differencing observations between receivers. Meanwhile, multipath is highly environment dependent, so it is less predictable than other major GPS error sources. It can have a major impact on differential carrier phase positioning systems when baselines are relatively short (<10 km).

4.4.1   The Multipath Model Built by the University of Calgary

A sophisticated model of multipath has been developed by the Department of Geomatics Engineering at the University of Calgary (Ray 2000, Ryan 2000). This model (called the UofC model below) is built based on the mechanism of the multipath generation. It contains three major parts: simulation of reflecting environment, simulation of antenna gain pattern, and simulation of tracking loop (both code and carrier).

In the first part of the model, a group of reflectors are configured to establish a specific reflecting environment. Users can set up the position, size, and shape of each reflector, as well as its reflecting coefficient (i.e., the strength of reflection). Once the reflecting environment is built, the number, strength and direction of the reflected signals can be determined, together with the path delays with respect to the direct signal.

In the second part of the model, users can simulate the gain patterns of different antennas, such as an omni-directional antenna, or a choke-ring antenna, which has a low gain at low elevations. Scaled by the antenna gain the strength of the received multipath signal can be decided. The impact of multipath on the range measurement still depends on the type of the tracking loop used in a GPS receiver. If a narrow-correlator is adopted, the

impact of multipath on observations can be largely reduced. Therefore, in the third part of the model, users may choose different types of tracking loops to simulate the applied receivers.

Here is an example of the option file used for multipath simulation in the UofC model.

```
MULTIPATH SIMULATION ANALYSIS PROGRAM VERSION 2.0

START WEEK NUMBER                  :        1023
START TIME (GPS SECONDS)           :        0.0
STOP WEEK NUMBER                   :        1023
STOP TIME  (GPS SECONDS)           :        86400.0
DATA INTERVAL (SECONDS)            :        5.0
MASK ANGLE (DEGREES)               :        5.0
ANTENNA GAIN PATTERN MODE          :        INTERNAL
CORRELATOR SPACING CHIPS (E-P)     :        0.50
DLL DISCRIMINATOR (COHERENT/DOT):           DOT
REFLECTOR INPUT FILENAME           :        e:\files\source\multisim\reflection.txt
```

The "internal" gain pattern means the gain pattern for the Sensor System's GPS patch antenna model S67-1575-Series (Sensor Web) is employed. A wide-correlator receiver with a non-coherent delay lock loop (DLL) is simulated for testing. The "reflector input file" defines the environment over which multipath simulations will be conducted. Several records are defined in this file and each record includes the coordinates (local level frame) of vertices of a reflecting polygon (or point). The polygons defined within this file are to determine whether there are any valid multipath reflection points for every visible satellite. The current simulation program can handle 50 separate polygons, each with a maximum of 10 vertices. An example of the description of a rectangular reflector is given as follows.

```
REFLECTOR MODE                :  PLANE
REFLECTOR MAGNITUDE           :  0.5

  40.000000   10.000000   0.000000
  40.000000   10.000000  10.000000
 -40.000000   10.000000  10.000000
 -40.000000   10.000000   0.000000
```

Reflector Mode: The reflecting surface can be either a fixed reflection, meaning that a reflection is forced from this point regardless of the geometry, or the reflector is a plane reflector defined by a polygon.

Reflector Magnitude: This specifies the magnitude of the reflection coefficient for the given reflector. This is used to scale the magnitude of the reflected signal, the valid value is from 0.0 to 1.0.

Figure 4.35 shows the procedure of multipath simulation. The detailed explanation can be obtained from Ryan (2000).
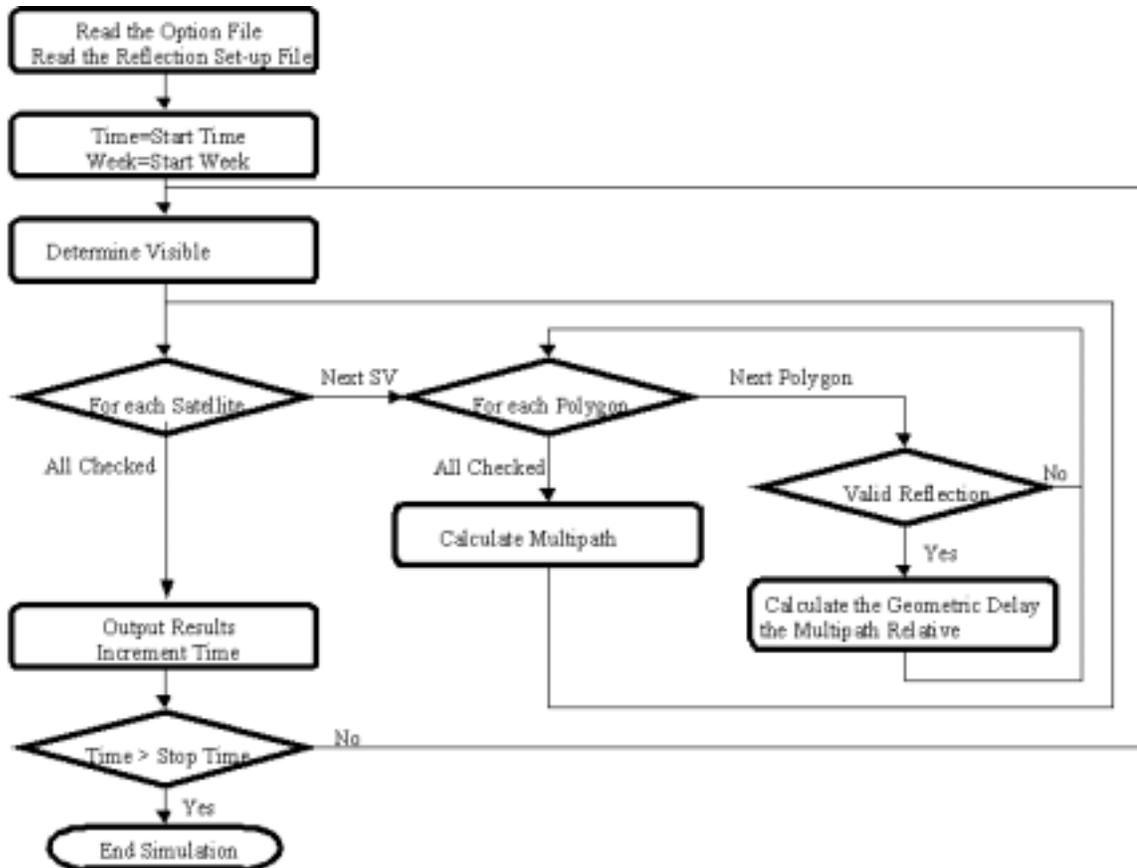
Figure 4.35 Multipath simulation program flowchart of UofC model

4.4.2   Simplification of the UofC Model

The UofC model is simplified in this research. In the simplified model, the simplest reflecting environment is defined. It is an infinite ground plane which has different reflecting coefficients (strength) at different reflecting points. According to this configuration, the user antenna can always get one and only one reflected signal from the ground. Thus, the complicated computation of detecting valid reflected signals, which

includes searching for reflecting points and possible blockage by the reflectors, can be omitted. Furthermore, it is much easier to compute the path delay of a reflected signal.

Since the satellite is far away from the user, the signal arriving at a reflecting point can be treated as parallel to the signal arriving at an antenna. Thus, the path delay of the reflected signal can be determined without using information of a satellite position, See Figure 4.36.

$$\ell_{delay} = \frac{h}{\sin E}(1 - \cos 2E)$$ (4.43)

$$\ell_{delay} = 2h \sin E$$

where h is the height of an antenna with respect to the ground plane,

E is the elevation angle of a satellite.
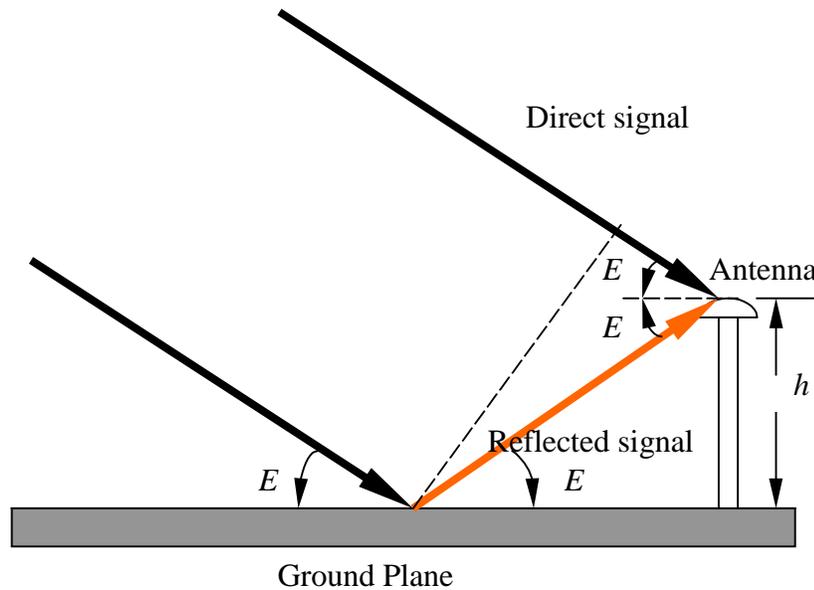
Figure 4.36 Computation of the path delay in the simplified UofC model

Figure 4.37 gives an example of multipath (both code and carrier) generated by the simplified model for a static platform. The height of the antenna is 1.5 m. The reflecting

coefficient is a random variable with average distribution from 0.8 to 1.0, which results in the high frequency component of the multipath.
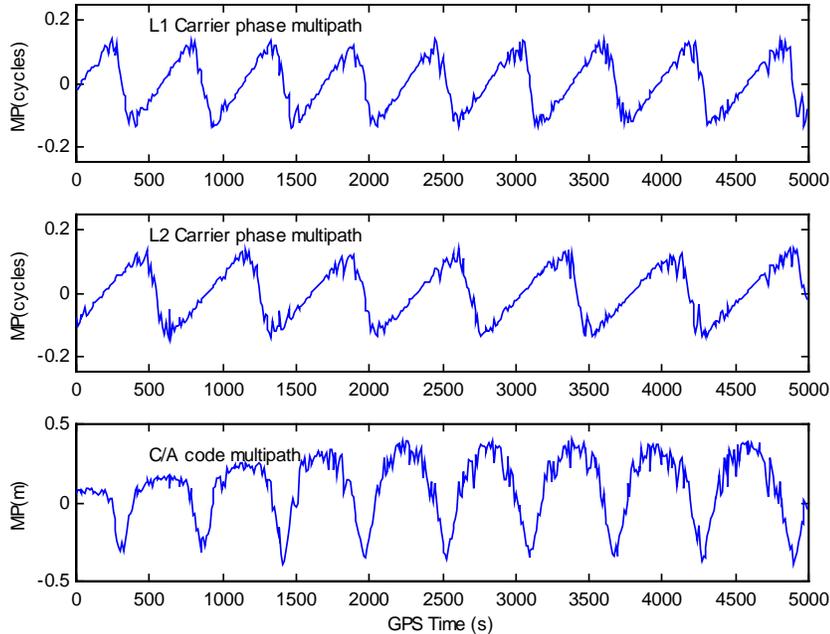
Figure 4.37 Simulated static multipath by the simplified UofC model

4.4.3 Kinematic Multipath Model

The advantage of the UofC model is that it can give a very accurate estimation of multipath if the reflecting scenario can be very well reproduced. The geometry of satellites and the dynamics of an observer can cause the multipath to change in a way analogous to a real situation. However, the high accuracy of the modeling is obtained at the expense of the increased complexity of the model.

The UofC model is suitable for simulations of static applications because it has a relatively stable environment of reflection which may be re-established in simulation. For kinematic simulation, however, the environment can change dramatically with time; thus, it is impossible to re-construct the reflection scenario along the trajectory of a moving platform. To simulate multipath for kinematic applications, the statistical method is used instead. The kinematic multipath error can be treated as a Gauss-Markov random process

(SATNAV, 1998), with the time constant dependent on the dynamics of the moving platform. Generally, the higher the dynamic, the shorter the time constant.

In kinematic multipath modeling, the multipath of the code, L1 carrier phase and L2 carrier phase are assumed independent, which seems to contradict our experience. The L1 and L2 carrier phase multipath are always considered to have the same multipath phase delay but different frequencies due to the difference of their wavelengths (Braasch, 1996). These experiences are generally based on the assumption of a single reflected signal, while in practice the multipath is usually a mixture of multiple reflected signals with different strengths and phase delays. The strength and phase delay of each reflected signal is also a function of time. Therefore, the multipath phase delay actually behaves randomly. From a statistical perspective, the multipath on L1 appears to be independent of the multipath on L2. As a result, the important issue in simulations is to make sure that the multipath error generated for L1 is statistically independent of the multipath error for L2. In the simulator, this is accomplished simply by using different 'seeds' for the random number generator.

Here is the method to generate a first-order Gauss-Markov process for kinematic multipath simulations:

$$X_{k+1} = e^{-\beta(t_{k+1}-t_k)} X_k + w_k \tag{4.44}$$

where X_k is the multipath error,

w_k is a Gaussian white noise to drive the model.

The variance of w_k is as follows:

$$q_k = \sigma^2 \left[1 - e^{-2\beta(t_{k+1}-t_k)} \right] \tag{4.45}$$

where σ^2 is the variance of the multipath required in simulations,

$1/\beta$ is the time constant of the random process.

In kinematic applications of car positioning, the time constant can be set to several seconds (Nayak, 2000).

To check the independence of the simulated multipath, a statistical test is conducted. The cross-correlation of two simulated multipath sequences is computed using the following equation:

$$c_{ij} = \frac{1}{N\sigma_i\sigma_j} \sum_{k=1}^{N} x_{i,k} x_{j,k}$$ (4.46)

where N is the length of the simulated sequence,

σ_i and σ_j are the standard deviations of the ith and j^{th} sequences, respectively,

x_i and x_j are the simulated multipath sequences.

The test conditions are listed in Table 4.1.

Table 4.1 Test conditions of correlation of simulated multipath sequences

Number of sequence	20
The time constant of multipath	10 s
The length of each sequence	10000 s
The data rate	One sample/s

Figure 4.38 gives results of the cross-correlation test. There are 190 cross-correlations calculated between any two of the 20 simulated sequences. The mean cross-correlation between simulated sequences is only 0.024. In some extreme cases, the cross-correlation can be up to 0.1 but it is still acceptable for simulations.

Figure 4.39 shows an example of the simulated kinematic multipath with a time constant of 10 s, which is quite different from the static multipath generated by the simplified

UofC model. Because of the difference of error properties, both the UofC model and this kinematic model are employed in the software simulator.
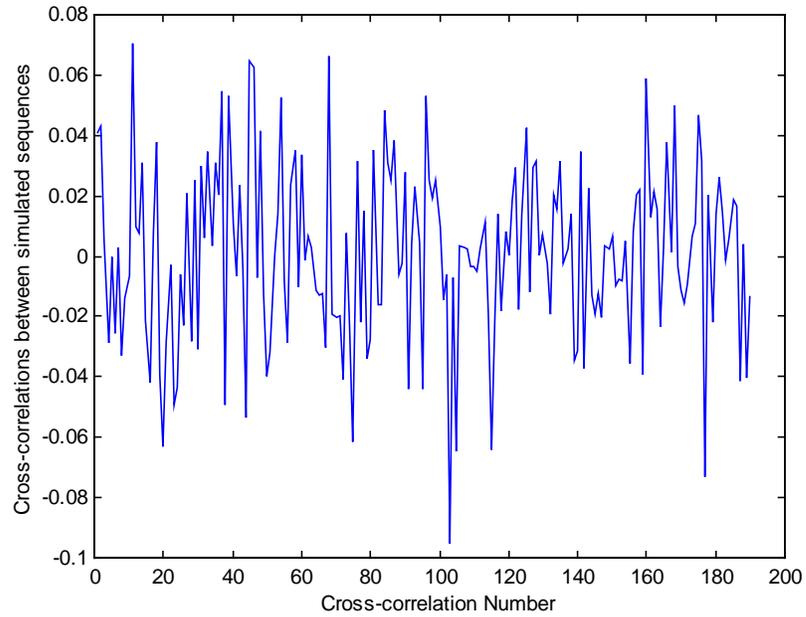
Figure 4.38 Cross-correlations between the simulated multipath sequences
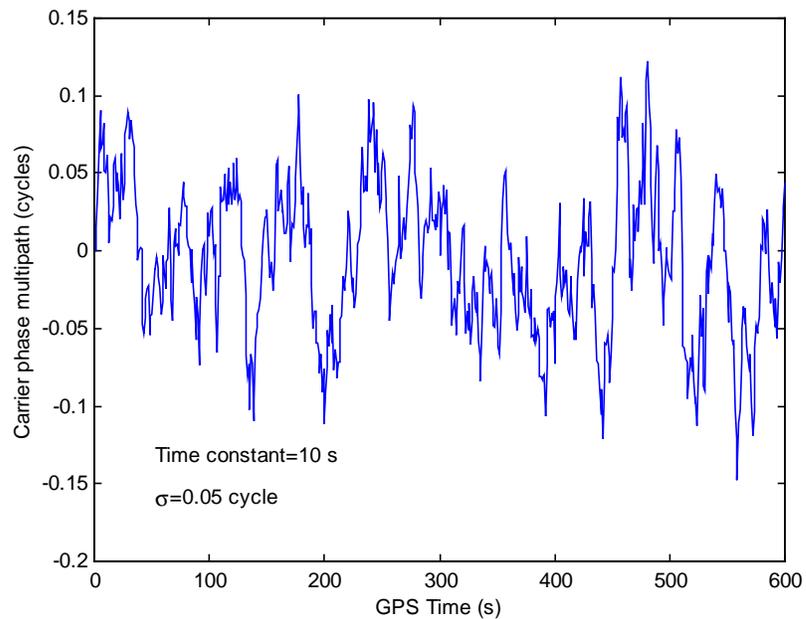
Figure 4.39 Example of simulated multipath for kinematic applications

4.5 SIMULATION OF SA

SA used to be the largest error source for stand-alone positioning which could degrade the horizontal positioning accuracy to 100 m (95%) and vertical accuracy to 150 m (95%). However, it did not affect differential GPS because SA (clock dithering) could be eliminated by differencing between two receivers if the measurement latency could be ignored. As discussed in Chapter 3, the relative positioning accuracy is degraded by the position error in the moving reference; therefore, SA is only simulated to test the degradation of relative positioning in case of large reference errors.

In this section, the discussion is focused on the simulation of satellite clock dithering (δ-error). Much effort has been put into modeling the effect of SA. Generally, models in existence can be categorized into the three following types (Van Graas and Braasch, 1996):

Second-Order Gauss-Markov Model: It can generate both range and range rate errors, but the error is noisier than the actual value (Studenny, 1993).

Autoregressive Model: It can only generate range error, but matches well to the measured SA data (Braasch et al, 1993).

Analytic Model: This model integrates both advantages of above models, but there is no direct relation between the choice of the model parameters and the output data. Noise levels must be set empirically (Lear et al, 1992).

Since the high accuracy of modeling is not a major concern for this research, an 11th order AR model is chosen according to the result derived by Braasch et al (1993). The corresponding AR coefficients and the variance of the driving noise input for an output data rate of one-second are given below.

```
a[1]  = -1.36192741558063
a[2]  = -0.15866710938728
a[3]  = +0.13545921610672
a[4]  = +0.21501267664869
a[5]  = +0.30061078095966
a[6]  = -0.12390183286070
a[7]  = +0.10063573000351
a[8]  = +0.02694677520401
a[9]=   -0.12898590228866
a[10]= +0.05083106570666
a[11]= -0.05600186282898
```
$$\sigma_e^2 = 1.6993 \times 10^{-3}$$

An example of the simulated SA in the range measurement is shown in Figure 4.40.
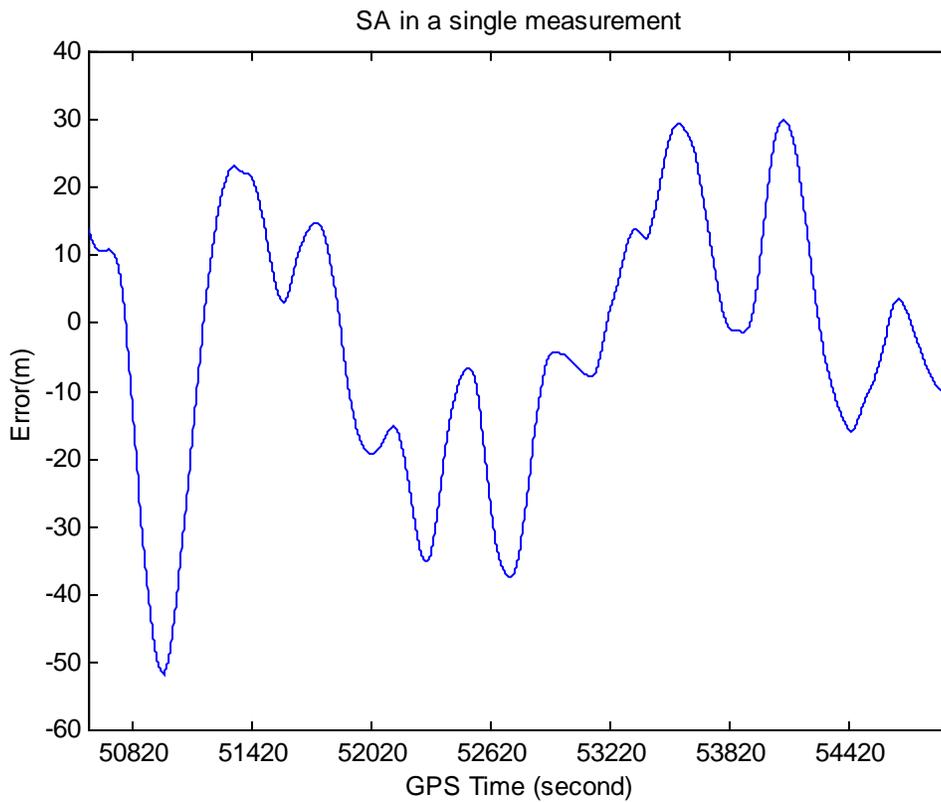
Figure 4.40 Example of the simulated SA (clock dithering)

5 DESIGN OF SCENARIOS FOR SIMULATION

5.1 CONSTELLATION SIMULATOR

5.1.1 Description of Constellation Simulator

In a GPS software simulator, the constellation generator is necessary to simulate the position of GPS satellites which are used to generate range measurements and some satellite-elevation dependent errors.

To describe a satellite orbit, six Keplerian elements are required, including:

a: semi-major axis

e: eccentricity

i: inclination angle of orbit with respect to the equatorial plane

ω: argument of perigee

Ω: right ascension of ascending node

E: eccentric anomaly (true or mean anomaly used alternatively)

In this simulator, for simplification, a circular orbit is used instead of an elliptical orbit. This is a good approximation because the eccentricity of the real GPS orbit is very small (about 0.01). For a circular orbit, only four parameters are necessary, because the eccentricity and argument of perigee can always be treated as 0. Consequently, the computation of position and the velocity of satellite become easier.

The position of a satellite on the orbit plane is:

$$r_o = \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} = r \begin{bmatrix} \cos E \\ \sin E \\ 0 \end{bmatrix}$$

(5.1)

where r is the radius of a circular orbit,

E is equivalent to the mean anomaly and can be computed using the following equation.

$$E = E_0 + n(t - t_0)$$
(5.2)

where E_0 is the eccentric anomaly at a reference epoch t_0,

n is the angular velocity of a satellite, which is a constant for circular orbit. It can be written as:

$$n = \sqrt{\frac{GM_E}{r^3}}$$
(5.3)

where G is the universal gravitational constant,

M_E is the mass of the Earth.

Observations of satellites are usually performed on the Earth. The coordinates of stations are conveniently referred to an Earth-fixed frame (e.g., WGS84). Therefore, a coordinate transformation is necessary to convert the satellite position from an orbit plane to WGS84.

$$r_e = Rr_0$$
(5.4)

where r_e is the position vector of a satellite in WGS84,

R is the rotational matrix for coordinate conversion.

Because z_o is always 0, the rotational matrix can be simplified as:

$$R = \begin{bmatrix} \cos\Omega' & -\sin\Omega'\cos i \\ \sin\Omega' & \cos\Omega'\cos i \\ 0 & \sin i \end{bmatrix}$$
(5.5)

where $\Omega' = \Omega - \dot{\Omega}_e(t - t_0)$

$\dot{\Omega}_e$ is the Earth rotational rate.

The R matrix is also used to get the satellite velocity in WGS84.

$$\dot{r}_e = R\dot{r}_0 \tag{5.6}$$

In order to compute the position and velocity of a satellite, four parameters must be initialized, namely (r, i, Ω, E_0). The inclination angle and period of GPS satellites are always treated as constants: thus, they are actually known to us. In simulation, i is set up to 55° and the radius r can be derived from the period of a GPS satellite using the following equation:

$$r = \left(\frac{GMT^2}{4\pi^2} \right)^{\frac{1}{3}} \tag{5.7}$$

where $T = 11^h 58^m$ is the period of a GPS satellite. As a result, $r = 26560.623$ km.

The selection of right ascension and mean anomaly is referred to a standard 24-satellite constellation (Massatt and Zeitzew, 1998 and Massatt, 1991), which is listed in Table 5.1. The relative positions of GPS satellites at a reference epoch are shown in Figure 5.1.

