OPTIMAL ON-AIRPORT MONITORING OF THE INTEGRITY OF GPS-BASED LANDING SYSTEMS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Gang Xie

March 2004

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Per K. Enge, Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Donald C. Cox

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Sam Pullen

Approved for the University Committee on Graduate Studies.

Abstract

The Global Positioning System (GPS) is a satellite-based radio navigation system. The Local Area Augmentation System (LAAS) is a version of Differential GPS (DGPS) designed to reliably support aircraft precision approaches. The Integrity Monitor Testbed (IMT) is a prototype of the LAAS Ground Facility (LGF) that is used to evaluate whether the LGF can meet system integrity requirements. To insure high integrity, the IMT has a variety of monitors to detect all possible failures. It also contains a failure-handling logic, known as Executive Monitoring (EXM), to exclude faulty measurements and recover once the failure disappears.

Spatial ionospheric gradients are major threats to the LAAS. These ionospheric anomalies can be modeled as sharp "wave fronts" in which a linear change in vertical ionosphere delay occurs over a short horizontal distance. One focus of this thesis is exploring methods to quickly detect ionospheric gradients given the required low probability of false alarm.

The first part of the thesis introduces GPS, LAAS, and the IMT and explains the algorithms and functionalities of IMT integrity monitors in detail. It then analyzes the failure responses of the integrity monitors under the most general measurement failure model. This analysis not only qualitatively maps the integrity monitors into the entire failure space, but also provides a tool to quantitatively compare the performance of different integrity monitors.

In addition, the analysis examines the limitations of the existing monitors in detecting small but hazardous ionospheric gradients. The divergence Cumulative Sum (CUSUM) method is then derived and assessed. It can reduce the time required to detect marginal ionospheric gradients by about 30%. With the divergence CUSUM method implemented in the IMT, system integrity and performance are greatly improved.

Different monitors can respond to the same failures. The last part of this thesis shows that the combination of these different monitors can detect certain failures more quickly than any individual monitor. This idea leads to a new method, called failure-specific testing, which can significantly improve the detection speed of small failures, including ionospheric gradients. Used as aids to diagnosis and exclusion, failure-specific tests can soften the current EXM logic so that the LAAS can meet the tighter continuity requirements that apply to Category III precision landings.

Acknowledgments

First and foremost, I would like to thank my advisor, Professor Per Enge, for offering me the opportunity to work in this exciting field. He gave me the freedom to pursue my dream and the guidance to complete it. His continuous encouragement and support in every aspect are immeasurable and paramount in accomplishing the thesis and degree.

I appreciate my co-advisor, Professor Donald Cox, for his great help in my academic development. He has served as my Ph.D. qualifying examiner, thesis defense committee member, instructor of two courses, and reader of this thesis. Without his help, I could not have finished my doctorate program so smoothly.

My special thanks go to the Stanford LAAS team leader, Dr. Sam Pullen, for his constant and tremendous assistance throughout my research. He not only generously helped me move forward but also patiently supervised every step I took. He has played a major role in this work. His corrections and comments improved this thesis significantly.

It is my great honor to have Professor John T. Gill III as my defense committee chair. His help on my career development is appreciated.

I am grateful to my former advisor, Professor Peter Levin, for introducing me to the GPS world and his continued support.

I am thankful to all my colleagues in the Stanford GPS, WAAS, and LAAS laboratories, specially the IMT research group members: Ming Luo, Dennis Akos, Jiyun Lee, Ludde Normark, Sasha Mitelman, and Guttorm Opshaug. Their help, friendship, and expertise in GPS are valuable in helping me to overcome any possible challenges. My talented friends and classmates in the Department of Electrical Engineering (EE) also extend my good fortune. I want to thank Stanford University for assembling so many wonderful people.

Thanks go to the Federal Aviation Administration Satellite Navigation Program Office for sponsoring this research. The Department of EE and the Department of Aeronautics and Astronautics (AA) also deserve thanks. Particularly, Sally Gressens (AA), Yasmeen Husain (AA), Diane Shankle (EE), Jeff Wade, Rolando Villalobos, and James Campbell have greatly assisted in making things run more smoothly for me as a Ph.D. student.

Thanks also go to Fiona Walter for proofreading this thesis. Her comments should greatly contribute to its readability.

Finally, on a more personal note, I would like to thank my parents, Zhuzhen and Yixian Xie, my parents-in-law, Suhua and Yiming Liu, and my wife, Yue Liu, for their deep love and persistent support.

Table of Contents

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Global Positioning System

1.1.1 System Overview

The Global Positioning System (GPS) is a satellite-based radio navigation system funded and operated by the United States Department of Defense (DoD). It provides instantaneous position, velocity and time (PVT) information almost anywhere on Earth at any time and in any weather. Though the system was designed for the U.S. military, today there are over twenty million GPS users worldwide [15].

GPS offers two kinds of service: the Precise Positioning Service (PPS) and the Standard Positioning Service (SPS) [42]. PPS includes a feature, called Anti-Spoofing (AS), and can be accessed only by DoD-authorized users equipped with appropriate encryption keys. However, SPS is open to all civil users. Selective Availability (SA), which intentionally degraded the SPS signals, was deactivated on May 2, 2000. With SA off, a stand-alone GPS user can typically estimate position with an accuracy of better than 10 meters and time to better than 100 nanoseconds [15].

GPS is comprised of three main components [42]:

1. *Space Segment*: The baseline GPS constellation consists of 24 satellites orbiting the Earth in almost circular orbits at an altitude of 20,200 km and a period of nearly twelve hours. Each satellite repeats the same track and crosses the same Earth-fixed point every two orbits. These space vehicles (SV) are arranged in six orbital planes with four primary satellite slots in each orbit. Keplerian elements that fully describe the motion of a GPS satellite can be found in Appendix A. Each satellite broadcasts time-tagged ranging signals and navigation data.
2. *Control Segment*: GPS is monitored and operated by the GPS Joint Program Office Operational Control Segment. Five monitor stations, distributed around the

world, continuously track the satellites in view, and transmit raw data and navigation messages to the Master Control Station (MCS) via ground and satellite links. The MCS, located in Colorado Springs, computes the ephemerides and clock parameters that are uploaded to the satellites via an S-band radio frequency link from one of several dedicated ground antennas at least once a day.

3. *User Segment*: GPS receivers track and decode the signals from the satellites. A GPS receiver computes the location of the satellites based on their ephemerides and also measures the distance to the satellites based on the travel time of the radio signals. The receiver then deduces its own location based on a simple mathematical principle called trilateration in three-dimensional space. Accurate timing is the key to measuring distance to satellites. Atomic clocks carried aboard the satellites are nearly perfect and nearly perfectly synchronized. In order to use inexpensive quartz oscillators, the receivers can utilize an extra satellite range measurement. With the distance measurements from at least four satellites, not only can the receiver calculate its position, but the receiver can also remove its clock bias.

1.1.2 Signals

Each GPS satellite transmits signals on two L-band frequencies: f_{L1} at 1575.42 MHz and f_{L2} at 1227.60 MHz [42]. The first waveform in Figure 1-1 shows the carrier f_{L1}. More frequencies are being added for both civil and military purposes [15,22]. The third frequency (L5) will be at 1176.45 MHz.

Each satellite transmits two different ranging codes: a coarse/acquisition (C/A) pseudo-random noise (PRN) code modulating the L1 carrier phase, and a precision (encrypted) [P(Y)] code modulating both the L1 and L2 carrier phases [42]. Each C/A code repeats every 1023 bits (or chips) in one millisecond, or equivalently, at a chipping rate of 1.023 Mcps. The second waveform in Figure 1-1 shows a segment of the C/A code. A P(Y)-code is an extremely long sequence (about 10^{14} chips). Each GPS satellite is assigned a unique PRN code; thus we can identify specific GPS satellites by their PRN numbers.

The auto- and cross-correlation properties of these spread spectrum codes enable GPS ranging. Specifically, the auto-correlation function of each code has only one main peak. This main peak of the auto-correlation function helps the GPS receiver acquire the GPS signals. The slope of the peak directly determines ranging precision. Additionally, the cross-correlation between any two codes has no significant peaks. The lack of correlation between different PRN codes enables the satellites to transmit on the same frequencies simultaneously without interfering with each other.

Both GPS frequencies are also modulated by binary navigation messages transmitted at a rate of 50 bits per second (bps) [42]. The third waveform in Figure 1-1 shows several navigation data bits. One message consists of five 300-bit subframes, and it contains the satellite health status, ephemerides, clock bias parameters, and almanacs. Ephemeris data describe the current SV orbits. Almanacs are the reduced-precision versions of ephemeris data for all SVs. Appendix A describes the navigation message contents in detail.

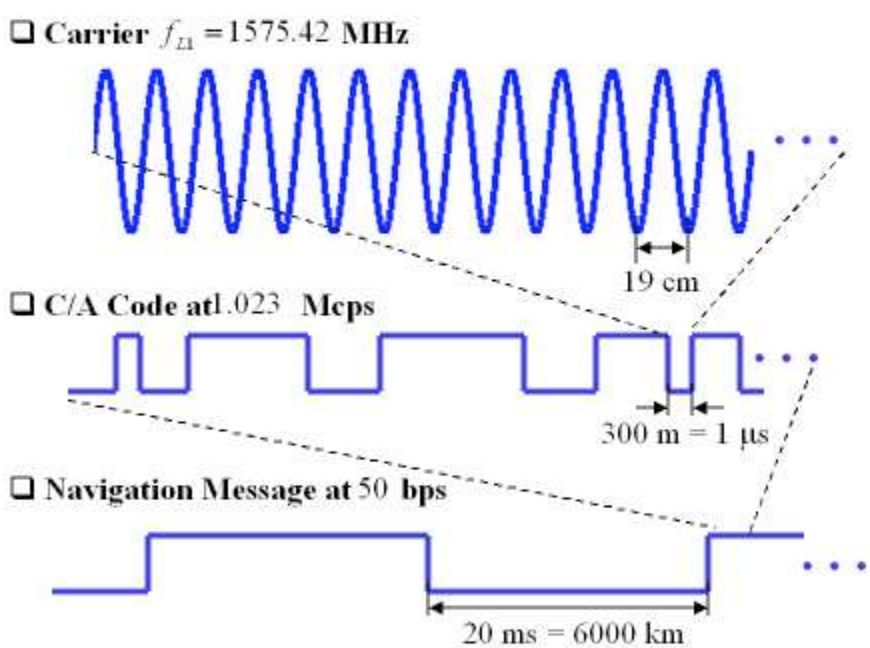

Figure 1-1 GPS Signal Structure

The structure of these three signal components, i.e., carriers, ranging codes, and navigation data, is diagrammed in Figure 1-1 [42]. A code and a message are combined using modulo-2 addition, and then the composite binary signal modulates the carrier with binary phase shift keying (BPSK). The unit "meter" is often interchanged with "second" for convenience in this thesis – the distinction between them is simply the scaling factor (c) that represents the speed of light in a vacuum (about 3×10^8 m/s).

1.1.3 Measurements and Position Estimation

GPS receivers track satellites, decode navigation messages, and produce code and carrier phase measurements for PVT determination.

The GPS signal acquisition process consists of a search for both PRN code shift and local carrier frequency offset. After the receiver "smells" a signal, the receiver continues code tracking using a delay lock loop (DLL) and phase tracking using a phase lock loop (PLL) [5,42].

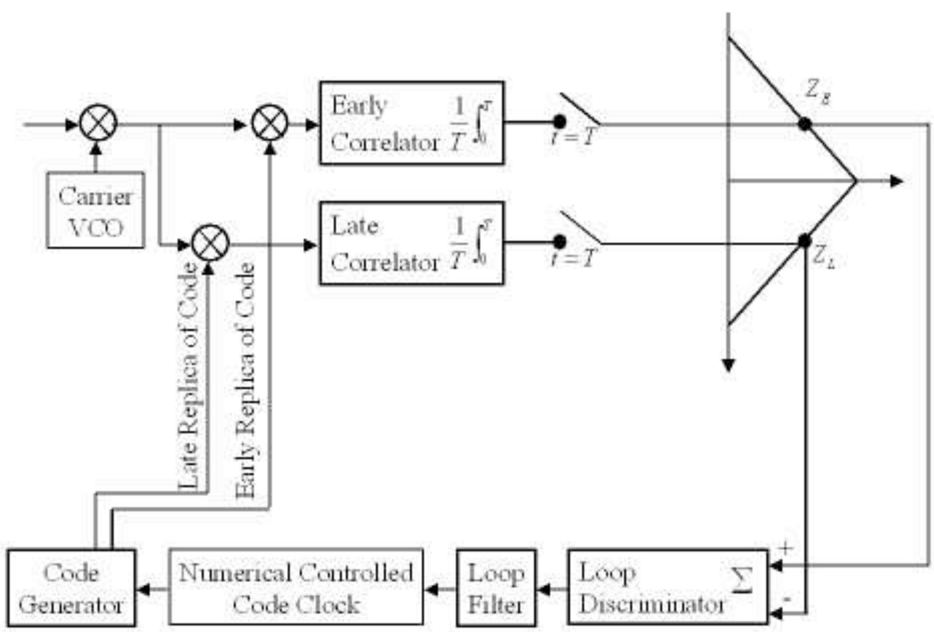

Figure 1-2 Delay Look Loop

Figure 1-2 shows the DLL structure [42]. The DLL correlates the received signal with a slightly earlier replica of the signal and a slightly later replica of the signal. If the received signal is locked, the early correlator samples the auto-correlation function peak on the rising edge, and the late correlator samples the falling edge of the peak. A discriminator function is formed by differencing the early and late correlation functions (Z_E and Z_L in Figure 1-2). The DLL maintains code lock by feeding back the output of the discriminator to the local code generator such that the discriminator output is driven to zero [5]. The difference between the signal reception time determined by the receiver clock and the transmission time "marked" on the signal is the apparent transit time of the signal from the satellite to the receiver. Pseudorange, ρ, is this measured transit time multiplied by the speed of light. It is not the true range from the satellite to the receiver since the receiver clock is not synchronized with the satellite clock.

Having removed the PRN code from the signal, the GPS receiver continues phase tracking with a phase lock loop [42]. It locally generates a carrier frequency and tries to exactly match the frequency and phase of the incoming signal. Once the carrier phase is locked, the navigation data bits can be readily extracted. The carrier phase measurement, ϕ, is the difference between the phases of the receiver-generated carrier and the carrier received from a satellite. The receiver can make out a partial cycle, but it has no idea of the number of whole cycles between the satellite and the receiver. The distance between the satellite and the receiver is equal to an unknown number of whole cycles plus the measured fractional cycle. This unknown number of whole cycles is named the "integrity ambiguity." The derivative of ϕ is the Doppler measurement that can be utilized to determine the receiver's velocity.

Code and carrier phase measurements can be modeled as [42]:

$$\rho = r + c(b_u - B^s) + I + T + M + \varepsilon_\rho \quad (1\text{-}1)$$

$$\phi = r + c(b_u - B^s) - I + T + N\lambda + \varepsilon_\phi \quad (1\text{-}2)$$

where

- ρ is measured code phase measurement, or pseudorange.
- ϕ is measured carrier phase measurement.
- r is true range from a satellite to a receiver.
- b_u is receiver (or user) clock bias. Both b_u and user position must be solved for.
- B^s is satellite clock bias. Both B^s and b_u represent the amount of time by which the satellite and receiver clocks are advanced relative to GPS Time (GPST). The satellite clock bias is modeled as a quadratic function of time, and the parameters of this model are estimated by MCS and uploaded to the satellites. They are broadcast as part of the navigation message [42].
- I is ionospheric delay.
- T is tropospheric delay.
- M is multipath error. I, T, and M will be discussed in Section 1.1.4 in detail.
- N is integer ambiguity. It is generally unknown to users, but it can be solved for using several techniques if needed [10,29]. The problem of precise relative positioning with carrier phase is actually a problem of integer ambiguity resolution [42].
- λ is carrier wavelength. For L1, $\lambda_{L1} = \dfrac{c}{f_{L1}} \approx 19$ cm.
- ε_ρ represents other code phase measurement errors.
- ε_ϕ represents other carrier phase measurement errors.

If ρ_c is the pseudorange obtained after accounting for the satellite clock offset and the other measurement errors, then Equation (1-1) can be rewritten for the "corrected" pseudorange measurements, ρ_c, as:

$$\rho_c \approx r + c \cdot b_u = \sqrt{(x^{(n)} - x)^2 + (y^{(n)} - y)^2 + (z^{(n)} - z)^2} + c \cdot b_u \qquad (1\text{-}3)$$

where the position of the nth satellite, ($x^{(n)}$, $y^{(n)}$, $z^{(n)}$), is calculated based on its ephemeris, and the user position, (x, y, z), is unknown. Together with the unknown receiver clock bias, b_u, there are four unknowns in Equation (1-3). These four unknowns can be solved for if measurements from at least four satellites in view are available at a single measurement epoch.

We can solve these equations by first linearizing them at initial estimates, then solving for corrections by the least-squares method (the system of equations is over-determined if there are more than four measurements), and finally improving the solution estimates with these corrections [42]. This process can be carried out iteratively, and the position estimates often converge quickly in two to four iterations.

1.1.4 Error Sources

Code and carrier phase measurements contain a variety of biases and errors, as indicated in Equations (1-1) and (1-2). These errors reduce the positioning accuracy. Figure 1-3 shows the major GPS measurement errors. They can be grouped basically into three sources: satellite-related, receiver-related, and propagation medium-related errors [42].

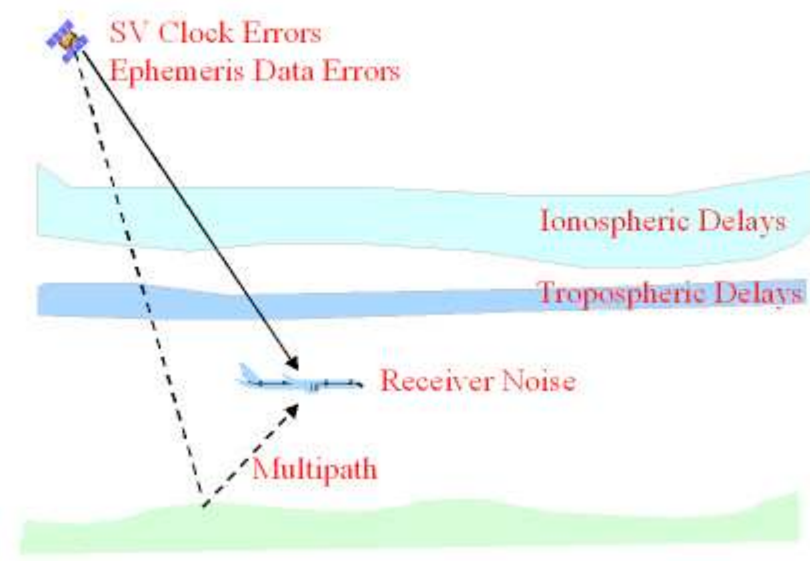

Figure 1-3 GPS Measurement Error Sources

- *Satellite-related errors*:
  1. *Satellite ephemeris*: The true position and velocity of a satellite are different from the estimates computed by the Control Segment. The ranging error due to the satellite ephemeris data error is about 1 – 2 m in the root mean square (rms) sense.
  2. *Satellite clock*: The satellite clock bias that cannot be corrected by the GPS Control Segment causes about 1 – 2 m ranging error in the rms sense.
- *Propagation medium-related errors*:
  3. *Tropospheric delay*: The troposphere includes the segment of Earth's atmosphere that is up to 8 – 13 km above sea level. Dry gases and water vapor in the troposphere can refract GPS signals. The tropospheric delay can be estimated using atmospheric models [42]. If not corrected, the resulting error is about 2 m at sea level for satellites directly overhead (it is larger for low-elevation satellites).
  4. *Ionospheric delay*: The ionosphere is a layer of ionized gases in Earth's atmosphere that lies approximately 50 km to 1000 km above the surface. The ionosphere affects GPS signal propagation by delaying code phase measurements while advancing carrier phase measurements. Opposite signs thus appear in front of the ionosphere error component, I, in Equations (1-1) and (1-2). In addition to this code-carrier divergence property, the ionosphere is also dispersive, i.e., the magnitude of the ionospheric delay is inversely proportional to the signal frequency. The ranging error, depending on the satellite elevation, is about 2 – 10 m due to the ionospheric delay. The GPS navigation data contains an ionospheric model that can remove approximately half of this delay [18].
- *Receiver-related errors*:
  5. *Multipath*: Reflected signals from surfaces near GPS antennas can interfere with the direct signals from the satellites. Because there are always reflecting surfaces in practice, multipath is hard to avoid. Using a multipath limiting antenna and placing it away from severe reflective environments can mitigate multipath [48]. Multipath affects both code and carrier phase measurements. It

causes about 1 – 5 m error in pseudorange measurements. On the other hand, only about 1 – 5 cm carrier phase measurement error is due to multipath. This is so small that it is not separately listed in Equation (1-2). Instead, ε_ϕ is assumed to contain the multipath component.

6. *Receiver noise*: Thermal noise, multi-access interference, and signal quantization noise can cause receiver error [42]. With modern microprocessor and chip technology, a GPS receiver introduces less than 0.5 m code measurement error and about 1 – 2 mm carrier phase measurement error due to receiver noise.

1.2 Differential GPS

As noted before, the positioning accuracy of stand-alone SPS users (with SA off) is about 10 m horizontally and 15 m vertically. With differential GPS (DGPS), much higher positioning accuracy can be achieved.