Table 5.1 Parameters of a GPS 24-satellite constellation

Slot	Mean Anomaly (degree)	Right Ascension (degree)	Slot	Mean Anomaly (degree)	Right Ascension (degree)
A3	11.68	272.85	D1	135.27	92.85
A4	41.81	272.85	D4	167.36	92.85
A2	161.79	272.85	D2	265.45	92.85
A1	268.13	272.85	D3	35.16	92.85
B1	80.96	332.85	E1	197.05	152.85
B2	173.34	332.85	E2	302.6	152.85
B4	204.38	332.85	E4	333.69	152.85
B3	309.98	332.85	E3	66.07	152.85
C1	111.88	32.85	F1	238.89	212.85
C4	241.57	32.85	F2	345.23	212.85
C3	339.67	32.85	F3	105.21	212.85
C2	11.8	32.85	F4	135.35	212.85
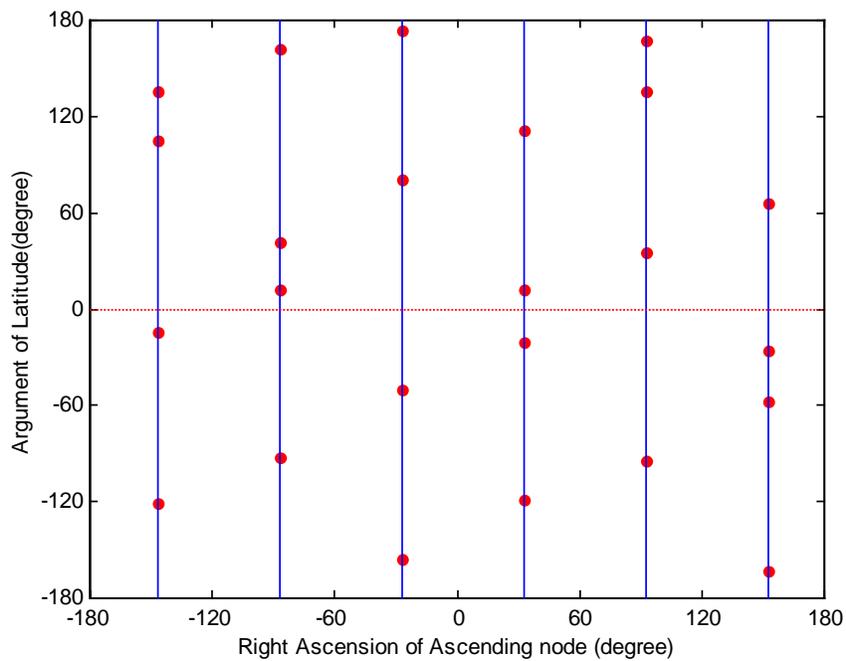
Figure 5.1 Six-plane 24-satellite constellation

5.1.2 Test of the Constellation Simulator

To check whether the constellation is properly built several tests are performed to show the satellite coverage and geometry.

[1] The Ground Track

From Figure 5.2, it can be seen that after one sidereal day ($23^h 56^m$), the satellite returns to the local zenith of an observer. Thus, the radius and period of the simulated GPS orbit are correctly configured.

[2] GPS Satellite Geometry and Visibility

All simulation tests of MultiKin were conducted in a region centred at (51°04' N, 114°07' W), which is close to the University of Calgary. The field tests were performed in the same area. Figure 5.3 and Figure 5.4 show the satellite geometry and visibility in the testing area.
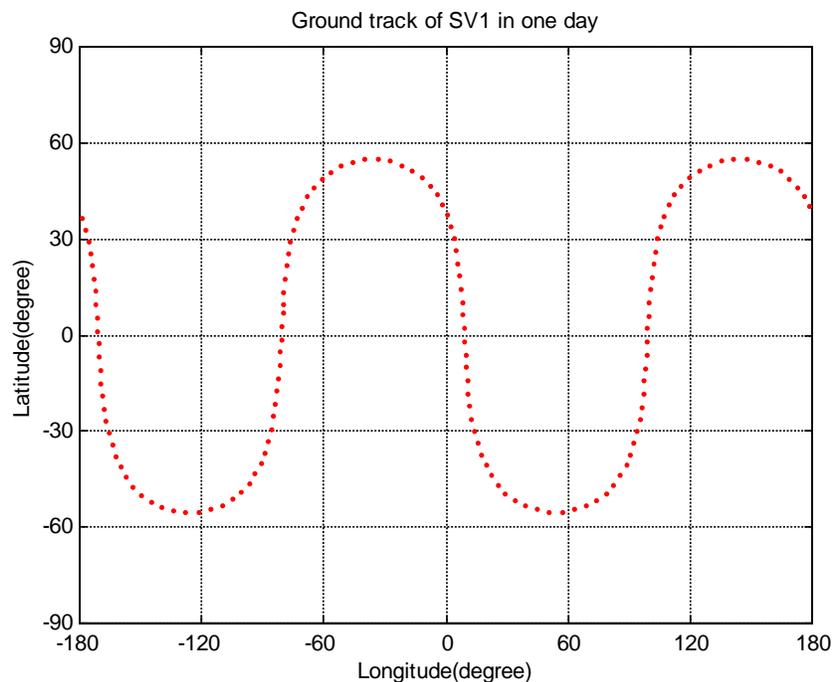
Figure 5.2 Ground track of a GPS satellite in one sidereal day
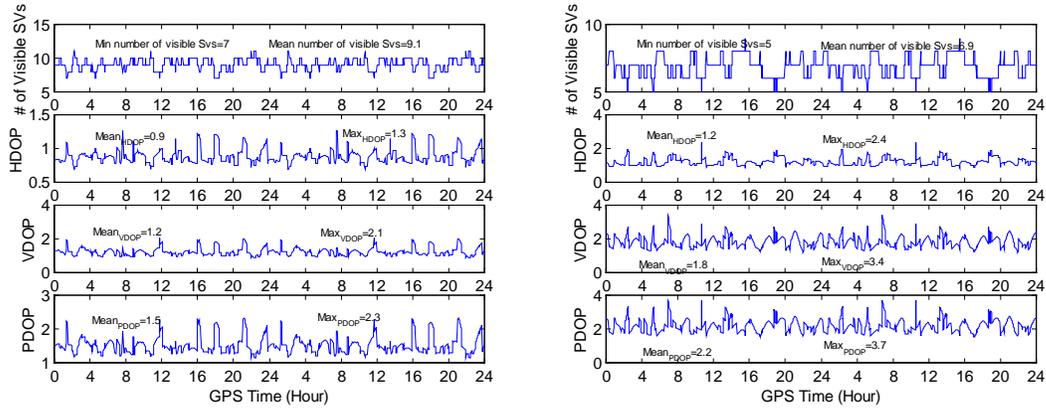
Figure 5.3 Number and DOP of satellites (GPS-24) observed at 51°04' N, 114°07' W (cutoff angle: 0º vs. 10º)
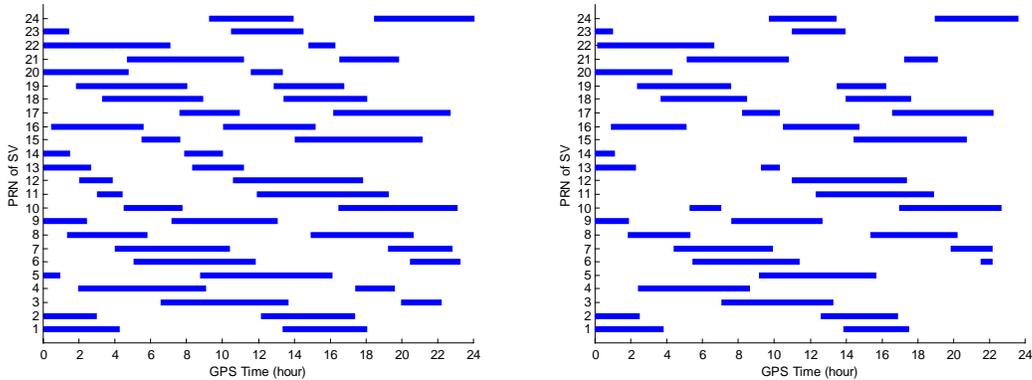
Figure 5.4 Visibility of satellites (GPS-24) observed at 51°04' N, 114°07' W (cutoff angle: 0º vs. 10º)

5.1.3 Enhanced GPS Constellation

Presently, there are 28 operational GPS satellites in space. Therefore, the real coverage of GPS is better than the 24-satellite constellation. According to some suggestions of GPS modernization, a six-plane 30-satellite constellation with the same semi-major axis and inclination as the current GPS constellation is suggested, which requires the smallest modification of current GPS constellation when compared to other schemes (Massatt and Zeitzew, 1998). Herein, this scheme is also simulated in our research to investigate the impact of satellite geometry and visibility on the performance of MultiKin. The selection

of right ascension and mean anomaly is listed in Table 5.2 (Massatt and Zeitzew, 1998), The relative position of GPS satellites at the reference epoch is shown in Figure 5.5.

Table 5.2 Parameters of a six-plane GPS 30-satellite constellation

Slot	Mean Anomaly (degree)	Right Ascension (degree)	Slot	Mean Anomaly (degree)	Right Ascension (degree)
A3	20.15	272.85	D1	138.63	92.85
A4	44.88	272.85	D4	167.88	92.85
A2	161.88	272.85	D2	255.93	92.85
A1	270.63	272.85	D3	37.33	92.85
A5	134.35	272.85	D5	283.68	92.85
B1	71.93	332.85	E1	193.93	152.85
B2	182.53	332.85	E2	307.43	152.85
B4	214.38	332.85	E4	331.81	152.85
B3	318.43	332.85	E3	51.48	152.85
B5	294.05	332.85	E5	83.33	152.85
C1	97.98	32.85	F1	245.71	212.85
C4	228.53	32.85	F2	355.23	212.85
C3	342.18	32.85	F3	103.98	212.85
C2	9.93	32.85	F4	131.41	212.85
C5	127.23	32.85	F5	220.98	212.85
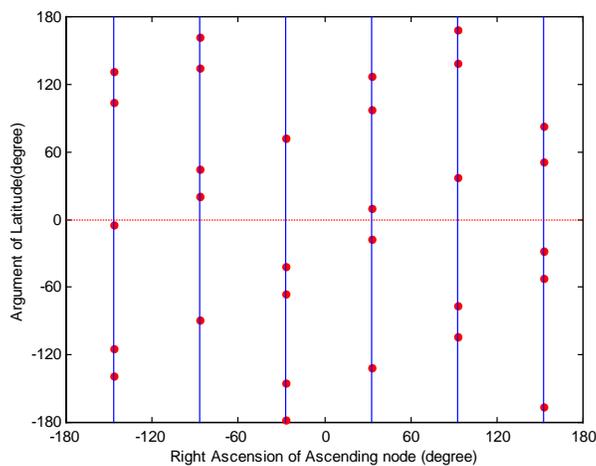
Figure 5.5 Six-plane 30-satellite constellation
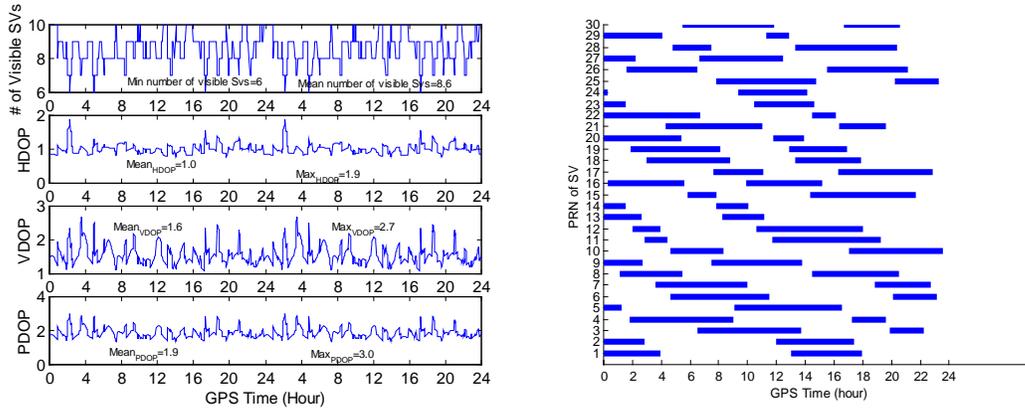
Figure 5.6 Geometry and visibility of satellites (GPS-30) observed at 51° 04' N, 114° 07' W (cutoff angle=10º)

Compared with the GPS-24 constellation (Figure 5.3, Figure 5.4) the satellite geometry and visibility have largely been improved in the GPS-30 constellation, see Figure 5.6. Table 5.3 gives a summary of the comparison.

Table 5.3 Comparison of GPS-24 and GPS-30, observed at 51° 04' N, 114° 07' W, cutoff angle=10º, observation time period: one GPS day

		GPS-24	GPS-30
DOP (mean)	HDOP	1.2	1.0
	VDOP	1.8	1.6
	PDOP	2.2	1.9
Number of Visible Satellites	Mean	6.9	8.6
	Minimum	5	6

5.2   TRAJECTORY SIMULATOR

Figure 5.7 shows the trajectory of a moving platform. The vehicle moves clockwise along the track and the velocity of each platform is a random walk process with the mean value of 5.0 m/s.
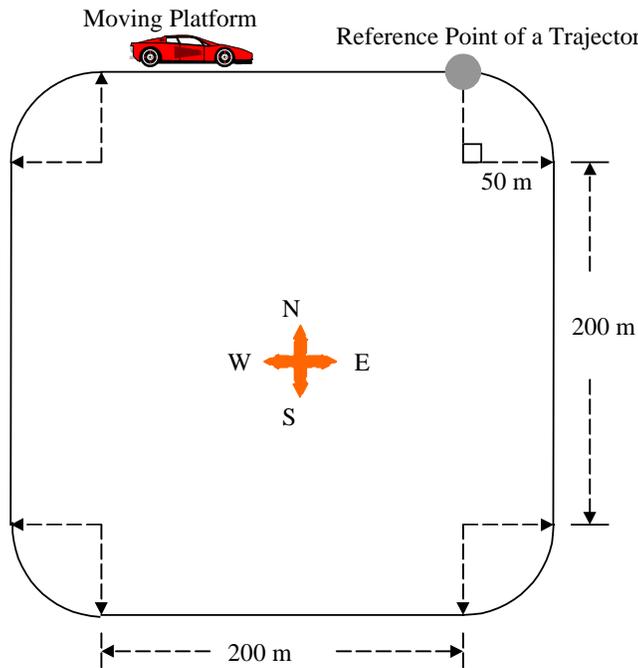
Figure 5.7 Trajectory of a moving platform

Since the trajectory is closed and the inter-platform distance (1.5 km~35 km) is much larger than the radius of the trajectory (150 m), the inter-platform distance can be approximated as unchanged. With this configuration, the impact of baseline lengths on MultiKin can be separately studied. Furthermore, the processing procedure is also simplified because Delaunay triangulation does not need to be refreshed during testing.

To test the impact of the number of moving platforms on the effectiveness of MultiKin, up to 10 platforms are configured in simulation. Figure 5.8 gives the position of those 10 moving platforms. The position of the central platform is (51° 04' N, 114° 07' W, 1100 m). The baseline lengths vary in different scenarios, which allows users to study the effect of spatially correlated differential errors.
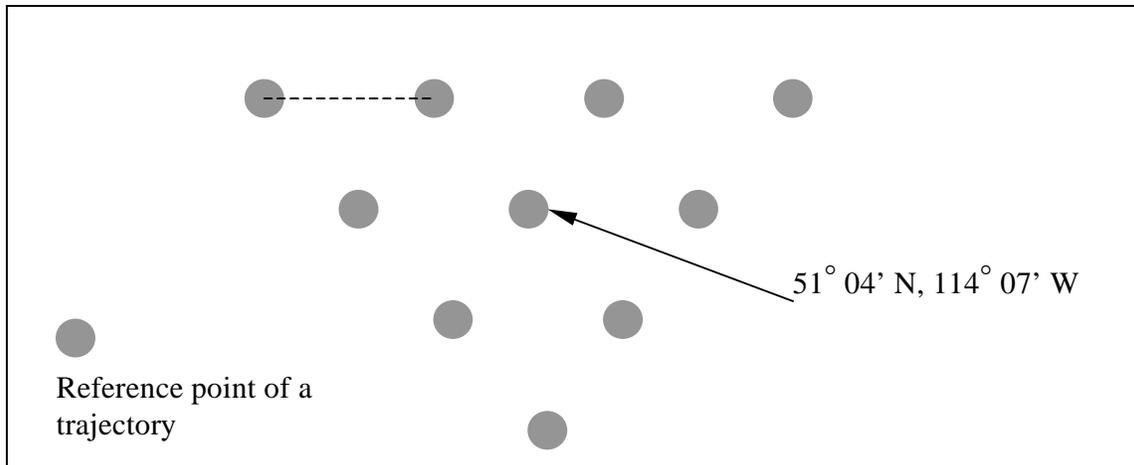
Figure 5.8 Position of moving platforms

5.3 OBSERVATION SIMULATION USING GPS SOFTWARE SIMULATOR

To generate necessary observations for a moving platform, the GPS simulator must be properly initialized. This task is accomplished by setting up the following parameters:

[1] Trajectory Generator: type of trajectory, vehicle dynamic, simulation area, etc.

[2] Constellation Generator: parameters of constellation, rejected satellites, cutoff elevation angle, etc.

[3] Error Simulator:

Ionospheric error: GIMs, variance of TEC increment, etc.

Tropospheric error: simulation area, meteorological data

Orbital error: coefficients of AR model, magnitude of SA (ε-error), channel with SA

SA: coefficients of AR model, on/off switch

Multipath: magnitude and model type (kinematic or static)

Receiver noise: magnitude

After initialization, the GPS simulator outputs observations using procedures as follows:

[1] Generate position and velocity of a GPS platform.

[2] Generate position (wrong) of visible satellites using constellation generator and user position.

[3] Generate true satellite position using orbital error generator.

[4] Generate other GPS errors.

[5] Generate pseudorange measurements for both code and carrier phase, and then add ambiguity terms to carrier phase measurements

The GPS software simulator also has the function of performing some statistical tests of generated GPS errors. This function is used to verify whether the parameters of the GPS software simulator are properly set up in the initialization step. The statistical mean and variance of absolute and double difference GPS error are compared with the setup values. If they do not match, the simulating procedure has to be restarted and some parameters may need adjustment.

It is worth noting that the software simulator does not refresh the ephemeris every two hours because the simulated orbital parameters are time-invariant. The constellation parameters defined in Table 5.2 are put in an ephemeris file. GPS data processing software, such as FLYKIN™, obtains the orbital parameters from this file and computes the apparent position of satellites, using the method described in section 5.1.1.

5.4   DESIGN OF SCENARIOS FOR SIMULATION

The following parameters are set up during the initialization of the GPS simulator for testing MultiKin.

Testing Period: For all scenarios, the testing period is from 50930 s to 55730 s (GPS seconds in week), which is Calgary local time from 7:10 am to 8:30 am. The GPS data rate is one Hz. The reason for choosing this testing period is that the satellite visibility

and geometry are very good and stable in this period. This allows us to study the effect of satellite geometry and visibility separately, because setting up poorer satellite geometry can be performed by simply rejecting some satellite observations without changing the testing period. A change of testing period could also result in changes of satellite visibility and geometry; however, atmospheric errors could change accordingly.

Baseline length: To separately study the effect of baseline length (i.e., magnitude of differential errors), all baseline lengths in one scenario are almost the same. However, they may differ from one scenario to another. There are five different baseline lengths configured in the simulation: 1.5 km, 5 km, 10 km, 20 km, and 35 km.

Satellite Geometry: There are three types of satellite geometry configured in the testing scenarios. They differ from each other by the constellation and rejection of observations. The cutoff elevation angle for all configurations is 10°. Table 5.4 gives the parameters of these configurations.

Table 5.4 Configuration of satellite geometry in simulation tests

	Constellation	Rejected satellites
Reduced Geometry	24-satellite	[2 15 16 19]
Full Geometry	24-satellite	None
Enhanced Geometry	30-satellite	None

Comparisons of satellite visibility and geometry of different scenarios are presented in Figure 5.9 and Figure 5.10. Three low-elevation and one mid-elevation satellites are rejected in the reduced geometry, which causes the mean PDOP to increase from 2.0 to 2.9. Meanwhile, the number of visible satellites is also reduced from eight to five. For the enhanced geometry, two new satellites are introduced, which improves both the satellite geometry and visibility. It should be noticed that the VDOP of the enhanced geometry is poorer than that of the full geometry from 52500 s to 54000 s (GPS time). This is because

the orbital parameters of the first 24 satellites in the enhanced constellation are slightly different from those in the standard 24-satellite constellation. However, statistically, the enhanced geometry is still better than the full geometry during the testing period, especially for the HDOP and visibility. The impact of these geometry changes is investigated in Chapter 6.
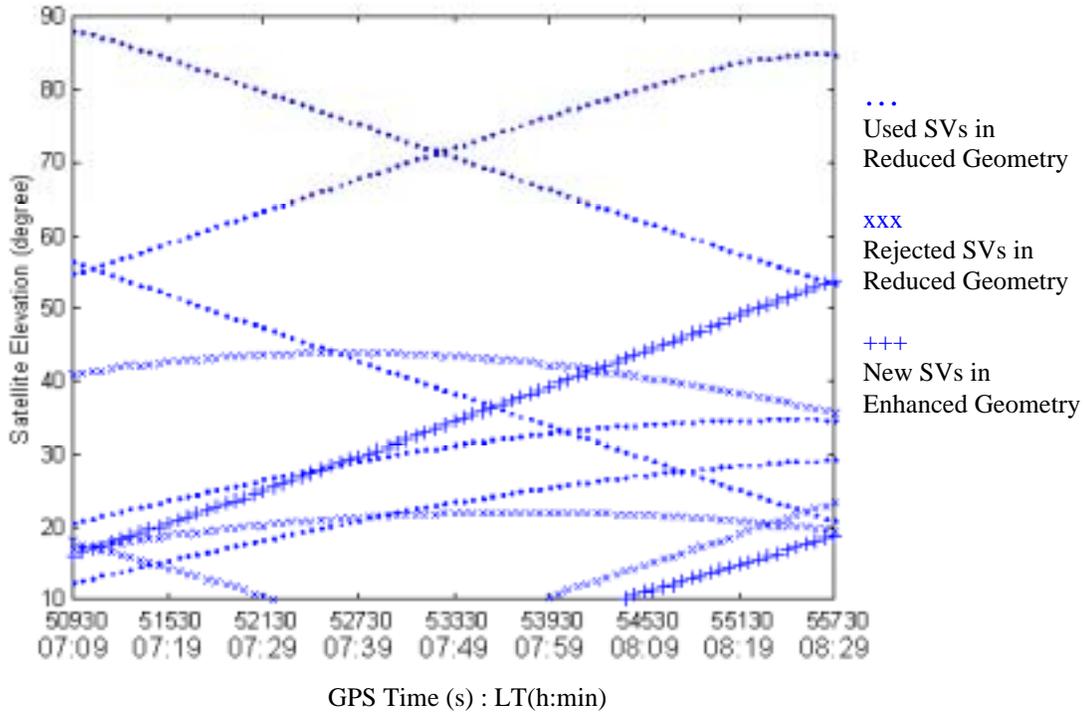
Figure 5.9 Satellite elevation and visibility in different scenarios
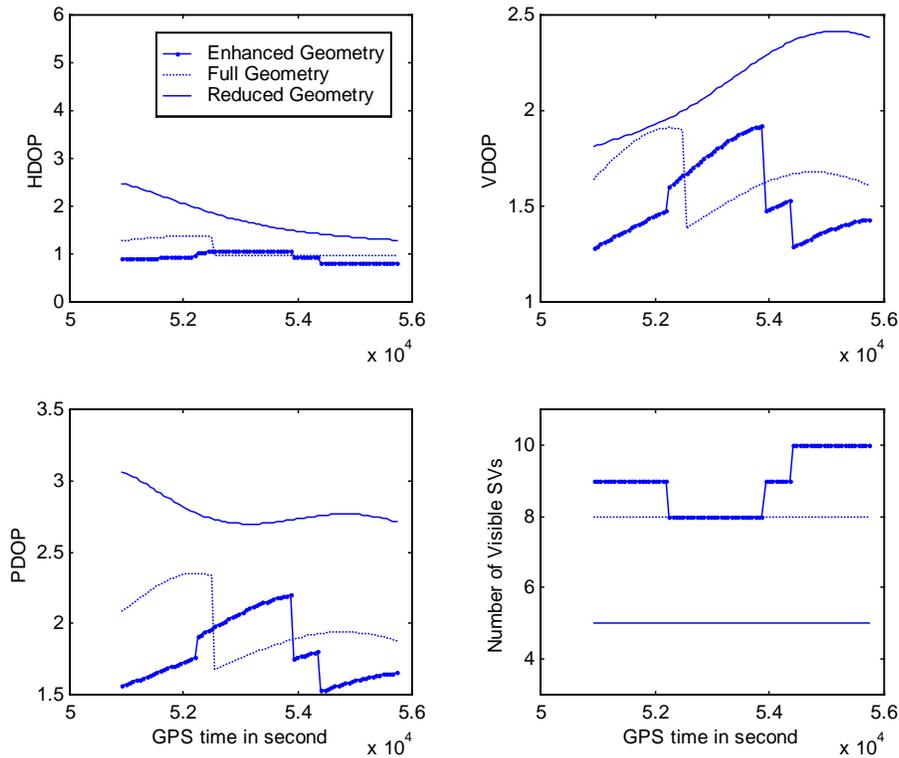
Figure 5.10 Visibility and geometry of different satellite configurations

The Error Magnitude: The error magnitude is one of the most important parameters affecting the performance of a GPS system. To fully test MultiKin, several levels of errors are set up. All the errors given below are in terms of RMS. The magnitudes of receiver noise, multipath and SA are given for single range measurements, while the atmospheric and orbital errors are double differenced quantities.

Receiver Noise: In the simulation tests, the receiver noise for C/A code is one metre. The carrier phase noise in L1 and L2 are set up to 0.01 cycle. These are typical values for commonly used GPS receivers.

Multipath: Multipath is the dominant error in differential GPS systems, when the baseline is short, or when other differential errors are reduced by modeling or by phase combination. Three levels of multipath are tested: small (1.0 mm), medium (3.8 mm), and large (1.0 cm). These are values for the L1 carrier phase. The magnitudes of the L2 carrier multipath and the C/A code multipath are accordingly set up by multipath models.

Generally, the L2 carrier phase multipath has the same magnitude (in cycle) as the L1 carrier phase multipath, while the C/A code multipath is 100 times larger than the L1 carrier phase multipath (in length).

Ionospheric Error: Three different levels of ionospheric conditions are tested, according to what is shown in Table 5.5.

Table 5.5 Setup of ionospheric error

Strength of ionosphere activity	Relative double differenced error
Quiet	1.5~2.5 ppm
Active	3.5~5.5 ppm
Strong	> 8.0 ppm

The absolute double difference range errors vary with baseline lengths.

Tropospheric Error: Two different scenarios are designed to simulate the typical and strong tropospheric errors. In the typical scenario, the relative tropospheric error is about 2.3 ppm, while for the strong scenario, it is 4.8 ppm, which assumes that the local temperature is very high (45 °C) and the relative humidity varies largely both in space and time.

Orbital Error: The magnitudes of along-track, cross-track and radial errors are set up to 3.97 m, 2.50 m, and 0.73 m respectively, which are derived from the statistical results of the actual orbital errors. The impact of large orbital biases is also considered in the simulation tests. In some scenarios, a 50 m bias is added in the along-track error for some satellites to simulate blunders in ephemeris parameters (mean anomaly).

SA: In most of the scenarios, the clock dithering (δ-error) is switched on for all the satellites and it is independent from one channel to another. The variance of clock dithering is 25.6 m. In some accuracy tests, SA is switched off to reduce the positioning error of the moving reference.

6 RESULTS AND ANALYSIS OF MULTIKIN

6.1 PARAMETERS OF PERFORMANCE

To evaluate MutiKin under different scenarios, several parameters are defined for evaluating the test performance.

TTAF (Time to True Ambiguities Fixed): Due to large measurement errors or poor satellite visibility, ambiguities can be fixed to incorrect values. Although FLYKIN™ and MultiKin can detect and correct such wrong fixes using a ratio test and ambiguity constraints, these wrong fixes will lengthen the time of correct resolution. Furthermore, the wrong fixes seriously degrade positioning accuracy. Unless ambiguities are correctly fixed, centimetre-level accuracy cannot be obtained. Therefore, *TTAF* is used as a parameter to show how long it takes to correctly fix ambiguities and to assure the positioning accuracy at the centimetre level.

TAF (Time to Ambiguities Fixed): In field tests, the correctness of ambiguities cannot be guaranteed, even when baselines are very short. Therefore, only the time to fix ambiguities can be measured. However, in the field tests, some reference stations with precisely known coordinates are set up to provide more redundancy and more constraints to check the correctness of the ambiguity resolution of a moving baseline. Therefore, *TAF* can be approximated as *TTAF*.

TSR (Time Saving Rate): This parameter represents the improvement of time to fix integer ambiguities induced by MultiKin. *TSR* is defined as

$$TSR = \frac{\overline{TTAF_i(Unconstrained)} - \overline{TTAF(MultiKin)}}{\overline{TTAF_i(Unconstrained)}} \tag{6.1}$$

The unconstrained method is also called the single baseline method. The higher the *TSR*, the more efficient MultiKin is. In the field tests, *TSR* is computed using *TAF*.

TDW (Time to Detect Wrong fixes): When the noise level becomes higher, the probability of incorrectly fixed ambiguities increases. Generally, wrong fixes will

generate a large bias in the position domain; thus, positioning accuracy is degraded. The ambiguity monitoring module (ratio test and constraint) is designed to detect wrong fixes. It is always expected that wrong fixes can be detected as soon as possible. The parameter used to evaluate the capability of the ambiguity monitoring module to detect wrong fixes is defined as *TDW*. It is actually a reliability indicator of the ambiguity resolution process.

NUB (Number of Unfixed Baselines): With the increase of baseline lengths, the differential errors increase. When the errors are larger than a certain level, it is difficult or even impossible to fix the integer ambiguities. Furthermore, in practice, due to the requirement of high positioning accuracy in real-time kinematic applications, only when the ambiguities can be fixed within a certain period, does the integer solution benefit the positioning accuracy. Therefore, in the simulation tests, a time limitation of ambiguity resolution is set up as 1200 s. If ambiguities cannot be fixed within this period, the baseline is an unfixed baseline. The total number of the unfixed baselines during a simulation test is called *NUB*.

LSB (Longest Solvable Baseline): Generally, the longer the baseline, the more difficult it is to fix ambiguities. When baselines are long enough to cause the *NUB* to reach a certain threshold, the baseline is defined as the *LSB*. It indicates the limitation of the functional distance for the applied ambiguity resolution method. Herein, the *NUB* threshold is defined as 10% of the total number of the tested baselines.

6.2 SIMULATION TESTS: DESIGN, RESULTS AND ANALYSIS

The simulation tests of MultiKin can be categorized into three different types: efficiency tests, reliability tests, and accuracy tests. In the last section of this chapter, some field tests will also be presented to show the performance of MultiKin in the case of real GPS scenarios.

6.2.1 Efficiency Tests

The efficiency and reliability tests consist of multiple trials. The first trial starts at 50930 s (GPS time). The following trials start at 15 s after the beginning of the previous trial.

Each trial lasts for at most 1200 s. If the true ambiguity can be fixed within 1200 s, the trial is treated as a successful trial, the *TTAF* is recorded and a new trial is started. Meanwhile, if any wrong fixes occur during this period, the *TDW* is also recorded. If the true ambiguity cannot be fixed within 1200 s, the trial of ambiguity fixing fails and no *TTAF* and *TDW* is recorded. The whole testing period is 4800 s; 1200 s are reserved for the final trial. Thus, the effective testing period is 3600 s, which can contain 240 overlapping simulation trials.

6.2.1.1 Test 1: Impact of Number of Platforms

Objective: This test is to evaluate the effect of the number of platforms on the effectiveness of MultiKin, i.e., whether increasing the number of moving platforms can further improve the efficiency of MultiKin.

Design of scenarios: The testing scenarios are listed in Table 6.1. The baseline lengths vary with scenarios. Each scenario was tested using three, four and 10 platforms. The three-platform constraint contains three baselines and one triangle. The four-platform constraint contains five baselines and two triangles. The 10-platform constraint contains 18 baseline and nine triangles. The positions of the selected moving platforms are shown in Figure 6.1.

Table 6.1 Simulated scenarios for Test 1

Scenario	Rx (single)	Mp (single)	DD Ion	DD Trop	DD Orbit	SA δ-error	Geometry	Observable	Total DD error	Length of Baseline
1	0.01 cycle/ 1.9 mm	0.02 cycle/ 3.8 mm	0.014 cycle/ 1.8 ppm	0.018 cycle/ 2.3 ppm	0.00046 cycle	On 25 m	PDOP= 2.0 Full geometry	L1	0.047 cycle/ 8.9 mm	1.5 km
2			0.06 cycle/ 2.3 ppm	0.061 cycle/ 2.3 ppm	0.0012 cycle			L1	0.078 cycle/ 1.5 cm	5 km
3			0.094 cycle/ 1.8 ppm	0.12 cycle/ 2.3 ppm	0.0036 cycle			WL	0.08 cycle/ 6.9 cm	10 km
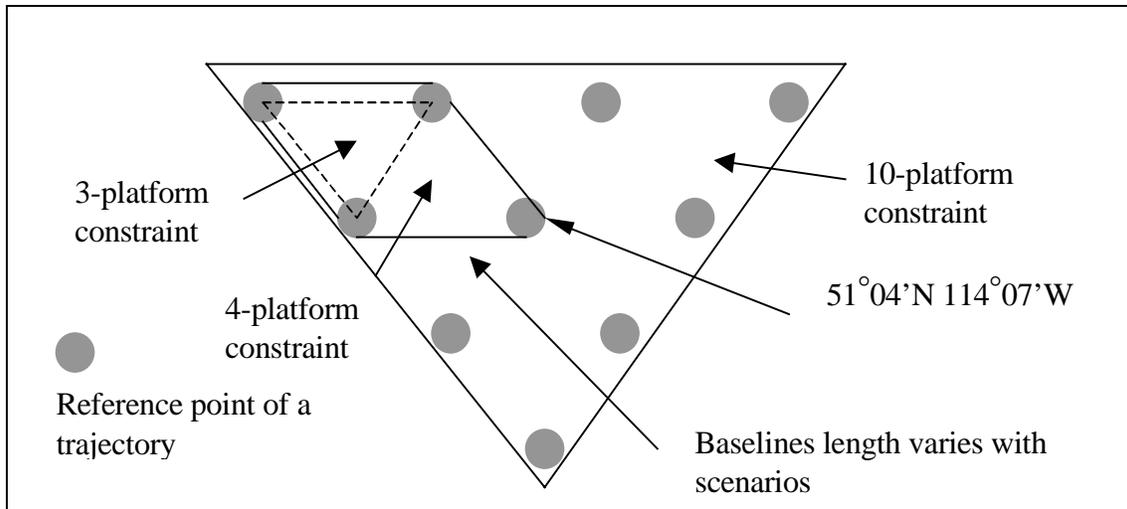
Figure 6.1 Selection of moving platforms in Test 1

Testing results in Figure 6.2 show that with the number of constrained platforms increased, the average time to fix ambiguities keeps decreasing. Therefore, increasing the number of platforms can further improve the efficiency of MultiKin. Moreover, by adding more platforms into the configuration, the time to fix ambiguities for a specific baseline is usually reduced, see Figure 6.3. Only a few exceptions (three out of 240) can be observed where the 10-platform constraint has a longer time to fix ambiguities than the four-platform constraint does. This is because the wrong fixes of the neighboring triangles result in the fixed solution of baseline 1 being reset by constraints. However, statistically, the 10-platform constraint is still slightly better than the four or three-platform constraint.

Testing results also show that although the efficiency increases by increasing the number of constrained platforms, the improvement rate diminishes (See Figure 6.4). The four-platform constraint almost has the same *TSR* as the 10-platform constraint in all scenarios. This effect can be explained by Figure 6.5.
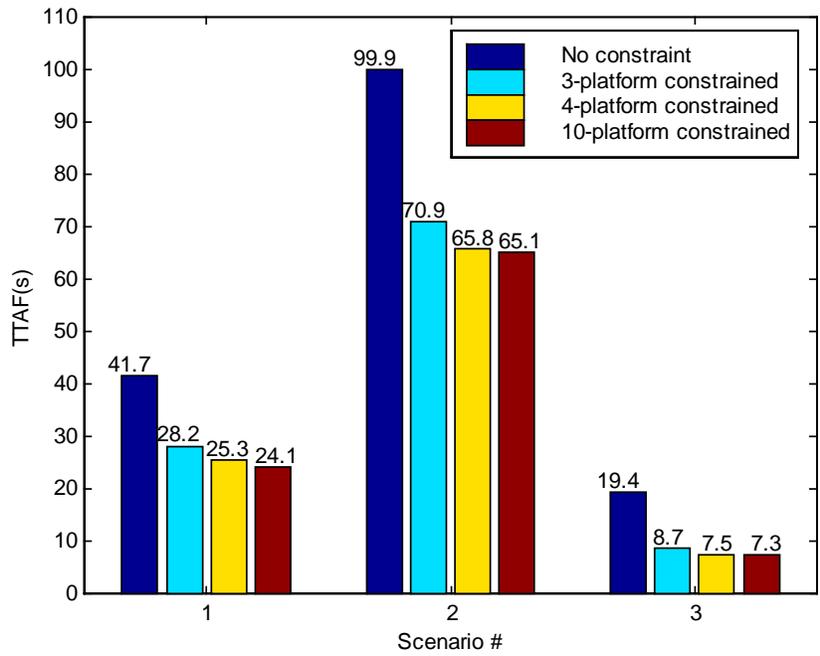
Figure 6.2 Time to true ambiguity fixed for Test 1
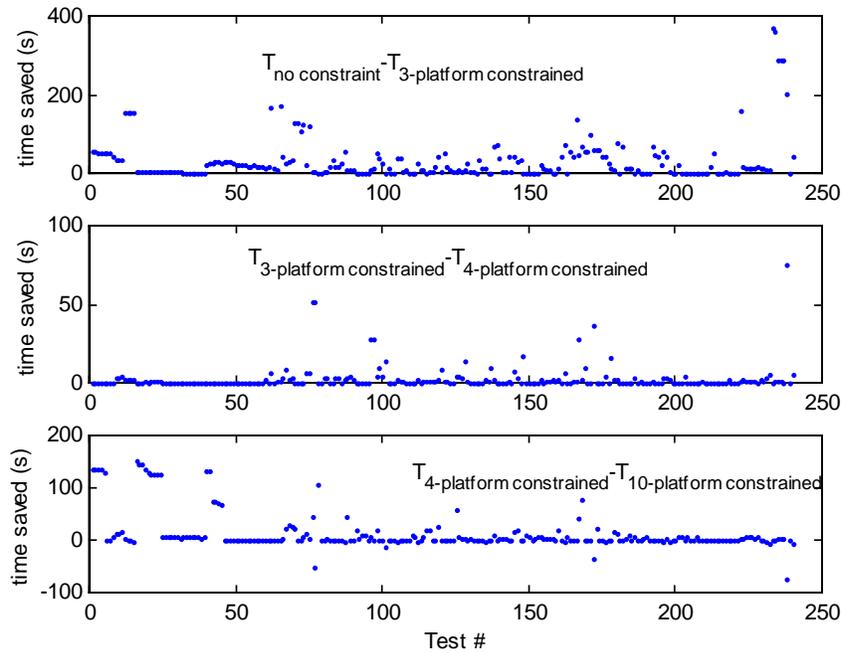
Figure 6.3 Increased improvement of *TTAF* for baseline 1 in Test 1
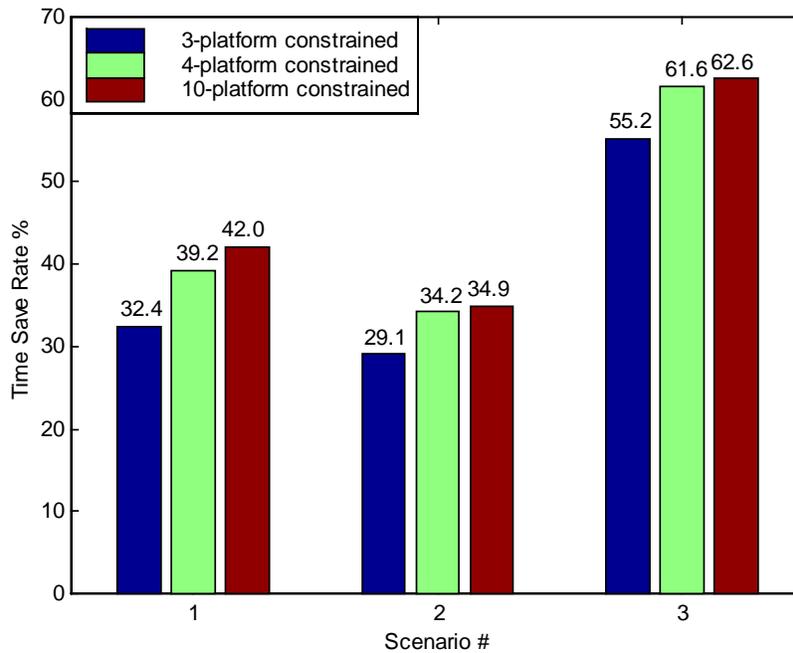
Figure 6.4 Efficiency comparison of multi-platform constraints
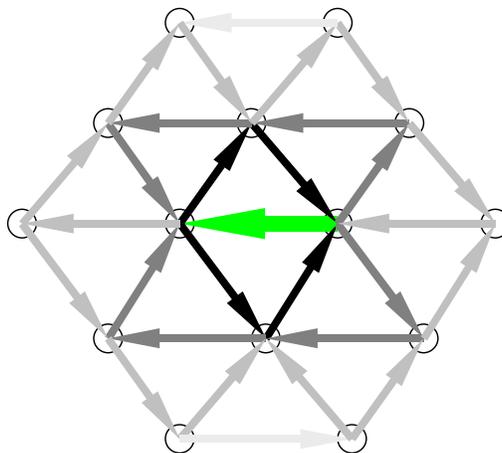
Figure 6.5 Interrelation of baselines on ambiguity resolution

The solution of a baseline is directly related to those baselines within the same triangles, while the baseline which is not in the same triangle can only have an indirect effect. The farther the baseline, the less it can help. In Figure 6.5, the baseline with the darkest color has the strongest effect on the ambiguity resolution of the bold baseline (in the centre).

From Figure 6.4, it can be concluded that the major efficiency improvement is generated by the direct triangular constraint, while those indirect constraints can only result in a minor improvement. Based on this conclusion, for all the following tests, a six-platform constraint was used, which is chosen as the best compromise between the four-platform and 10-platform constraint. This small configuration can simplify the data processing and give a good estimation of the maximum efficiency improvement generated by MultiKin. The six-platform constraints are configured as shown in Figure 6.6, where nine baselines and four triangles are included.

$51°04'N\ 114°07'W$

Reference point of a trajectory
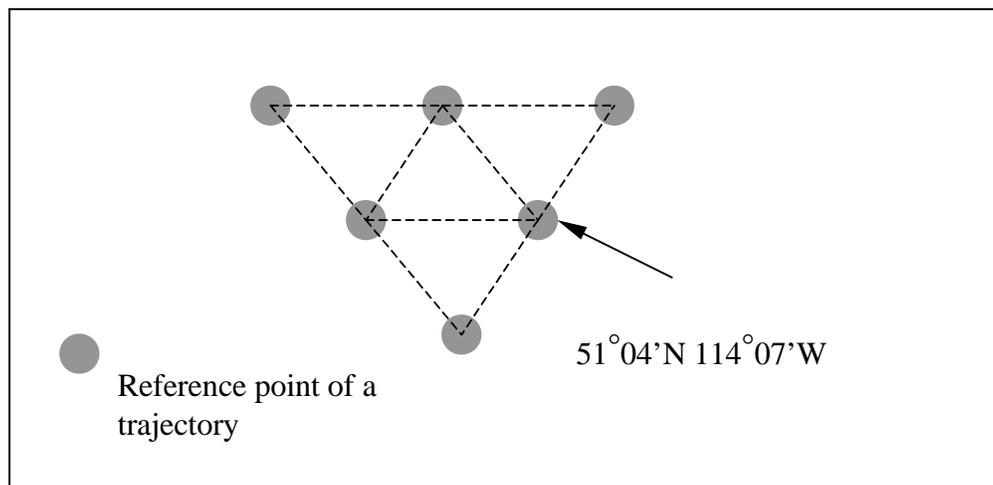
Figure 6.6 Configuration of six-platform constraint

GPS errors in the double differenced measurements have the largest impact on ambiguity resolution. The following tests (Test 2 to 7) were conducted to test the impact of different errors on the efficiency of MultiKin.

6.2.1.2   Test 2: Impact of Multipath

Objective: To test the effect of multipath on the efficiency of MultiKin.

Design of scenarios: The testing conditions of Test 2 are listed in Table 6.2. To separately study the effect of multipath, baseline lengths are the same in all simulated scenarios, i.e., the spatially correlated errors have the same magnitudes. Thus, the possible efficiency change of MultiKin under different scenarios should be only caused

by the change of multipath. Since multipath is the dominant error for short-baseline resolution, the baseline length is set up to 1.5 km. In addition, multipath behaves quite differently between the static and kinematic applications. Both multipath models discussed in Chapter 4 are tested.

Table 6.2 Simulated scenarios for Test 2

Scenario	Rx (single)	MP (single)	DD Ion	DD Trop	DD Orbit	SA δ-error	Geometry	Observable	Total DD error	Length of Baseline
1	0.01 cycle/ 1.9mm	0.005 cycle/ 1mm[1]	0.014 cycle/ 1.8 ppm	0.018 cycle/ 2.3 ppm	0.00036 cycle / 0.05 ppm	On 25 m	PDOP= 2.0	L1	0.027 cycle/ 5.13 mm	1.5 km
2		0.02 cycle/ 3.8mm[1]					Full geometry		0.047 cycle/ 8.93 mm	
3		0.02 cycle/ 3.8mm[2]							0.047 cycle/ 8.93 mm	
4		0.05 cycle/ 1cm[1]							0.1 cycle/ 1.9 cm	

[1] Kinematic model, [2] Static model

Figure 6.7 and Figure 6.8 show the results of Test 2. It is obvious that the *TTAF* increases with increasing multipath. However, Test 2 also shows two interesting results:

[1] The efficiency of MultiKin decreases with increasing multipath,

[2] The static multipath results in a longer *TTAF* and a lower efficiency of MultiKin than the kinematic multipath of the same magnitude.

For the first point, the ambiguity resolution in MultiKin still depends on the single baseline resolution. The ambiguity constraints can only speed up ambiguity resolution when the sufficiency test in MultiKin is passed (see section 3.4.2). When the error magnitude is increased, the output ratio from the distinguishing test will become smaller; thus, it will take a relatively longer time to pass the sufficiency test. An example of fixing ambiguities in the case of extremely large errors can help to understand the testing results. When the GPS errors are very large, the correct ambiguities cannot be fixed even using constraints. Therefore, no improvement can be obtained by MultiKin. More tests

were conducted for other GPS errors to check whether the improvement of MultiKin is reduced by increasing GPS errors.
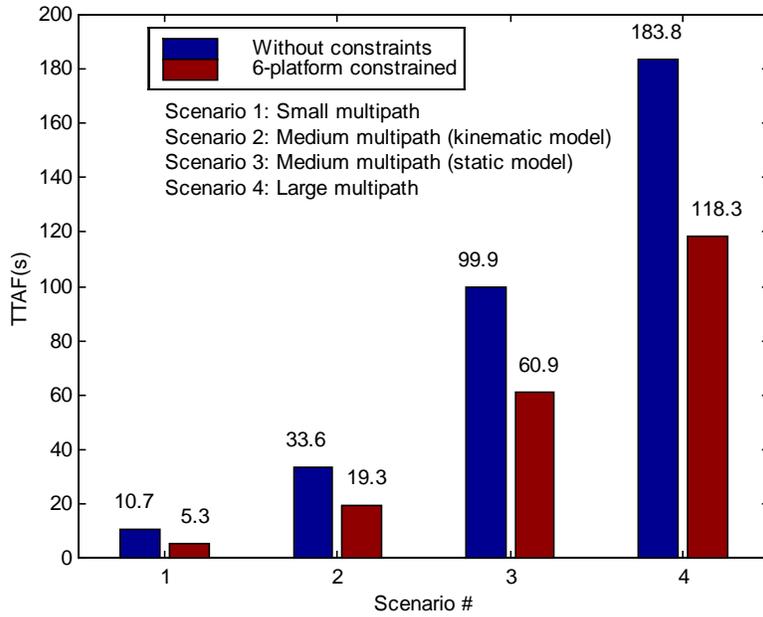
Figure 6.7 Time to true ambiguity fixed for Test 2
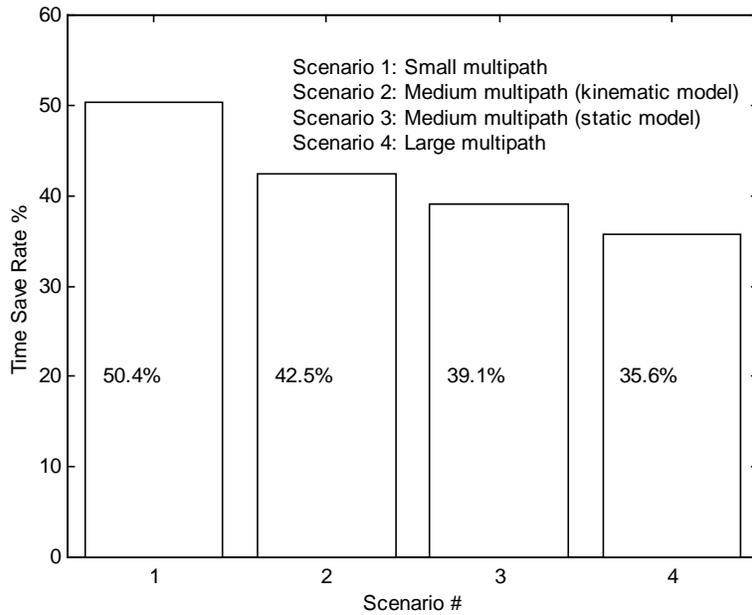
Figure 6.8 Comparison of TSR for Test 2

The second point can be explained by the difference of the statistical properties between the kinematic multipath and the static multipath. The kinematic multipath is simulated by a Gauss-Markov process with a time constant of a few seconds. The static multipath changes with the satellite geometry slowly and regularly. Since the kinematic multipath is less self-correlated than the static multipath, the Kalman filter can work more effectively in the case of the kinematic multipath. It is worth mentioning that for the fourth scenario, there are eight trials failing to fix ambiguities within 1200 s when no constraint is applied. However, after using the six-platform constraint, all ambiguities can be fixed within 755 s. Therefore the decreased *NUB* also shows the efficiency improvement of MultiKin.

6.2.1.3   Test 3: Impact of Ionospheric Error

Objective: To test the effect of the ionospheric error on the efficiency of MultiKin.

Design of scenarios: Ionospheric error is one of the largest error sources in relative positioning. When the baseline is long or the solar activity is strong, the residual ionospheric error can be large enough to cause difficulty in ambiguity resolution. The following scenarios (see Table 6.3) have the same baseline length; other spatially correlated errors are kept at the same level in different scenarios. Thus, the effect of the ionospheric error can be separately studied. Three scenarios are generated to simulate quiet, active and strong ionospheric activities. Because the L1 ambiguities cannot be fixed in the case of strong ionospheric activity, the widelane observables are used for all the tests.

Table 6.3 Simulated scenarios for Test 3

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error (WL)	Length of Baseline
1	0.01 cycle/ 1.9mm	0.02 cycle/ 3.8mm	0.094 cycle/ 1.8ppm	0.12 cycle/ 2.3ppm	0.0036 cycle/ 0.07 ppm	On 25m	PDOP= 2.0 Full geometry	WL	0.078 cycle/ 6.7cm	10km
2			0.23 cycle/ 4.4ppm						0.1 cycle/ 8.6cm	
3			0.45 cycle/ 8.6ppm						0.13 cycle/ 11.2cm	

The result of Test 3 is analogous to Test 2. As shown in Figure 6.9 and Figure 6.10, the *TTAF* increases and *TSR* decreases with increasing the ionospheric errors. When the ionosphere is quiet, a very large improvement of efficiency (66.9%) can be achieved by using MultiKin. While the ionospheric activity is strong, the *TSR* decreases to 46.8%.
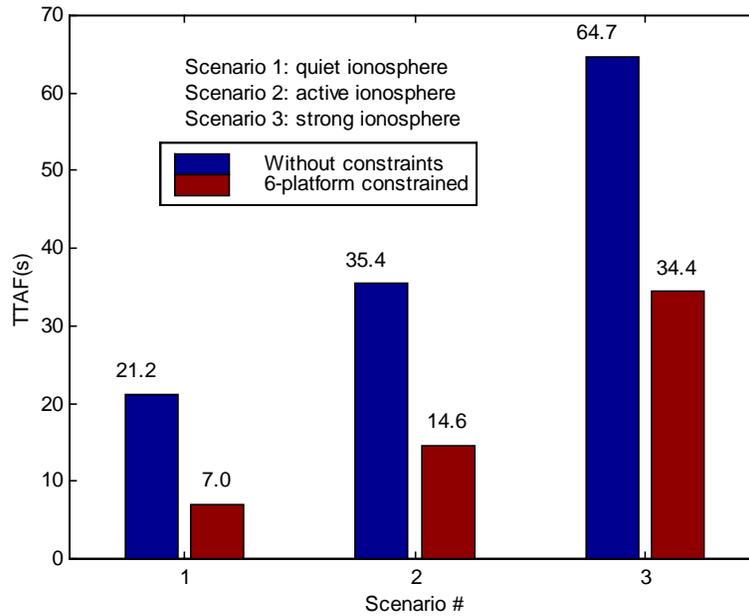
Figure 6.9 Time to true ambiguity fixed for Test 3
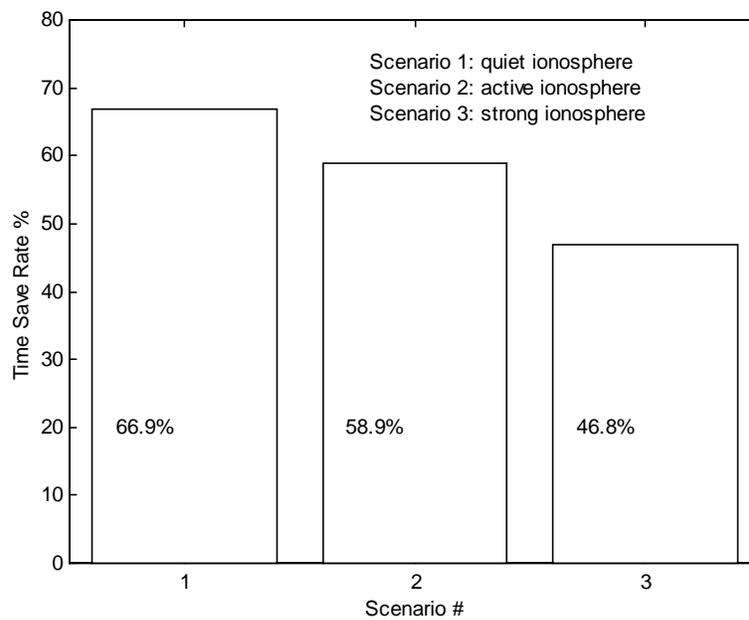
Figure 6.10 Comparison of TSR for Test 3

6.2.1.4 Test 4: Impact of Tropospheric Error

Objective: To test the effect of tropospheric errors on the efficiency of MultiKin.