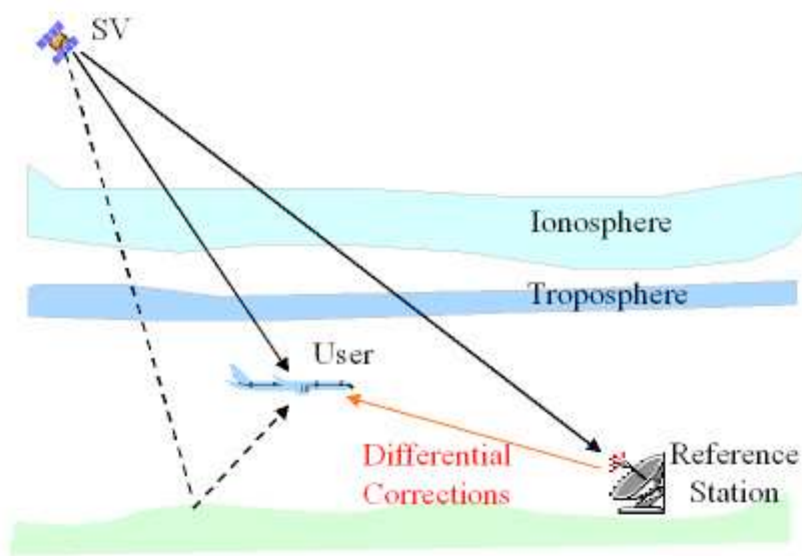

Figure 1-4 Differential GPS Configuration

DGPS takes advantage of the fact that differential users in the same region have common measurement errors in general. As shown in Figure 1-4, a reference station at a known location calculates the difference between the measured and expected pseudoranges, and sends differential corrections to users in this region via a communication link. The users can eliminate the majority of pseudorange error and achieve a higher accuracy with DGPS. Table 1-1 summarizes these errors in GPS and error mitigation in DGPS [42]. In differential mode, the user's distance from the reference receiver is assumed to be tens of kilometers and signal latency is tens of seconds.

Error Source	Error Size in GPS	Residual Error in DGPS
Satellite Clock Model	1 – 2 m (rms)	0.0 m
Satellite Ephemeris Prediction	1 – 2 m (rms)	0.1 m (rms)
Tropospheric Delay	≈ 2 m in zenith direction at sea level	0.2 m (rms) plus altitude effect
Ionospheric Delay	2 – 10 m in zenith direction	0.2 m (rms)
Multipath	Code: 1 – 5 m Carrier: 1 – 5 cm	Uncorrelated between antennas
Receiver Noise	Code: 0.5 m (rms) Carrier 1 – 2 mm (rms)	Uncorrelated between receivers

Table 1-1 A Summary of Measurement Errors in GPS and DGPS

Since the reference and user receivers are tracking signals from the same GPS satellite at the same time, satellite clock error is eliminated nearly completely in differential mode. Satellite ephemeris error can also be nearly eliminated for the same reason. Note that residual differential ephemeris error will grow as the separation between the lines of sight from the satellite to the user and reference station increases. When the user is 100 km away from the reference station, the residual range error is less than 0.05 cm [42].

DGPS can also reduce the range error due to ionospheric and tropospheric effects. Clearly, the corresponding residual differential error depends upon the spatial ionospheric and tropospheric correlation properties. When the separation between the reference and user receivers is large, the ionospheric and tropospheric propagation delays in these two signal paths can be very different.

DGPS effectively mitigates satellite clock error, satellite ephemeris error, ionospheric delay, and tropospheric delay in its service coverage. However, multipath and receiver noise are uncorrelated at the reference and user receivers, and therefore cannot be corrected by DGPS. On the contrary, the DGPS user has increased multipath and receiver noise since these errors from the reference and user receivers are combined.

1.3 Local Area Augmentation System

The U.S. Federal Aviation Administration (FAA) is developing the Local Area Augmentation System (LAAS) to support aircraft precision approaches. This local-area ground-based DGPS is responsible for generating and broadcasting carrier-smoothed code differential corrections and approach-path information to user aircrafts [81]. It also places the responsibility for detecting and alarming space-segment and ground-segment failures on the LAAS Ground Facility (LGF). The LGF must insure that all ranging sources for which LAAS corrections are broadcast are safe to use. If a failure occurs that threatens user safety, the LGF must detect and alert users (by not broadcasting corrections for the affected ranging source) within a certain time-to-alert.

Figure 1-5 describes how LAAS operates. The LGF installs four Reference Receivers (RR), RR antenna pairs, redundant Very High Frequency (VHF) Data Broadcast (VDB) equipment linked to a single VDB antenna, and equipment racks on the airport property where LAAS service is provided [79]. The LGF tracks, decodes, and monitors GPS satellite information and generates differential corrections. It also performs integrity checks on the generated corrections. The correction message, along with suitable integrity parameters and approach path information, is then broadcast to airborne users using the VDB from the ground-based transmitter. Pseudolites, which transmit GPS-like signals, may be deployed on the airport ground in order to improve satellite geometry [25,51].

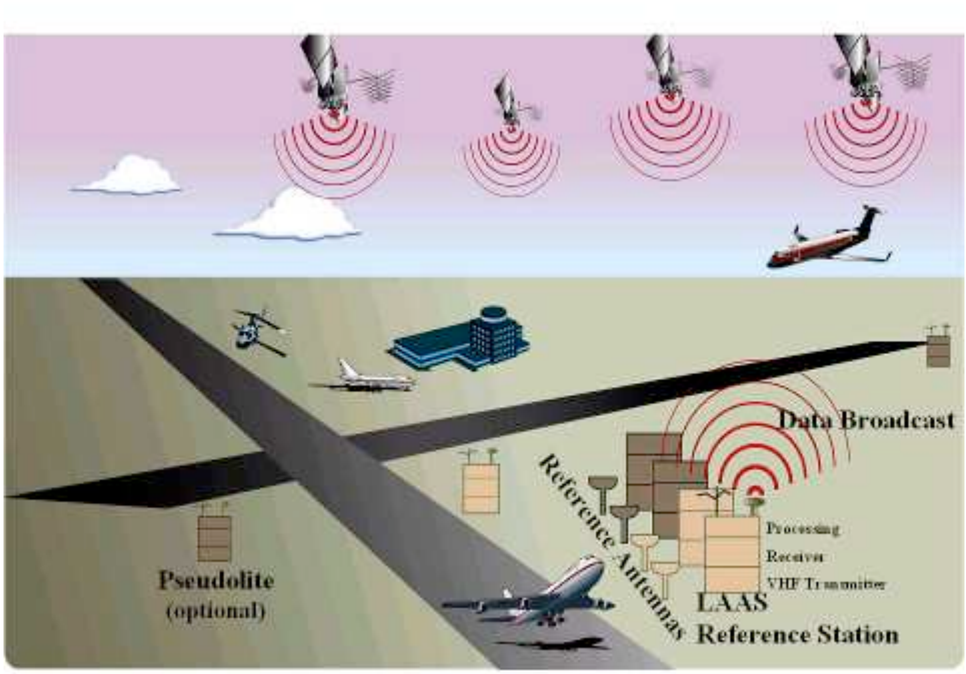

Figure 1-5 LAAS Components (Courtesy of FAA)

LAAS Precision Operation		CAT I	CAT II	CAT III
Accuracy [m] 95%	Horizontal	16.0	6.9	6.1
	Vertical	7.7	2.0	2.0
Integrity	Time-to-Alert [s]	3	2	2
	Alert Limit [m]	H: 40 V: 10-15	H: 17.3 V: 5.3	H: 15.5 V: 5.3
	P_{HMI} /Approach	2×10^{-7}	2×10^{-9}	2×10^{-9}
Continuity	Failure Rate	5×10^{-5} / approach	4×10^{-6} / 15 sec	10^{-7} / last 15 sec
Availability		0.99 - 0.99999	0.99 - 0.99999	0.99 - 0.99999

Table 1-2 Requirements for LAAS Precision Approach

The performance of all navigation systems, including LAAS, can be evaluated with respect to accuracy, integrity, continuity, and availability [16,69].

1. *Accuracy*: Measure of navigation output deviation from truth.
2. *Integrity*: Ability of a system to provide timely warnings when the system should not be used for navigation.
3. *Continuity*: Likelihood that the navigation system supports accuracy and integrity requirements for the duration of intended operation.
4. *Availability*: Percentage of time the system supports accuracy, integrity, and continuity requirements before approach is initiated.

The currently suggested requirements for LAAS precision approaches are summarized in Table 1-2 [16,19,81]. P_{HMI} is the probability of Hazardously Misleading Information (HMI). For example, if HMI causes a user vertical navigation error that exceeds a 10-meter Vertical Alert Limit (VAL), the LGF must detect the failure and alert the user within a 3-second time-to-alert (TTA) for CAT I LAAS precision approaches. Note that LAAS Categories (CAT) II and III have much stricter requirements both on integrity and continuity than LAAS CAT I.

The FAA commenced the LAAS CAT I development phase in April 2003 and expects to have this system operational in 2006. Based on results of the LAAS CAT II/III research and development efforts, the FAA will make a decision on the feasibility of awarding a full-scale development and production contract in early 2005 [78].

1.4 Integrity Monitor Testbed

1.4.1 Functions

The Stanford University Integrity Monitor Testbed (IMT) is a prototype of the LAAS Ground Facility (LGF) that is used to evaluate whether the LGF can meet the integrity requirements that apply to CAT I aircraft precision approach [33,48,49,64,67,89]. A comprehensive set of monitoring algorithms is implemented in the IMT in order to detect an array of possible failures in the GPS Space Segment, the GPS Ground Segment, and the LGF itself. Some failures may trigger more than one monitoring algorithm, so a complex failure-handling logic, called Executive Monitoring (EXM), is included in the LGF to coordinate different integrity monitors and to isolate faults. The details of the monitoring algorithms and the EXM will be discussed in Chapter 2.

1.4.2 Hardware Configuration

The LGF requires redundant DGPS reference receivers to be able to detect and exclude failures of individual receivers. Figure 1-6 shows the configuration of the three IMT antennas on the Stanford HEPL laboratory rooftop. Although limited by the space available on the rooftop, these antennas are sufficiently separated to minimize correlation between individual receiver multipath errors [30]. The current IMT employs three NovAtel OEM 4 receivers that are connected to three separated GPS Pinwheel antennas. Each receiver can track as many as twelve satellites simultaneously. A channel means one satellite tracked on one reference receiver (RR). For example, Figure 1-6 depicts a channel (RR 2, PRN 7), or simply channel (2, 7).

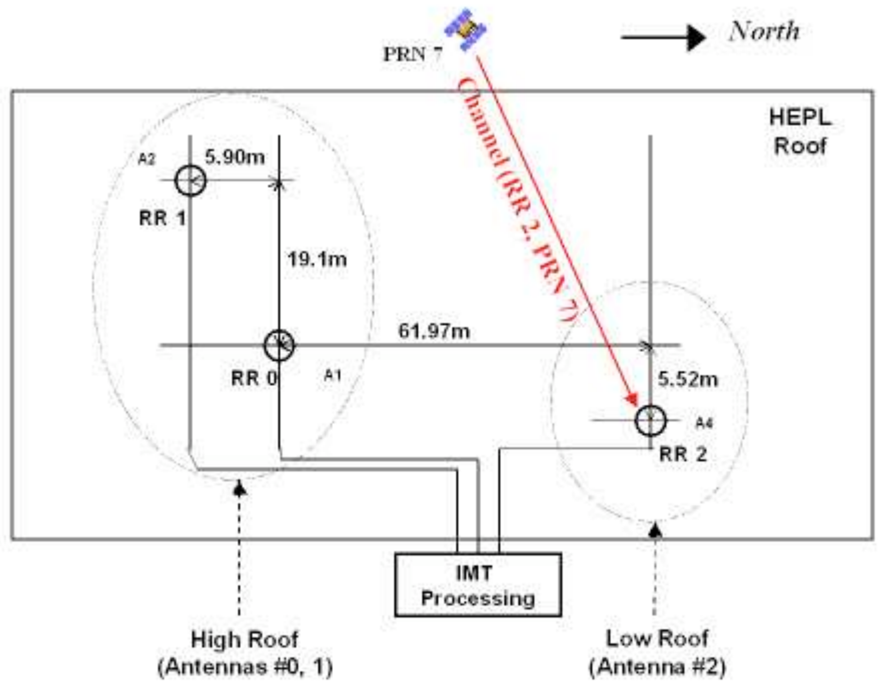

Figure 1-6 IMT Antennas Configuration

Each receiver samples GPS signals and provides receiver measurement packets every T_s seconds, where

$$T_s = 0.5 \tag{1-4}$$

in the current IMT architecture. In other words, the sampling rate is equal to two epochs per second. These GPS measurements are fed into the IMT processor for further calculations.

The Stanford IMT is configured by software in such a way that it can be used either in real-time or in post-processing. Figure 1-7 illustrates this mechanism. When Point 1 is closed, GPS receiver packets are directly fed into the IMT processing unit. This is referred to as real-time IMT operation. If Point 2 is closed, then the packets can be saved into a non-volatile storage device. Depending on whether Point A or B is closed, the saved data packets can be used to perform nominal or failure testing. This is called post-processing IMT operation. One fixed data set of receiver packets saved in storage can thus be processed repeatedly, and this makes debugging and testing easier.

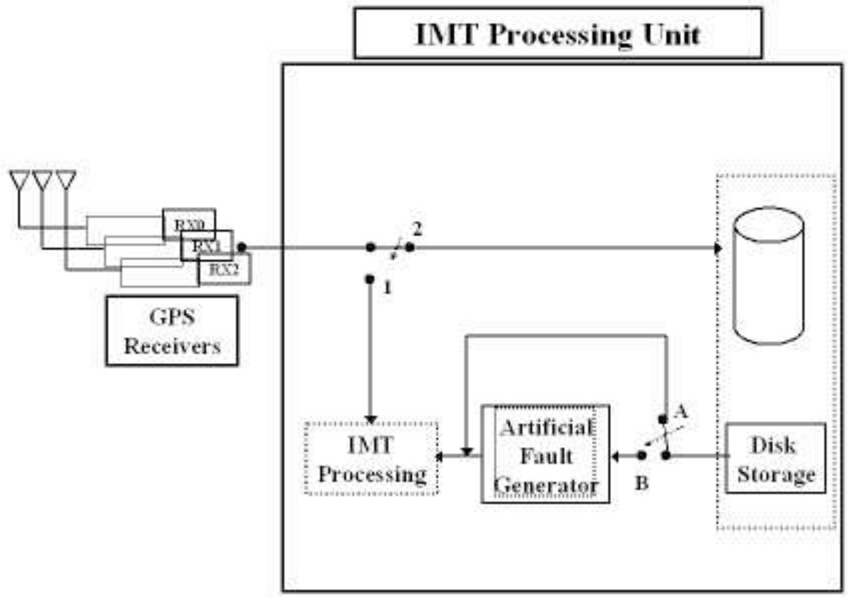

Figure 1-7 Real-Time vs. Post-Processing IMT

1.4.3 Verification Testing

LGF verification testing can be divided into two components: nominal testing and testing with simulated failures. Nominal testing is to check whether the IMT generates false alarm that endangers the system continuity. Nominal test results will be presented largely in Chapter 2. Because of the complexity of the monitoring algorithms, it is also necessary to demonstrate that the IMT can meet specified integrity requirements under a variety of possible failure situations. Failure tests will be discussed in Chapters 4 and 5, when the performance of newly designed integrity monitors is compared with that of existing ones. Additional failure test examples can be found in [28,33,34,89].

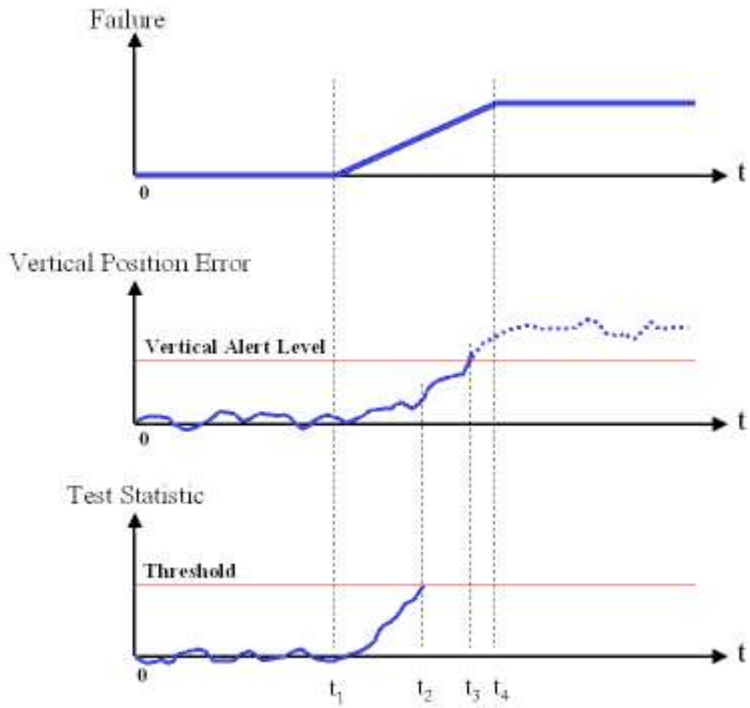

Figure 1-8 Failure and Failure Effects

To test the IMT response to failures, simulated errors are injected into the system. The top plot of Figure 1-8 shows a ramp failure commencing at time t_1. Without integrity monitors, the Vertical Position Error (VPE) of a LAAS user starts to increase at t_1 and finally exceeds a specified Vertical Alert Limit (VAL) at t_3, as illustrated in the middle plot. If the failure is not detected and the user is not alerted within a given time-to-alert, the ramp failure is said to cause Hazardously Misleading Information (HMI). The bottom plot of Figure 1-8 shows the response of an integrity monitor test statistic. The magnitude of the test statistic exceeds its threshold at t_2, and the failure is detected. The user will then be alerted not to use the failed measurements before the VPE reaches VAL. The figure demonstrates that the integrity function is to protect users from HMI. This dissertation concerns itself with the design of monitors to detect a variety of failures.

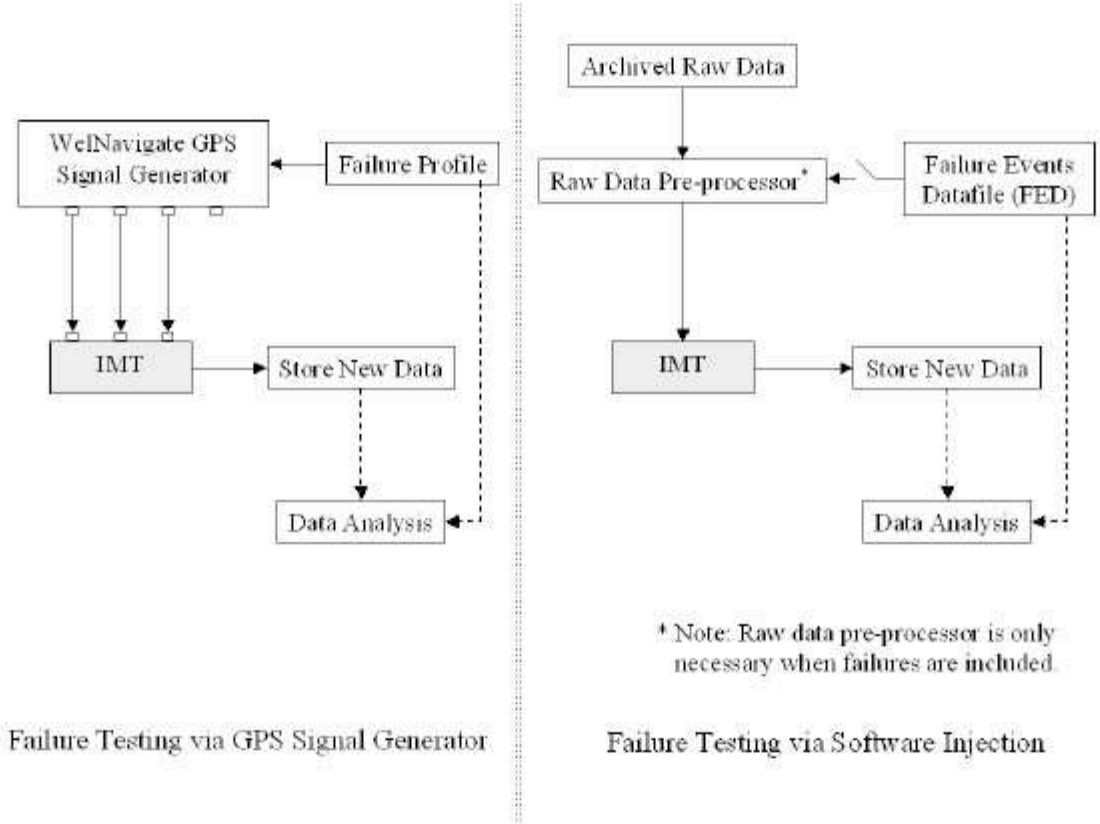

Figure 1-9 Two Failure Testing Methods

Two methods are used to inject failures into the IMT, as illustrated in Figure 1-9. One is to program a WelNavigate 40-channel GPS signal simulator to generate failed radio frequency signals that are then fed into the IMT. However, as already indicated in Figure 1-7, the post-processing IMT offers another way to inject failures. We have a program that can modify stored IMT receiver measurements collected under nominal conditions to inject failures after the fact, and then we let the IMT post-process these packets. Meanwhile, it is also possible to directly inject failures during the IMT processing by changing the IMT source code properly.

In this thesis, the post-processing version of the IMT is used to carry out all tests on a 24-hour nominal GPS data set collected on March 10, 2003.

1.5 Previous Work

The U.S. FAA has been developing the LAAS as the primary navigation system to support Category I, II and, III aircraft precision approaches. There have been many studies done on LAAS. Dr. Enge discussed the fundamental operation of the LAAS and established the connection between accuracy, integrity, and continuity [16].

In order to meet the stringent integrity requirement shown in Table 1-2, LAAS needs to implement a comprehensive set of integrity monitoring algorithms in order to detect a wide range of possible failures. Many monitoring algorithms have been designed. For example, the code/carrier integrity monitoring technique was developed in [75]. Signal quality monitoring was investigated to target deformations of the C/A-code broadcast by GPS satellites [1,61,82]. Several methods were developed to validate the GPS ephemeris and clock data for each satellite [36,51]. Besides integrity monitors in the range domain, an integrity monitor in the position domain was also studied [27]. Dr. Pullen, Dr. Pervan, and their colleagues built the original LAAS Ground Facility (LGF) prototype, known as the Integrity Monitor Testbed (IMT), and used it to evaluate whether the LGF can meet the CAT I integrity requirements. My work relies on this testbed.

The ionosphere provides the most worrisome challenge to LAAS and the LAAS integrity algorithms. Christie obtained statistical bounds for the ionospheric decorrelation effects on a LAAS architecture [9]. An ionospheric anomaly observed from Wide Area Augmentation System (WAAS) supertruth data on April 6, 2000 was reported by Datta-Barua [12]. Dehel then confirmed this ionospheric anomaly based on independent data from the International GPS Service (IGS) [13]. Luo discussed the ionosphere threat model and assessed the ionospheric impact on LAAS [32]. Her simulations showed that ionospheric gradients could potentially lead to vertical positioning errors as large as 19 meters for LAAS users if no integrity monitors are used to detect the gradients. A major focus of this thesis is exploring methods to more quickly detect ionospheric gradients.

The divergence Cumulative Sum (CUSUM) method is developed to quickly detect marginal ionospheric gradients in this thesis. The CUSUM method was invented in 1954

[50]. It has been widely applied in continuous inspection for quality control. Pullen used CUSUMs to validate protection level overbounds for GPS augmentation systems [66]. The CUSUM method was also successfully used for LGF error sigma-mean monitoring by Lee [28].

1.6 Contributions

Three major contributions are described in detail in the text of this thesis. Brief summaries of these efforts are provided here. Figure 1-10 shows the relationship among these contributions. The first contribution is a comprehensive analysis of the existing LAAS monitor suite. This analysis uncovers the limitations of the existing monitors. The Cumulative Sum (CUSUM) method is therefore added to the monitor suite to improve the detection of ionospheric gradients. It is my second contribution. Based on the results of the comprehensive analysis, combining proper monitor test statistics to form failure-specific tests, as the third contribution, can further improve the system integrity.

1.6.1 First Comprehensive Analysis of Monitor Suite

In this thesis, a generic failure model, in which a step failure is injected into both the code and carrier measurements of a GPS receiver channel (a channel is a satellite tracked on one receiver), is created. The failure responses of every integrity monitor proposed for the LGF are derived analytically using this general measurement failure model. The analysis includes channel code step failures, channel phase step failures, satellite clock step failures, satellite clock ramp failures, receiver clock step failures, and ionospheric gradients. My work identifies which integrity monitor can detect which failure modes. It delivers a clear overall picture of each monitor's function against the entire set of possible threats. These results offer valuable information on how to construct effective failure-specific test (FST) statistics. More importantly, the failure analyses provide a tool to evaluate quantitatively and compare the performance of the IMT integrity monitors.

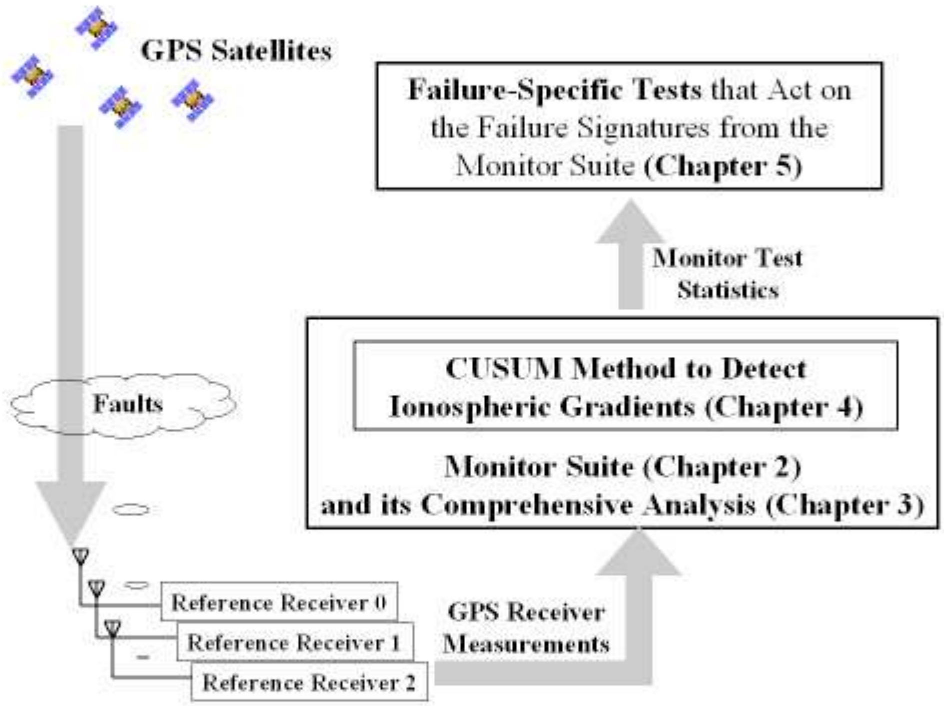

Figure 1-10 Relationship among Contributions

1.6.2 Cumulative Sum (CUSUM) Method to Detect Ionospheric Gradients

The generic Cumulative Sum (CUSUM) quality-assurance method is applied in this thesis to the detection of unusual ionosphere divergence and the divergence CUSUM method is developed that is able to quickly detect marginal ionospheric gradients. Failure tests demonstrate that the divergence CUSUM method is faster in detecting small ionospheric gradients than existing methods, such as the acceleration-ramp-step, code-carrier divergence, and innovation tests. The CUSUM method improves upon the detection speed of the existing code-carrier divergence method by about 30%. When combined, the divergence CUSUM and acceleration-ramp-step monitors offer the lowest bounds on detection time. This method can and should be utilized in future precision approaches and landings.

1.6.3 Combining Monitors for Faster Detection of Specific Failures

Some failures excite multiple monitors, and these failures are best detected by combining the monitors. Monitor combinations detect certain failures more quickly than any individual monitor [70]. These combinations yield powerful failure-specific tests (FSTs), which are pioneered in this dissertation for LAAS. This thesis first gives a mathematical derivation to prove that the FST constructed in an ideal case is able to detect failures more quickly than individual monitors with the same probability of false alarm. Unfortunately, the sum of squared normalized IMT test statistics is not χ^2-distributed in practice. A γ -distribution overbounding method is developed in this thesis to determine the threshold for FSTs. Using the results of the failure analyses, two FST statistics are constructed and tested with IMT data. These test statistics can significantly improve the detection speed of small satellite clock failures and ionospheric gradients. Fault-specific tests are needed so that the LAAS can meet the tighter continuity requirements that apply to Category III precision landings.

1.7 Thesis Outline

Chapter 2 discusses the IMT functions, including the IMT integrity monitoring algorithms and the EXM, in detail. Chapter 3 creates a generic measurement failure model and then analytically derives the failure responses of integrity monitors under the generic and several other known failure modes. Chapter 4 explores the divergence CUSUM method that is able to quickly detect marginal ionospheric gradients. Chapter 5 theoretically demonstrates the idea of failure-specific tests (FST) and develops two FST statistics. Chapter 6 summarizes the accomplished work and provides some thoughts on the direction for future research. This outline is also illustrated in Figure 1-10.

Chapter 2 Integrity Monitor Testbed

2.1 IMT Functions

As noted in the introduction, the IMT is an LGF prototype that is used to evaluate whether or not the LGF can meet integrity requirements. This chapter will explain the IMT functions in detail.

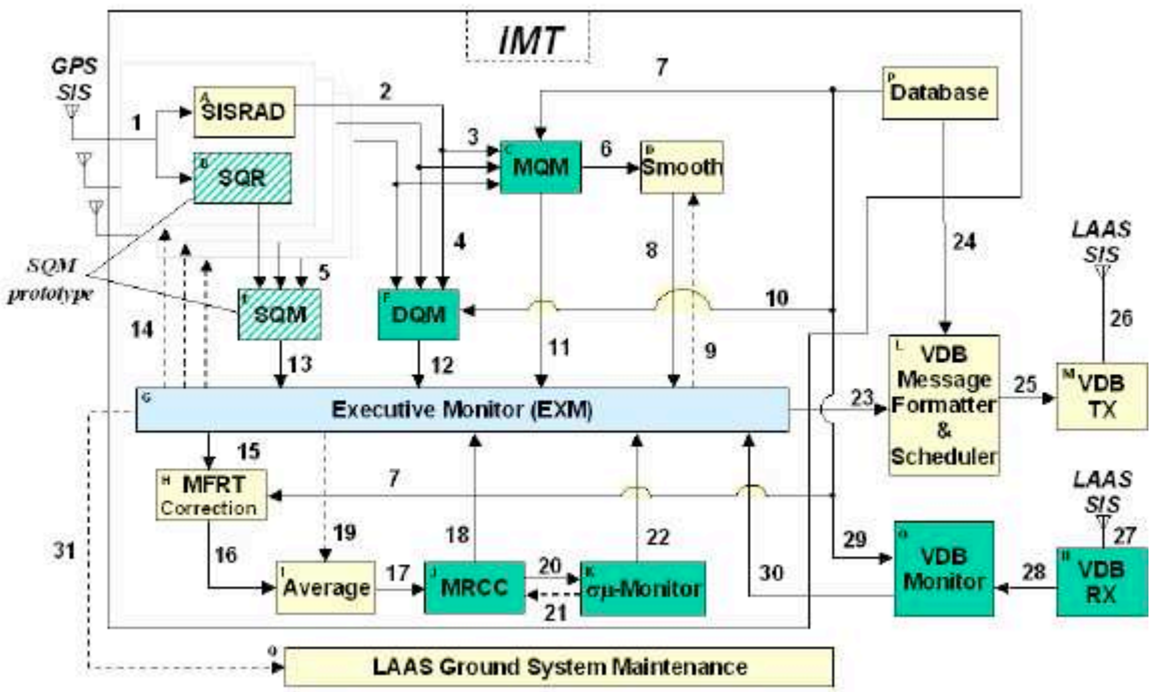

Figure 2-1 IMT Functional Flow Diagram

Figure 2-1 is the functional flow diagram of the IMT. The processing core in the IMT is divided into three parts [81,89]:

1. Nominal processing, including carrier smoothing and the calculation of differential corrections, as will be explained in Section 2.3.

2. A variety of integrity monitor algorithms that are grouped into Signal Quality Monitoring (SQM), Data Quality Monitoring (DQM), Measurement Quality Monitoring (MQM), Multiple Reference Consistency Check (MRCC), σμ-monitor, and Message Field Range Test (MFRT) classes. These are targeted at detecting a wide range of possible failures in the GPS Signal in Space (SIS) or in the LGF itself. SQM targets deformations of the C/A-code broadcast by GPS satellites [1,61,82]. DQM checks the validity of the GPS ephemeris and clock data for each satellite that rises in view of the LGF and at each time new navigation data messages are broadcast [36]. MQM confirms the consistency of the pseudorange and carrier-phase measurements over the last few epochs to detect sudden step and any other rapid errors [26,89]. MRCC examines the consistency of corrections for each satellite across all reference receivers [26,56,89]. The σμ-monitor helps ensure a Gaussian distribution for the correction error with zero mean and that the broadcast σ_{pr_gnd} overbounds the actual errors in the broadcast differential corrections [7,59,66]. MFRT verifies that the computed averaged pseudorange corrections and correction rates fit within the message field bounds [26].
3. Executive Monitoring (EXM) includes complex failure-handling logic. Each integrity monitor may generate one flag (failure) per channel or per satellite, and it is the EXM that coordinates all the monitors and combines these flags. It then isolates failed measurements detected by individual monitors at the system level and reintroduces these measurements after the failure is clearly determined to be over. The first phase of EXM (EXM-I) excludes measurements flagged by any of the "QM" monitors. The measurements that survive EXM-I are then used to compute corrections and B-values, and the second phase of EXM (EXM-II) tries a series of exclusion steps based on MRCC, the σμ-monitor, and MFRT flags [26,64].

If the integrity monitors are not able to detect a hazardous failure and the corresponding erroneous measurements are not excluded by the EXM, the failure can enter the calculation of differential corrections and cause Hazardously Misleading Information

(HMI) to be contained in the corrections. These monitor algorithms, EXM-I, and EXM-II will be explained in approximately chronological processing order. In addition to the IMT processing core, the LGF also contains other essential components, such as VHF Data Broadcast (VDB), through which the LAAS corrections will be broadcast to users. These components are not part of the IMT, which is focused on integrity monitoring and failure resolution, so they will not be discussed in this thesis.

2.2 Gaussian Overbounding Method to Determine Thresholds

Each LGF integrity monitor calculates test statistics, compares the values of its test statistics to the corresponding thresholds, and then decides if a failure is happening. Proper valued thresholds are often the key to achieving the desired performance of the monitor. Therefore, before the integrity monitor algorithms are explained, it is important to first explain the general threshold determination method utilized by the IMT.

Some thresholds can be determined theoretically. However, in most cases, theoretical models for the Probability Density Function (PDF) of a test statistic that are accurate beyond two to three standard deviations do not exist. In addition, the observed PDF obtained from nominal testing is often not very close to that of any well-known statistical distribution. Without an analytical PDF model for a test statistic, we will encounter difficulties in determining its threshold with a specified false alarm probability.

To overcome these difficulties, we typically calculate the threshold from a Gaussian distribution that overbounds the two tails of the apparent (observed from data) distribution. Gaussian distributions are used since avionics assume that measurement errors are Gaussian [39,40,41]. The code-carrier divergence test statistic will be used here as an illustration since the divergence rate is analyzed in the greatest detail in this dissertation. Divergence (or, more exactly, divergence rate) is the estimated change rate of ionosphere delay, I, over time. Section 2.4.3 will discuss the divergence in detail. Moreover, the overbounding procedure outlined here will be applied to most of the integrity monitor test statistics – not just the divergence test statistic.

The threshold determination method consists of the following steps:

1. Run the IMT on one or more sets of pre-stored, nominal data, and collect nominal divergence results shown as dots in Figure 2-2-A. These results are from all channels on the three IMT reference receivers, since the same threshold will be used for all the channels (recall that a channel is one satellite tracked on one receiver). However, if a test statistic has different distributions on different receivers, then different thresholds should be derived and applied for each receiver. For example, the SQM received signal power test has different thresholds for different receivers, as will be discussed in Section 2.4.2.
2. Calculate the sample standard deviation, σ_{elev}, of the test statistic in nine elevation bins of 10 degrees each, and interpolate them at other elevations with a reasonable high-order polynomial curve (a 4th-order polynomial curve is used in this case), as shown by the dashed line in Figure 2-2-B. The binning is due to the fact that the distribution of divergence and most of the other test statistics in the IMT is satellite elevation angle dependent. Measurement errors tend to grow at low elevation angles because multipath is more severe and the obliquity factors for the ionosphere and the troposphere are larger. Thus, test statistics for low elevation satellites have greater variability. Divergence is elevation angle dependent due to the obliquity factor [42]. The same process is repeated to calculate the sample means, μ_{elev}, of the test statistic. For the purpose of threshold derivation, the divergence means are assumed to be zero for two reasons. First, the actual means are very close to zero and will be closer to zero if a longer data set is tested. Second, the means, if nonzero, are very small, compared to the standard deviations. Nominal testing shows that most of the IMT test statistics have zero means.