Design of scenarios: Generally, tropospheric errors have less effect on relative positioning than ionospheric errors. However, in some extreme situations, the tropospheric errors can be very large. In the following tests, see Table 6.4, a scenario with very large tropospheric effect (4.8 ppm) is generated where the temperature is very high (45 °C) and relative humidity is largely varied. The baseline length is constant to keep other errors relatively unchanged.

Table 6.4 Simulated scenarios for Test 4

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error (L1)	Length of Baseline
1	0.01 cycle/ 1.9mm	0.02 cycle/ 3.8mm	0.11 cycle/ 2.1 ppm	0.12 cycle/ 2.3 ppm	0.0036 cycle/ 0.07 ppm	On 25m	PDOP= 2.0 Full geometry	L1	0.12 cycle/ 2.3 cm	10 km
2				0.25 cycle/ 4.8 ppm					0.24 cycle/ 4.6 cm	

The testing results (see Figure 6.11 and Figure 6.12) repeat the same conclusion reached in Tests 2 and 3. The *TSR* of MultiKin decreases from 44.4% to 34.2%, with tropospheric errors increased from 2.3 ppm to 4.8 ppm. For the unconstrained method, 287 baselines were not fixed within 20 minutes. These represent 13.3% of all 2160 tested baselines. For MultiKin, however, only 27 baselines were not fixed, which is ten times less than the result using the unconstrained method.

It should be noted that all the tested platforms have the similar height in simulations. If the height difference of the applied platforms is large, such as for the aeronautic applications, the impact of tropopsheric errors can be larger; thus, the efficiency improvement of MultiKin can be further reduced.
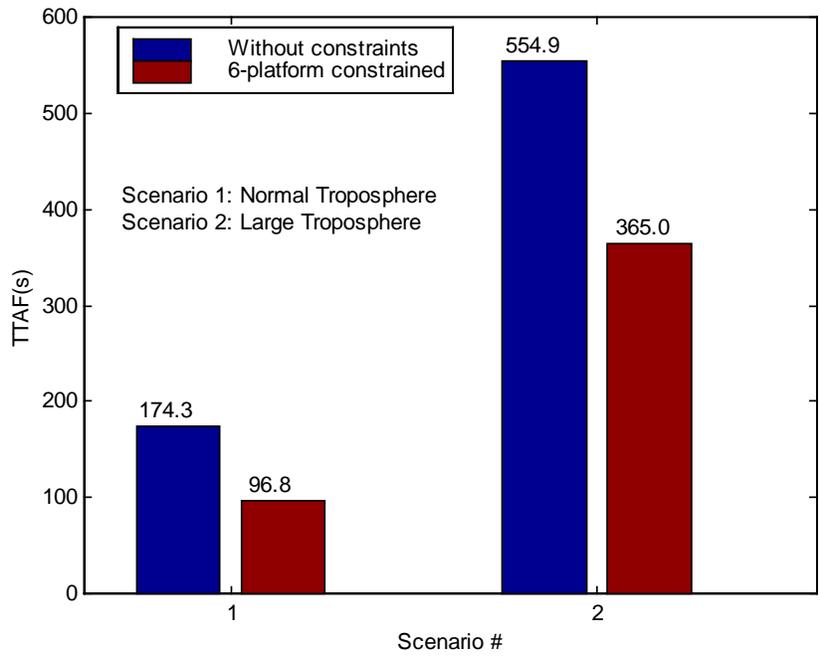
Figure 6.11 Time to true ambiguity fixed for Test 4
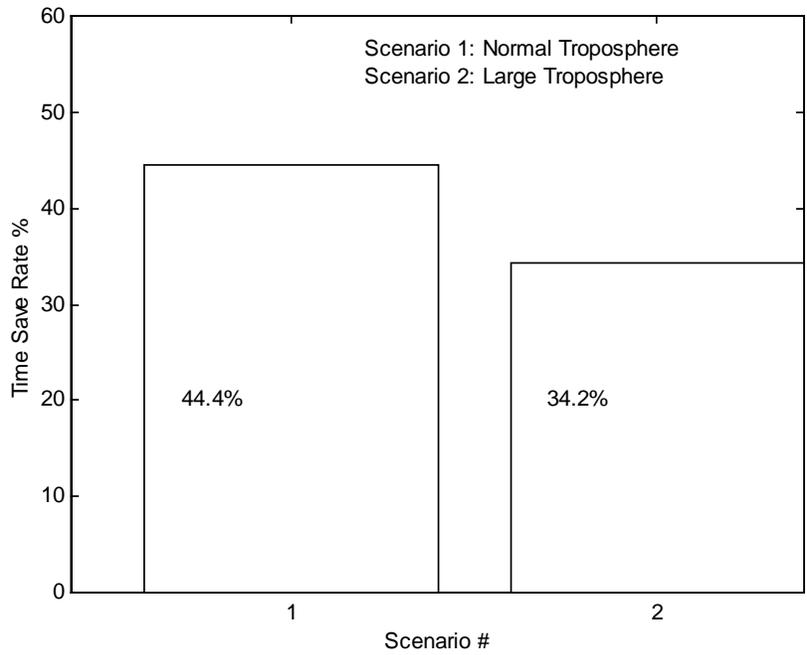
Figure 6.12 Comparison of TSR for Test 4

6.2.1.5 Test 5: Impact of Orbital Error (Large orbital bias)

Objective: To test the effect of large orbital biases on the efficiency of MultiKin.

Design of scenarios: Normal orbital errors have the least effect on relative positioning of short baselines when compared to other GPS errors. Since the RMS of the simulated orbital errors (normal) is less than 5.0 m, the maximum differential orbital error is less than 0.25 ppm. This has only a minor effect on ambiguity resolution when the baseline length is relatively short (10 km). However, if a large orbital bias (a few tens of metres) occurs, the effect of orbital error cannot be neglected. Although the orbital bias occurs at a very low frequency, it is still necessary to understand its effect of on the efficiency of MultiKin. This is because the large orbital bias can seriously degrade the effectiveness of positioning.

The following scenarios (see Table 6.5) are designed to investigate the effect of large orbital biases on MultiKin. The magnitudes of other GPS errors are kept constant in different scenarios. Since the large orbital biases seldom occur to all the GPS satellites, a 50 m bias was only added to several satellites in the along-track channel, which is equivalent to a 0.37 arcsecond bias of mean anomaly in ephemeris.

Table 6.5 Simulated scenarios for Test 5

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error (L1)	Length of Baseline
1	0.01 cycle/ 1.9 mm	0.02 cycle/ 3.8 mm	0.098 cycle/ 1.8 ppm	0.12 cycle/ 2.3 ppm	0.0036 cycle[1]	On 25 m	PDOP= 2.0 Full geometry	L1	0.12 cycle/ 2.3 cm	10 km
2					0.085 cycle[2]				0.12 cycle/ 2.3 cm	
3					0.03 cycle[3]				0.13 cycle/ 2.5 cm	
4					0.092 cycle[4]				0.15 cycle/ 3.0 cm	

[1]: Normal orbital error; [2]: Large bias on the base satellite; [3]: Large bias on a non-base satellite; [4]: Large bias on the base and one non-base satellite

Figure 6.13 and Figure 6.14 show the results of the simulation tests, which are similar to the above tests. The efficiency improvement rate decreases as errors increase. Both the unconstrained method and MultiKin show better performance when the large orbital biases are applied to a non-base satellite than to the base satellite. This is because when the large orbital biases are applied to a non-base satellite, they only affect the double difference measurements related to that satellite; whereas when the large orbital biases are applied to the base satellite, they affect all the double difference measurements. However, the difference between Scenarios 2 and 3 is not large. This is because FLYKIN™ adopts the strategy of "all fixed or all float" in ambiguity fixing. In other words, all the ambiguities must be fixed at the same time, since FLYKIN™ cannot fix some of the ambiguities to integers while keep others as float numbers. Therefore, even if only one double difference measurement is ruined by the large orbital errors, it still delays the fixing time of other ambiguities. The performance attenuation of Scenario 4 compared to Scenario 3 is larger than the performance attenuation of Scenario 3 compared to Scenario 2. That is because the orbital error is doubled in Scenario 4 when compared to Scenario 3.
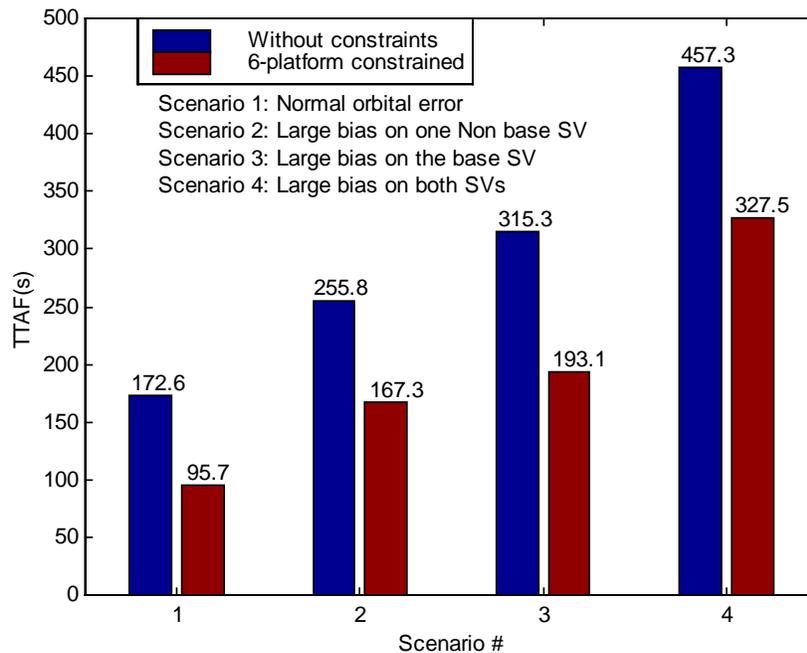
Figure 6.13 Time to true ambiguity fixed for Test 5
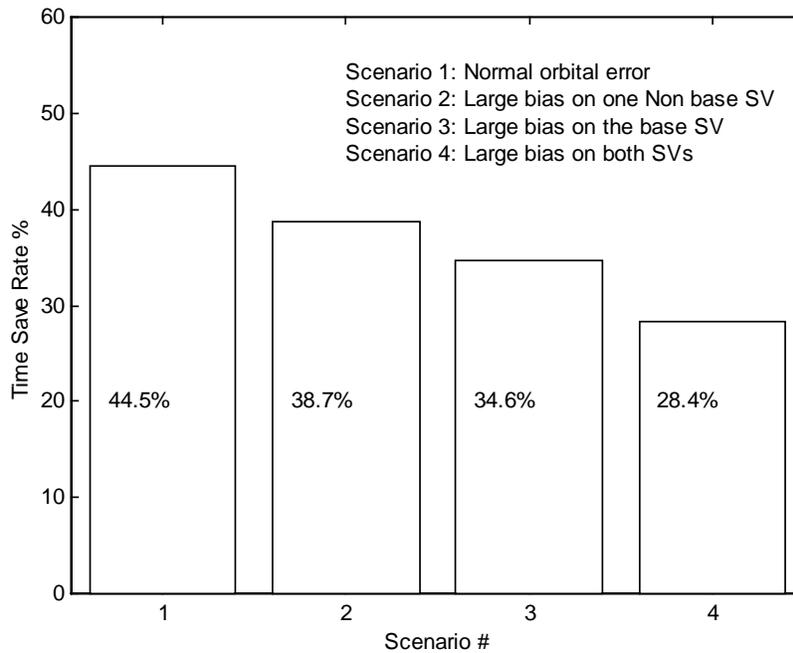
Figure 6.14 Comparison of TSR for Test 5

The comparisons of *TTAF* and *TSR* only show part of the efficiency improvement generated by MultiKin. Table 6.6 gives the comparison of *NUB* for the different scenarios. The improvement induced by MultiKin in the case of Scenario 4 is very significant. When large orbital biases are applied to two visible satellites, the 10 km baseline is actually unsolvable if just using FLYKIN™. However, MultiKin can easily deal with this situation and give fast and correct integer ambiguity solutions.

Table 6.6 Comparison of *NUB* in Test 5

Scenario / NUB / Method	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Unconstrained	0 / 0%	10 / 0.46%	12 / 0.56%	256 / 11.85%
six-platform constrained MultiKin	0	0	0	0

6.2.1.6   Test 6 : Impact of Baseline Length

Objective: To test the general effect of the error magnitude on the effectiveness of MultiKin.

Design of scenarios: Usually, errors in double difference measurements increase with the baseline length. In this test, the relative magnitude of spatially correlated error is constant in all scenarios (ionospheric error: 4.4 ppm, tropospheric error: 2.3 ppm, orbital error: 0.06 ppm). Increasing baseline length increases the absolute magnitudes of all correlated errors. The widelane observables were used to keep all the baselines solvable. It can be seen from Table 6.7 that the double differenced widelane errors increase from 0.1 cycle to 0.30 cycle when the baseline lengths increase from 10 km to 35 km.

Table 6.7 Simulated scenarios for Test 5

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error (WL)	Length of Baseline
1	0.01 cycle/ 1.9 mm	0.02 cycle/ 3.8 mm	4.4ppm	2.3ppm	0.06 ppm	On 25m	PDOP= 2.0 Full geometry	WL	0.10 cycle/ 8.6 cm	10 km
2									0.17 cycle/ 14.2 cm	20 km
3									0.30 cycle/ 25.9 cm	35 km

Figure 6.15 and Figure 6.16 show the results of Test 5. It once again supports the conclusions of the previous efficiency tests. The improvement of MultiKin decreases with increasing errors. In Scenario 3, the *NUB* is 61 for the unconstrained method, while for MultiKin, all the baselines are solvable.
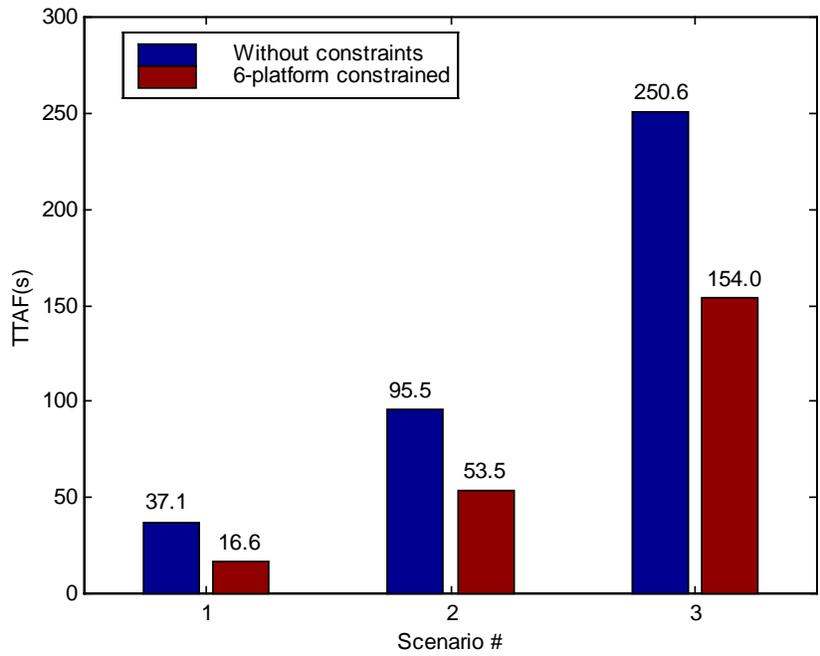
Figure 6.15 Time to true ambiguity fixed for Test 6
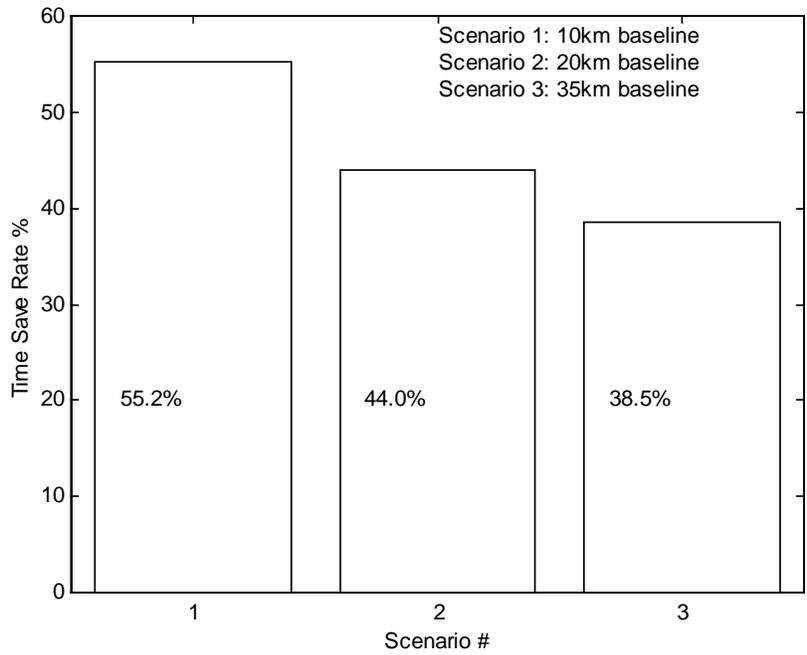
Figure 6.16 Comparison of TSR for Test 6

6.2.1.7   Test 7: Impact of Observables

Objective: To test the effect of using different observables on MultiKin.

Design of scenarios: In this test (see Table 6.8) only one scenario is generated by the software GPS simulator, but in the processing software, the L1 and widelane observables are used respectively to compare the efficiency improvement generated by MultiKin.

Table 6.8 Simulated scenarios for Test 7

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error	Length of Baseline
1	0.01 cycle/ 1.9mm	0.02 cycle/ 3.8mm	0.094 cycle/ 1.8ppm	0.12 cycle/ 2.3ppm	0.0036 cycle	On 25m	PDOP= 2.0 Full geometry	WL	0.078 cycle/ 6.7cm	10km
2								L1	0.12 cycle/ 2.3cm	

The results of simulation tests show that the use of the widelane observables leads to not only faster (see Figure 6.17) but also more efficient (see Figure 6.18) ambiguity resolution than the use of the L1 observables. This can be explained by comparing the total double differenced errors in the widelane and L1 observables, where the relative errors in widelane (0.078 cycle) are smaller than the relative errors in L1 (0.12 cycle). This is because the use of the widelane observables reduces the differenced ionospheric errors (in cycles) by 71.7%, and the differenced tropospheric and orbital errors (in cycles) by 77.9%. A detailed derivation can be found in Chapter 2.

Since the higher efficiency of the widelane observables results from the reduced errors (in cycles) of observations, Test 7 supports the results of all the previous efficiency tests.
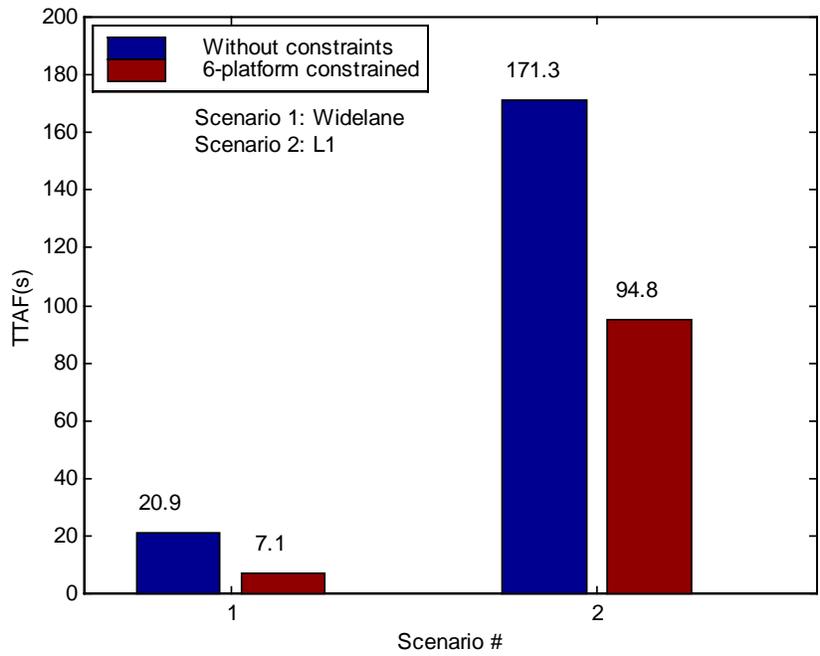
Figure 6.17 Time to true ambiguity fixed for Test 7
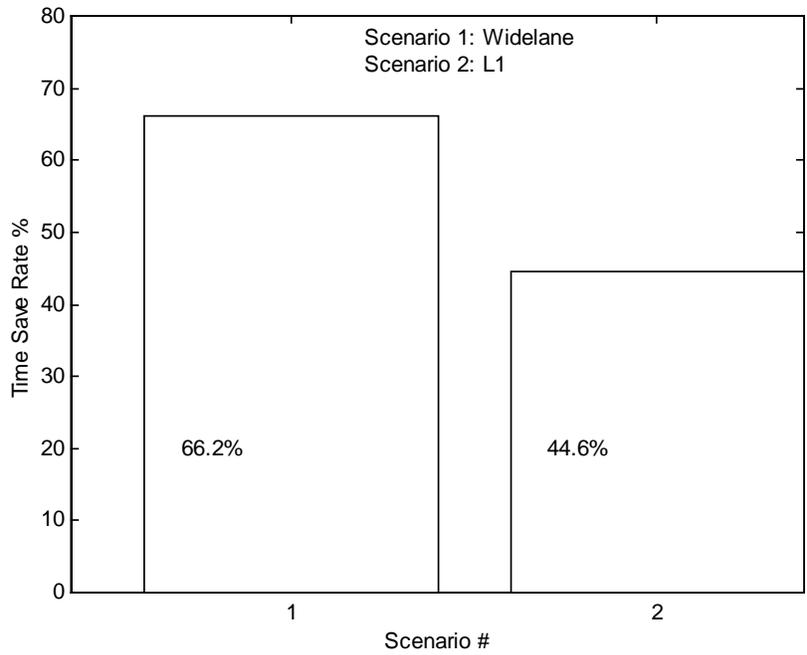
Figure 6.18 Comparison of TSR for Test 7

6.2.1.8 Test 8: Impact of Satellite Geometry and Visibility

Objective: To test the effect of satellite geometry and visibility on the effectiveness of MultiKin.

Design of scenarios: Table 6.9 lists all the scenarios in this test. Various simulated errors are kept at the same level. The lengths of baselines are constant and the resulting total double differenced errors are almost at the same level. The only difference is the number and geometry of visible satellites. A detailed description of "Full geometry", "Reduced geometry" and "Enhanced geometry" can be found in Chapter 5.

Testing results in Figure 6.19 and Figure 6.20 show that the number of visible satellites has a very large impact on the efficiency of ambiguity resolution. It is well known that only three double differenced ambiguities are independent. If more than four satellites are observed, redundancy can be obtained in data processing. The greater the number of visible satellites, the more redundancies there are for ambiguity resolution. The redundancies of the above three scenarios are n, $4n$, and $5n$ respectively, where n is the the number of the observation epochs. When the redundancy increases, both the ambiguity fixing time and the efficiency of MultiKin can be improved. Therefore, it can be expected that MultiKin will show better performance in an open-sky testing area, where the number of commonly visible satellites is large.