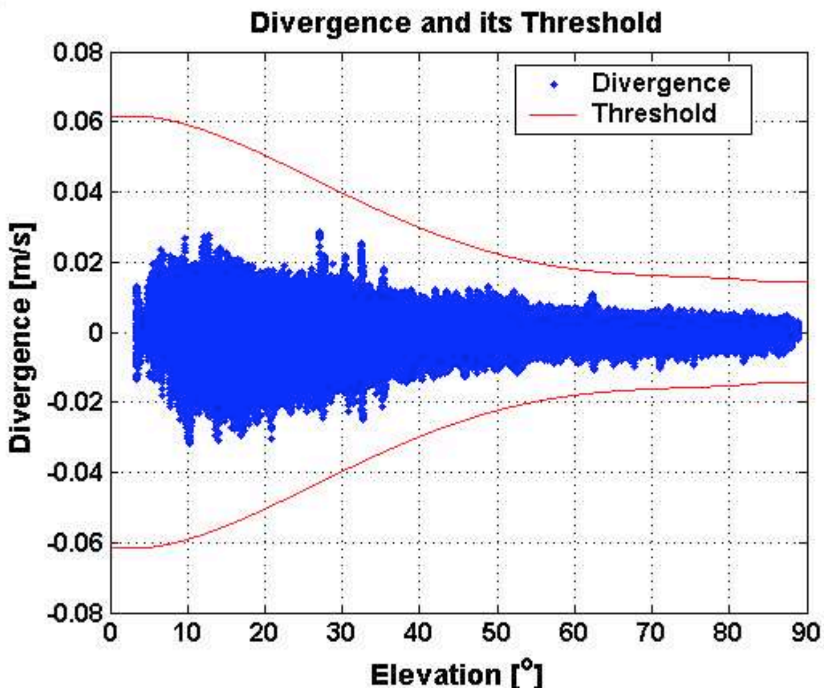

Figure 2-2-A

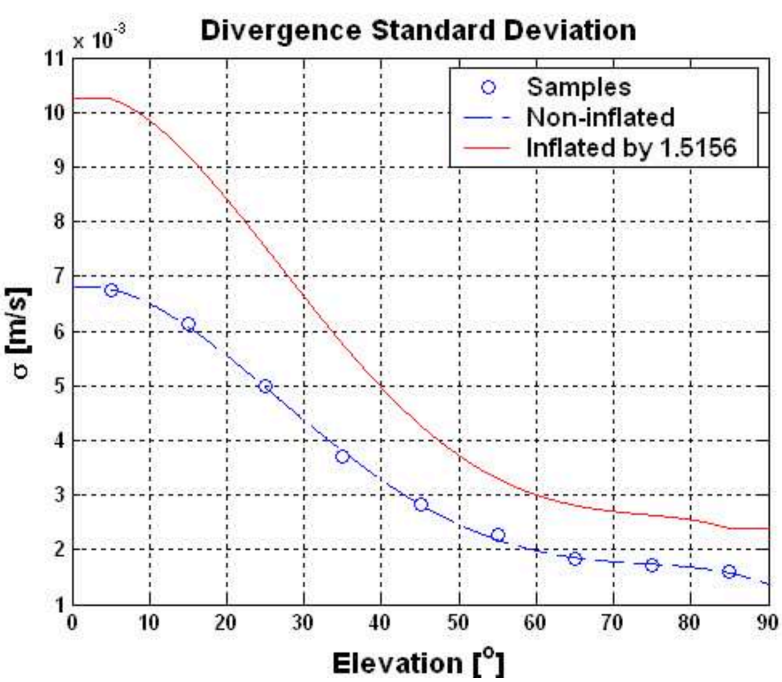

Figure 2-2-B

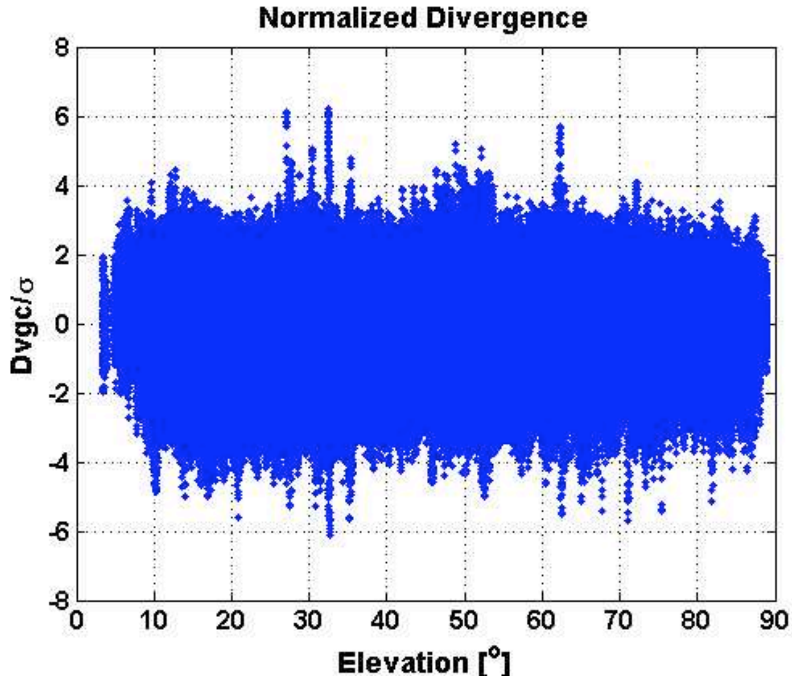

Figure 2-2-C

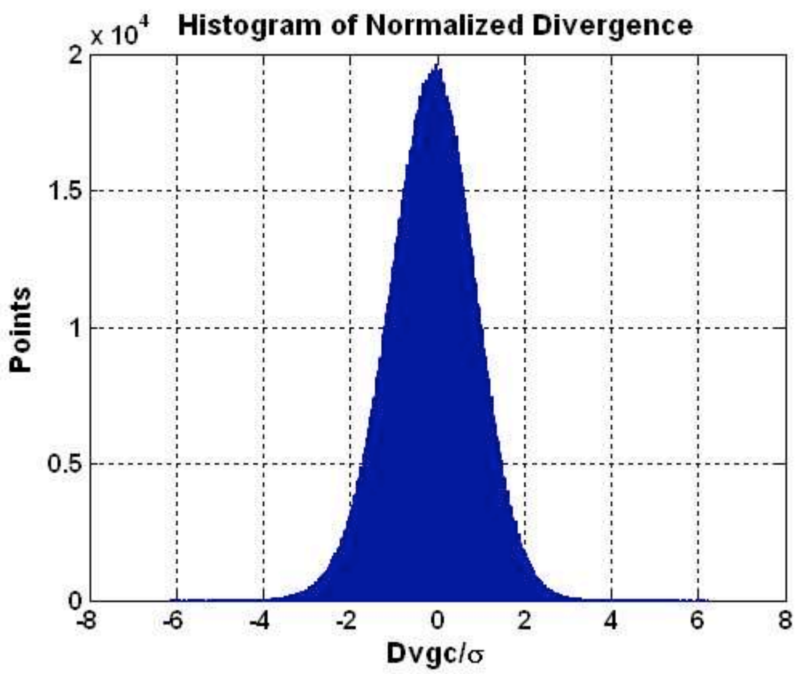

Figure 2-2-D

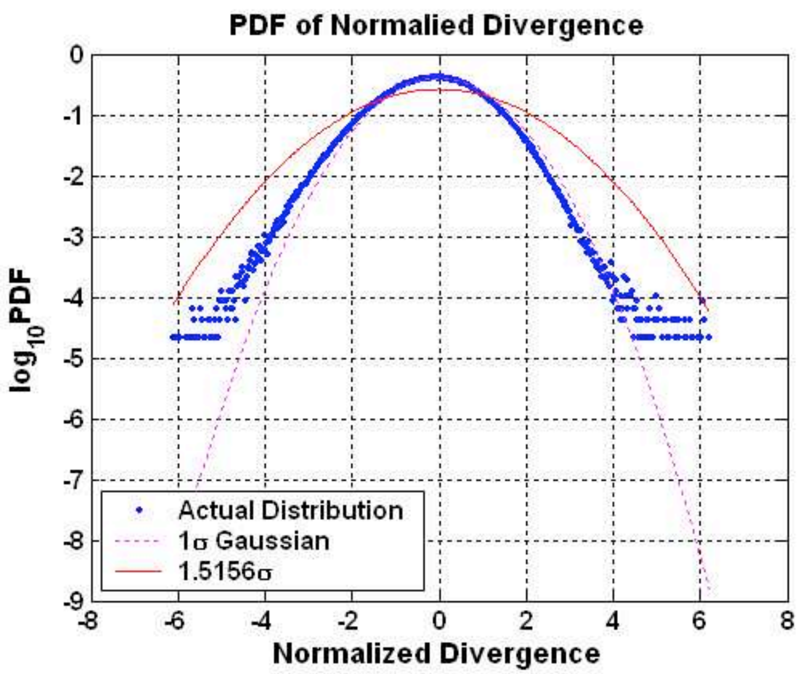

Figure 2-2-E

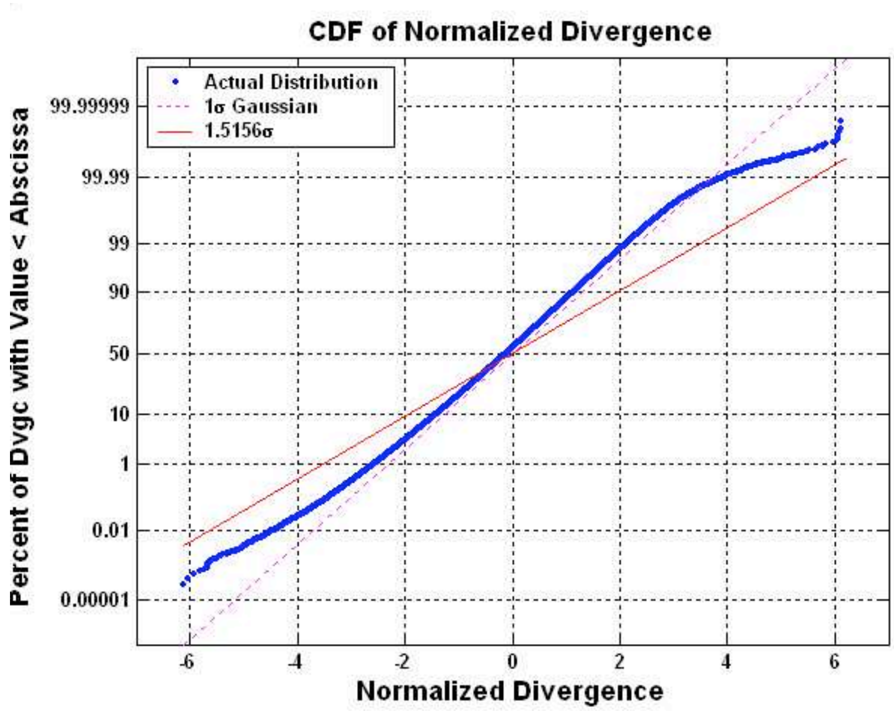

Figure 2-2-F

Figure 2-2 Threshold Determination Method

3. Subtract μ_{elev} from each data point, and then divide the result by its standard deviation, σ_{elev}, at the corresponding elevation. Figure 2-2-C is the divergence after this normalization process. It shows that the distribution of the normalized divergence is quite consistent at different elevations. This indicates excellent μ_{elev} and σ_{elev} interpolations achieved in Step 2.
4. Compute the distribution of the normalized divergence from its histogram. Figure 2-2-D is a plot of the histogram of the normalized divergence using 500 bins. When the number of points in each bin is divided by the total number of points in all bins, the histogram is transformed into the discretized, apparent PDF as shown by the dotted line in Figure 2-2-E, where the y-axis has a logarithmic scale.
5. Compute the standard deviation of the normalized divergence. To protect continuity, a zero-mean Gaussian distribution with an inflated standard deviation is used to overbound the tails of the data distribution. In other words, in each bin at the two tails, the desired Gaussian distribution has a larger probability (i.e., the integral of its PDF over the range of the corresponding bin) than the discretized, apparent data distribution. The standard deviation inflation factor f should be minimized during the overbounding process in order to protect integrity. The process is accomplished numerically in Matlab. As demonstrated in Figure 2-2-E, the solid line overbounds the tails of the dotted blue line. The inflation factor, f, in this example is equal to 1.5156. Figure 2-2-F illustrates that the actual Cumulative Density Function (CDF) is also overbounded by the Gaussian distribution. In this figure, a non-linear transformation is used to change the CDF of a Gaussian variable into a straight line whose slope is related to the standard deviation of the Gaussian distribution. This figure suggests that the value of f can be smaller if we apply the overbound method directly to the CDF. This CDF overbound method is widely used [74].
6. Multiply the original standard deviations by f. The final standard deviation of the divergence test statistic is considered to be $f\sigma_{elev}$, as shown by the solid line in Figure 2-2-B.

7. Finally, the elevation-dependent threshold is modeled by

$$Threshold_{elev} = \mu_{elev} \pm 6 f \sigma_{elev} \tag{2-1}$$

The solid line in Figure 2-2-A is the divergence threshold. The figure shows that the magnitude of all divergence points is below the corresponding threshold, as expected from nominal testing.

Since a Gaussian variable, $X \sim N(\mu_{elev}, f^2 \sigma_{elev}^2)$, overbounds the actual distribution, the probability of false alarm with the threshold from Equation (2-1) is less than

$$\Pr\left(\left|x - \mu_{elev}\right| > 6 f \sigma_{elev}\right) = 2Q(6) = 1.9732 \times 10^{-9} \tag{2-2}$$

The overall continuity requirement in Table 1-2 is subdivided into many sub-allocations for the many different tests [64,81]. With the threshold at a 6-sigma level for each test, the system meets the overall continuity requirement.

2.3 Carrier Smoothing and Pseudorange Corrections

Carrier smoothing and the generation of differential corrections are the backbone of LGF processing under nominal conditions. For each channel (m,n) at epoch k, the IMT applies carrier smoothing to reduce raw pseudorange measurement errors with the following filter [81]:

$$\rho_{s,m,n}(k) = \frac{1}{N_s}\rho_{m,n}(k) + \frac{N_s - 1}{N_s}\left(\rho_{s,m,n}(k-1) + \phi_{m,n}(k) - \phi_{m,n}(k-1)\right) \tag{2-3}$$

where

$$N_s = \tau_s / T_s = 100/0.5 = 200 \tag{2-4}$$

The smoothing filter uses a time constant, τ_s, of 100 seconds, and it assumes the satellite is continuously tracked on the receiver. If this is not the case, the filter needs to be reset when receiver m loses phase lock on satellite n or when demanded by EXM (i.e., due to a monitor alert). Figure 2-3 shows the single difference of raw pseudorange measurements, ρ, and smoothed pseudorange, ρ_s, from Channel (RR 0, PRN 7) for 30 seconds. Note that ρ_s is smoother than ρ after carrier smoothing. Because the elevation angle of PRN 7 reaches as high as 85^o when passing over Stanford University, this satellite will be frequently used as a preferred example.

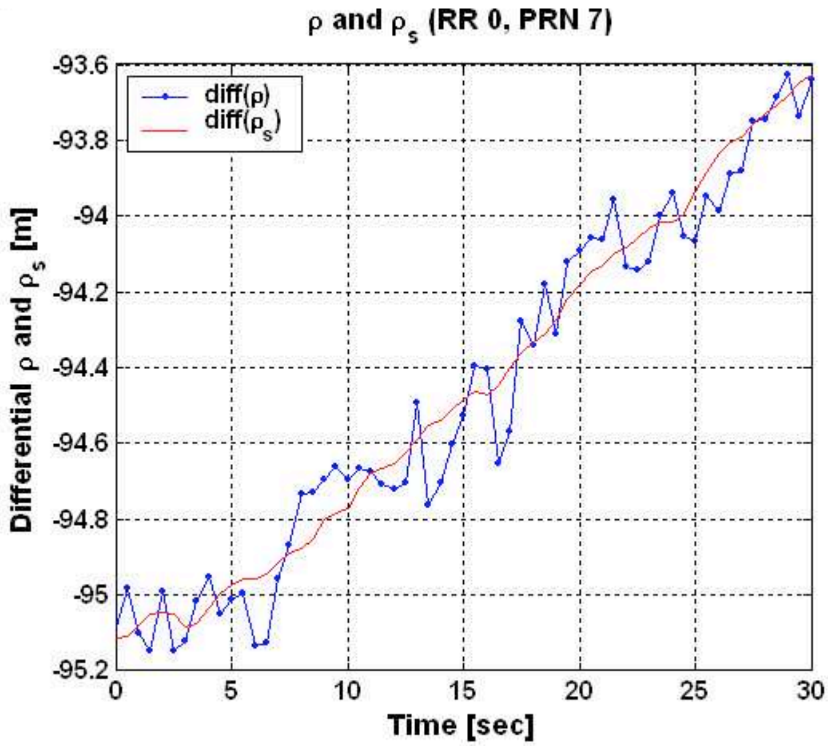

Figure 2-3 Pseudorange and Smoothed Pseudorange

In the IMT, corrections are generated on both pseudorange and carrier-phase measurements [26]:

$$\rho_{sc,m,n}(k) = \rho_{s,m,n}(k) - R_{m,n}(k) - \tau_{m,n}(k) \tag{2-5}$$

$$\phi_{c,m,n}(k) = \phi_{m,n}(k) - R_{m,n}(k) - \tau_{m,n}(k) - \phi_{ci,m,n}(0) \tag{2-6}$$

where ρ_{sc} is the smoothed pseudorange correction, and ϕ_c is the raw carrier correction. Geometric range, $R_{m,n}$, is the distance from reference antenna m to satellite n. Using the known location of the reference antenna, both $R_{m,n}$ and the satellite clock correction, $\tau_{m,n}$, are computed from DQM-approved satellite navigation data. The initial intermediate raw carrier correction (at epoch 0), $\phi_{ci,m,n}(0)$, is equal to $\phi_{m,n}(0) - R_{m,n}(0) - \tau_{m,n}(0)$. Equation (2-6) makes the value of $\phi_{c,m,n}$ smaller, and it also cancels the integer ambiguity in $\phi_{m,n}$ since the phase maintains lock starting at epoch 0. If a cycle slip happens at one epoch, then that epoch is regarded as epoch 0 from that time on.

2.4 Signal Quality Monitoring (SQM)

The purpose of SQM is to detect and identify anomalies in the received GPS ranging signals, including satellite signal anomalies and local interference. It consists of three functions: the correlation peak symmetry test, the received signal power test, and the code-carrier divergence test [81].

2.4.1 Correlation Peak Symmetry Test

The primary focus of SQM is to ensure that the correlation peak has sufficient symmetry. Reference receivers that incorporate SQM report C/A-code correlation measurements at several different correlator spacings. These measurements are processed by the SQM algorithms to determine if the ideal triangular C/A-code correlation shape has been appreciably altered by the presence of signal-deformation failures, otherwise known as "evil waveforms." The SQM prototype was developed separate from the IMT, and it has been integrated into the IMT over the past few years. The details of its algorithms and performance can be found in [1,17,43,60,61,62].

2.4.2 Received Signal Power Test

SQM has a separate algorithm to confirm that received satellite signal power is within SPS specifications [77]. Signal power levels significantly below specified levels can increase ranging errors and may result in integrity risk. The signal-power monitor takes an average of reported receiver carrier-to-noise power density ratio, C/N_0, for each channel (m,n) at the current epoch, k, and at the previous epoch, $k-1$ [89]:

$$C/N_{0\_avg,m,n}(k) = \frac{1}{2}\left(C/N_{0,m,n}(k-1) + C/N_{0,m,n}(k)\right) \tag{2-7}$$

The calculated $C/N_{0\_avg,m,n}(k)$ is then compared with a threshold value. If $C/N_{0\_avg,m,n}(k)$ is larger than the threshold, it passes this signal power test. Otherwise, a flag is generated for this channel.

Different thresholds are used for different receivers because of different antenna sites, antenna gain variations, different cable losses, and different hardware configurations. Though the distribution of signal power, C/N_{0_avg}, in $[dB \cdot Hz]$, shown as the dotted line in Figure 2-4-A, is unlikely to be Gaussian, its threshold is still set at $\mu_{elev} - 6f\sigma_{elev}$ using the method discussed in Section 2.2. Its standard deviation inflation factor equals 1.7451 for RR 0. Figure 2-4-B shows the thresholds for the three receivers. Figure 2-4-C shows nominal C/N_{0_avg} results received from all satellites on RR 0, while Figure 2-4-D shows nominal C/N_{0_avg} results from a channel. All points are well above the corresponding thresholds.

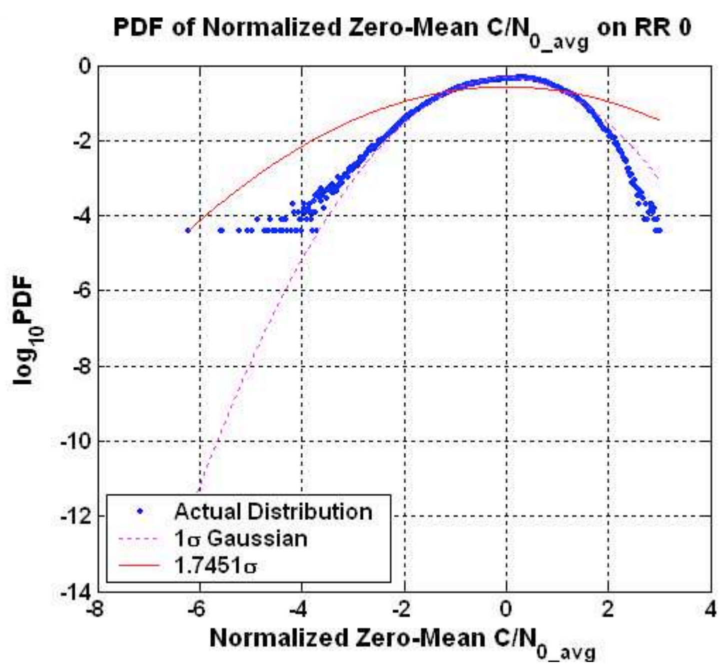

Figure 2-4-A

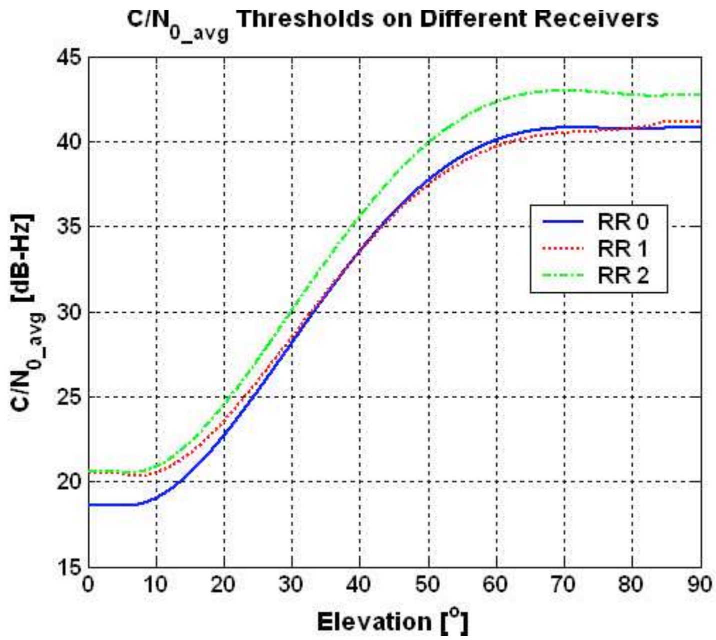

Figure 2-4-B

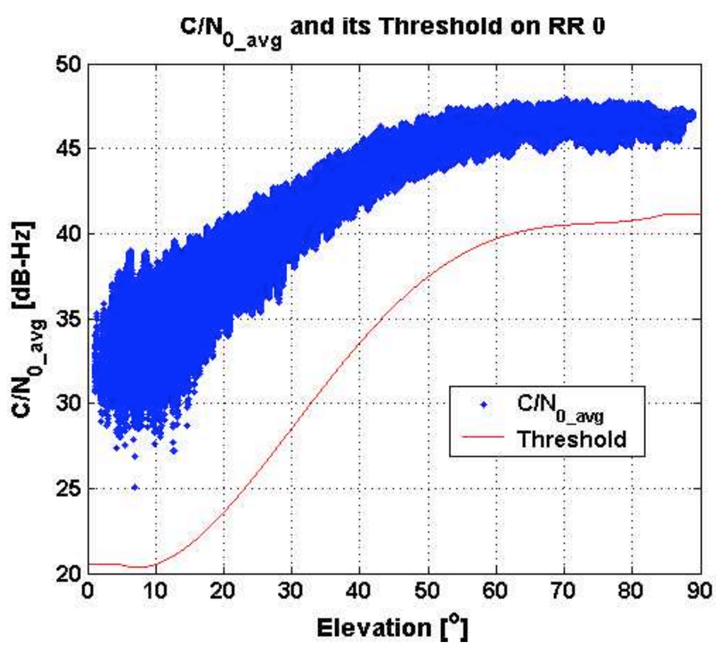

Figure 2-4-C

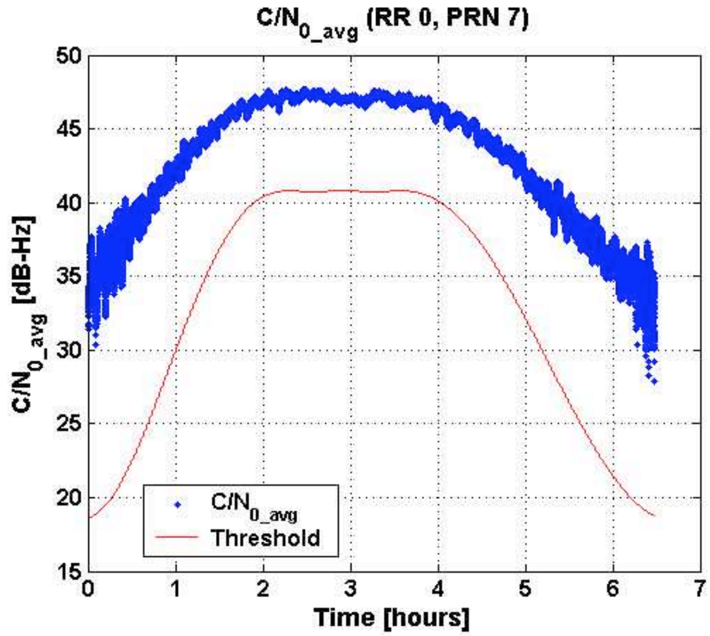

Figure 2-4-D

Figure 2-4 Received Signal Power Thresholds and Nominal Test Results

In this test, the same threshold is applied to all satellites tracked on the same receiver. Variations among GPS satellites may cause a wide range of measured C/N_{0_avg} at a given elevation angle, as demonstrated in Figure 2-4-C. The received signal power can be more precisely monitored in the future if different thresholds are derived and used for different satellites even on the same receiver.

As can be seen in Figure 2-4-D, when the satellite elevation is low, C/N_{0_avg} is also low and has a large variance, which makes it difficult to tightly monitor the signal power of satellites when they first rise into view of the LGF. Averaging C/N_0 measurements over a 100-second interval, as suggested in [20], has been tried in the IMT, but because of the dominance of long-delay multipath on the IMT antennas at low elevation angles, the C/N_0 variance reduction is small.

2.4.3 Code-Carrier Divergence Test

The code-carrier divergence test is used to detect ionospheric storms and ensure that the divergence of code and carrier for any given satellite is sufficiently small. It uses a geometric moving averaging (GMA) method to estimate the code-carrier divergence, $Dvgc_{m,n}(k)$ [81]:

$$Dvgc_{m,n}(k) = \frac{\tau_d - T_s}{\tau_d} Dvgc_{m,n}(k-1) + \frac{1}{\tau_d} dz_{m,n}(k) \tag{2-8}$$

where the time constant of averaging, τ_d, equals 200 seconds, the IMT GPS measurement update rate $T_s = 0.5$ seconds as in Equation (1-4), and z is defined as the raw code-minus-carrier measurement. From Equations (1-1) and (1-2) on Channel (m,n),

$$\begin{aligned} z_{m,n}(k) &= \rho_{m,n}(k) - \phi_{m,n}(k) \\ &= 2I_{m,n}(k) + M_{m,n}(k) - N_{m,n}(k)\lambda \end{aligned} \tag{2-9}$$

and

$$\begin{aligned} dz_{m,n}(k) &= z_{m,n}(k) - z_{m,n}(k-1) \\ &= 2\big(I_{m,n}(k) - I_{m,n}(k-1)\big) + \big(M_{m,n}(k) - M_{m,n}(k-1)\big) \\ &\quad - \big(N_{m,n}(k) - N_{m,n}(k-1)\big)\lambda \\ &= \dot{I}_{m,n}(k) + \big(M_{m,n}(k) - M_{m,n}(k-1)\big) \end{aligned} \tag{2-10}$$

The estimated ionospheric gradient is equal to:

$$\dot{I}_{m,n}(k) = \frac{I_{m,n}(k) - I_{m,n}(k-1)}{T_s} \tag{2-11}$$

The integer ambiguity, $N_{m,n}$, is cancelled in Equation (2-10) since no cycle slip is assumed. If a cycle slip is detected by another monitor, the GMA filter needs to be reset.

One slight improvement in convergence speed of the GMA method is to use a variable time constant, $\tau(k)$, instead of a fixed value when a new satellite rises into view or filter reset occurs after a satellite is temporarily excluded [53], i.e.,

$$Dvgc_{m,n}(k) = \frac{\tau(k) - T_s}{\tau(k)} Dvgc_{m,n}(k-1) + \frac{1}{\tau(k)} dz_{m,n}(k) \tag{2-12}$$

with

$$\tau(k) = \begin{cases} kT_s, & k < \tau_d / T_s \\ \tau_d, & else \end{cases} \tag{2-13}$$

With varying time constant, $\tau(k)$, used in the first τ_d seconds, divergence should converge faster. However, in practice, the calculated noisy divergence results in the first τ_d seconds are ignored, regardless of whether Equation (2-8) or (2-12) is implemented.

The standard deviation inflation factor is equal to 1.5156, as already shown in Figure 2-2-E on page 29. All nominal divergence test results shown in Figure 2-2-A are well bounded by the thresholds, as expected. Figure 2-5 illustrates nominal divergence test results from one channel.