Table 6.9 Simulated scenarios for Test 8

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error	Length of Baseline
1	0.01 cycle/ 1.9 mm	0.02 cycle/ 3.8 cm	0.052 cycle/ 2.0 ppm	0.06 cycle/ 2.3 ppm	0.001 cycle	On 25 m	PDOP =1.7 Enhanced geometry	L1	0.064 cycle/ 1.2 cm	5 km
2							PDOP= 2.0 Full geometry		0.064 cycle/ 1.2 cm	
3							PDOP= 2.9 Reduced geometry		0.065 cycle/ 1.2 cm	
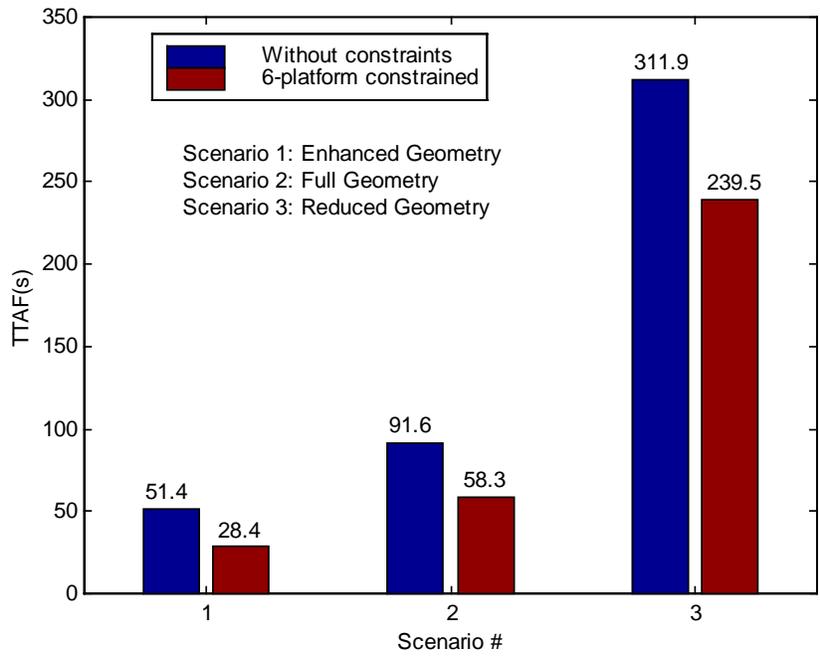
Figure 6.19 Time to true ambiguity fixed for Test 8
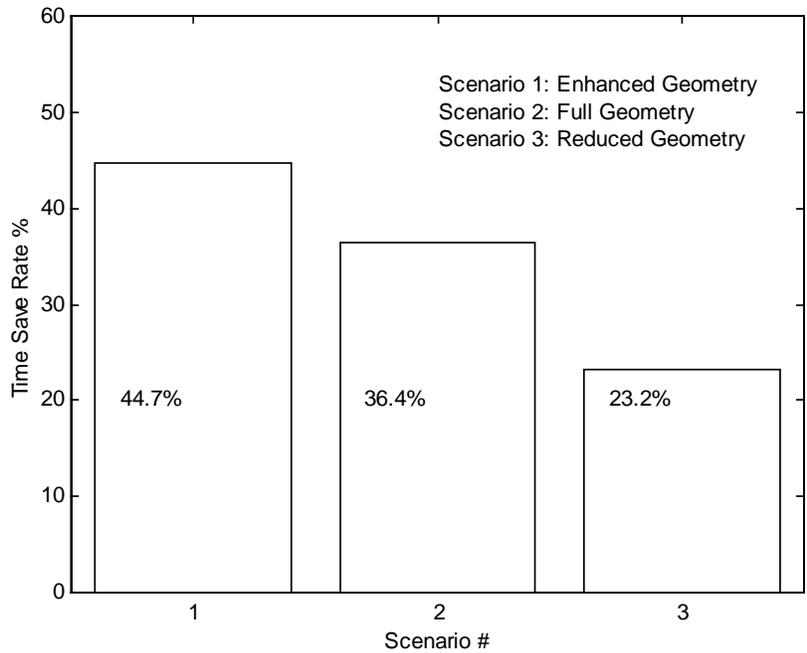
Figure 6.20 Comparison of TSR for Test 8

6.2.1.9   *Longest Solvable Baseline for L1 Observables*

Objective: To test the efficiency of MultiKin in the case of very large differential errors.

Design of scenarios: In this test, two scenarios are set up with baseline lengths of 10 km and 15 km. The ionospheric activity is active (4.0 ppm). Other GPS errors are at normal levels. To test the efficiency of MultiKin when used for high-accuracy (centimetre level) positioning, the L1 observable is used. The detailed parameters are listed in Table 6.10.

Table 6.10 Simulated scenarios for Test 9

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error (L1)	Length of Baseline
1	0.01 cycle/ 1.9 mm	0.02 cycle/ 3.8 mm	0.23 cycle/ 4.0 ppm	0.12 cycle/ 2.3 ppm	0.0027 cycle/ 0.05 ppm	On 25m	PDOP= 2.0 Full geometry	L1	0.22 cycle/ 4.4 cm	10 km
2			0.35 cycle/ 4.0 ppm	0.18 cycle / 2.3 ppm	0.004 cycle/ 0.05 ppm				0.26 cycle/ 5.1 cm	15 km

Testing results (Figure 6.21 and Figure 6.22) show that the performance of both the unconstrained method and MultiKin degrades very fast over this range of distance. Table 6.11 gives the results of *NUB* for the two scenarios. According to the definition of *LSB* given in section 6.1, i.e., the unfixed rate is less than 10%, the *LSB* for the unconstrained method is about 11 km, while for MultiKin, it is about 13 km. These distances are estimated by linear interpolation. It can be concluded that MultiKin can increase the functional distance of ambiguity resolution; however, the increased distance depends on the error magnitudes. Large differential errors can seriously degrade the improvement by MultiKin.

Table 6.11 *NUB* for Test 9

NUB Method \ Scenario	1	2
Unconstrained	43	753
MultiKin	0	378
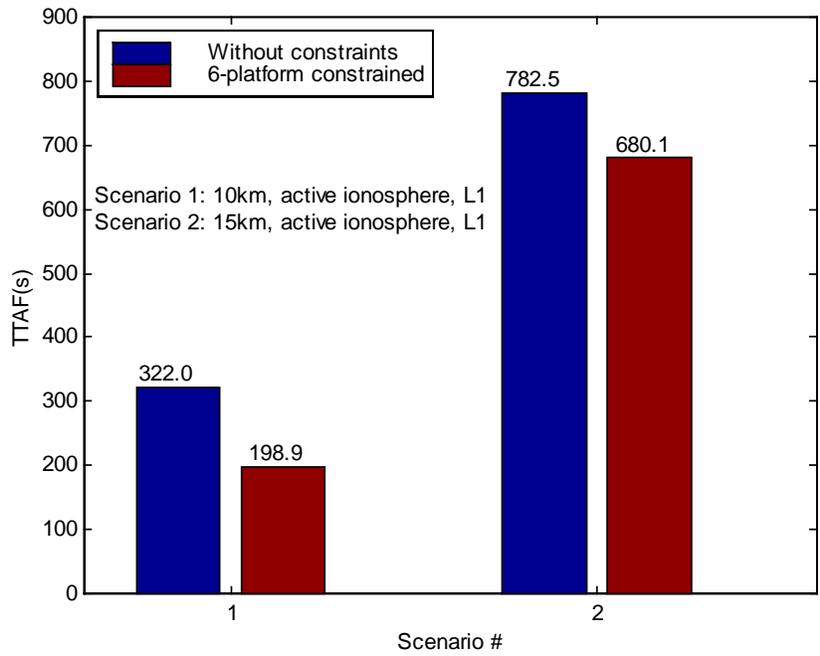
Figure 6.21 Time to true ambiguity fixed for Test 9
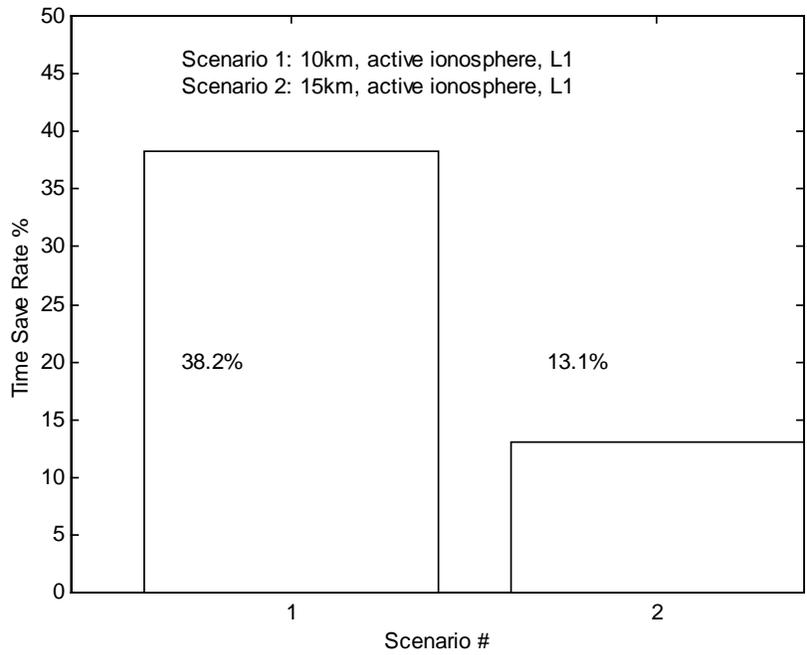
Figure 6.22 Comparison of TSR for Test 9

6.2.2 Accuracy Test

Objective: The constrained method used in MultiKin only affects the ambiguity fixing time because it does not adjust the Kalman filter in FLYKIN™. The accuracy of positioning a single-baseline is not affected by using ambiguity constraints. Herein, the accuracy test is designed to understand the following four points:

[1] Derived accuracy of "virtual baselines",

[2] Extra relative positioning error caused by the "moving reference" error,

[3] Impact of using different observables on positioning accuracy,

[4] Impact of using different solutions on positioning accuracy.

As discussed in Chapter 3, a virtual baseline means a virtual linkage between two moving platforms. The relative position vector of a virtual baseline is not directly computed from the double differenced observations of the two platforms, but derived from the sum of the position vectors along the shortest path between the two moving platforms.

In the accuracy test, a 10-platform configuration is tested. Platform 1 is the reference platform. The red arrow is the position vector (Delaunay edge) directly computed by MultiKin. Therefore, only the relative positions of platform 4 and 7 are directly derived by FLYKIN™. All others are derived by the shortest path algorithm. The relative positioning errors are computed as follows:

[1] Compute the sum of relative position vectors ($\vec{r}_{Ref,i}$) between the reference platform and a rover platform i in the Earth frame (WGS84). Figure 6.23 shows the baselines used for computing relative positions for all rover platforms, e.g., $\vec{r}_{Ref,8} = -\vec{r}_{41} + \vec{r}_{40} + \vec{r}_{08}$, where Ref=1.

[2] Obtain the absolute position of the reference platform $\vec{r}_{Ref}$ using the stand-alone-positioning module in the modified version of FLYKIN™.

[3] Compute the coordinate of the ith rover platform in the local-level frame centred at the computed position ($\vec{r}_{Ref}$) of the reference platform: $\vec{r}_{Ref,i}^{LL}$.

[4] Compute the true coordinate of the ith rover platform in the local-level frame centred at the true position of reference platform (included in the trajectory files): $\vec{R}_{Ref,i}^{LL}$.

[5] Compute the relative positioning errors (the difference between $\vec{R}_{Ref,i}^{LL}$ and $\vec{r}_{Ref,i}^{LL}$).
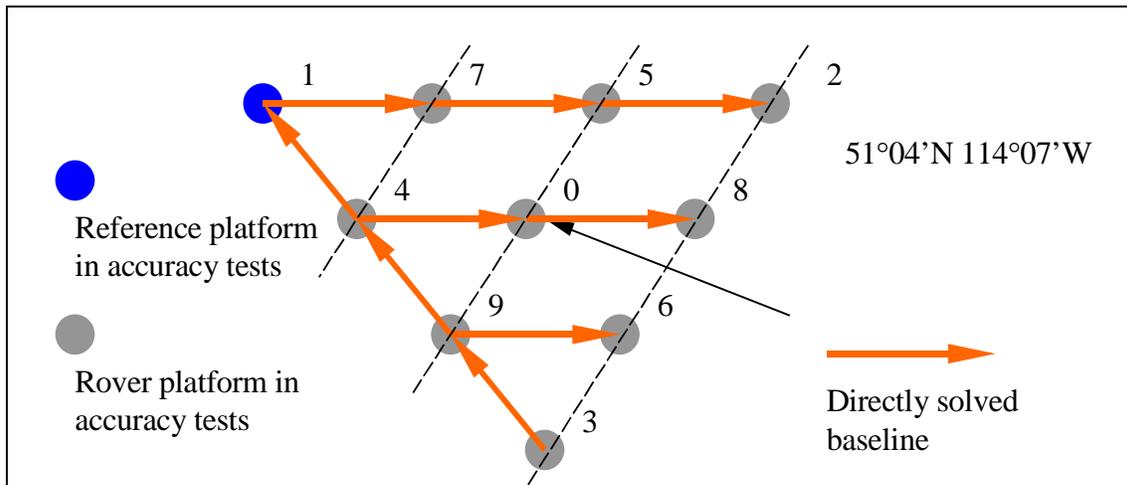
Figure 6.23 The shortest paths from the reference platform to all rover platforms

Design of scenarios: Table 6.12 and Table 6.13 list five scenarios which are designed to test the positioning accuracy in the case of different error magnitudes, baseline lengths, satellite geometries and observable types. The testing period is from 50930 s to 55730 s (GPS time). The magnitude of GPS errors is set at a normal level. The five scenarios described in Table 6.12 were run twice, namely, once in fixed ambiguity mode, and once in float ambiguity mode. In the case of the float mode, a filter convergence period of 1,000 s is used. Although float solutions have poorer positioning accuracy than fixed solutions, they are much more robust than fixed solutions. This is because integer solutions can be incorrectly fixed and the wrong fixes can seriously degrade the relative positioning accuracy. Therefore, in some applications, if float solutions can meet the accuracy requirement, they may be preferred to integer solutions.

Table 6.12 Simulated scenarios for Test 10

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Total DD error	Length of Baseline
1	0.01 cycle/ 1.9mm	0.02 cycle/ 3.8mm	0.014 cycle/ 1.8 ppm	0.018 cycle/ 2.3 ppm	0.00055 cycle/ 0.07ppm	On 25 m / and then off for all five scenarios	PDOP= 2.0 Full geometry	L1	0.047 cycle/ 0.89 cm	1. 5km
2			0.047 cycle/ 1.8 ppm	0.06 cycle/ 2.3 ppm	0.0018 cycle/ 0.07ppm				0.064 cycle/ 1.5 cm	5 km
3			0.47 cycle/ 1.8 ppm	0.06 cycle/ 2.3 ppm	0.0018 cycle/ 0.07ppm		PDOP =2.8 Reduced geometry		0.063 cycle/ 1.2 cm	
4			0.094 cycle/ 1.8 ppm	0.12 cycle/ 2.3 ppm	0.0036 cycle/ 0.07ppm		PDOP= 2.0 Full geometry		0.12 cycle/ 2.3 cm	10 km
5								WL	0.078 cycle/ 6.7 cm	

Table 6.13 Length of tested baselines and virtual baselines

Length of baseline / Scenario / Platform #	1	2	3	4	5
1	Reference platform				
4	1.5 km	5.0 km		10.0 km	
7	1.5 km	5.0 km		10.0 km	
5[1]	3.0 km	10.0 km		20.0 km	
0[1]	2.6 km	8.7 km		17.3 km	
9[1]	3.0 km	10.0 km		20.0 km	
2[1]	4.5 km	15.0 km		30.0 km	
8[1]	4.0 km	13.2 km		26.4 km	
6[1]	4.0 km	13.2 km		26.4 km	
3[1]	4.5 km	15.0 km		30.0 km	

[1] Virtual baseline

In the first group of tests, the integer ambiguities are set to the correct values. They are treated as constant and not updated by the Kalman filter in FLYKIN™. In the second group of tests, the ambiguities are updated by Kalman filtering but the ambiguity search module is switched off during the tests. Thus, the positioning accuracy is always related to the float solutions.

These tests are first conducted with SA switched on. The resulting range error is 25.6m (RMS). Then the tests are repeated with SA switched off. The comparison of these tests can help to understand the accuracy degradation of relative positioning caused by a moving reference error.

From test results shown from Table 6.14 to Table 6.17, the following conclusions can be drawn:

[1] The accuracy of the "virtual baseline" derived by the shortest path algorithm is close to the accuracy of the real baseline with the same length. For instance, referring to Table 6.14, the accuracy of a 10 km virtual baseline between platform 1 and 5 in Scenario 2 is 5.57 cm, while the accuracy of a 10 km real baseline between platform 1 and 7 in Scenario 4 is 5.81 cm.

[2] Poor satellite geometry lowers the accuracy of stand-alone positioning and corresponds to increased errors in the relative positions. For instance, in Scenario 2 (Table 6.14), the stand-alone positioning accuracy is 25.1 m, and the relative positioning accuracy of baseline 1-4 is 2.98 cm. In Scenario 3 (Table 6.14), the poor satellite geometry degrades the stand-alone positioning accuracy to 55.2 m and hence degrades the relative positioning accuracy of baseline 1-4 to 5.75 cm.

[3] The use of the widelane observable (integer solution) can support a 10 cm level relative positioning accuracy for a 10 km baseline whether SA is on or off, see Table 6.14 and Table 6.16.

[4] The L1 (integer solution) observable can support 10 cm level positioning accuracy for a 20 km baseline (virtual) when SA is on (see Table 6.14, Scenario 4, baseline 1-5

and baseline 1-9), and a 30 km baseline (virtual) when SA is off (see Table 6.16, Scenario 4, baseline 1-2, 1-3).

[5] The L1 (float solution) observable can support centimetre level positioning accuracy over distances between 10 km and 20 km (see results of scenario 2 and 4 in Table 6.15 and Table 6.17) when GPS errors are at the normal level.

[6] Switching off SA is beneficial to increasing relative positioning accuracy. For instance, the relative positioning error of the five-km baseline 1-4 is decreased from 2.98 cm to 1.97 cm after SA was switched off in the software simulator.

Table 6.14 Positioning accuracy (3D RMS) of fixed solution (SA on)

Scenario / Positioning Accuracy / Platform #	1	2	3	4	5
1 (stand-alone, horizontal)	29.2 m	25.1 m	55.2 m	27.6 m	27.6 m
4	1.57cm	2.98 cm	5.75 cm	5.27 cm	10.1 cm
7	1.59 cm	2.94 cm	5.34 cm	5.81 cm	9.90 cm
5	2.24 cm	5.57 cm	9.40 cm	11.3 cm	14.3 cm
0	2.02 cm	5.11 cm	8.28 cm	9.30 cm	12.8 cm
9	2.23 cm	5.66 cm	10.1 cm	10.0 cm	14.5 cm
2	2.99 cm	8.82 cm	14.2 cm	14.5 cm	19.16 cm
8	2.71 cm	7.35 cm	12.1 cm	13.4 cm	17.6 cm
6	2.74 cm	7.41 cm	11.9 cm	14.3 cm	17.8 cm
3	3.03 cm	8.11 cm	13.7 cm	16.7 cm	19.9 cm

Table 6.15 Positioning accuracy (3D RMS) of float solution (SA on)

Scenario / Positioning Accuracy / Platform #	1	2	3	4	5
1 (stand-alone, horizontal)	29.2 m	25.1 m	55.2 m	27.6 m	27.6 m
4	4.99cm	5.06 cm	8.73 cm	6.31 cm	11.0 cm
7	2.91 cm	5.00 cm	6.64 cm	6.04 cm	11.4 cm
5	4.13 cm	7.87 cm	11.0 cm	13.2 cm	17.4 cm
0	3.75 cm	8.95 cm	8.43 cm	12.4 cm	15.3 cm
9	4.24 cm	8.13 cm	10.1 cm	15.3 cm	21.3 cm
2	5.76 cm	10.7 cm	16.8 cm	16.1 cm	21.8 cm
8	5.30 cm	11.1 cm	14.9 cm	13.6 cm	18.7 cm
6	5.78 cm	10.1 cm	14.5 cm	16.2 cm	21.6 cm
3	4.60 cm	9.81 cm	18.5 cm	17.2 cm	21.3 cm

Table 6.16 Positioning accuracy (3D RMS) of fixed solution (SA off)

Positioning Accuracy / Platform # \ Scenario	1	2	3	4	5
1 (stand-alone, horizontal)	2.60 m	2.10 m	4.05 m	2.46 m	2.47 m
4	1.37cm	1.97 cm	2.13 cm	4.62 cm	10.4 cm
7	1.40 cm	2.03 cm	2.25 cm	4.06 cm	9.56 cm
5	1.59 cm	3.78 cm	3.81 cm	7.52 cm	13.1 cm
0	1.57 cm	3.61 cm	3.72 cm	7.72 cm	13.2 cm
9	1.60 cm	4.01 cm	4.31 cm	8.11 cm	14.6 cm
2	1.91 cm	5.47 cm	5.75 cm	10.1 cm	19.4 cm
8	1.88 cm	5.11 cm	5.22 cm	9.88 cm	17.4 cm
6	1.82 cm	4.95 cm	4.98 cm	10.4 cm	16.8 cm
3	1.93 cm	5.32 cm	5.58 cm	10.4 cm	16.7 cm

Table 6.17 Positioning accuracy of (3D RMS) float solution (SA off)

Positioning Accuracy / Platform # \ Scenario	1	2	3	4	5
1 (stand-alone, horizontal)	2.60 m	2.10 m	4.05 m	2.46 m	2.47 m
4	3.51 cm	4.03 cm	6.76 cm	7.98 cm	11.6 cm
7	3.03 cm	4.62 cm	8.81 cm	7.94 cm	11.6 cm
5	3.67 cm	6.65 cm	10.4 cm	16.1 cm	14.8 cm
0	5.04 cm	6.77 cm	9.31 cm	14.2 cm	15.4 cm
9	4.08 cm	6.98 cm	13.5 cm	13.2 cm	15.4 cm
2	5.29 cm	8.12 cm	15.1 cm	18.2 cm	21.4 cm
8	3.79 cm	7.67 cm	10.6 cm	21.0 cm	21.4 cm
6	5.51 cm	7.98 cm	9.64 cm	17.2 cm	18.7 cm
3	7.88 cm	8.03 cm	10.2 cm	15.2 cm	19.0 cm

6.2.3 Reliability Test

Objective: To test the reliability improvement of MultiKin to detect incorrectly fixed ambiguities.

Design of scenarios: Three scenarios used in the efficiency test are re-tested here to get statistical data concerning *TDW*, see Table 6.18. These three scenarios have different error magnitudes, baseline lengths, satellite geometry, and observable types.

Table 6.18 Simulated scenarios for Test 10

Scenario	Rx (single)	MP (single)	DD Ion (L1)	DD Trop (L1)	DD Orbit (L1)	SA δ-error	Geometry	Observable	Length of Baseline
1	0.01 cycle	0.02 cycle	4.4 ppm	2.3 ppm	Normal	On 25m	Full geometry	L1	10 km
2			2.0 ppm				Reduced geometry	L1	5 km
3			4.7 ppm				Full geometry	WL	35 km

Figure 6.24 shows the distribution of resulting *TDW*s of the tested scenarios. MultiKin can detect the wrong fixes faster than the unconstrained method. In addition, MultiKin can always detect the wrong fixes within 1200 s for the tested scenarios (see Table 6.19). For the unconstrained method, though the number of undetectable wrong fixes can be as large as 11.6%, even when the optimal ratio test is applied in FLYKIN™ for error detection.

MultiKin performs well in detecting wrong fixes. This is beneficial to the reliability and accuracy of positioning. Therefore, it can be concluded that MultiKin can achieve high accuracy faster than the unconstrained method.

Scenario 1

Scenario 2
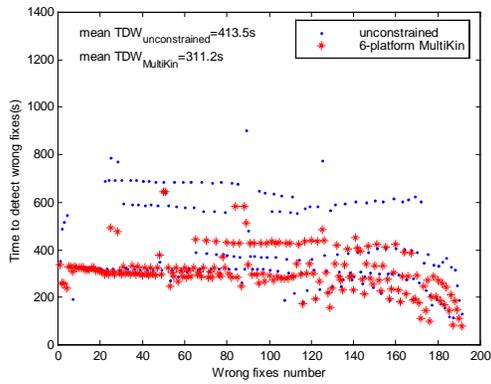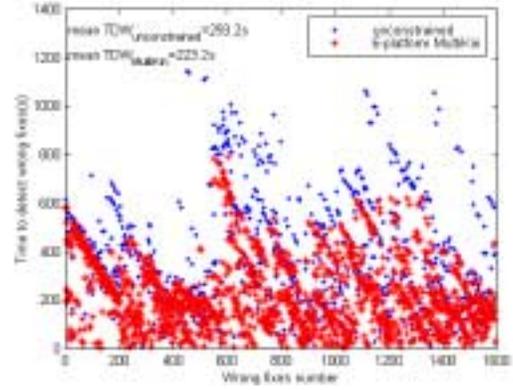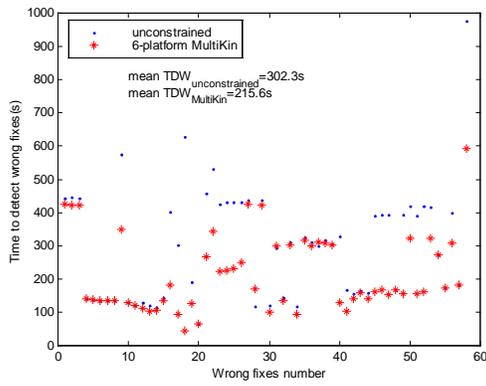
Scenario 3

Figure 6.24 *TDW* **for Test 10**

Table 6.19 Performance of detecting wrong fixes: the unconstrained method vs. MultiKin

Scenario	Total Number of wrong fixes	Mean TDW (s)		Undetected wrong fixes	
		Unconstrained	MultiKin	Unconstrained	MultiKin
1	58	302.3	215.6	0	0
2	1615	293.2	223.2	22 / 1.4%	0
3	216	413.5	311.2	25 / 11.6%	0

6.3   FIELD TESTS: DESIGN, RESULTS AND ANALYSIS

Although MultiKin has been fully tested by simulation tests, it is still necessary to perform some field tests, because the scenarios generated by the software simulator deviate from real situations to some extent. If the conclusions drawn from simulation tests can be confirmed by field tests, they will be more convincing. In the following sections, two field tests with different testing conditions are discussed.

6.3.1   Field Test 1

The first field test is a short baseline test. In this test, the baseline lengths are limited within a few hundred metres; thus, the resulting double differenced errors are very small. This field test is designed to understand the performance of MultiKin in some benign cases.