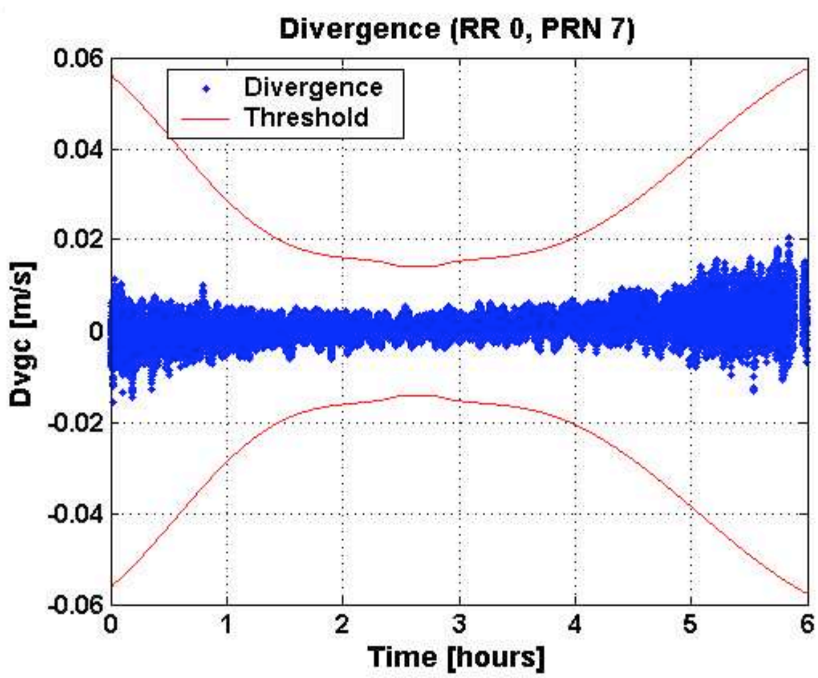

Figure 2-5 Nominal Code-Carrier Divergence Test Results from Channel (0, 7)

2.5 Data Quality Monitoring (DQM)

The purpose of DQM is to verify that satellite navigation data is of sufficient fidelity. DQM continually checks the validity of the GPS ephemeris and clock parameters for each satellite that rises into view of the LGF [36,65]. DQM works in tandem with the MFRT function, as will be explained in Section 2.10.

For a newly risen satellite, DQM compares the satellite orbit locations over the next 6 hours at 5-minute intervals based on the received new ephemeris message to those generated from the most recent almanac message. Ephemeris data describe the orbit of a satellite, while the almanac is a subset of the clock and ephemeris data that covers all satellites in the GPS constellation but with reduced precision. DQM then insures that the ephemeris-based satellite positions agree with the almanac-based positions to within 7000 meters (this is limited by the accuracy of the almanac). Figure 2-6-A shows that it takes DQM about 18 seconds to finish validating a new ephemeris by examining two points every epoch (except for the first epoch). There are 71 points total that span $70 \times 5 = 350$ minutes or about 6 hours. All points are below the threshold of 7000 meters. Status Value 4 in Figure 2-6-C represents the ephemeris validation process for newly risen satellites using this ephemeris-almanac test.

When navigation messages are updated (typically every two hours), DQM compares satellite position based on old and new ephemeris to insure that the new ephemeris is consistent with the old, validated over the past 2 hours and the upcoming 2 hours. Figure 2-6-B again shows that it takes about 18 seconds to finish validating a new ephemeris by examining two points every epoch (except for the first epoch). A total of 71 samples of the satellite position difference are calculated with a 206-second interval between any two neighboring samples. They span $70 \times 206 = 14420$ seconds, or nearly 4 hours. All results are below the threshold of 250 meters, as expected. Status Value 3 in Figure 2-6-C represents the ephemeris re-validation process using this ephemeris-ephemeris test. In fact, DQM also compares the satellite positions based on the received new ephemeris message and the most recent almanac message for updated navigation messages, as it does for a newly risen satellite.

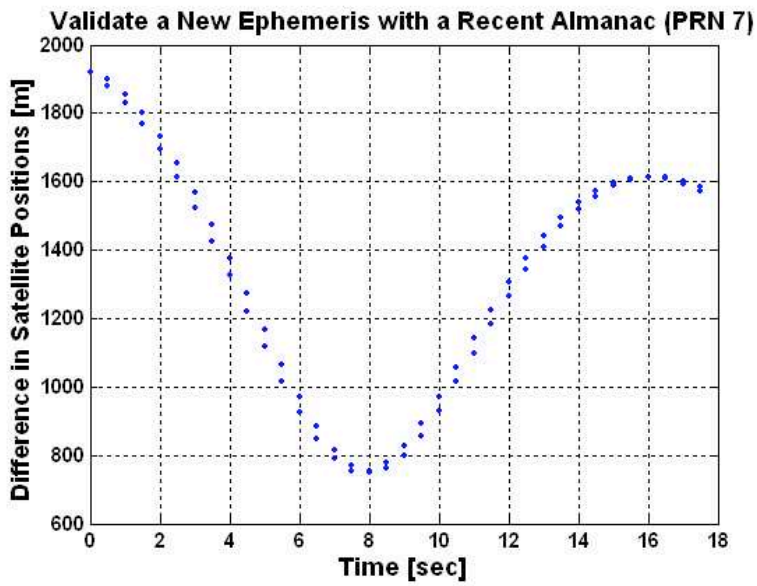

Figure 2-6-A

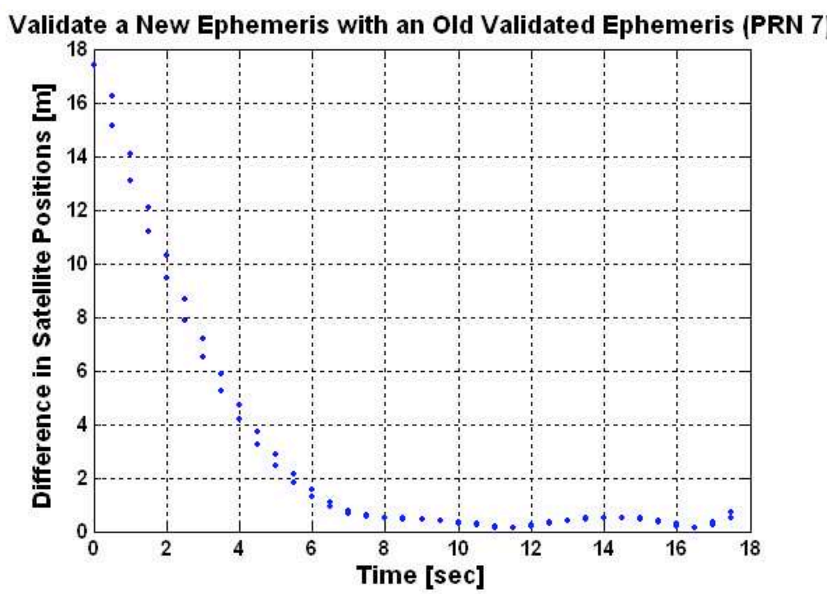

Figure 2-6-B

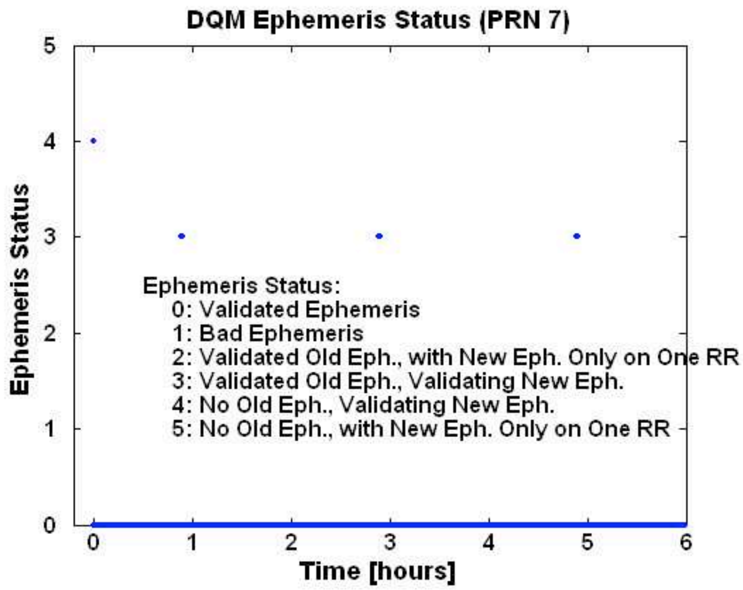

Figure 2-6-C

Figure 2-6 DQM Ephemeris Status of a Typical Satellite and the Validation of a New Ephemeris

Figure 2-6-C shows how the health status of a satellite's ephemeris evolves. When a satellite rises and becomes visible to the IMT, there is no old ephemeris but only an almanac. DQM begins its validation process (with Status Value 4) when at least two reference receivers have separately obtained identical new navigation data. After the initial ephemeris is validated (Status Value 0), the navigation data is updated once every two hours (and the Status Value jumps to 3 for three times in six hours). If the Status is 4 or 5, it means that the IMT has no validated ephemeris for a satellite. Status 1 implies a bad ephemeris, and the corresponding satellite should be excluded. In all other cases (0, 2, and 3), there is always a validated ephemeris, and the messages from this satellite are safe to use.

The Yesterday-minus-Today Ephemeris (YE-TE) test is a more precise validation of ephemerides for newly risen satellites than the ephemeris-almanac test. The concept of the YE-TE test is to confirm that today's broadcast ephemeris data for each GPS satellite is correct by comparison with the most recently validated ephemeris data. The comparison can be based on satellite positions computed from the old and new ephemeris or the individual orbit parameters of the old and new messages. The YE-TE test has become a part of DQM, and the details of this test can be found in [65].

2.6 Measurement Quality Monitoring (MQM)

MQM confirms the consistency of the pseudorange and carrier-phase measurements over the last few epochs to detect sudden step and any other impulsive errors due to GPS SIS clock anomalies or LGF receiver failures. It consists of three monitors: the receiver lock time check, the carrier acceleration-ramp-step test, and the carrier-smoothed code innovation test. MQM generates a flag for the appropriate channel if it detects a failure in the measurements from the channel.

2.6.1 Receiver Lock Time Check

This check ensures continuous receiver phase lock on each channel by computing the numerical difference of the lock times reported by each reference receiver. The top plot in Figure 2-7 shows typical nominal lock time results reported from one GPS receiver channel. A long straight line in the top plot indicates that the receiver keeps track of the satellite for about six hours. The bottom plot is the zoomed-in version of the final period (about half an hour) when the satellite sets. It shows that the receiver frequently loses track of satellites at low elevation angles.

This type of failure is most likely not hazardous, and it frequently occurs when the receiver tends to track satellites at low elevation angles, as observed in Figure 2-7. On the other hand, cycle slips that are not reported by the receiver can be hazardous. One cycle is only 19 cm long, so single-cycle slips do not result in 7.7-meter vertical positioning errors or HMI. Hazardous multiple-cycle slips would be detected quickly by the acceleration-ramp-step test.

The IMT implements this check in such a way that EXM will not reject measurements from a channel as hazardous even if this check fails. Instead, the IMT initializes and resets several memory buffers properly before using it, including re-initializing its carrier smoothing filter discussed in Section 2.3. Not generating flags from this monitor protects continuity.

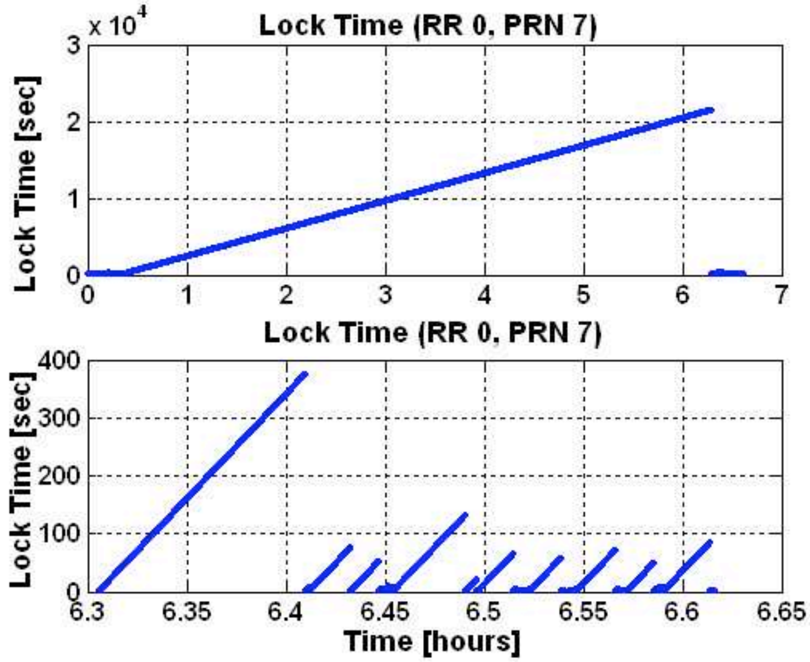

Figure 2-7 Nominal Lock Time Test Results from Channel (0, 7)

2.6.2 Carrier Acceleration-Ramp-Step Test

The carrier acceleration-ramp-step test is designed to detect impulses, steps, ramps, excessive accelerations, or other rapid changes in carrier phase measurements.

On channel (RR m, PRN n) at each epoch k, the last 10 continuous epochs (i.e., from epoch $k-9$ to epoch k) of ϕ^*, written as $\phi^*_{m,n}$, are calculated first by the following formula:

$$\phi^*_{m,n}(k) = \phi_{c,m,n}(k) - \frac{1}{N_m} \sum_{j \in S_m(k)} \phi_{c,m,j}(k) \tag{2-14}$$

where $\phi_{c,m,n}$ is calculated from Equation (2-6). The subtraction in Equation (2-14) cancels any possible receiver clock drifts. S_m is the set of N_m satellites tracked on receiver m and should be exactly the same in calculating all these ten fitting points of $\phi^*_{m,n}(k)$;

otherwise, it will be meaningless to fit a curve based on these ten points and it will be difficult to determine whether a rapid change in $\phi^*_{m,n}$ is caused by a measurement failure or due to a change in S_m.

These ten fitting points, $\phi^*_{m,n}(k)$, are used to fit the following quadratic model with the least-squares method [81]:

$$\phi^*_{m,n}(k,t) = \frac{d^2\phi^*_{m,n}(k,t)}{dt^2}\cdot\frac{t^2}{2} + \frac{d\phi^*_{m,n}(k,t)}{dt}\cdot t + \phi^*_{0,m,n}(k) \tag{2-15}$$

In the current IMT implementation, $t = 0$ is set at epoch $k-9$, $t = T_s$ is set at epoch $k-8$, and so on. The acceleration and ramp are defined as the coefficients in the curve:

$$Acc_{m,n}(k) \equiv \frac{d^2\phi^*_{m,n}(k,t)}{dt^2} \tag{2-16}$$

$$Ramp_{m,n}(k) \equiv \frac{d\phi^*_{m,n}(k,t)}{dt} \tag{2-17}$$

Besides the acceleration and ramp, a third test statistic expresses the apparent change (or "step") in the latest measurement and is defined as:

$$Step_{m,n}(k) \equiv \phi^*_{m,n}(k) - \phi^*_{m,n}(k-1, 10\cdot T_s) \tag{2-18}$$

where $\phi^*_{m,n}(k)$ is the actual value at the current epoch k from Equation (2-16), and $\phi^*_{m,n}(k-1, 10\cdot T_s)$ is the predicted value at $10\cdot T_s$ from the fitting curve (2-15) modeled at epoch $k-1$.

The solid line in Figure 2-8-A is the Gaussian distribution with an inflated σ used to determine the threshold for acceleration. Its standard deviation inflation factor equals 1.6514. Figure 2-8-B illustrates that all nominal acceleration test results from three receivers are well bounded by these thresholds, as expected. Typical nominal results from one channel are showed in Figure 2-8-C. Thresholds and nominal test results of ramp and step are shown in Figures 2-9 and 2-10, respectively.

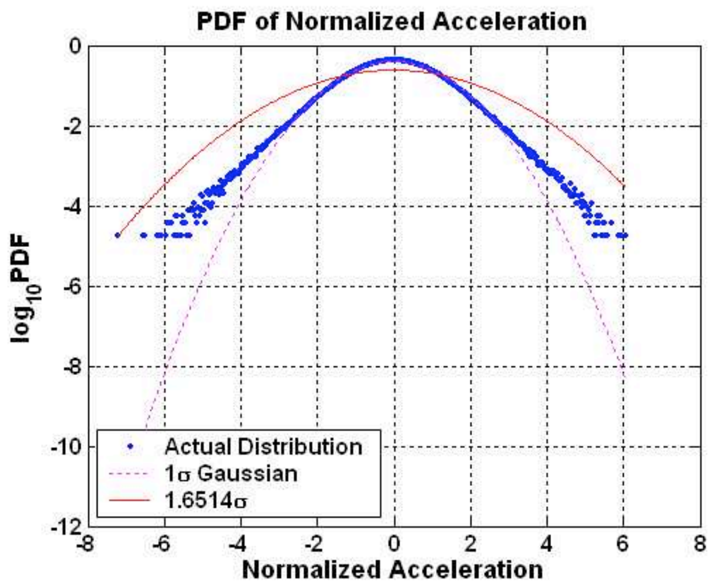

Figure 2-8-A

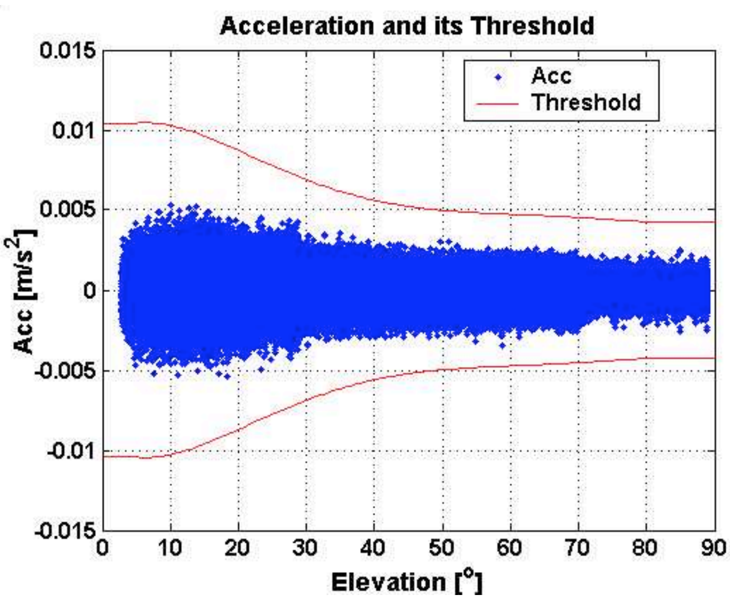

Figure 2-8-B

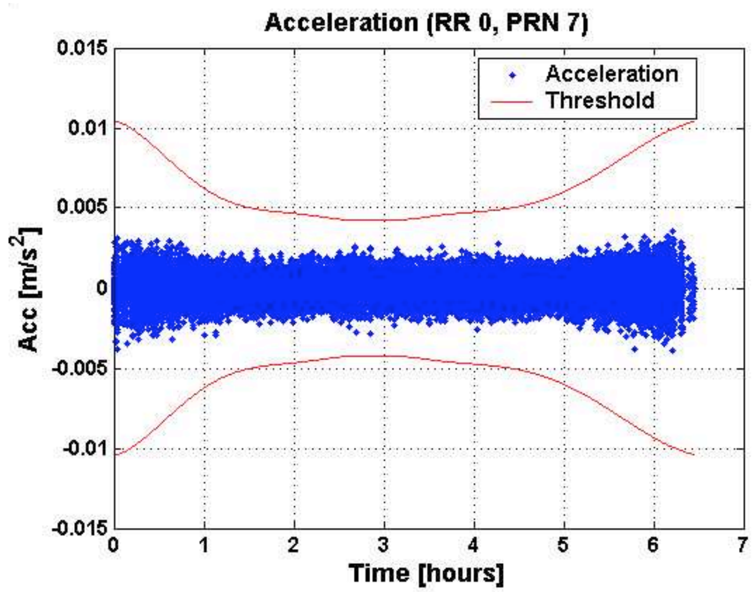

Figure 2-8-C

Figure 2-8 MQM Acceleration Threshold and Nominal Test Results

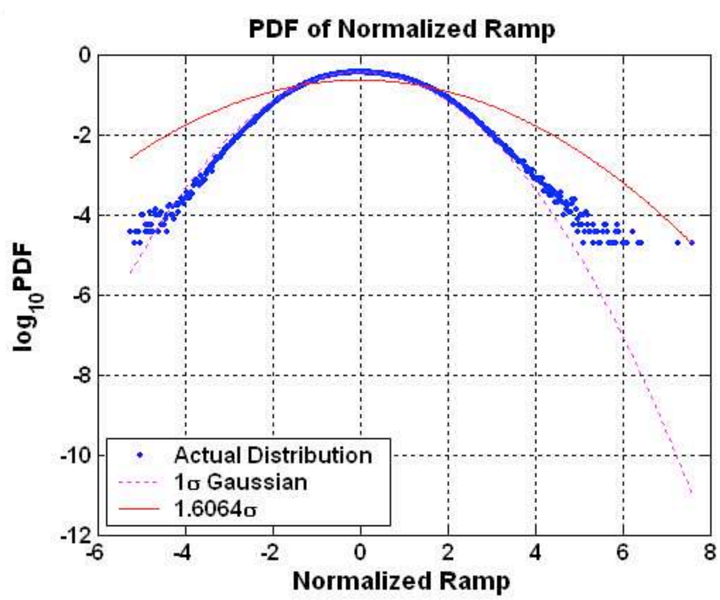

Figure 2-9-A

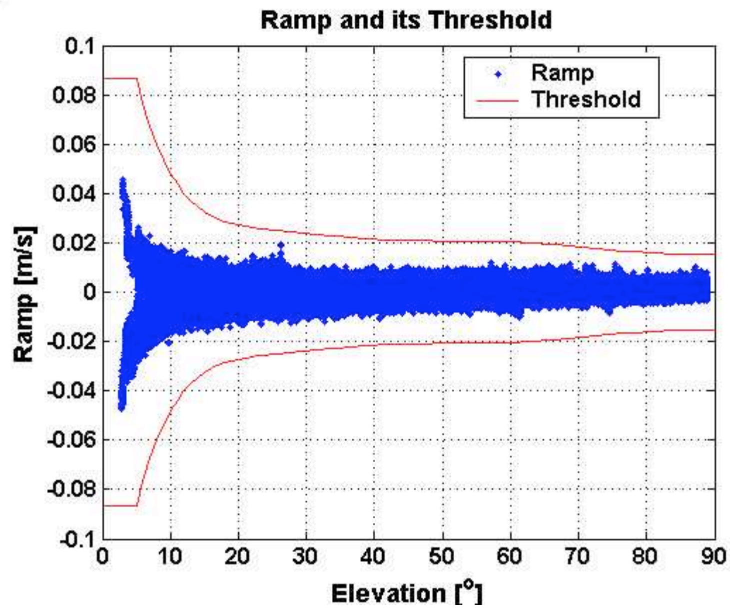

Figure 2-9-B

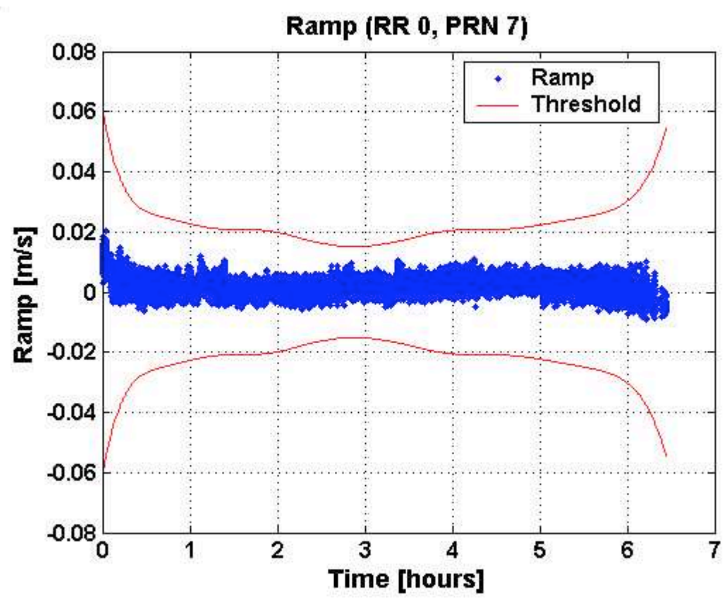

Figure 2-9-C

Figure 2-9 MQM Ramp Threshold and Nominal Test Results

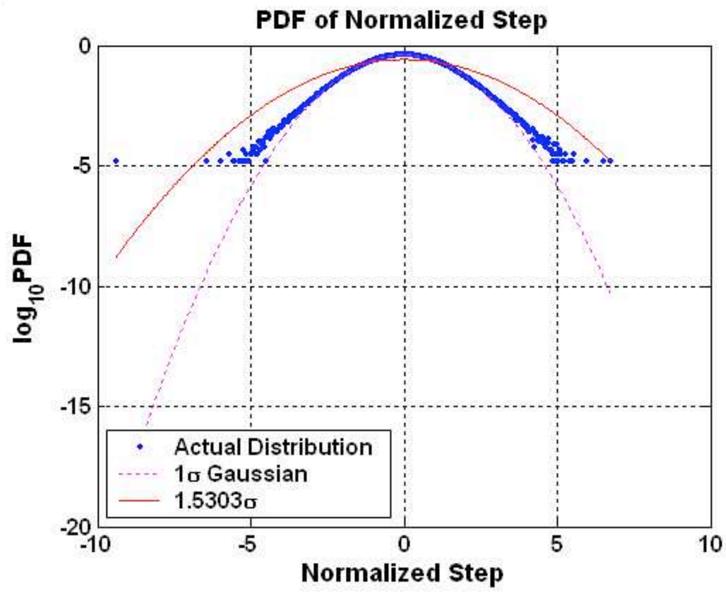

Figure 2-10-A

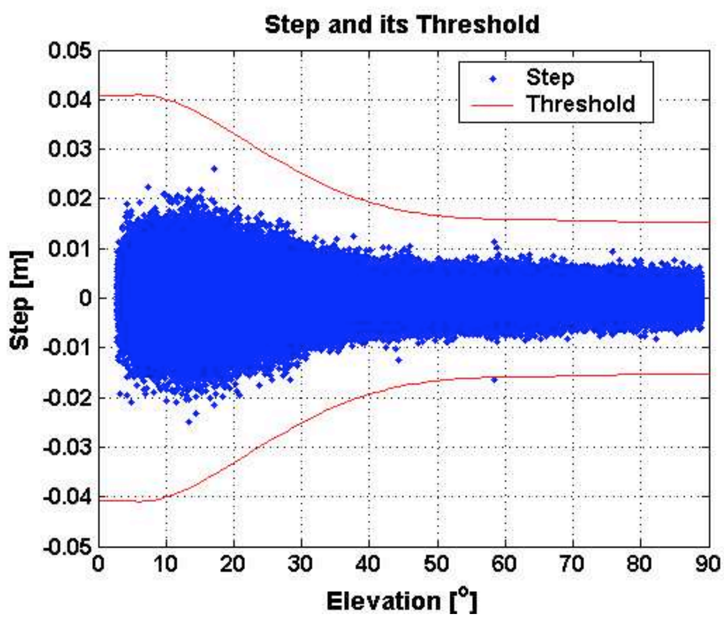

Figure 2-10-B

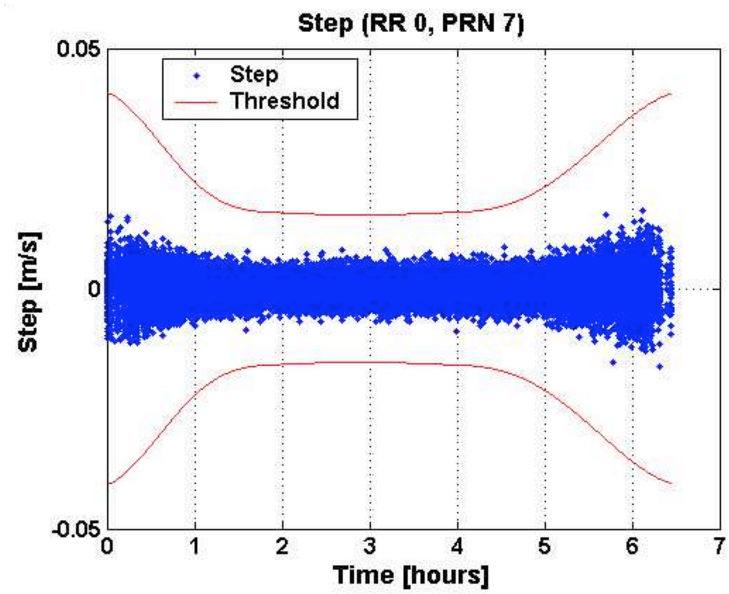

Figure 2-10-C

Figure 2-10 MQM Step Threshold and Nominal Test Results

If at least one of the acceleration, ramp, and step test statistics on a channel exceeds the corresponding threshold, the channel fails and is flagged for later exclusion by EXM-I. If a channel contains a huge measurement failure, the calculated $\phi^*_{m,n}$ for other non-failed channels will be polluted by the failure through the averaging function in Equation (2-14). The failure may then possibly cause multiple flags even on non-failed channels. In order to avoid these false alarms and to protect continuity, determination logic needs to be implemented within the acceleration-ramp-step calculation.

Figure 2-11 is the diagram of the determination logic in the acceleration-ramp-step calculation. After calculating acceleration, ramp and step test statistics for all channels on a given reference receiver, flags can be generated simply by comparing their magnitudes to their corresponding thresholds. If there is no flag, then all channels pass the test; otherwise, potential false alarm needs to be differentiated from real flags. The logic does this by finding the maximum flagged acceleration, ramp, and step test statistics among all channels first. In other words, it finds the maximum acceleration value if there are one or more acceleration flags. Similarly, it finds at most two maximums from the ramp and step test statistics. If there is only one maximum or if two or three maximums occur on the same channel, that channel is believed to contain failed measurements and is excluded. The acceleration-ramp-step calculation is then repeated right after the exclusion. If these maximums occur on different channels, MQM simply retains the calculated results and exits its logic, instead of excluding multiple channels that could damage the system's availability. This does not mean that these channels will be approved. Instead, EXM-I will more accurately continue with these flags based on information from other integrity monitors. It is quite rare to have failures happening on multiple channels at the same time, so the number of iterations in this logic usually is not more than two.

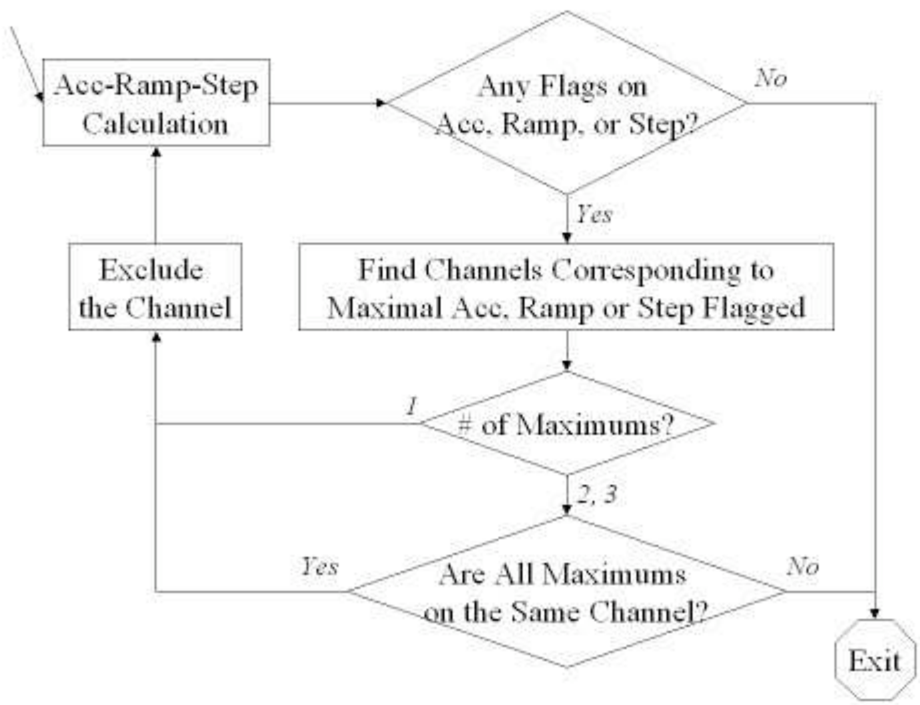

Figure 2-11 Acceleration-Ramp-Step Determination Logic

2.6.3 Carrier-Smoothed Code (CSC) Innovation Test

The CSC innovation test is used to detect impulse and step errors on raw pseudorange measurements. The innovation test statistic is defined as [81]

$$Inno_{m,n}(k) \equiv \rho_{m,n}(k) - \left(\rho_{s,m,n}(k-1) + \phi_{m,n}(k) - \phi_{m,n}(k-1)\right) \qquad (2\text{-}19)$$

where $\rho_{s,m,n}$ is the carrier-smoothed code filter output from Equation (2-3). Chapter 4 shows that the innovation and divergence tests are mathematically equivalent except for different smoothing time constants and scaling factors. This is one of the findings in this thesis, and will be further discussed in Section 4.1.

Plot (A) in Figure 2-12 is a Gaussian distribution used to determine the threshold of this test. The standard deviation inflation factor equals 1.7686. Plot (B) shows that all nominal innovation results from three receivers are well bounded by the threshold, as expected. Typical nominal results from one channel are shown in Plot (C).

If two or all of the innovations in three successive epochs are over thresholds, a real flag will be generated from this CSC innovation test, and the channel will be excluded by EXM-I. Otherwise, if only the current epoch's innovation is over the threshold, the measurement can still be used, but the smoothing-filter update for this epoch does not use its raw pseudorange. Instead of using Equation (2-3) on page 31, the code is smoothed by

$$\rho_{s,m,n}(k) = \rho_{s,m,n}(k-1) + \phi_{m,n}(k) - \phi_{m,n}(k-1) \tag{2-20}$$

Since the possible carrier phase measurement error is believed to be much smaller than the code error at the current epoch, Equation (2-20) updates the carrier-smoothed code based only on the carrier phase measurement. Thus, the erroneous code measurement is rejected, but the smoothing filter does not need to be reset.

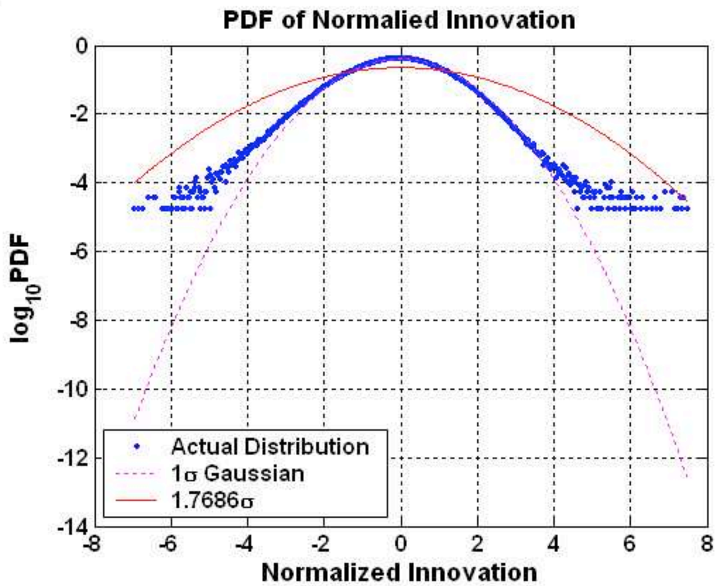

Figure 2-12-A

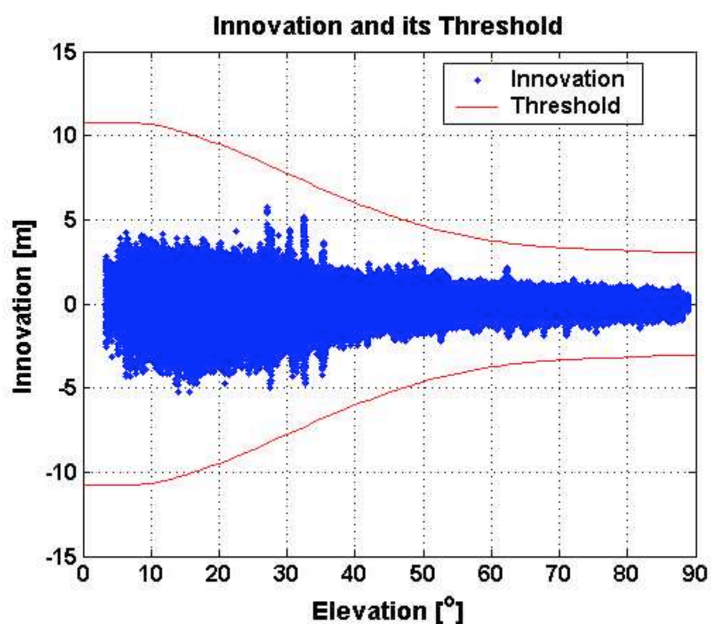

Figure 2-12-B

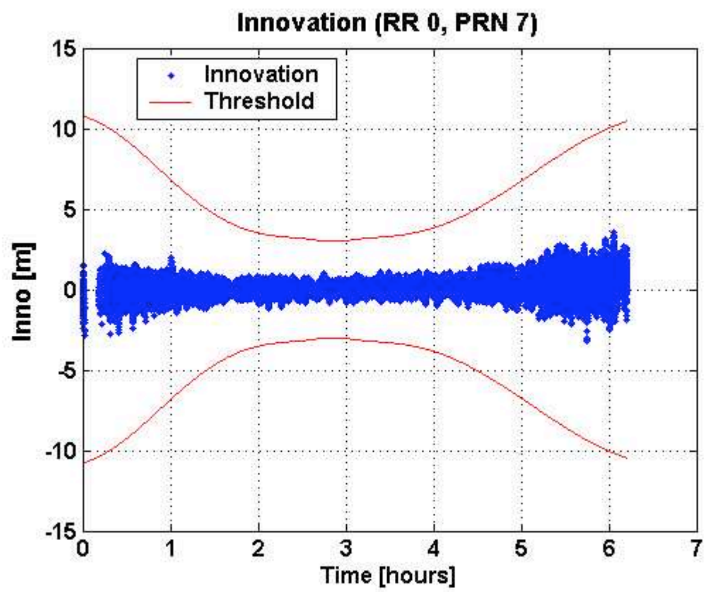

Figure 2-12-C

Figure 2-12 MQM CSC Innovation Threshold and Nominal Test Results

2.7 Phase One of Executive Monitoring (EXM-I)

In the previous sections, the SQM, DQM, and MQM integrity monitor algorithms were discussed. Each monitor algorithm is targeted at detecting certain failures and may generate one flag per receiver-satellite pair. This section will explain how the first phase of EXM (EXM-I) acts on any quality monitor (QM) flag and excludes measurements with at least one flag from any of SQM, DQM, and MQM. The measurements that survive EXM-I can then enter the second phase (the functions below the EXM block in Figure 2-1 on page 23).

In the current IMT implementation, two matrices, called tracking (**T**) and decision (**D**) matrices, are created to support EXM. Each entry in the matrices is for a single channel (a satellite tracked on one reference receiver). The **T**-matrix records which satellites are physically tracked on each receiver. The **D**-matrix is constructed by combing the three QM flags in a logical-OR operation (i.e., a flag in any QM algorithm leads to a flag in **D**).

The details of EXM-I fault handling logic and its 11 isolation action cases are given in [26]. All but two of these cases can be regarded as combinations of three fundamental situations:

1. Flag on a single satellite on a single reference receiver;
2. Flags on a single satellite and multiple reference receivers;
3. Flags on multiple satellites on a single reference receiver.

Resolution is straightforward in Case 1: the single flagged measurement is excluded. Cases 2 and 3, where multiple flags exist on a single satellite or reference receiver, lead to the exclusion of that satellite or receiver entirely. When combinations of Cases 2 and 3 occur, all suspect satellites and receivers are excluded – no attempt is made to guess whether a satellite or receiver is the cause. These rules are conservative but make sense when not enough redundancy is present to clearly determine which elements have failed. The LGF continuity risk probabilities allocated to satellite or LGF hardware failures are on the order of 10^{-6} per 15 seconds, and the EXM exclusion logic is based on the assumption that failures and fault-free alarm are truly rare. Therefore, any cases with multiple flags should be treated very cautiously.

PRN #	1	2	3	4	11	12	21	31
RR 0	√	√	√	√	√	√	×	√
RR 1	√	√	√	√	√	√	√	/
RR 2	√	√	√	√	×	/	√	√

√: good ×: bad / : not tracked

Figure 2-13 An Example of Selecting a Common Set

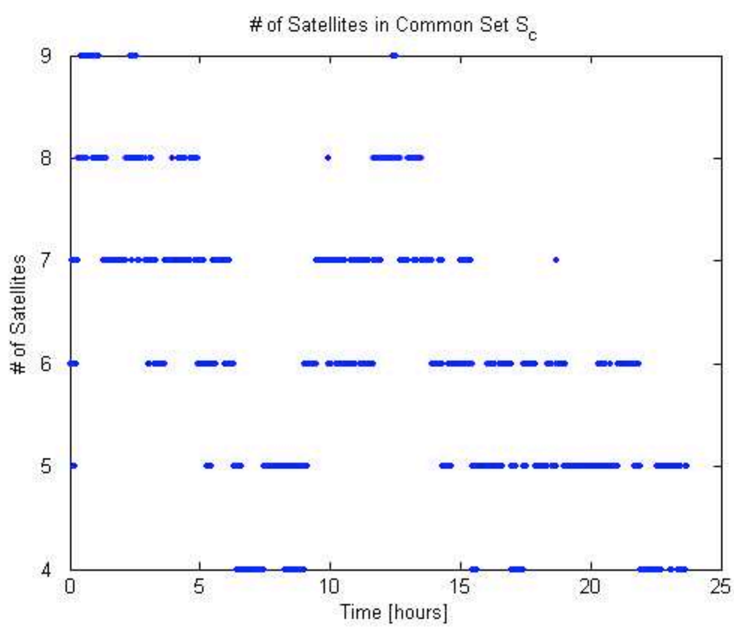

Figure 2-14 An Example of the Number of Satellites in the Common Set

A common set of visible satellites that survive EXM-I is selected so that the receiver clock adjustment calculations (as will be discussed in Section 2.8) use the same set of satellites. The common set consists of at least four satellites that are all tracked on at least two common receivers. A common set that consists of all three receivers has higher priority than one consisting of two receivers. For example, Figure 2-13 shows a table similar to the form of the **D** and **T**-matrices. The common set that the IMT will select is the one circled by the solid line instead of the dashed line. If fewer than four satellites are tracked by all three receivers, the largest common set that can be formed from any two receivers is chosen. If a common set of at least four satellites cannot be approved, and this event was not predicted by the LGF "constellation alert" algorithm (which considers known satellite outages), all measurements are excluded, and the IMT is reset. Figure 2-14 shows an example of the number of satellites in the common set over time. These satellites are tracked on all three receivers.

2.8 Multiple Reference Consistency Check (MRCC)

After EXM-I, the remaining measurements are used to compute candidate corrections. Next, B-values are computed to express the consistency of these corrections for each satellite across all reference receivers. MRCC is a process of computing and checking these B-values to be able to broadcast them to users and to isolate any receivers or receiver channels that create anomalously large errors in the corrections.

After computing the individual corrections for pseudorange and carrier phase for each channel using Equations (2-5) and (2-6), a receiver clock adjusted correction is computed as follows to allow measurements to be compared across receivers [81]:

$$\rho_{sca,m,n}(k) = \rho_{sc,m,n}(k) - \frac{1}{N_c(k)} \sum_{j \in S_c(k)} \rho_{sc,m,j}(k) \tag{2-21}$$

$$\phi_{ca,m,n}(k) = \phi_{c,m,n}(k) - \frac{1}{N_c(k)} \sum_{j \in S_c(k)} \phi_{c,m,j}(k) \tag{2-22}$$

The receiver clock adjustment also makes the broadcast corrections as close as possible to zero in order to minimize the number of bits required to send. S_c denotes a set of N_c

ranging sources in the maximum common set (as explained in Section 2.7). The same set of satellites should be used in Equations (2-21) and (2-22).