6.3.1.1   Design of Field Test 1

Moving Platforms: Four cars

GPS Equipment: Four NovAtel Millennium receivers, each mounted on a car (three receivers have choke-ring antenna)

Two Ashtech Z-12 receivers on two reference stations

Data Collector: Five Compaq Pentium 166 notebook PCs and one desktop PC

GPS Data Description: One Hz data rate, C/A code, Doppler, L1 and L2 carrier phase

Testing Period: 1998-Nov-09, 9:30 am -11:30 am

Location: Main Parking lot at Market Mall, Calgary

Vehicle Dynamic: Low speed driving (<30 km/h)

Test Area: <1 km×1 km , Flat and open sky

Configurations of GPS platforms: See Figure 6.25.
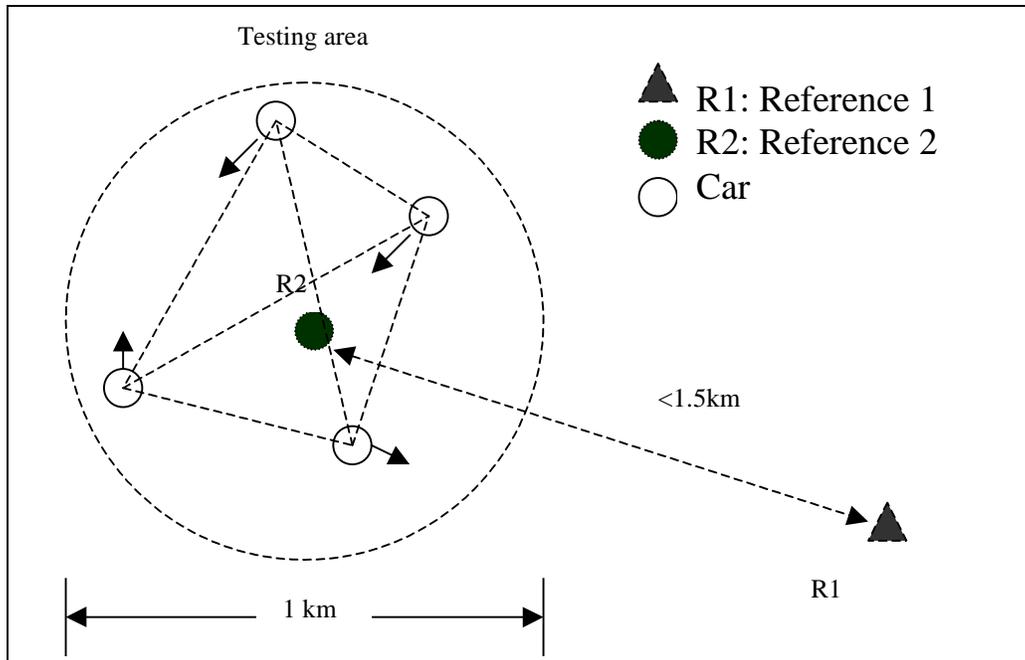
Figure 6.25 Scenario for field test

Two reference stations were set up. One was on the roof of the Engineering building at the University of Calgary (Precisely known coordinate (WGS84): 51 ° 04 ′ 45.80932 ″ *N,* 114 ° 07 ′ 57.99118 ″ *W,* 1116.776 m), the other was at the main parking lot of Market Mall. The reference stations are not necessary for the application of MultiKin. In this field test, they were used to estimate the positioning accuracy and check the correctness of the integer ambiguities.

6.3.1.2 *Real Test Conditions*

In the test, the data of Car 4 was paused for about 15 minutes due to a cable connection problem; therefore, the data from only three cars were used in data processing.

Although the field test was conducted in an open sky area, lamps at the parking lot and mall buildings sometimes obstructed GPS signals. Figure 6.26 shows the visibility of common satellites by the three cars. It can be seen that the number of visible satellites varies considerably. Sometimes, only one common satellite is observed. However, the average number of visible satellites is more than seven during the test period. Thus, the overall observation redundancy is still very good.

Figure 6.27 shows the trajectories of the tested cars. They moved within the designed range of the testing area. The largest inter-platform distance is 610 m. The average inter-platform distance is less than 150 m. Thus, in this test, the effect of spatially correlated errors can be ignored. All these cars were equipped with choke-ring antennas so the multipath effect was largely reduced. According to the statistical results, the RMS of the observation residuals is 0.35 cm.
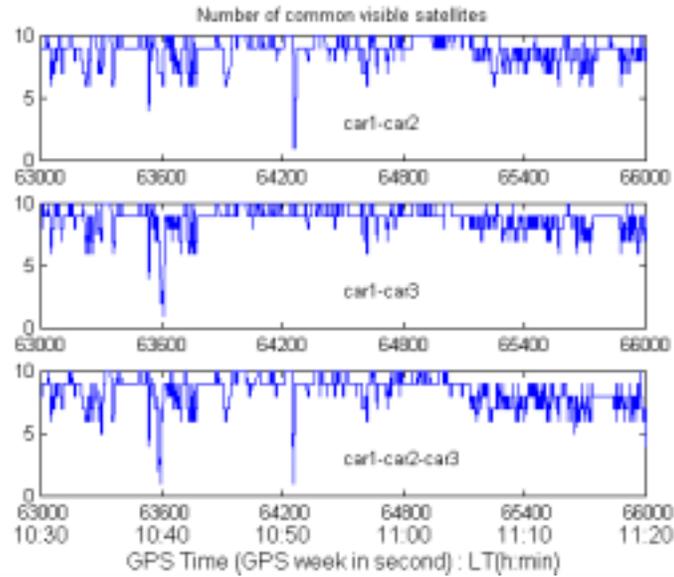
Figure 6.26 Satellite visibility in the testing area
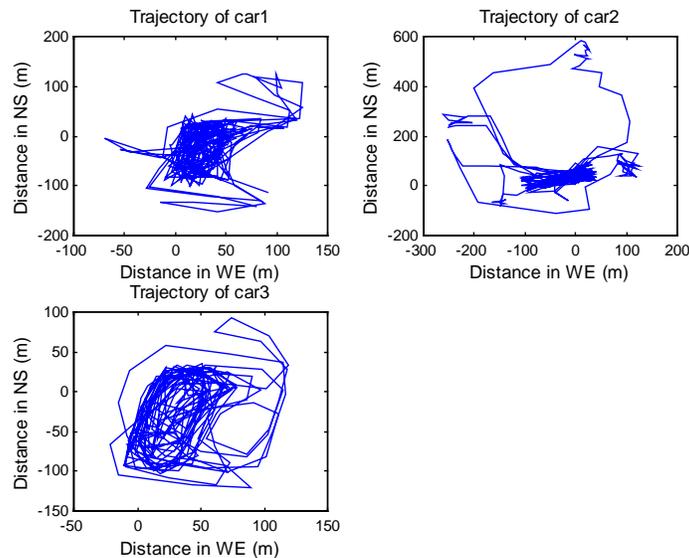
Figure 6.27 Trajectories of moving platforms

6.3.1.3   Estimation of Positioning Accuracy and Verification of Integer Ambiguities

Procedure 1: Estimation of Positioning Precision

[1] Use the modified version of FLYKIN™ (with stand-alone positioning module) to compute the relative position vector between Car 1 and Car 2. $\vec{r}_{12}^{LL}$ is in the local level frame centred at $\vec{r}_1$. $\vec{r}_1$ is the approximated position of Car 1 estimated by the stand-alone positioning module.

[2] Use the data from the reference station and the original version of FLYKIN™ to compute the precise position of Car 1: $\vec{R}_1$.

[3] Use the original version of FLYKIN™ and $\vec{R}_1$ to compute the relative position vector between Car 1 and Car 2, i.e., $\vec{R}_{12}^{LL}$, in the local level frame centred at $\vec{R}_1$.

[4] The residual of $\vec{R}_{12}^{LL}$ and $\vec{r}_{12}^{LL}$ can be approximated as the relative positioning error. The difference between $\vec{R}_{12}^{LL}$ and $\vec{r}_{12}^{LL}$ can be treated as the extra positioning error caused by reference error.

Procedure 2: Ambiguity Checking

[1] Fix the integer ambiguities of a moving baseline between Car 1 and Car 2 (N_{12}) using the modified version of FLYKIN™.

[2] Fix the integer ambiguities of baselines $R1$ and Car 1 ($N_{R1,1}$) using the original version of FLYKIN™.

[3] Fix the integer ambiguities of baselines $R1$ and Car 2 ($N_{R1,2}$) using the original version of FLYKIN™.

[4] Use triangular ambiguity constraint to check the closure of ambiguities $N_{R1,1}$, $N_{R1,2}$ and N_{12}.

[5] Repeat steps [2] to [4], but using reference station *R2*.

If N_{12} meets the closure of both constraints in [4] and [5], it will be treated as the correct integer solution. The above procedures were used to check all the moving baselines.

6.3.1.4 Results of Field Test 1

The field tests consist of multiple tests. The first test started at 63000 s (GPS time) which was Calgary local time 10:30 am. The following tests started at 15 s after the beginning of the previous test. Each test lasted for at most 300 s. If the true ambiguity can be fixed within 300 s, the test is treated as a successful test, the *TAF* is recorded and a new test is started. Meanwhile, if any wrong fix happens during this period, the *TDW* is also recorded. If the true ambiguity cannot be fixed within 300 s, the trial of ambiguity fixing fails and no *TAF* and *TDW* is recorded. The whole testing period is 3300 s; 300 s are reserved for the final test. Thus, the effective testing period is 3000 s, which contains 200 overlapping tests. It should be noted that the maximum testing period is 300 s which is much less than that of the simulation tests. This is because the baselines were very short in these field tests, i.e., the differential errors were small and multipath was also reduced by using choke-ring antenna, ambiguities could be fixed very quickly.

Table 6.20 shows the efficiency improvement of MultiKin for the field test. The fixed integer ambiguities are verified by the ambiguity checking procedure. No wrong fixes were detected. The efficiency improvement induced by MultiKin is relatively high (*TSR*=61.3%). This is because the short inter-platform distances and the application of chokering antennas limited the observation error at a very low level. Therefore, it can be concluded that MultiKin can work well in real GPS scenarios (benign case).

The RMSs of stand-alone and relative positioning error are given in Table 6.21. When using the stand-alone module to position a car, the RMS of the horizontal positioning error is about 30 m because SA was still on at that time. The RMS of relative positioning is about 0.71 cm, which is very good, because of slight measurement errors. The degradation of relative positioning precision is only about 0.07 cm, which is small

enough to be ignored. This is because the extra relative positioning error is proportional to the inter-platform distance, which was quite short in the field test.

Table 6.20 Comparison of TAF between the unconstrained method and MultiKin (L1 observables)

	Unconstrained	MultiKin	TSR
Car1-Car2	23.9 s	10.2 s	61.3%
Car2-Car3	38.8 s	12.6 s	
Car3-Car1	24.1 s	10.8 s	

Table 6.21 Relative positioning precision in the field test (L1 observables)

Baseline	Horizontal stand-alone positioning accuracy of Car i (2D RMS)	Residual of $\vec{r}_{ij}^{LL}$ (3D RMS)	Residual of $\vec{R}_{ij}^{LL}$ (3D RMS)	$\vec{r}_{ij}^{LL}$ - $\vec{R}_{ij}^{LL}$ (3D RMS)
Car1-Car2	31.3 m	0.71 cm	0.71 cm	0.05 cm
Car2-Car3	32.2 m	0.72 cm	0.71 cm	0.07 cm
Car3-Car1	30.6 m	0.71 m	0.70 cm	0.07 cm

6.3.2   Field Test 2

The second field test is a longer baseline test. In this test, the baseline lengths vary between three and seven kilometers. Furthermore, the ionosphere is currently active (solar maximum: mid 2000). Thus, the resulting double differenced errors are larger than those in Field Test 1. Test 2 is designed to understand the performance of MultiKin for the case of larger differential errors and configurations.

6.3.2.1 Field Test Design

Moving Platforms: Five cars

GPS Receivers: Three NovAtel OEM3 L1/L2 receivers (one was used at the reference station), three NovAtel OEM4 L1/L2 receivers

Antennas: Five NovAtel Pinwheel antennas, one choke-ring antenna (at the reference station)

Data Collector: Four Panasonic Pentium III notebook PCs, one Acer Pentium III notebook PC, and one desktop PC (at reference station)

GPS Data Description: 1.0 Hz data rate, C/A code, Doppler, L1 and L2 carrier phase

Testing Period: 2000-Dec-21, 10:05 am -12:45 am

Vehicle Dynamic: Low speed driving (<50 km/h)

Test Area: 4 km $\times$ 6 km, Flat and open sky, (See Figure 6.28)

Temperature: -15ºC

A reference station was set up on the roof of the Engineering building at the University of Calgary (Precisely known coordinate (WGS84): 51 $°$ 04 $'$ 45.94126 $''$ *N,* 114 $°$ 07 $'$ 58.29947 $''$ *W,* 1116.877 m). This reference station was also used to independently estimate the positioning accuracy and to check the correctness of the integer ambiguities. In this test, five cars moved in five separate areas around the reference station. The distances between the reference station and cars vary between two and three kilometres. The distances between cars vary between three and seven kilometres.
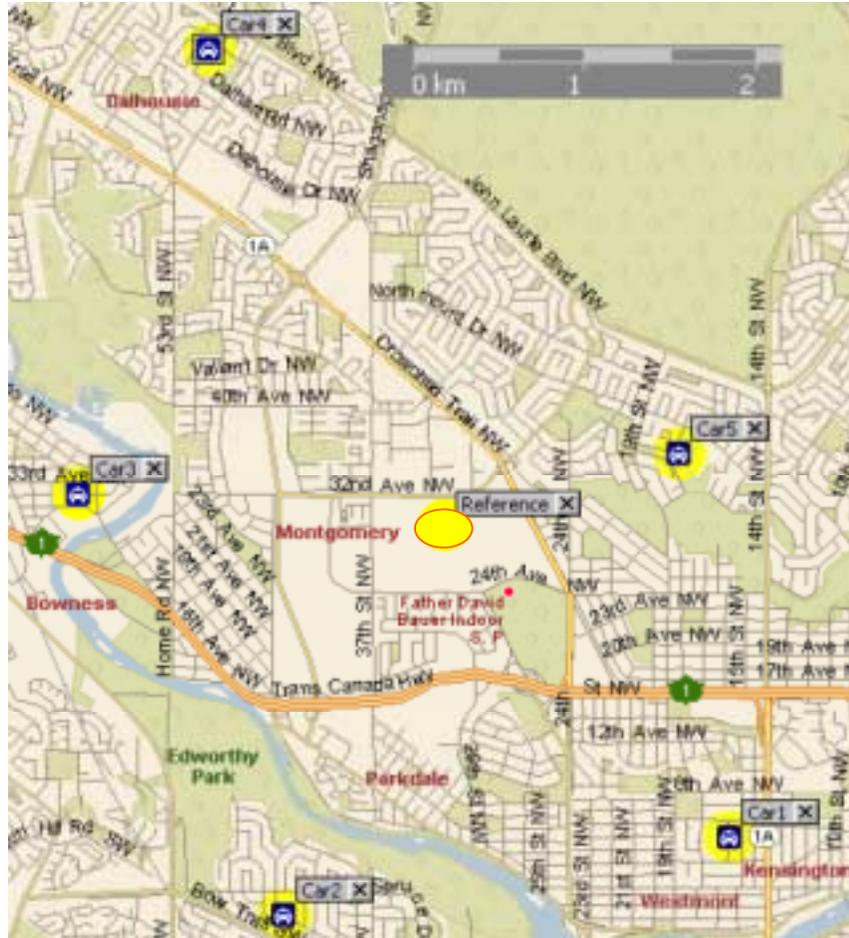
Figure 6.28 Map of Field Test 2

6.3.2.2 Analysis of Testing Conditions

The test started at 10:05 am, December 21, 2000. After arriving at the testing sites, each car stopped for about 10 minutes to perform a static initialization. This procedure is not necessary for MultiKin, but was performed for independent integer ambiguity verification and position estimation. The static initialization was from 10:30 am to 10:40 am (GPS time: 408600 s to 409200 s). The kinematic test started at 10:40 am and ended at 12:20 pm (GPS time: 409200 s to 415200 s).

Figure 6.29 shows the longitude variations of each moving platform as a function of time. The static initialization can be observed from the unchanged longitudes of the cars. Periodic oscillations in longitude can be observed during the kinematic test, which means that these cars (except Car 5) were circling at the testing sites. This also implies that the

inter-platform distances did not change much during the testing period. Although the position changes of Car 5 were relatively larger than others, these changes did not cause an obvious variation of the overall testing geometry. Therefore, the dynamic Delaunay triangulation is not necessary for the test.
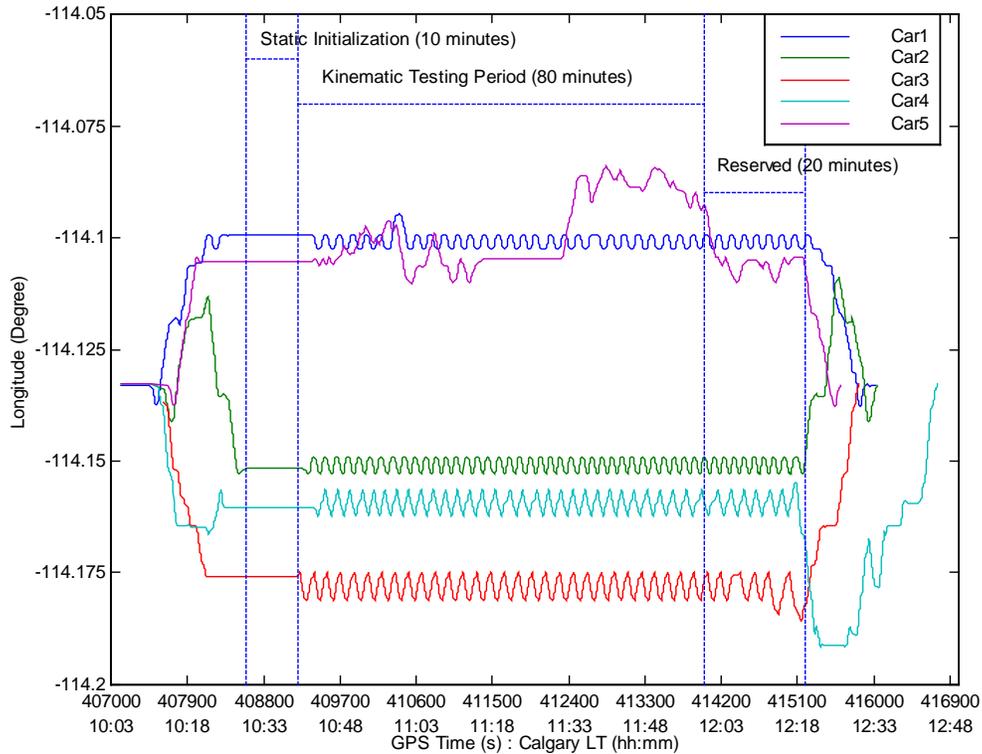
Figure 6.29 Longitudes of moving platforms in Field Test 2

Seven baselines were selected using the Delaunay triangulation. Figure 6.30 shows the selected baselines and the real trajectory of each car during the kinematic portion of the test. Figure 6.31 shows the temporal variations and distribution of baseline lengths. It can be observed that baseline lengths vary from 3000 m to 7000 m; thus, the effect of spatially correlated errors is expected to be significant and to result in noticeable performance degradation. The RMS of double differenced L1 residuals is 0.65 cm, which is larger than that of Field Test 1 (0.31 cm).
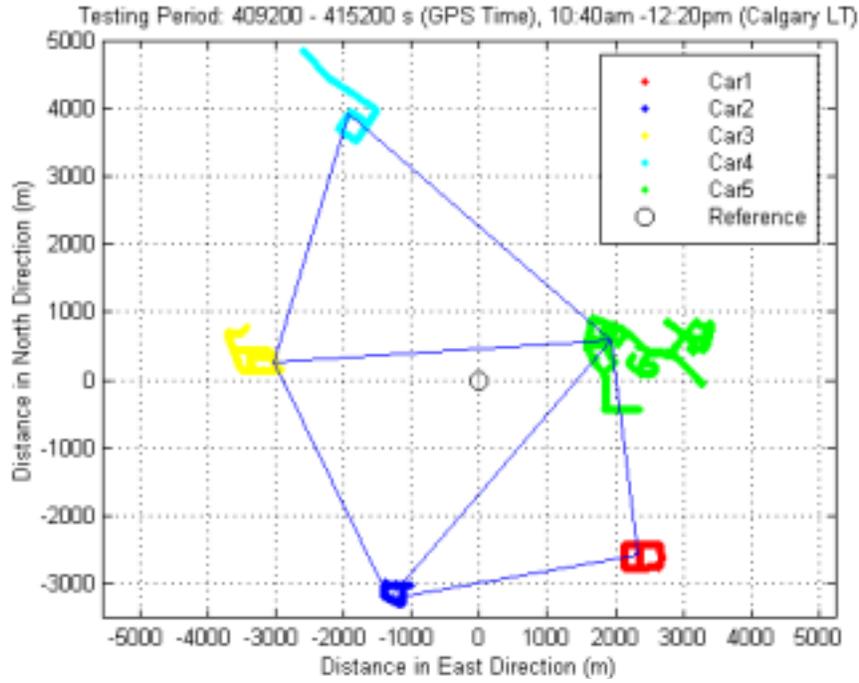
Figure 6.30 Selected baselines and trajectory of each moving platform
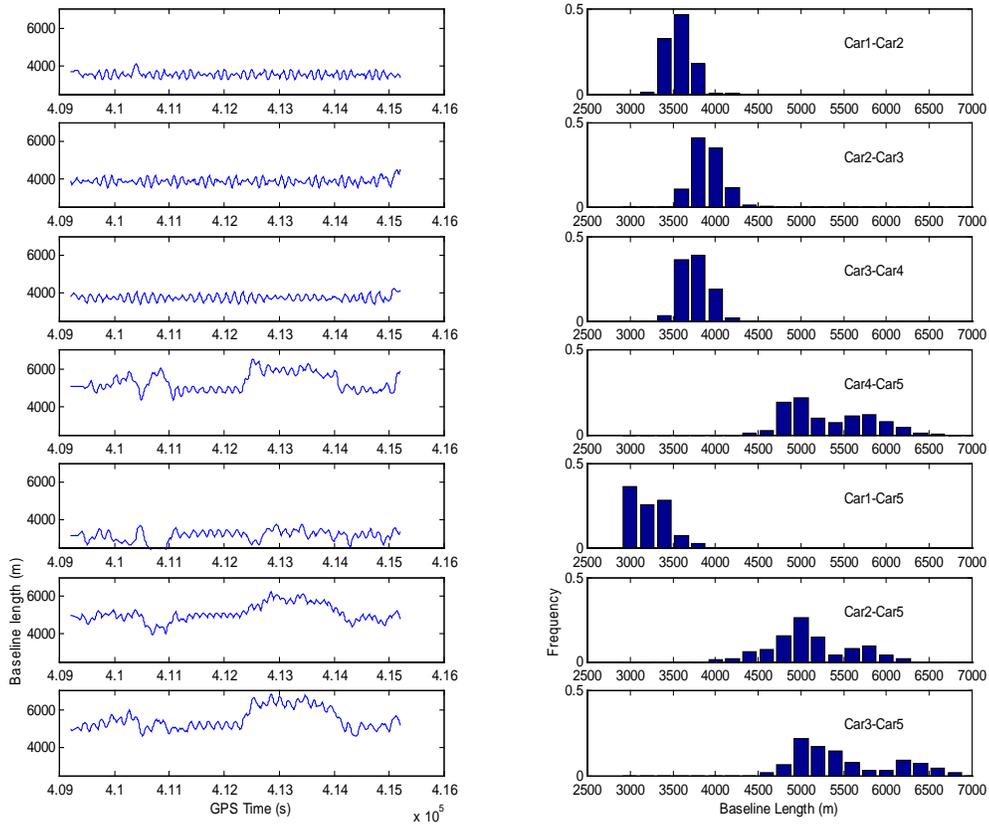
Figure 6.31 Temporal variations and distributions of baseline lengths

185

There were few obstructions at elevation higher than 25° to mask signals. The average satellite visibility is good for each baseline. Figure 6.32 shows the temporal variations and distributions of satellite visibility observed for each baseline. In most cases, eight to nine satellites could be commonly seen from any two cars. This provides a good redundancy for ambiguity resolution. However, the satellite visibility varies considerably, due to signal masking at elevations lower than 25° due to trees and residences.
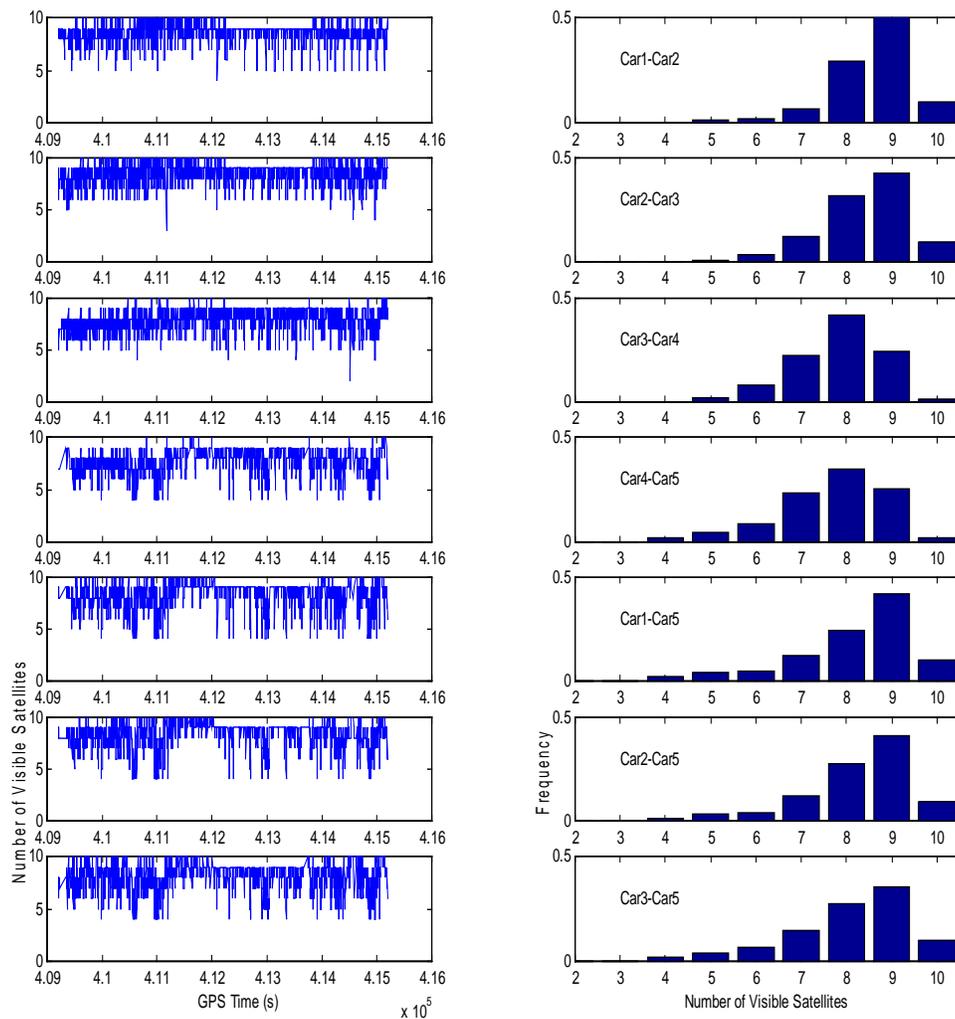
Figure 6.32 Satellite visibility for the selected baselines

6.3.2.3   Results Analysis

[1] Stand-alone Positioning Accuracy

Stand-alone positioning accuracy depends on both DOP and measurement errors. Table 6.22 lists the resulting DOP and positioning error (L1) for each car. To estimate errors in stand-alone positioning, the precise trajectory of each car must be known first with respect to the WGS84 reference station coordinates. These were computed using the original version of FLYKIN ™ and the double differenced measurements between each moving platform and the static reference station, whose coordinates in WGS84 were known. Since the HDOP is very good (<1.5) and SA has been turned off, the horizontal positioning accuracy is high. This implies that the accuracy degradation in relative positioning can be greatly reduced, because an error of one metre induces only a 0.2 ppm baseline error.