B-values are generated by the following formulas [81]:

$$B_{\rho,m,n}(k) = \frac{1}{M_n(k)} \sum_{i \in S_n(k)} \rho_{sca,i,n}(k) - \frac{1}{M_n(k)-1} \sum_{\substack{i \in S_n(k) \\ i \neq m}} \rho_{sca,i,n}(k) \tag{2-23}$$

$$\begin{aligned} B_{\phi,m,n}(k) = {} & \frac{1}{M_n(k)} \sum_{i \in S_n(k)} \left(\phi_{ca,i,n}(k) - \phi_{ca,i,n}(0)\right) \\ & - \frac{1}{M_n(k)-1} \sum_{\substack{i \in S_n(k) \\ i \neq m}} \left(\phi_{ca,i,n}(k) - \phi_{ca,i,n}(0)\right) \end{aligned} \tag{2-24}$$

where $\phi_{ca,m,n}(0)$ is evaluated at the first measurement epoch for channel (m,n). The B-values need to be reset every time the common set, S_c, changes or when Receiver m loses tracking on Satellite n. Note that the sum of all B-values from a satellite is exactly equal to zero.

B-values are best thought of as estimates of pseudorange error under the hypothesis that a given reference receiver has failed. The first term in the right side of Equation (2-23) is the averaged satellite pseudorange correction, $\rho_{corr,n}(k)$, as will be discussed in Section 2.10. In the case that a given reference receiver has failed, the best estimate of the "true" pseudorange correction is the correction generated by taking the average of all receivers except the one hypothesized to have failed. This is the second term in the right side of Equation (2-23). Note that the calculation of the first term, $\rho_{corr,n}(k)$, includes the supposedly failed receiver. The difference between these two terms is the B-value that is broadcast to users so that they can compute the position error bound that applies to this hypothesis [33,56].

Not every channel can have a B-value. The following rules can be used to determine whether a channel has a B-value:

1. If S_c consists of three receivers, all channels in S_c will have B-values. Satellites tracked on two receivers can have B-values as well.

2. If S_c consists of two receivers, only channels in S_c will have B-values.
3. If S_c does not exist, then no B-values (or pseudorange corrections as discussed in Section 2.10) can be generated and no LAAS service can be provided at this moment.

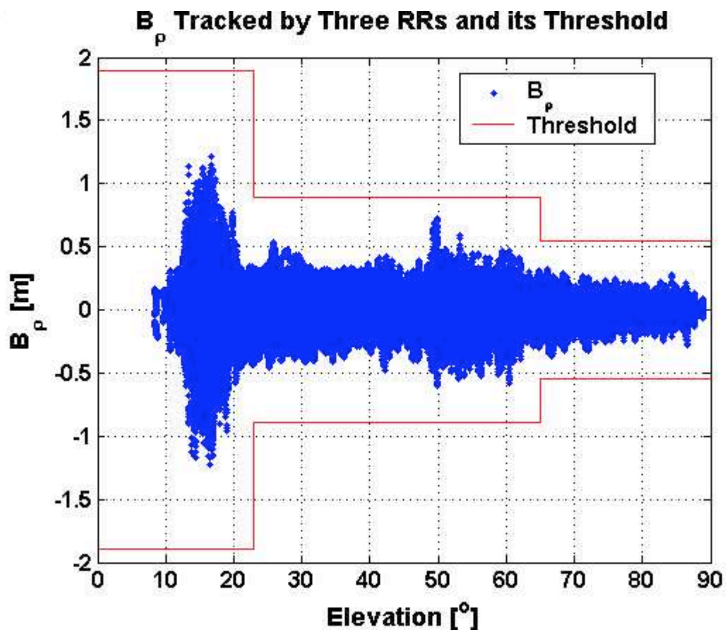

Figure 2-15-A

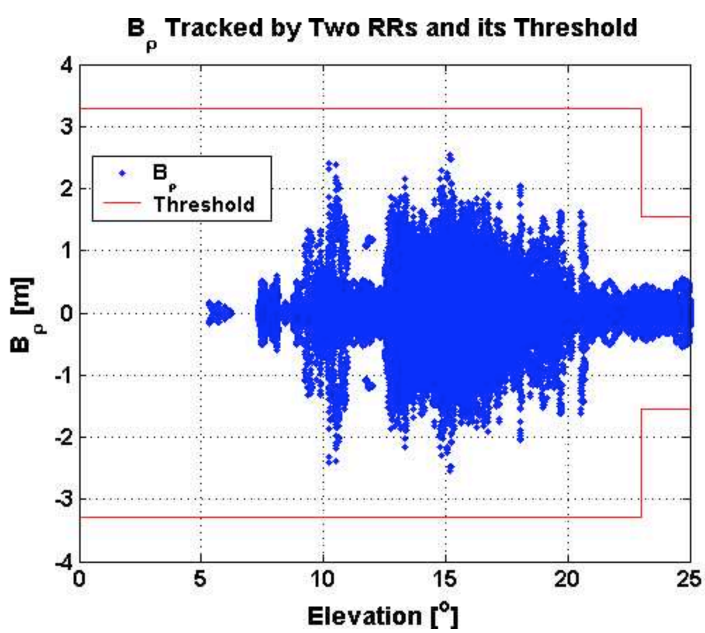

Figure 2-15-B

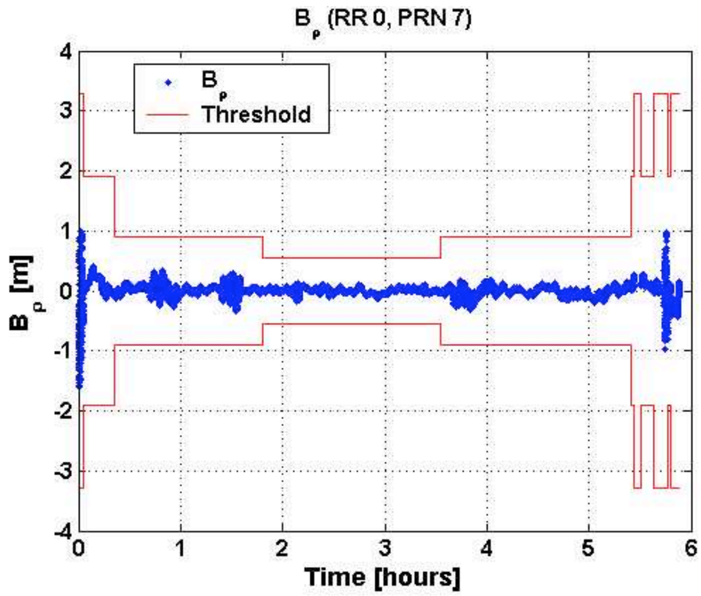

Figure 2-15-C

Figure 2-15 B_ρ Threshold and Nominal Test Results

The nominal test results and thresholds of B_ρ are illustrated in Figure 2-15. Plot (A) shows B_ρ from those satellites tracked by three receivers. We use a three-step function to model its thresholds, and the magnitude of each step is obtained by the Gaussian overbounding method described in Section 2.2. Plot (B) shows B_ρ from satellites that have only two B_ρ, and the threshold is about $\sqrt{3}$ times larger than in Plot (A) at the same elevation. Plot (C) shows an example of B_ρ from Channel (0, 7). The threshold frequently jumps up and down during the last 30 minutes, since PRN 7 sets during this period of time causing the receivers to frequently lose tracking on this satellite at low elevations.

B_ϕ is another type of B-value based on carrier-phase measurements. Calculating B_ϕ is not required in CAT I LAAS, since CAT I LAAS does not broadcast carrier phase corrections. The current IMT also calculates B_ϕ for research purposes. It assumes that all calculated B_ϕ can never exceed the threshold for B_ϕ and that there is no flag generated from the B_ϕ test.

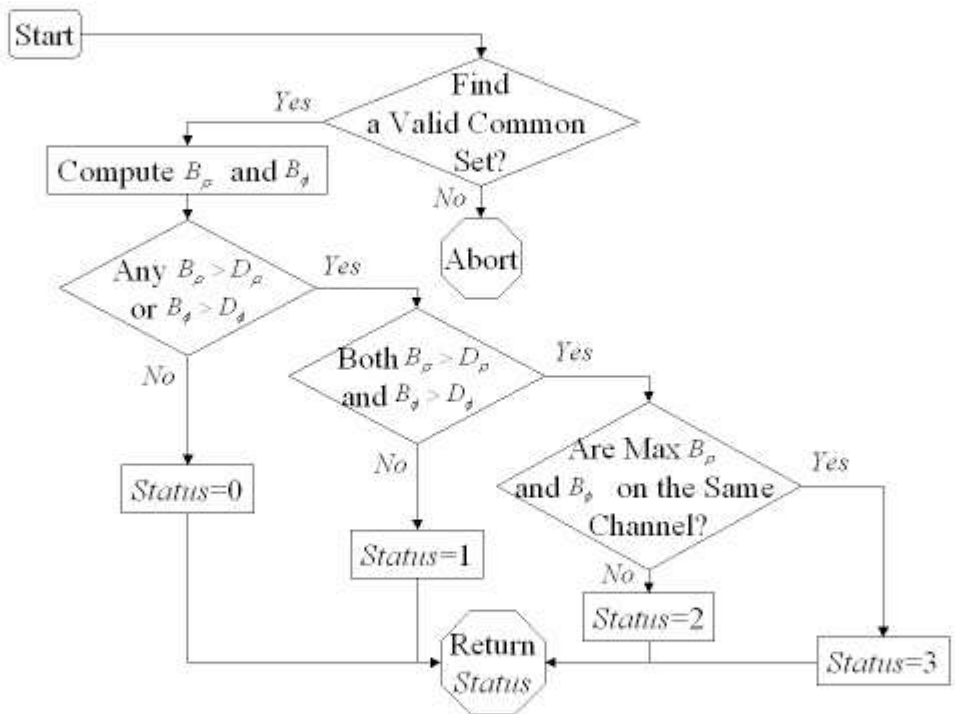

Figure 2-16 MRCC Fault Detection Flowchart

After B-values for both pseudorange and carrier-phase measurements are computed, the next step of MRCC is to compare each B-value with its threshold. If any B-values exceed their thresholds, the IMT finds the largest B_ρ and B_ϕ that are over the thresholds (currently there is no B_ϕ flag; therefore, no maximum B_ϕ needs to be found). Figure 2-16 is the diagram of the MRCC fault detection process, which has some similarities with the MQM determination logic in Figure 2-11. This process basically determines whether there are any B-values over threshold, whether they are large B-values of different types (B_ρ and B_ϕ), and whether the different types of the largest B-values are on the same channel.

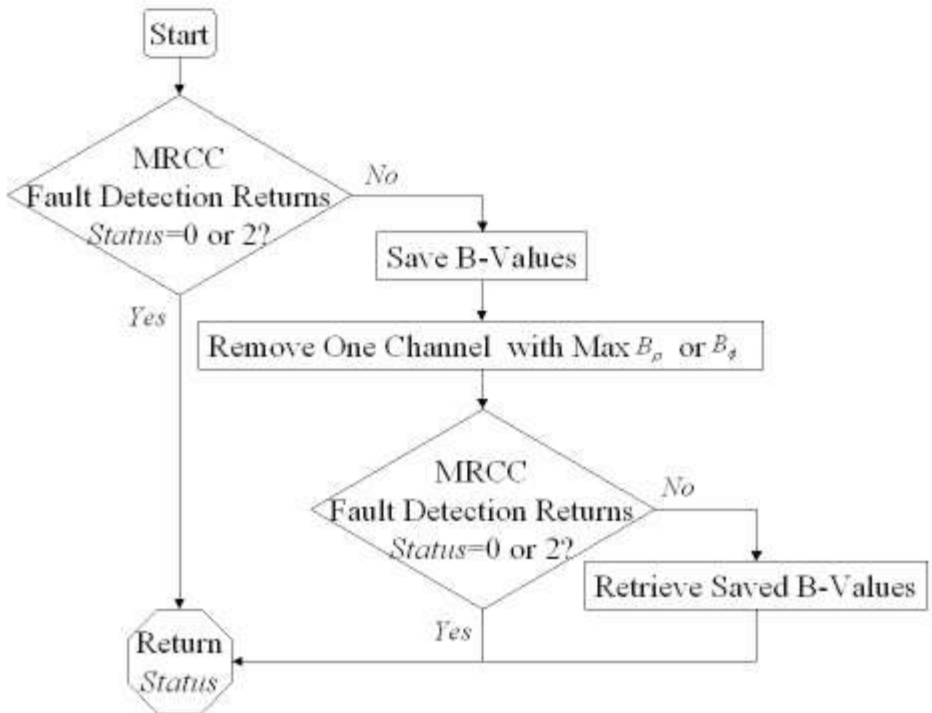

Figure 2-17 EXM-II Pre-Screen Flowchart

The MRCC isolation procedure, which is also called the EXM-II "pre-screen," picks up the detection status returned from Figure 2-16 and continues to isolate errors. Figure 2-17 illustrates this procedure. If MRCC fault detection returns a status of 0, i.e., no B-values are over threshold, then the MRCC isolation procedure does nothing. If the status is 2, i.e., the largest B_ρ and B_ϕ are not on the same channel, the pre-screen is not able to handle this complex situation. MRCC then passes the task to the similar, but more comprehensive, logic of EXM-II, as will be discussed in Section 2.11. In all other cases, it removes the channel with the largest B-value and then re-computes B-values with a reduced common set. If no MRCC flags remain, the pre-screen was successful. The reason for the pre-screen is that a single measurement fault will affect multiple B-values due to clock adjustment and correction averaging. If MRCC flags still exist, it means that the pre-screen may have mishandled the failure, so the original B-values are retrieved and MRCC also passes the task to EXM-II.

2.9 σμ-Monitor

The B-values resulting from MRCC are used as inputs to the σμ-monitor. This monitor is designed to guarantee that the pseudorange correction error distribution is bounded by a zero-mean Gaussian distribution with a broadcast σ_{pr_gnd} value. Two different monitors, the estimation and Cumulative Sum (CUSUM) algorithms, have been implemented for both mean and sigma violations. The details of these monitors are covered in [28], and the CUSUM algorithm will be explained in Chapter 4.

2.10 Message Field Range Test (MFRT)

The MFRT is executed as the last step in EXM-II before measurements are approved for broadcast. It confirms that the computed averaged pseudorange corrections, ρ_{corr}, are within a bound of ± 125 meters, and that the correction rates, $R_{\rho_{corr}}$, are also within a threshold of ± 0.8 m/s [64]. These thresholds are determined based on nominal data from low-elevation satellites using the method discussed in Section 2.2, and then the same thresholds are applied at other elevations to protect continuity.

The term ρ_{corr} is actually the first term in the right side of Equation (2-23), i.e.,

$$\rho_{corr,n}(k) = \frac{1}{M_n(k)} \sum_{i \in S_n(k)} \rho_{sca,i,n}(k) \tag{2-25}$$

$R_{\rho_{corr}}$ is the change rate of ρ_{corr} and is computed by

$$R_{\rho_{corr},n}(k) = \frac{1}{T_s}\left(\rho_{corr,n}(k) - \rho_{corr,n}(k-1)\right) \tag{2-26}$$

Recall from Chapter 1 that CAT I LAAS has a Vertical Alert Limit (VAL) of 10 meters. Consequently, carrier phase corrections are not required. However, they may be needed for CAT II and III because of their more demanding VAL of 5.3 meters. In this case, carrier-phase corrections, ϕ_{corr}, can be similarly calculated by

$$\phi_{corr,n}(k) = \frac{1}{M_n(k)} \sum_{i \in S_n(k)} \left(\phi_{ca,i,n}(k) - \phi_{ca,i,n}(0)\right) \tag{2-27}$$

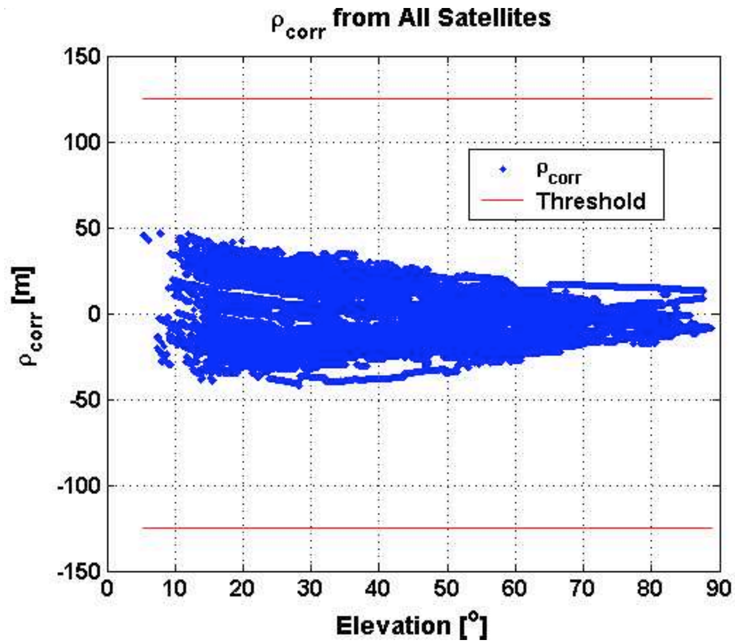

Figure 2-18-A

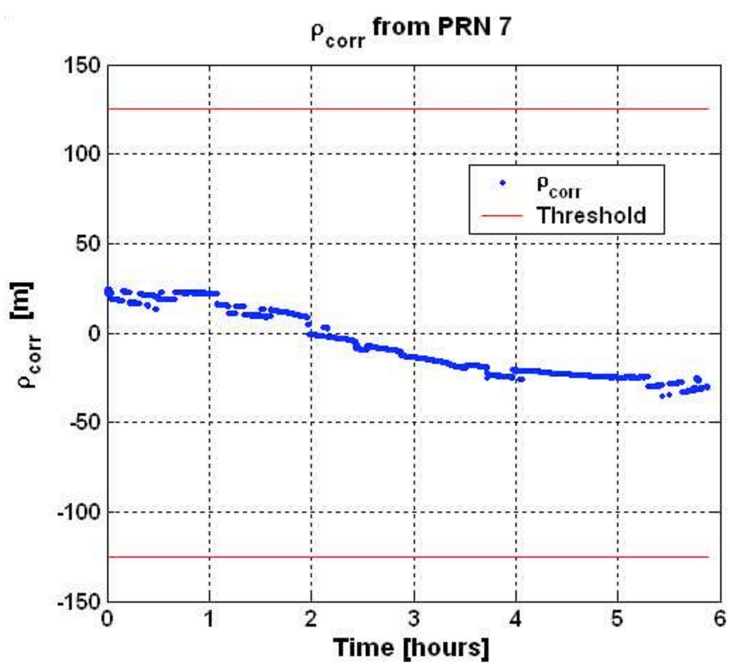

Figure 2-18-B

Figure 2-18 Nominal ρ_{corr} Test Results

Nominal ρ_{corr} test results are shows in Figure 2-18. Plot (A) in Figure 2-18 indicates that ρ_{corr} appears elevation dependent, though currently a constant threshold is applied. Plot (B) shows gradual changes in the actual satellite clock, ionosphere, troposphere (no ionosphere or troposphere corrections are applied), and IMT multipath punctuated by occasional jumps. Small jumps (typically under one meter) occur when the satellite navigation data changes and is revalidated by DQM. These jumps occur on the aircraft as well (it switches ephemeris on the same epoch) and are thus not a concern. However, several back-and-forth jumps appear to be caused by changes in the common set used to perform clock corrections. When a low-elevation satellite is tracked on two of three IMT receivers but loses and regains lock on the third receiver, it comes in and out of the common set as currently constructed, which means that the ionosphere and troposphere errors on the low-elevation satellite (which could be tens of meters near solar maximum) come in and out of the satellite clock correction, causing ρ_{corr} jumps of as high as several meters.

Common-set-driven jumps in ρ_{corr} do not threaten users because the corrections for all approved satellites jump by the same amount; thus, the common change in user pseudoranges after applying LGF corrections goes entirely into the user clock bias solution and does not affect user position solutions. However, frequent common-set-changes stress EXM processing and should be reduced to the extent possible. The current IMT limits the common set to satellites above 10 degrees elevation in order to minimize the in-and-out common set issue.

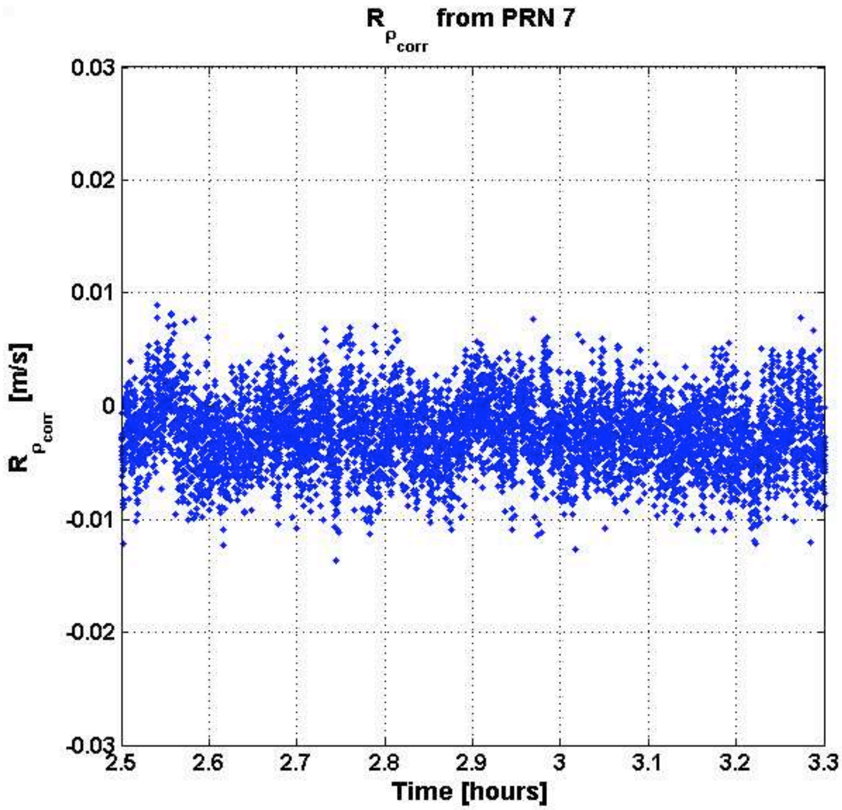

Figure 2-19 Nominal $R_{\rho_{corr}}$ Test Results

All these jumps in ρ_{corr} can result in large $R_{\rho_{corr}}$ amplitudes which we are trying to avoid with some patches in the algorithm and software. Figure 2-19 shows a segment of nominal $R_{\rho_{corr}}$ test results that are well bounded by the threshold ± 0.8 m/s.