Figure 6.33 and Figure 6.34 show the temporal DOP variations and the absolute positioning errors observed at Car 1. The periodic DOP variations are caused by signal masking of low satellites. During most periods of the kinematic test, Car 1 circled around a school zone; thus, the masking sequences repeated every few minutes.

Table 6.22 DOP and stand-alone positioning errors (L1) in Field Test 2

	Car 1	Car 2	Car 3	Car 4	Car 5
Mean HDOP	1.14	1.12	1.18	1.33	1.28
Mean VDOP	1.63	1.64	1.68	1.94	1.61
Horizontal Error(m): RMS	1.88	2.25	2.14	2.16	2.51
Vertical Error(m): RMS	9.36	9.40	9.76	8.99	9.01
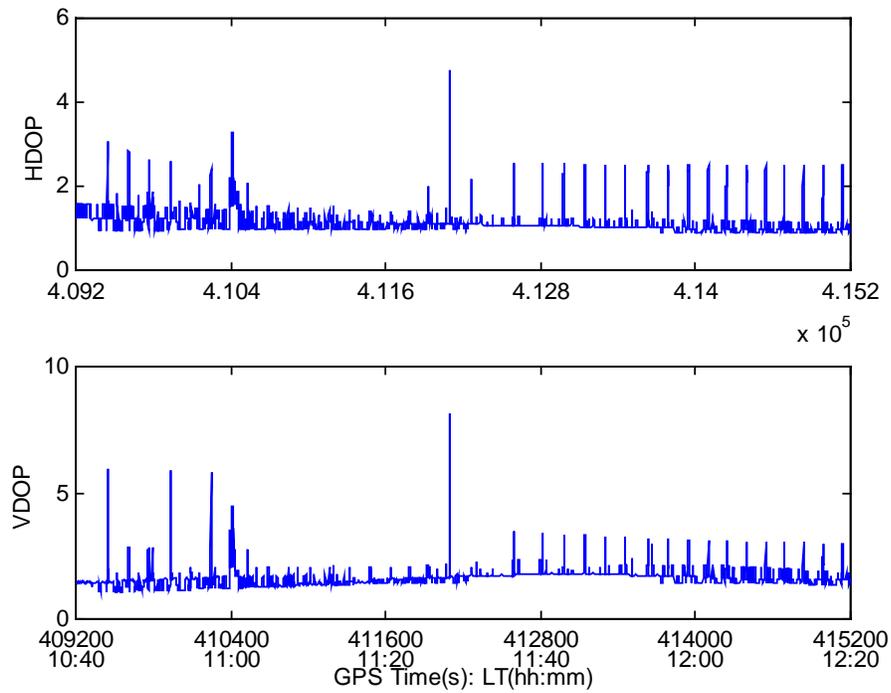
Figure 6.33 DOP observed at Car 1
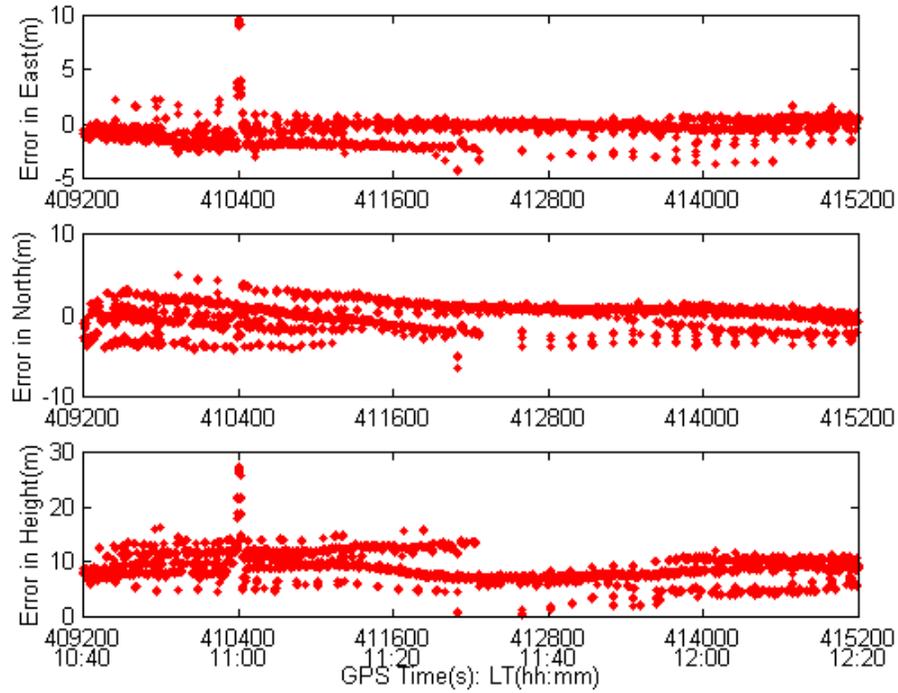
Figure 6.34 Stand-alone positioning error of Car 1

[2] Efficiency Improvement of MultiKin

The efficiency tests are also based on multiple trials. The first trial starts at 409200 s (GPS time). The following trials start at 15 s after the beginning of the previous trial. Each trial lasts for at most 1200 s. If the ambiguities can be fixed within 1200 s, the trial is treated as a successful one, the *TAF* is recorded and a new trial is started. If the ambiguities cannot be fixed within 1200 s, the trial of ambiguity fixing fails and no data is recorded. The entire data sequence spans 6000 s; 1200 s are reserved for the final trial. Thus, the effective testing period is 4800 s, which contains 320 overlapping trials.

The first group of efficiency tests was conducted to check the impact of platform numbers on MultiKin. Results with three, four, and five-platform constraints were compared with those from the unconstrained method. In the three-platform configuration, Cars 1, 2 and 5 were used. In the four-platform configuration, Car 4 was added to the three-platform configuration. The five-platform configuration consists of all the moving vehicles. The *TAFs* of Baselines Car 1-Car 2, Car 2-Car 5, and Car 5-Car 1 are used to estimate the efficiency improvement induced by ambiguity constraints. The procedure defined in Section 6.3.1.3 was used to check the correctness of the fixed ambiguities. No wrong fix was detected by the ambiguity checking procedure.

Table 6.23 shows the results of the efficiency tests using L1 observables. *TSR(3)* is computed using the *TAF*s of only three baselines. *TSR* is computed using the *TAF*s of all the baselines in the configuration. These results validate the results of the simulation tests. First, the ambiguity constraints can speed up the ambiguity resolution. By using ambiguity constraints, the time-to-fix can be reduced by 29% and the number of unfixed baselines is significantly reduced (unconstrained *NUB* 12.8% vs. MultiKin *NUB* 0). Second, increasing the number of constrained platforms can further improve the efficiency of ambiguity resolution, but the improvement rate diminishes. For instance, when the four-platform constraints are used, the *TSR(3)* increases from 22.4% to 28.0%. the five-platform constraints can only incrementally improve *TSR(3)* by another 1%. The results also show that the increased errors can degrade the performance of MultiKin in the L1 mode. This can be seen by comparing the *TSR* of this test with that of Test 1. In

the first field test, the *TSR* (three-platform constraints) reaches 61.3%, with double difference measurement residuals of 0.35 cm (L1). In the current test, the *TSR(3)* is 22.4% due to the larger L1 measurement errors (0.65 cm).

Table 6.23 Efficiency improvement of MultiKin in Field Test 2 (L1 observable)

Baseline # / *TAF/ NUB* / Constraints	Car1-2	Car2-5	Car5-1	*TSR(3)*	*TSR*
Unconstrained	555.9 s / 12.8%	458.8 s / 4.1%	414.8 s / 2.8%		
Three-platform constrained	457.3 s / 0	332.1 s / 0	319.5 s / 0	22.4 %	22.4 %
Four-platform constrained	418.5 s / 0	310.6 s / 0	300.5 s / 0	28.0 %	28.6 %
Five-platform constrained	412.6 s / 0	305.7 s / 0	296.2 s / 0	29.0 %	28.9 %

The *TAF*s of this field test are much larger than those of Field Test 1. This results not only from the increased measurement errors but also from the rapidly changing satellite visibility. The rapid variation in the satellite visibility means low satellite tracking is frequently interrupted. When a satellite is re-acquired, its new ambiguity has to be determined just like it was tracked for the first time; the Kalman filter in FLYKIN™ has to be reset to estimate the float ambiguity and its variance. Thus, the rapid change in satellite visibility results in frequent Kalman filter resets and causes difficulty in the filter's convergence. Thus, the time-to-fix increases significantly.

The above efficiency tests were repeated using the widelane observables (WL). Table 6.24 shows the results. Surprisingly, the use of widelane observables does not improve the efficiency of MultiKin. Furthermore, the use of WL observables does not speed up ambiguity resolution of a single baseline. For instance, in the case of the unconstrained method, the average *TAF* of Baseline Car 1-2 using WL observables (621.4 s) is longer

than that using the L1 observables (555.9 s), which contradicts previous experience. However the reason for this is a relatively higher percentage of L2 carrier phase cycle slips on lower satellites due to signal masking.

Table 6.24 Efficiency improvement of MultiKin in Field Test 2 (WL observables)

Baseline # *TAF/* *NUB* Constraints	Car1-2	Car2-5	Car5-1	*TSR(3)*	*TSR*
Unconstrained	621.4 s / 12.8%	483.7 s / 4.1%	327.1 s / 2.5%		
Three-platform constrained	523.1 s / 0	379.6 s / 0	266.4 s / 0	18.4 %	18.4 %
Four-platform constrained	500.3 s / 0	362.5 s / 0	251.7 s / 0	22.2 %	21.5 %
Five-platform constrained	493.2 s / 0	358.5 s / 0	246.1 s / 0	23.3 %	22.4 %

Table 6.25 shows the number of cycle slips in the L1 and widelane observables. It can be seen that there are much more cycle slips in the widelane observables. The difference is caused by a higher number of cycle slips in the L2 observables. The reason why there are many cycle slips in the L2 observations is that the signal strength of L2 is weaker than that of L1. Moreover, the semi-codeless tracking technique of the L2 carrier phase has at least 14 dB loss in signal to noise ratio with respect to the direct P code correlation. Therefore, the L2 phase lock loop has more difficulty maintaining lock under signal masking conditions.

Since there are more cycle slips in the WL observables, the Kalman filter will be reset more frequently than in the case of L1 observables. Thus, the time-to-fix will increase due to the frequent filter reset.

Table 6.25 Number of cycle slips in the L1 and WL observables

	Car1-2	Car2-5	Car5-1
Number of Epochs with cycle slip (L1)	22	59	54
Number of Epochs with cycle slip (WL)	333	227	223

[3] Relative Positioning Accuracy

The Procedure 1 defined in Section 6.3.1.3 was used to estimate the relative positioning accuracy in Field Test 2. It can be seen from Table 6.26 that sub-decimetre accuracy was achieved in this configuration, which has an average baseline length of 5 km. From Table 6.26, it can also be concluded that the accuracy degradation in relative positions caused by the reference error is very small, because the baselines are relatively short (< 7 km) and the stand-alone positioning accuracy (< 3 m) is greatly improved, now that SA is off.

6.3.3 Field Test Summary

From the above two field tests, it can be found that the conclusions drawn from the simulation tests are validated by the field tests. For instance, the improvement rate of ambiguity resolution efficiency diminishes with an increasing number of platforms. Performance decreases with an increasing magnitude of the differential errors. Since these two field tests were conducted under quite different conditions, such as the configuration, baseline lengths, error magnitudes, etc., it can be concluded that the simulation tests are valid for evaluating the performance of MultiKin, which also implies that the design of error models in the software GPS simulator is valid.

Table 6.26 Relative positioning accuracy in Field Test 2 (L1 fixed solution)

Baseline	Horizontal stand-alone positioning accuracy of Car i (RMS)	Residual of $\bar{r}_{ij}^{LL}$ (3D RMS)	Residual of $\bar{R}_{ij}^{LL}$ (3D RMS)	$\bar{r}_{ij}^{LL}$ - $\bar{R}_{ij}^{LL}$ (3D RMS)
Car1-Car2	1.88 m	6.42 cm	6.29 cm	0.20 cm
Car2-Car3	2.25m	5.67 cm	5.55 cm	0.21cm
Car3-Car4	2.14m	4.96 cm	4.87 cm	0.14 cm
Car4-Car5	2.16m	6.59 cm	6.44 cm	0.30 cm
Car1-Car5	1.88 m	4.73 cm	4.69 cm	0.12 cm
Car2-Car5	2.25m	6.58 cm	6.45 cm	0.29 cm
Car3-Car5	2.14m	6.39 cm	6.30 cm	0.17 cm

7   OPERATIONAL CONSIDERATIONS

To use the MultiKin method operationally in real-time, three issues must be considered, namely:

[1] The data processing capability for GPS data from multiple platforms,

[2] The data processing structure,

[3] The type of data link.

The requirements and options are discussed in the following sections.

7.1   DATA PROCESSING CAPABILITY

MultiKin was tested on a desktop PC with an Intel Pentium® II 400 MHz CPU and 64 MB memory. It was programmed to process GPS data from up to 10 moving platforms, which corresponds to 18 baselines and nine triangular constraints. However, these numbers can easily be increased to any level as long as the processor can handle it in real-time. According to the results of a test conducted to analyze processing speed, it is estimated that the above processor can actually process up to 50 moving platforms, i.e., about 140 moving baselines (by Delaunay triangulation) in real-time with a GPS data rate of one Hz. Increasing the GPS rate decreases the maximum number of platforms that can be processed.

The above data processing capability is derived assuming that all the CPU time is used for data processing. In practice, some of the CPU time is taken to download data from the communication link interface, decode the received data, identify the source of the data, convert the data to recognizable format for MultiKin, pre-process the data, and output the results. When the data increases, the data processing time also increases. Therefore, to design a practical system, the extra processing time must be considered when estimating the capability of data processing.

7.2 DATA PROCESSING STRUCTURE

To operationally implement MultiKin, the transmission of the measured data is indispensable. However, the adopted communication link depends on the data processing structure.

Two kinds of structures are usually adopted for data processing, as shown in Figure 7.1. In the first structure, the data of multiple platforms is processed at a data processing centre. This centre does not have to be a GPS station, but it must be equipped with a wireless data link for receiving observation data from the moving GPS platforms. In this structure, only the central station runs MultiKin to compute the relative position of the surrounding moving platforms. This structure is called the central processing structure (system) in the following discussion.

The Central Processing Structure

Central station
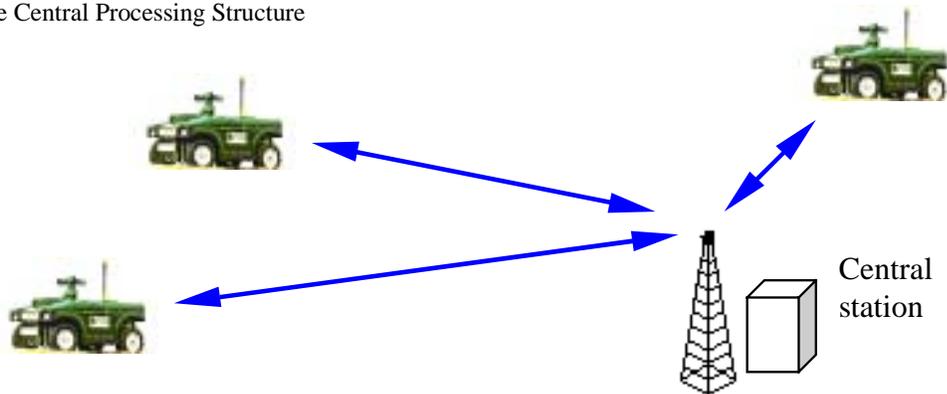
The Distributed Processing Structure
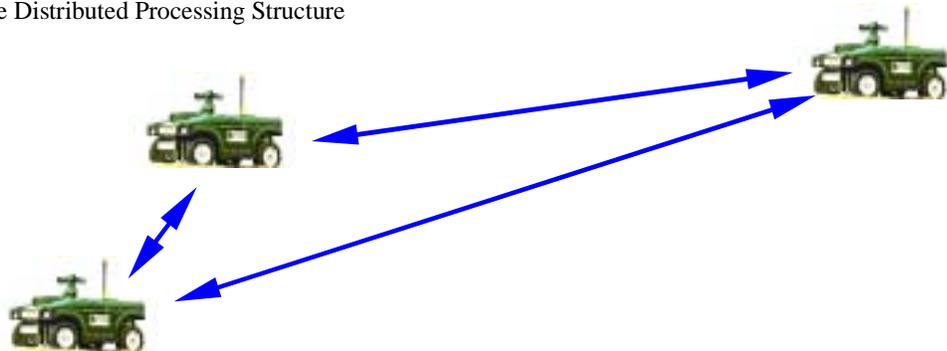
Figure 7.1 Comparison of two data processing structures

In the second structure, every moving platform is equipped with a data processor to run MultiKin. It broadcasts its own measurement, receives other measurements and processes data on a local computer. For simplification, this system is called the distributed processing structure (system) in the sequel. However, this is different from the usual distributed processing system in a computer network, because the processing procedure on each platform is independent and does not need to cooperate with the computation procedure performed on other platforms.

A central station can be equipped with a high-quality data processor, which allows MultiKin to process a large number of moving platforms. Thus, the average cost of data processing can be lowered. However, if a moving platform needs to know the relative position of others, it has to acquire the processing results from the central station via a wireless data link.

Another problem for a central processing system is its reliability. If the central station crashes, the entire system ceases operation because the surrounding platforms cannot obtain the position data. A common solution to this problem is to have a backup system. When the main processor is out of service, the backup system can at once take over its job to keep the system running.

For a distributed processing structure, each platform must have a data processing capability. If the platform is equipped with a high-performance processor, the implementation costs will increase considerably. However, if the platform is equipped with a low-cost processor, the limited data processing capability can cause difficulty in processing all the data from surrounding platforms. In addition, the data processed on each platform is the same, and the processing procedure is just repeated from one platform to another. The major advantage of the distributed processing structure is that no processed results need to be transmitted.

The reliability problem also exists in the distributed processing structure. Although the breakdown of the data processor on a moving platform does not affect other data processing units, the faulty platform does not know the relative position of the others. In some applications, this can be very dangerous. A possible solution to this problem is to

allow a faulty platform to acquire position data from the other platforms via the wireless data link. Because the surrounding platforms all process the same data, they work like a multiple-backup system for each other. The difficulty in this solution is an increase in complexity of the system, especially the communication system. A very complicated communication protocol must be designed to let the faulty platform know the number of nearby platforms and their status, and then to select a nearby platform to acquire processing results.

The realization of both data processing structures depends on the data link. In the following section, a detailed discussion about the requirement and options of the data link is presented.

7.3   DATA LINK

A data link is necessary for data exchange between platforms. Because of the high mobility of the platforms, a wireless data link must be used. Before the discussion of the selection of a data link, the requirement of MultiKin for a data link is first studied.

7.3.1   Capability Requirement of a Data Link

For the sake of discussion, a 12-channel GPS receiver is assumed as the raw data collector. It usually outputs measurement from at most 11 channels and the other channel is used for satellite searching. Each channel outputs C/A code pseudorange, Doppler, L1 and L2 carrier phase, satellite number and GPS time. For time and measurements, a double-precision (64 bits) float number is usually required. For the satellite number, eight bits are enough. Therefore, the maximum number of bits required for representing one epoch of data is as follows:

$$N_u = 11 \times (64 \times 4 + 8) + 64 = 2968(bits)$$

Number of channels

Satellite Number

GPS Time

C/A code, Doppler,
L1, L2 carrier phase

Considering the extra bandwidth for error control, signal synchronization and signaling control, the real number of transmitted bits can be doubled. If the GPS data rate is one Hz, the required speed for real-time transmitting data of one moving platform should reach 6.0 *kbps* (kilobits per second), denoted by r_{upload}.

If the data processing results is broadcast by a central station or transmitted to a breakdown platform, this will take up additional channel resources. Assume that the broadcast data contains the GPS time, 3D relative user position and identification number of the corresponding moving platform. A double-precision float number is used to represent time and position, and eight bits are used for the platform identification number. Then the total number of bits for representing one-platform results is

$$N_d = 64 \times 4 + 8 = 264(bits)$$
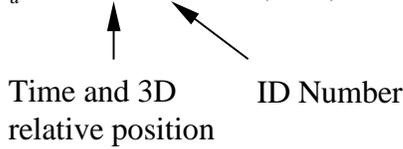
Time and 3D
relative position   ID Number

Considering the extra bandwidth for error control, signal synchronization and signaling control, the practical capability requirement for transmitting the results of one platform reaches 600 *bps*, denoted by $r_{download}$.

Now the total required capability of an upload and a download data link can be estimated using the following equations:

$$R_{upload} = Nr_{upload} = 6N(kbps) \qquad (7.1)$$

$$R_{download} = Nr_{download} = 0.6N(kbps)$$

where N is the number of moving platforms.

7.3.2   Type of Data Link

Usually, there are two types of wireless channels: broadcast channels (contention channels) and contention-free channels.

In the broadcast channel, when a station is transmitting data, all other stations can "hear" the data. The transmitting station places the destination address into the transmitted message to designate the recipient. The major problem of using this channel is the effectiveness of the protocol of channel accessing. Because the transmitting channel is unique, when several stations try to transmit data at the same time, the transmitted messages will collide with each other. This will result in a transmission failure. Since the 1970s when the first convention protocol for broadcast channel, ALOHA, was adopted on the campus network at the University of Hawaii, many multiple access protocols have been developed (Tanenbaum, 1996), such as slot-ALOHA (S-ALOHA), Carrier Sensing Multiple Access (CSMA), and CSMA with Collision Detection (CSMA/CD). Now the data throughput has been much improved.

Another problem of a broadcast channel is its bandwidth. To guarantee real-time data processing, all the moving platforms must transmit their GPS data within one processing epoch (usually, it is the inverse of the GPS data rate). If the number of moving platforms is very large, e.g., N=50, according to the Equation (7.1), the upload broadcast channel must have a bandwidth of 300 *kbps*. In other words, each moving platform must be equipped with a wireless modem with a speed of 300 *kbps*. This is not a realistic requirement for a wireless data link over the distances required by the current application.

For a contention-free channel, Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA) or Code Division Multiple Access (CDMA) technique are usually used to partition a wide-band channel into several sub-channels. One sub-channel is only used for communication between two end-users. Others cannot interfere with or even listen to their communication. When using this type of data link for uploading data, the capability of a sub-channel only needs to meet the requirement of r_{upload}, and will not be affected by the number of moving platforms. Table 7.1 lists the capability requirements of a data link for case of different processing structures.

Table 7.1 Comparison of data links for different data processing structures

	Broadcast Channel			
	Upload		Download	
	Number of channel required	Capability of each channel	Number of channel required	Capability of each channel
Central	1	$N*6kbps$	1	$N*0.6kbps$
Distributed	1	$N*6\ kbps$	$0^{[1]}$	
	Contention-free Channel			
Central	N	$6kbps$	N	$N*0.6kbps$
Distributed	$N(N-1)/2$	$6kbps$	$0^{[2]}$	

[1] If the result transmission function to the faulty platform is required, it is the same as the central processing structure.

[2] If the result transmission function to the faulty platform is required, the number of channel depends on the number of breakdown platforms. The capability of a single channel is the same as the central processing structure.

7.4 SUGGESTED IMPLEMENTATION OF MULTIKIN SYSTEM

The distributed processing structure is not recommended herein. Although it saves the procedure of transmitting data results, it largely increases the costs of data processing. In addition, it cannot use the contention-free channel to transmit data to other platforms because too many sub-channels must be taken, see Table 7.1. However, if a broadcast channel for data transmission is used, when the number of platforms increases, the bandwidth and communication equipment requirements become critical. Therefore, a central processing system is suggested herein for implementation of MultiKin.

The unit cost of procuring a high-quality data processor decreases as the number of moving platforms increases. Therefore, in a central processing system, the major problem is how to optimally select the wireless data link. From the comparison in Table 7.1, it can

conclude that the best solution maybe to use the contention-free channel for data uploading and the broadcast channel for data downloading.

Firstly, when a platform uploads data to the central platform, it only needs a low-speed wireless modem (about 6.0 *kbps*) to transmit the data. The speed of the modem has nothing to do with the number of platforms in the configuration. Admittedly, the receiver in the central station must have the capability to receive data from all of the moving platforms, because the total input data rate is $N*6.0$ *kbps*. However, this requirement is relatively easier to meet than equipping all of the moving platforms with high speed transceivers.

Secondly, the broadcast channel should be used instead of a contention-free channel because when the contention-free channel is used, all of the channels are transmitting the same data results. This is not cost-effective when the number of moving platforms is large. Since only the central station uses the broadcast channel to transmit data results, no collision control needs to be considered in the communication protocol. Thus, the communication system can be substantially simplified.

A very important consideration for implementation is the number of moving platforms, i.e., the scale of the configuration. This is affected by the following elements:

[1] Effectiveness of MultiKin

[2] Processing capability of MultiKin

[3] Capability of data processor

[4] Capability of data link

As shown in Chapter 6, the improvement rate diminishes with the increase in the number of platforms. A 10-platform constraint has almost the same performance as a four-platform constraint. Thus, enlarging the scale of configuration does not generate a corresponding increase in effectiveness.

The current MultiKin software can easily be upgraded to support processing of any number of moving platforms. It is only limited by the capability of the data processor. Since the computer used herein can process data from 50 platforms in one second, it is believed that it is possible to process GPS data from 100 platforms in real time using a computer with a higher speed CPU and a larger memory. Thus, the only remaining limitation of configuration scale is the capability of the data link.

This limitation does not exist in the upload data link because a moving platform only needs a 6.0 *kbps* wireless modem for data transmission. This requirement can be easily satisfied because many applied wireless modems can support this data rate. For instance, the GSM modem can support 9.6*kbps* data transmission and the CDPD modem (Cellular Digital Packet Data) can support 19.2 *kbps.* Therefore, the real limitation can only come from the download data link.

For broadcasting data over distances of 10 km with a high speed, the proper frequency bands are VHF and UHF. However, most of the VHF and UHF modems can only support a transmission rate of up to 9600 *bps*, i.e., it can only broadcast positions from 16 platforms in one second.

There are several ways to increase the configuration scale without increasing the capacity of the data link. First, if the position transmission rate can be lowered, then more time can be used to broadcast additional position data. For instance, if GPS data is 0.5 Hz, the data from 32 platforms can be broadcast in real-time with a 9600 bps modem. Second, a data compression technique can be used to reduce the transmitted bit number. If the functional distance is limited to 10 km, then a 25-bit number is accurate enough to represent a distance at the millimetre level. Assuming the required time accuracy is at the level of tenths of seconds, a 20-bit number is accurate enough to describe the GPS time. Therefore, the total required number of bits for one-platform data can be compressed to N_d' :

$$N_d' = 25 \times 3 + 20 + 8 = 103(\text{ bits })$$

3D relative position   GPS time   User ID

Considering the extra bits for communication, the practical number of transmitted bits can be doubled to about 240. In this case, a modem with a speed of 9600 *bps* can transmit position data from 40 platforms in one second. According to the above analysis, 20 to 30 moving platforms can be included simultaneously.

In some applications, it is not necessary for a moving platform to know the relative position of the other platforms. The relative position data is only used for the central station to monitor and control the moving platforms. In this situation, the download data link is completely unnecessary. Without this limitation, the number of moving platforms that can be used simultaneously increases very significantly.

The final consideration for implementing MultiKin is the functional distance, which is limited by the functional distance of the broadcast radio and the required positioning precision. To increase the distance of the download link, the power of the transmitter at the central station must be increased. However, it is not necessary to increase the transmitting power of the moving platforms when the distance from a moving platform to a central station is increased, because some ground stations can be set up to relay the signal from a moving platform to the central station. From the test results described in Chapter 6, it can be seen that centimetre level positioning accuracy can be achieved for baselines with lengths of 20 to 30 km when the ionospheric activity is average. If the ionosphere is active, the longest solvable baseline for L1 is less than 10 km. Therefore, if high accuracy is emphasized, then the maximum functional distance suggested is 10 km for L1 and up to a few tens of kilometers for WL observables.

8 CONCLUSIONS AND RECOMMENDATIONS

8.1 GPS SOFTWARE SIMULATOR

The GPS software simulator developed herein was found to be very effective to fully evaluate the performance of MultiKin. The following six major GPS errors were simulated: ionospheric error, tropospheric error, orbital error, SA, multipath and receiver noise. Variation of the error model parameters allowed the generation of a wide-range of testing scenarios. For atmospheric error simulation, high spatial resolutions and temporal variations were emphasized. Given that multipath is an important error source for short baseline applications, two simulation models were proposed for the static and kinematic case, respectively. This proved necessary and effective due to the different multipath behavior for each case and the resulting impact on ambiguity resolution. The validity of the error models was verified by a satisfactory degree of consistency between the simulation and field test results.

8.2 MULTIKIN PERFORMANCE

From the simulation and field test results, the following conclusions can be made:

[1] Delaunay triangulation is very effective in constructing ambiguity constraints. It is the first time that a detailed discussion about optimally selecting constraints in a multi-platform configuration was presented. The resulting constraints can reduce the ambiguity resolution time by up to 66.9%. Furthermore, compared with the full constraint selection method, MultiKin only has a very small efficiency degradation (<1.3%), but is much more efficient from a computational aspect because the numbers of selected baselines and triangles only grow linearly with the number of platforms.

[2] The ratio test is much more effective in detecting wrong fixes than the residual test. When the error is relatively small, the ratio test can detect wrong fixes much faster than the residual test (118.4 s vs. 427.6 s). When the double difference error becomes larger, the ratio test not only maintains good performance in detecting wrong fixes but

also generates fewer false alarms than the residual test (0 vs. 4.2%). This is because *a priori* information about the absolute GPS errors is not necessary in the ratio test; thus, it can adapt better to a wide-range of testing conditions than the residual test.

[3] MultiKin is very effective in improving the efficiency and reliability of OTF ambiguity resolution. Compared with the single baseline method, MultiKin can

- fix ambiguities faster (*TSR* is from 13.1% to 66.9),
- fix more ambiguities in cases of large differential errors,
- fix ambiguities over longer distances, and
- detect wrong fixes much faster.

The degree of improvement depends on three elements, namely the number of constrained platforms, the magnitude of double difference errors, and satellite visibility.

Increasing the number of constrained platforms can further increase efficiency; however, the improvement rate diminishes with an increasing number of platforms. The 10-platform constraint has almost the same improvement as the four-platform constraint. This is because the baselines that are not in the same Delaunay triangle can only provide indirect benefits, and the benefits degrade rapidly with the separation of the baselines.

The magnitude of differential errors has a major impact on the efficiency of MultiKin. As the error magnitude increases, the efficiency improvement decreases. This effect has been repeatedly observed for all other GPS errors, such as the ionospheric error, the orbital error and the tropospheric error.

Satellite visibility is also very important for the efficiency improvement introduced by MultiKin. Poor satellite visibility decreases the observation redundancy and hence degrades the efficiency of MultiKin. The improvement in the case of a reduced geometry is only half of the improvement in the case of a full geometry (23.3% vs. 45.7%).

[4] Accuracy degradation in relative positions is proportional to the baseline length and the reference errors. Switching off SA greatly improves GPS stand-alone positioning accuracy (<10 m, horizontal) and therefore reduces the accuracy degradation. When baseline lengths are limited to within 10 km, the accuracy degradation can now be neglected.

[5] The use of the widelane observables or the float ambiguity solutions results in relatively poorer positioning accuracy. These can still provide a 10-cm level accuracy for 10-km baselines.

8.3 RECOMMENDATIONS

8.3.1 Additional Field Tests

Although MultiKin has been tested using many simulated scenarios, it is still necessary to conduct additional field tests to fully understand the performance and limitation of the method. This is mainly because the simulated errors often behave differently from real GPS errors and the real testing scenarios can be too complicated to be re-created in simulation. In addition, some parameters in MultiKin, such as the thresholds in the sufficiency test are set up according to the results of the simulation tests. When using the method with real data, these parameters may not be optimal. Thus, additional field tests would be helpful for adjusting these parameters for practical applications.

8.3.2 Development of More Sophisticated Error Models

Some problems remain in the models developed for the GPS software simulator. First, the ionospheric model is a static model, i.e., the TEC value of a grid point in the Sun-fixed frame is time-invariant. This assumption will not cause problems in kinematic applications, because the long-term properties of the ionosphere are not used in the simulations. However, for static applications, if a long-term test is conducted, e.g., simulation for deformation monitoring, the daily repeatability of ionospheric errors can be observed, which is not entirely true for real observations. Therefore, more work can be performed to induce short-term and long-term variations of the ionosphere into the combined model. The same problem exists for the tropospheric error simulation, because

only diurnal changes of temperature and relative humidity are simulated. In addition, to better simulate tropospheric errors, statistical tests of global and regional meteorological data should be performed to obtain more information on time and spatial variations of tropospheric errors.

Currently, the simulated GPS constellation is based on a 24-satellite scheme, whereas the real GPS constellation now contains more satellites, and their positions are different from the original design. It is suggested that the real ephemeris or almanac be used in the constellation generator to give a more realistic reproduction of the satellite geometry.

8.3.3 Optimization of MultiKin

8.3.3.1 Integrated Data Processing

Currently, the improvement of OTF ambiguity resolution in MultiKin results from the use of multiple triangular constraints. The observation data of each baseline is still processed independently. For future work, the integrated multiple platform data processing is recommended as the correlation between baselines could be taken into account. This may further improve the efficiency and reliability of ambiguity resolution. Correspondingly, the complexity of the data processing will be largely increased with an increasing number of platforms.

8.3.3.2 Dynamic Processing of MultiKin

MultiKin is tested based on a quasi-static configuration so the Delaunay triangulation is actually time-invariant in simulation tests. However, in real applications, the configuration can vary dramatically, through changes in baseline length, introduction of new platforms, or removal of some platforms. The reliable and smooth transition from one configuration to the next will significantly increase the data processing complexity. However, algorithms for processing dynamic configurations must be included in the MultiKin software.

REFERENCES

Al-Haifi, Y. M., S. J. Corbett and P. A. Cross (1998) Performance Evaluation of GPS Single-Epoch On-the-Fly Ambiguity Resolution. *NAVIGATION: Journal of the Institute of Navigation*, Vol. 44, No. 4: 479-487

Avis, D. and B. K. Bhattcharya (1983) Algorithms for Computing ℓ-Dimensional Voronio Diagrams and Their Duals. *Advances in Computing Research*, Editor: F. P. Preparata, Vol. I, JAI Press Inc. Greenwich CT.

Bern, M. and D. Eppstein (1992) Mesh Generation and Optimal Triangulation, F.K. Hwang and D. Z. Du, editors, *Computing in Euclidean Geometry.* World Scientific

Black, H.D. and A. Eisner (1984) Correcting Satellite Doppler Data for Tropospheric Effects. *Journal of Geophysical Research*, Vol. 89, No. D2: 2616-2626

Borge, T. and B. Forssell (1994), A New Real-Time Ambiguity Resolution Strategy Based on Polynomial Identification. *Proceedings of the 1994 International Symposium on Kinematic Systems in Geodesy, Geomatics, and Navigation* (*KIS-94*), Department of Geomatics Engineering, the University of Calgary: 233-240

Braasch, M. S., A. Fink, and K. Duffus (1993) Improved Modeling of GPS Selective Availability. *Proceedings of ION National Technical Meeting*, the Institute of Navigation, Alexandria, VA: 121-130

Chao, C. C. (1974) The Tropospheric Calibration Model for Mariner Mars, 1971, *JPL TR32-1587*, Jet Propulsion Laboratory, Pasadena, CA

Chen, D. (1993) Fast Ambiguity Search Filter (FASF): A Novel Concept for GPS Ambiguity Resolution. *Proceedings of ION GPS-93*, the Institute of Navigation, Alexandria, VA: 781-787

Chen, D. and G. Lachapelle (1995) A Comparison of the FASF and Least-Squares Search Algorithms for On-the-Fly Ambiguity Resolution. *NAVIGATION: Journal of the Institute of Navigation*, Vol. 42, No. 2: 371-390

Christie, J. R., P. Ko, A. Hansen, D. Dai, S. Pullen, B. S. Pervan, B. W. Parkinson (1999) The Effects of Local Ionospheric Decorrelation on LAAS: Theory and Experimental Results. *Proceedings of ION National Technical Meeting*, the Institute of Navigation, Alexandria, VA: 769-777

Collin, R. E. (1985) *Antennas and Radiowave Propagation*, McGraw-Hill Book Company, Toronto

Collins, J.P., R.B. Langley (1999), Normial and Extreme Error Performance of the UNB3 Tropospheric Delay Model. *Technical Report 204*, Geodesy and Geomatics Engineering, University of New Brunswick

Counselman, C.C. and S. A. Gourevitch (1981) Miniature Interferometer Terminals for Earth Surveying: Ambiguity and Multipath for the Global Positioning System. *IEEE Trans. on Geoscience and Remote Sensing*, Vol.19, No.4: 244-252

Dijkstra, E.W. (1959) A Note on Two Problems in Connection With Graphs, *Numerische Mathematik* Vol. 1: 269-271

Draganov, A., T. Cashin and J. Murray (1996) An Ionospheric Correction Algorithm for WAAS and Initial Test Results, *Proceedings of ION GPS-96*, the Institute of Navigation, Alexandria, VA: 789-797

Dvais, J.L., T. A. Herring, I. I. Shapiro, A. E. E. Rogers and G. Elgered (1985) Geodesy by Radio Interferometry: Effects of Atmospheric Modeling Errors on Estimates of Baseline Length. *Radio Science*, Vol. 20, No. 6: 1593-1607

El-Arini, M.B. Conker, T. Albertson, J.K. Reegan, J.A. Klobuchar and P. Doherty (1994) Comparison of Real-Time Ionospheric Algorithms for a GPS Wide-Area Augmentation System (WAAS). *NAVIGATION: Journal of the Institute of Navigation*, Vol. 41, No 4: 393-413

Enge, P. and A. J. Van Dierendonck (1996) *Global Positioning System: Theory and Applications*, Volume II, Chapter 4: Wide Area Augmentation System, American Institute of Aeronautics and Astronautics. Inc: 117-142

Erickson, C. (1992) An Analysis of Ambiguity Resolution Techniques for Rapid Static GPS Surveys Using Single Frequency Data, *Proceedings ION GPS-92*, the Institute of Navigation, Alexandria, VA: 453-462

FAA (1994) Specification for the Wide Area Augmentation System, FAA-E-2892

FAA (1997) Specification for the Wide Area Augmentation System, FAA-E-2892C (draft)

FLYKIN Web, University Technologies International Inc. website http://www.uti.ca/133.htm

Fortune, S. (1987) A Sweepline Algorithm for Voronoi Diagrams, *Algorithmica*, Vol. 2, No. 2: 153-174

Frei, E. and G. Beutler (1990), Rapid Static Positioning Based on the Fast Ambiguity Resolution Approach "FARA": Theory and First Results. *Manuscripta Geodaetica*, Vol. 15: 325-356

Georgiadou, Y and A. Kleusberg (1988) On Carrier Signal Multipath Effects in Relative GPS Positioning. *Manuscripta Geodaetica*, Vol. 13, No. 3: 172-179.

Goldhirsh, J. and W. J. Vogel (1989) Mobile Satellite System Fade Statistics for Shadowing and Multipath from Roadside Trace at UHF and L-Band. *IEEE Transactions on Antennas and Propagation*, Vol. 37, No. 4: 489-498

Han, S. and C. Rizos (1999) The Impact of Two Additional Civilian GPS Frequencies on Ambiguity Resolution Strategies, *Proceedings of ION Annual Meeting*, the Institute of Navigation, Alexandria, VA: 315-321

Hansen, A., T. Walter and P. Enge (1997) Ionospheric Correction Using Tomography, *Proceedings of ION GPS-97*, the Institute of Navigation, Alexandria, VA: 249-257

Hatch, R. (1990) Instantaneous Ambiguity Resolution, Kinematic Systems in Geodesy, Surveying and Remote Sensing. *IAG Symposia* 107, Springer-Verlag: 299-308

Haykin, S. (1996) *Adaptive Filter Theory*. Prentice-Hall.

Heppe, S. (2000) *Merging Navigation and Communications*, ION GPS-2000 Tutorial, the Institute of Navigation, Alexandria, VA

Hitschfeld, N. (1993) *Grid Generation for Three-Dimensional Non-Rectangular Semiconductor Devices*, Hartung-Gorre Press

Hopfield, H.S. (1969) Two-Quartic Tropospheric Refractivity Profile for Correcting Satellite Data. *Journal of Geophysical Research*, Vol. 74, No. 18: 4487-4499

JPL website, http://igscb.jpl.nasa.gov/

Kee, C. (1996) *Global Positioning System: Theory and Applications*, Volume II, Chapter 3: Wide Area Differential GPS, American Institute of Aeronautics and Astronautics. Inc.: 81-116

Kendall, M. G. and A. Stuart (1968) *The Advanced Theory of Statistics*, Griffin, London

Keong, J. (1999) Determining Heading and Pitch Using a Single Difference GPS/GLONASS Approach, Master Thesis. *UCGE Report 20134*, Geomatics Engineering, The University of Calgary

Klobuchar, J. A. (1996), *Global Positioning System: Theory and Applications*, Volume I, Chapter 12: Ionospheric Effects on GPS, American Institute of Aeronautics and Astronautics. Inc.: 485-516

Klobuchar, J. A., (1987) Ionospheric Time-Delay Algorithm for Single-Frequency GPS Users. *IEEE Transactions on Aerospace and Electronic System*, Vol. 23, No. 3: 332-338

Komjathy, A. and R. Langley (1996) An Assessment of Predicted and Measured Ionospheric Total Electron Content Using a Regional GPS Network. *Proceedings of ION National Technical Meeting*, the Institute of Navigation, Alexandria, VA: 615-624

Kumar, R. and P. Munjal (1998) Simulation Studies for Ionospheric Scintillation. *Proceedings of ION Annual Meeting*, the Institute of Navigation, Alexandria, VA: 303-315

Lachapelle, G. (1997) *ENGO 625 GPS Theory and Applications*. Geomatics Engineering, The University of Calgary

Lachapelle, G., C. Liu, and G. Lu (1993) Quadruple Single Frequency Receiver System for Ambiguity Resolution on the Fly. *Proceeding of ION GPS-93*, the Institute of Navigation, Alexandria, VA: 1167-1172

Lachapelle, G., H. Sun, M. E. Cannon, G. Lu (1994) Precise Aircraft-to Aircraft Positioning Using a Multiple Receiver Configuration. *Canadian Aeronautics and Space Journal*, Vol. 40, No. 2: 793-799

Lachapelle, G., M. E. Cannon, G. Lu (1992), High Precision GPS Navigation with Emphasis on Carrier Phase Ambiguity Resolution. *Marine Geodesy*, Vol.15, No.4: 253-269

Landau, H. and H. Euler (1992) On the Fly Ambiguity Resolution for Precise Differential Positioning. *Proceedings of ION GPS-92*, the Institute of Navigation, Alexandria, VA: 607-613

Lathi, B. P. (1992) *Linear Systems and Signals*. Berkeley-Cambridge Press, Carmichael, California

Lawson, C. L. (1977) Software for C^1 Surface Interpolation. Editor: J. R. Rice, *Mathematical Software,* Academic Press, New York, Vol. III: 161-194

Lear, W., M. Montez, L. Rater, and L. Zyla (1992) The Effect of Selective Availability on Orbit Space Vehicles Equipped with SPS GPS Receivers, *Proceedings of ION GPS-92*, the Institute of Navigation, Alexandria, VA: 825-840

Leick, A. (1995) *GPS Satellite Surveying, 2nd Edition*. John Wiley and Sons, Inc.

Lu, G. (1995) Development of a GPS Multi-antenna System for Attitude Determination. Pd. D. Thesis, *UCGE Report 20073*, Geomatics Engineering, The University of Calgary

Luo, N., G. Lachapelle (1999) Centimeter-level Relative Positioning of Distributed Moving Platforms Using GPS Carrier Phase Observables. *Proceedings of ION Annual Meeting*, the Institute of Navigation, Alexandria, VA: 307-313

Luo, N. (2000) Centimeter-level Relative Positioning of Multiple Moving Platforms Using Ambiguity Constraints. *Proceedings of ION GPS-2000*, the Institute of Navigation, Alexandria, VA, in press

Madar, G. L. (1990) Ambiguity Function Techniques for GPS Phase Initialization and Kinematic Solutions. *Proceedings of Second International Symposium on Precise Positioning System*, The Canadian Institute of Surveying and Mapping, Ottawa: 1233-1247

Mannucci, A.J., B.A. Iijima and B. D. Wilson (1997) Wide Area Ionospheric Delay Corrections under Ionospheric Storm Conditions. *Proceedings of ION National Technical Meeting*, the Institute of Navigation, Alexandria, VA: 871-881

Mannucci, A.J., B. D. Wilson and C. Edwards (1993) A New Method for Monitoring the Earth's Ionospheric Total Electron Content Using the GPS Global Network. *Proceedings of ION GPS-93*, the Institute of Navigation, Alexandria, VA: 1323-1332

Marini, J. W. (1972) Correction of Satellite Tracking data for an Arbitrary Tropospheric Profile. *Radio Science*, Vol. 7, No. 2: 223-231

Massatt, P. and M. Zeitzew, (1998) The GPS Constellation Design – Current and Projected. *Proceedings of ION National Technical Meeting*, the Institute of Navigation, Alexandria, VA: 435-445

Merriam, M. L. (1993) An Efficient Advancing Front Algorithm for Delaunay Triangulation. *Proceedings of AIAA 29th Aerospace Sciences Meeting*, Number 91-0792, Reno, USA

Nayak, R. (2000) Reliable and Continuous Urban Navigation Using Multiple GPS Antennas and Low Cost IMU. Master Thesis, *UCGE Report 20144*, Geomatics Engineering, The University of Calgary

Niell, A.E. (1993). A New Approach for the Hydrostatic Mapping Function. *Proceedings of the International Workshop for Reference Frame Establishment and Technical Development in Space Geodesy,* Communications Research Laboratory, Koganei, Tokyo, Japan: 61-68

Parkinson, B. W. (1996) *Global Positioning System: Theory and Applications*. Volume I, Chapter 1: Introduction and Heritage of NAVSTAR, the Global Positioning System, American Institute of Aeronautics and Astronautics. Inc.:3-28

Preparata, F. P. and M.I. Shamos, (1985) *Computational Geometry*. Springer

Rajan, V.T. (1991) Optimality of the Delaunay Triangulation in R^d. *Proceedings of 7th Annual Symposium on Computational Geometry*: 357-363

Raquet, J. F. (1997) A New Approach to GPS Carrier-Phase Ambiguity Resolution Using a Reference Receiver Network. *Proceedings of ION National Technical Meeting,* the Institute of Navigation, Alexandria, VA: 357-366.

Raquet, J.F. (1998) Development of a Method for Kinematic GPS Carrier-phase Ambiguity Resolution Using Multiple Reference Receivers. Ph.D. Thesis, *UCGE Report 20116*, Geomatics Engineering, The University of Calgary

Ray, J. K. (2000) Mitigation of GPS Code and Carrier Phase Multipath Effects Using a Multi-Antenna System. Ph.D. Thesis, *UCGE report 20136*. Geomatics Engineering, The University of Calgary

Remondi, B.W. (1991) Kinematic GPS Results Without Static Initialization, *NOAA Technical Memorandum NOS NGS-55*, Rockville, MD

Remondi, B.W. (1992) Real-time Centimeter-accuracy GPS: Initialization While in Motion (Warm Start versus Cold Start). *Proceedings of ION GPS-92*, the Institute of Navigation, Alexandria, VA: 1053-1061

Resch, G. M. (1984) Water Vapor Radiometry in Geodetic Applications. *Geodetic Refraction*, F. K. Brunner, Editor, Springer-Verlag

RTCA (1998) *Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment, RTCA/DO-229A, SC-159*. Washington, DC

Ryan, S. (2000) Multiple Blunders in Marine Application. *Proceedings of ION GPS-2000*, the Institute of Navigation, Alexandria, VA, In press.

Saastamoinen, J. (1972) Atmospheric Correction for the Troposphere and Stratosphere in Radio Ranging of Satellites. *Geophysics Monograph. Serie*s, American Geophysical Union, Washington, DC Vol. 15

Saastamoinen, J. (1973) Contribution to the theory of atmospheric refraction. *Bulletin Géodésiqué*, Vol. 107: 13-34

SATNAV (1998) SATNAV Toolbox Menu. GPSoft LLC

Schaer, S. (1997) How to Use CODE's Global Ionosphere Maps. http://www.aiub.unibe.ch/ionosphere.html, Astronomical Institute, University of Berne

Sensor Web, Sensor Systems Web Page, www.sensorantennas.com

Skone, S (1999) Wide Area Ionosphere Grid Modeling in Auroral Region. Ph.D. Thesis, *UCGE Report 20123*, Geomatics Engineering, The University of Calgary

Spilker Jr., J. J. (1996), *Global Positioning System: Theory and Applications*, Volume I, Chapter 13: Tropospheric Effects on GPS, American Institute of Aeronautics and Astronautics. Inc.: 517-546

Studenny, J. (1993) Simulation of a Second-order Gauss-Markov Process. *RTCA paper 148-93/SC159-424*

Su, P. and L.S. Drysdale (1995) A Comparison of Sequential Delaunay Triangulation Algorithms. *Proceedings of 11th Annual Symposium on Computational Geometry*, Vancouver, Canada: 61-70

Sun, H., M.E. Cannon, T.E. Melgard (1999) Real-time GPS Reference Network Carrier Phase Ambiguity Resolution. *Proceedings of ION National Technical Meeting*, the Institute of Navigation, Alexandria, VA: 193-199

Tang, C. (1996) Accuracy and Reliability of Various DGPS Approaches. Master Thesis, *UCGE Report 20095*, Geomatics Engineering, The University of Calgary

Tanenbaum, A. S. (1996) *Computer Networks 3rd Edition*. Prentice Hall

Teunissen, P.J.G. (1994) A New Method for Fast Carrier Phase Ambiguity Estimation. *Proceedings of the IEEE Position, Location, and Navigation Sympossium (IEEE PLANS-94)*, Las Vegas, Nevada: 562-573,

Van Graas, F. and M. S. Braasch (1996), *Global Positioning System: Theory and Applications*, Volume I, Chapter 17: Selective Availability, American Institute of Aeronautics and Astronautics. Inc.: 601-622

Van Nee, R.D.J. (1995) *Multipath and Multi-Transmitter Interference in Spread-Spectrum Communication and Navigation Systems*. Delft University Press

Wei, M. and K. P. Schwarz (1995) Fast Ambiguity Resolution Using An Integer Nonolinear Programming Method. *Proceedings of ION GPS-95*, the Institute of Navigation, Alexandria, VA: 1101-1110

Weisenburger, S. D. (1997) Effect of Constraints and Multiple Receivers for On-the-fly Ambiguity Resolution. Master Thesis, *UCGE Report 20109*, Geomatics Engineering, The University of Calgary

Welch, P. D. (1967) The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging over Short, Modified Periodograms. *IEEE Transactions on Audio Electroacoustics*, Vol. AU-15: 70-73

Wells, D. et al (1987) *Guide to GPS Positioning*. Canadian GPS Associates, Fredericton, New Brunswick

Zhang, J. (1999) Investigations into the Estimation of Residual Tropospheric Delays in a GPS Network. Master Thesis, *UCGE Report 20132*, Geomatics Engineering, The University of Calgary

Zumberge, J. F. and W. I. Bertiger (1996), *Global Positioning System: Theory and Applications*, Volume I, Chapter 16: Ephemeris and Clock Navigation Message Accuracy, American Institute of Aeronautics and Astronautics. Inc.: 585-600