2.11 Phase Two of Executive Monitoring (EXM-II)

Once corrections, B-values, σμ, and MRCC flags have been generated, the second phase of executive monitoring (EXM-II) is performed. This section will explain how EXM-II coordinates these flags and prevents faulty measurements from being sent to approaching aircraft by the LAAS VHF data broadcast.

The logic used to handle MRCC flags is complicated because each measurement exclusion attempt usually requires the IMT to reduce the common set, S_c. When this happens, corrections and B-values must be recomputed for all remaining measurements, and then EXM-II must confirm that this new, reduced set of measurements passes MRCC.

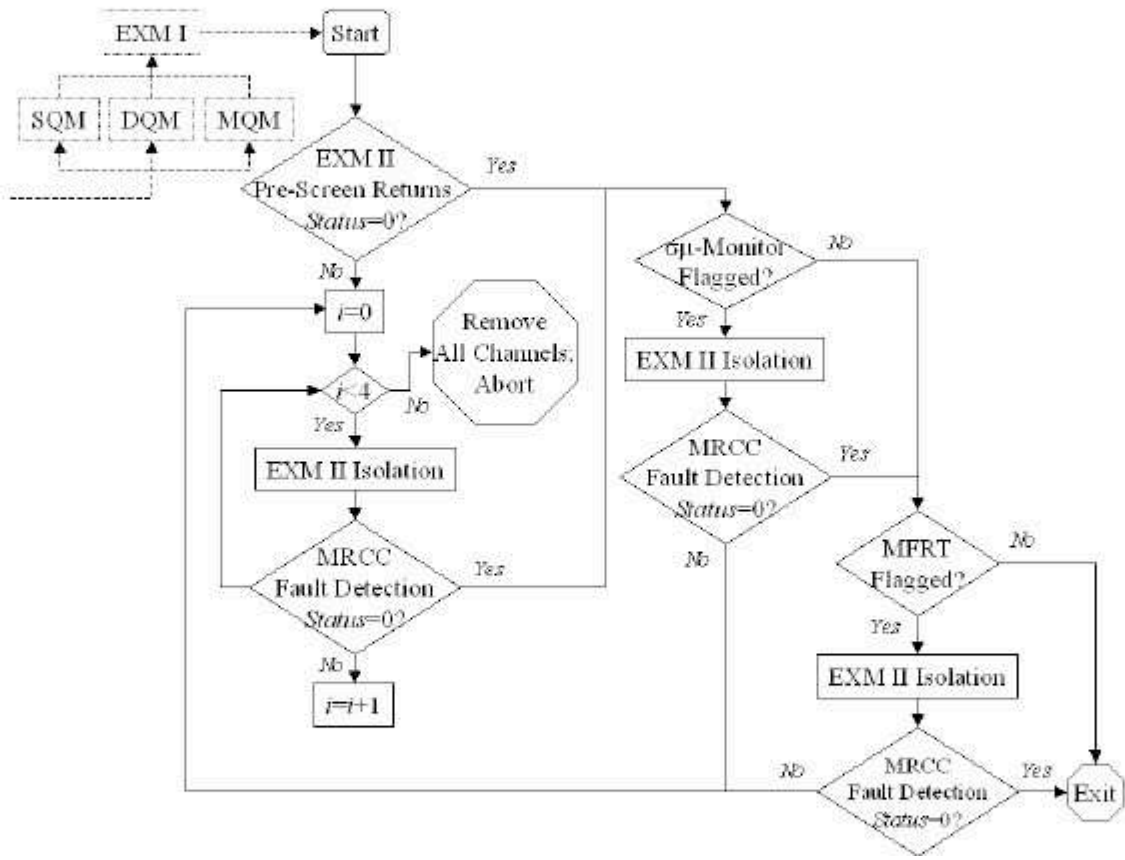

Figure 2-20 EXM-II Flowchart

Figure 2-20 shows a diagram of the EXM-II flowchart. In this diagram, the block named "EXM-II Pre-Screen" was explained in Figure 2-17 on page 61. When the EXM-II pre-screen fails to isolate failures, it then passes the task to the block named "EXM-II Isolation." The block named "EXM-II Isolation" operates using logic similar to EXM-I. In other words, single B-value flags on individual channels are removed, but if more than one B-value is flagged on a given satellite or receiver, the entire satellite or receiver is excluded. In order to bound the processing time required by EXM-II, the IMT exercises a limit of four attempts to exclude measurements and repeat the EXM-II isolation-MRCC fault detection process before it "gives up" and excludes all measurements. It should be very rare for EXM-II to need to make more than one or two exclusions. Meanwhile, all measurements must also pass the σμ-monitor and MFRT tests – any flags from these tests also necessitate EXM-II exclusions.

Finally, EXM-II is responsible for managing excluded ranging sources and/or receiver channels. If measurements are excluded due to failures detected by the σμ-monitor or MFRT, they will be excluded for the entire period in which the satellite in question passes over the LGF because, for the faults targeted by these monitors, it is difficult to verify during the current pass that a detected failure is no longer present. In most other cases, excluded measurements enter "self-recovery" mode, in which the carrier smoothing filters are re-initialized. During self-recovery, if the re-smoothed measurement passes all tests with thresholds tightened to 3-sigma levels, the measurement is declared safe for use again and is reintegrated into the set of usable measurements [26,33]. Tightened thresholds are also applied to those measurements that were excluded because of σμ-monitor or MFRT flags in the previous pass. Tightened thresholds at 3-sigma levels help insure that the probability of reintroducing a still failed measurement is below 1.94×10^{-9} [81], assuming the corresponding missed-detection probability, P_{MD}, is equal to 0.001. P_{MD} will be explained in Section 3.9.2 in detail. If a failed channel does not pass the tightened thresholds, it enters self-recovery mode again and repeats the recovery process. If self-recovery is failed a second time, the affected channel enters "external maintenance" mode and cannot be used again until the affected equipment is certified to be healthy by an external agency, such as FAA maintenance personnel [26].

Chapter 3 Monitor Step Responses

3.1 Objective

Integrity monitors are designed to detect certain hazardous failures. If they fail to detect a hazardous failure, erroneous measurements due to the failure can result in hazardously misleading information (HMI) in LAAS differential corrections and then lead to a large aircraft positioning error.

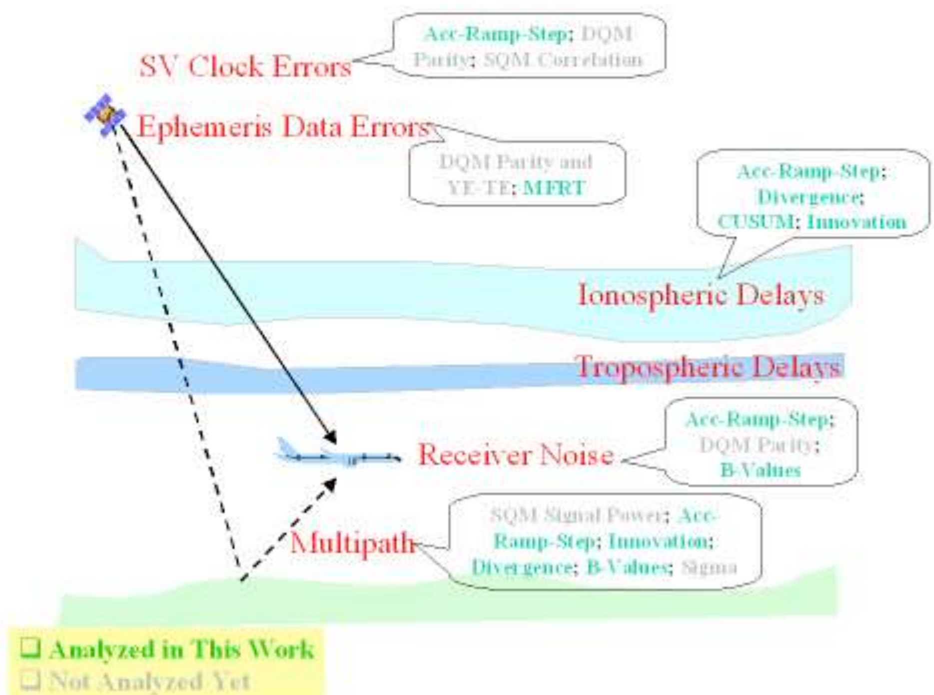

Figure 3-1 IMT Monitors in the Entire Set of Threats

In order to check whether the IMT can detect the entire set of possible threats, it is important to first identify which IMT integrity monitors can detect which failures. Figure 3-1 categorizes the integrity monitors into groups so that all integrity monitors in each group target the same failure source. For example, the acceleration-ramp-step, GMA divergence, innovation, and the CUSUM (developed in Chapter 4) monitors can all detect ionospheric gradients but with a different detection performance. Therefore, it is also important to examine the degree to which each monitor can reliably respond to various failure modes. Given the knowledge of each monitor's behavior under different failure conditions, the overall IMT system integrity and performance can be evaluated. Highlighted monitors in Figure 3-1 are analyzed in detail, while those in gray are not examined in this thesis due to the complexity of their algorithms or even lack of analytical solutions. The work here is the first comprehensive analysis of an integrity monitor suite.

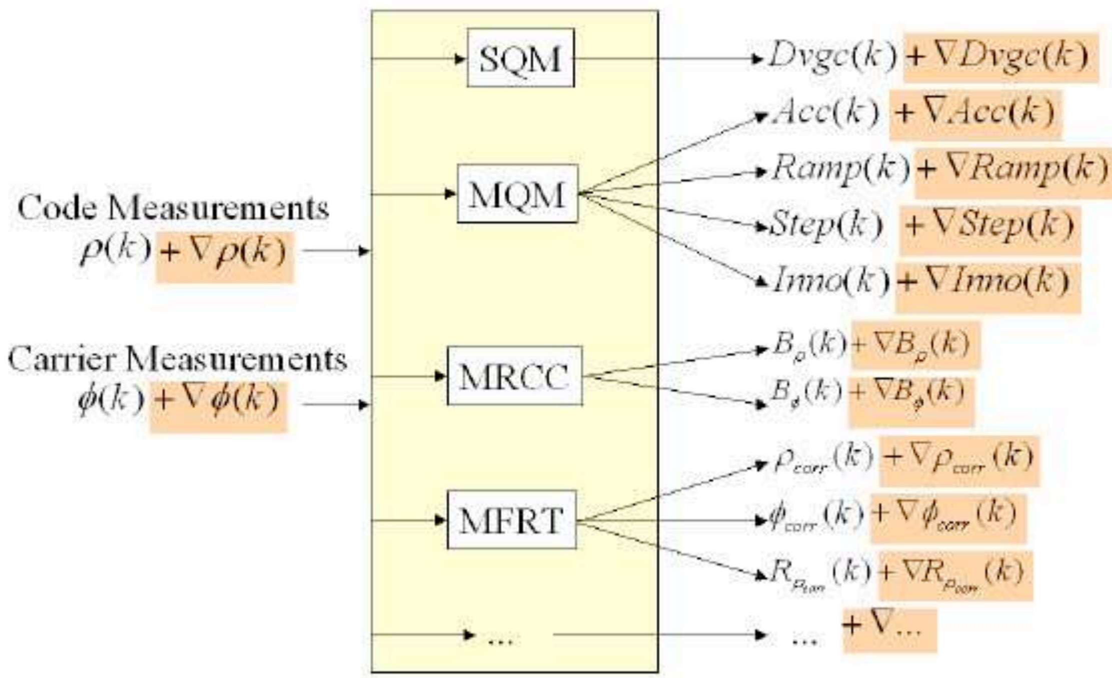

Figure 3-2 The Objective of Failure Effects Analysis

The IMT and its integrity monitors were discussed in Chapter 2. When code and carrier phase measurements are delivered into the IMT, a set of test statistics is calculated, as illustrated in Figure 3-2. However, it is often difficult to identify which monitors can detect what kinds of failures by just inspecting their algorithms and equations. A major reason for this is that the connections between threat symptoms and monitor test statistics are complicated and indirect. For example, the divergence test statistic, $Dvgc$, is the smoothed version of raw divergence, dz, where dz is the change rate of z, while z is code-minus-carrier, as expressed in Equations (2-8)-(2-10). A second example is the innovation test statistic from Equation (2-19). The innovation method depends on both code and carrier phase measurements; however, it actually never detects satellite clock failures that affect both code and carrier phase measurement errors. If the satellite clock failures lead to the same size errors in both code and carrier phase measurements, then the innovation test will not detect the fault, because it is based on carrier-smoothed code measurements. This chapter analyzes the ability to detect several known hazards across the set of IMT monitors.

Failure Modes	Channel (r, p)		Satellite p		Receiver r
	Code Meas.	Phase Meas.	Satellite Clock	Ionospheric Divergence	Receiver Clock
Acc, $Ramp$ and $Step$					
$Inno$					
$Dvgc$					
B_ρ					
B_ϕ					
ρ_{corr}					
ϕ_{corr}					
$R_{\rho_{corr}}$					

Table 3-1 Effects of Known Failure Modes on IMT Monitors (Goal)

As illustrated in Figure 3-2, this chapter derives the mathematical relationship between failure inputs ($\nabla\rho$ and $\nabla\phi$) and failure effects, highlighted in red, on these test statistics. By manipulating and simplifying these equations, the failure effects can be obtained directly in terms of $\nabla\rho$ and $\nabla\phi$. It is then possible to determine which monitors detect which failures. One goal of this chapter is to complete Table 3-1 analytically to present a clear picture about the responses of each test statistic (row) under each failure mode (column). More importantly, based on the analysis, the performance of integrity monitors can also be evaluated quantitatively. As will be seen in Chapter 5, this analysis also provides valuable information to construct effective failure-specific tests.

3.2 Channel Step Failures

Failures can occur in both the LAAS ground segment and the GPS space segment, and they can be in the form of a step, ramp, or any other complicated function. Thus, not all failures can be easily modeled analytically. In this chapter, our focus is to analyze failures in the range domain, since most failures appear as disturbances in code, ρ, and/or carrier, ϕ, measurements.

If a certain failure occurs at epoch k_0, a test statistic s on a channel (r, p), denoted by $s_{r,p}(k)$, can be written in the following general form:

$$s^{\times}_{r,p}(k) = s_{r,p}(k) + \nabla s_{r,p}(k) \tag{3-1}$$

i.e., the original nominal response, $s_{r,p}(k)$, becomes $s^{\times}_{r,p}(k)$ under the failure condition. Their difference, $\nabla s_{r,p}(k)$, is the failure effect and is of interest in this chapter. Since the failure starts at k_0, $\nabla s_{r,p}(k)$ can also be written as $\nabla s_{r,p}(k) \cdot u(k-k_0)$, where $u(k)$ is the discrete-time unit step function, i.e.,

$$u(k) = \begin{cases} 1, & k \geq 0 \\ 0, & else \end{cases} \tag{3-2}$$

The Dirac delta function, $\delta(k)$, is also used in this chapter, and its definition is:

$$\delta(k) = \begin{cases} 1, & k = 0 \\ 0, & else \end{cases} \tag{3-3}$$

If $\nabla s_{r,p}(k)$ is zero, we can easily conclude that the monitor is not designed to detect the corresponding failure. The monitor may, however, be capable of quickly detecting the failure if $\nabla s_{r,p}(k)$ changes sufficiently and immediately.

Note that in the derivations that follow, characters m, r, and i are used to index reference receivers, while n, p, and j index satellites.

3.2.1 General Failure Model

The analysis starts with a generic failure model in which a step failure is injected into both the code and carrier measurements of channel (r, p) starting from k_0, i.e.,

$$\rho^{\times}_{r,p}(k) = \rho_{r,p}(k) + \ell \cdot u(k - k_0) \tag{3-4}$$

$$\phi^{\times}_{r,p}(k) = \phi_{r,p}(k) + \theta \cdot u(k - k_0) \tag{3-5}$$

where the magnitudes ℓ and θ can be independent but do not need to be. This channel step failure model is simple, but it is very important. First, step or step-like failures do occur in practice (e.g., receiver carrier-phase cycle slips). Second, any arbitrary error can be mathematically constructed as the sum of a sequence of delayed step functions. In other words, after the step responses of all integrity monitors under this general failure model are obtained, responses under any other complicated failure condition can be calculated. Superposition can be used since linearity exists between failure inputs and responses for most integrity monitors, as will be seen later. Third, satellite and receiver failures can be decomposed to multiple channel failures as well. Therefore, the analysis of a step failure occurring on a single channel is the foundation of this whole chapter.

In this failure model, measurements from an individual channel are assumed not to interfere with those from other channels in the same GPS receiver. This assumption is reasonable since the possible interference among different channels can be nearly negligible compared to failures.

As pointed to in Section 2.3, carrier smoothing and the generation of differential corrections are the backbone of LGF processing under nominal conditions. Several integrity monitors are even designed to detect failures in carrier-smoothed code measurements, $\rho_{s,r,p}$. Therefore, the analysis begins with the calculation of $\rho_{s,r,p}$. Starting from Equation (2-3), $\rho_{s,r,p}$ under the failure becomes:

$$\rho^{\times}_{s,r,p}(k) = \rho_{s,r,p}(k) + \nabla\rho_{s,r,p}(k) \tag{3-6}$$

At $k = k_0$,

$$\begin{aligned}
\rho^{\times}_{s,r,p}(k_0) &= \frac{1}{N_s}\rho^{\times}_{r,p}(k_0) + \frac{N_s - 1}{N_s}\left(\rho^{\times}_{s,r,p}(k_0 - 1) + \phi^{\times}_{r,p}(k_0) - \phi^{\times}_{r,p}(k_0 - 1)\right) \\
&= \frac{1}{N_s}\left(\rho_{r,p}(k_0) + \ell\right) \\
&\quad + \frac{N_s - 1}{N_s}\left(\rho_{s,r,p}(k_0 - 1) + \nabla\rho_{s,r,p}(k_0 - 1) + \phi_{r,p}(k_0) + \theta - \phi_{r,p}(k_0 - 1)\right) \\
&= \rho_{s,r,p}(k_0) + \left(\ell - \frac{N_s - 1}{N_s}(\ell - \theta)\right)
\end{aligned} \tag{3-7}$$

At any epoch $k > k_0$,

$$\begin{aligned}
\rho^{\times}_{s,r,p}(k) &= \frac{1}{N_s}\rho^{\times}_{r,p}(k) + \frac{N_s - 1}{N_s}\left(\rho^{\times}_{s,r,p}(k - 1) + \phi^{\times}_{r,p}(k) - \phi^{\times}_{r,p}(k - 1)\right) \\
&= \frac{1}{N_s}\left(\rho_{r,p}(k) + \ell\right) \\
&\quad + \frac{N_s - 1}{N_s}\left(\rho_{s,r,p}(k - 1) + \nabla\rho_{s,r,p}(k - 1) + \phi_{r,p}(k) + \theta - \phi_{r,p}(k - 1) - \theta\right) \\
&= \rho_{s,r,p}(k) + \frac{1}{N_s}\left(\ell + (N_s - 1)\cdot\nabla\rho_{s,r,p}(k - 1)\right) \\
&= \rho_{s,r,p}(k) + \left(\ell + \left(\frac{N_s - 1}{N_s}\right)^{k-k_0}\left(\nabla\rho_{s,r,p}(k_0) - \ell\right)\right) \\
&= \rho_{s,r,p}(k) + \left(\ell - \left(\frac{N_s - 1}{N_s}\right)^{k-k_0+1}(\ell - \theta)\right)
\end{aligned} \tag{3-8}$$

Equations (3-7) and (3-8) together lead to:

$$\nabla\rho_{s,r,p}(k) = \left(\ell - \left(\frac{N_s - 1}{N_s}\right)^{k-k_0+1}(\ell - \theta)\right)\cdot u(k - k_0) \tag{3-9}$$

Equations (2-5) and (2-6) then become:

$$\begin{aligned}\rho^{\times}_{sc,r,p}(k) &= \rho^{\times}_{s,r,p}(k) - R^{\times}_{r,p}(k) - \tau^{\times}_{r,p}(k) \\ &= \rho_{s,r,p}(k) + \nabla\rho_{s,r,p}(k) - R_{r,p}(k) - \tau_{r,p}(k) \\ &= \rho_{sc,r,p}(k) + \nabla\rho_{s,r,p}(k)\end{aligned} \tag{3-10}$$

and

$$\begin{aligned}\phi^{\times}_{c,r,p}(k) &= \phi^{\times}_{r,p}(k) - R^{\times}_{r,p}(k) - \tau^{\times}_{r,p}(k) - \phi^{\times}_{ci,r,p}(k) \\ &= \phi_{r,p}(k) + \theta\cdot u(k-k_0) - R_{r,p}(k) - \tau_{r,p}(k) - \phi_{ci,r,p}(k) \\ &= \phi_{c,r,p}(k) + \theta\cdot u(k-k_0)\end{aligned} \tag{3-11}$$

Several test statistics, such as the acceleration-ramp-step, B_{ϕ}, and ϕ_{corr}, rely only on carrier phase measurements. Equation (3-11) confirms that they will not respond to code failures.

In the following sections, the failure effects on each integrity monitor will be analyzed under this general failure model.

3.2.2 Innovation Failure Effects

Equation (2-19) implies that innovation test statistics are channel-based; i.e., one channel's innovation is independent of any other channel measurements. Therefore, the most general failure has no impact on the innovation of other channels except the failed channel (r, p).

From Equations (2-19), (3-4)-(3-6) and (3-9), the innovation at epoch $k = k_0$ is:

$$\begin{aligned}Inno^{\times}_{r,p}(k_0) &= \rho^{\times}_{r,p}(k_0) - \left(\rho^{\times}_{s,r,p}(k_0-1) + \phi^{\times}_{r,p}(k_0) - \phi^{\times}_{r,p}(k_0-1)\right) \\ &= \rho_{r,p}(k_0) + \ell - \left(\rho_{s,r,p}(k_0-1) + \phi_{r,p}(k_0) + \theta - \phi_{r,p}(k_0-1)\right) \\ &= Inno_{r,p}(k_0) + (\ell - \theta)\end{aligned} \tag{3-12}$$

At any epoch $k > k_0$,

$$\begin{aligned}
Inno^{\times}_{r,p}(k) &= \rho^{\times}_{r,p}(k) - \left(\rho^{\times}_{s,r,p}(k-1) + \phi^{\times}_{r,p}(k) - \phi^{\times}_{r,p}(k-1)\right) \\
&= \rho_{r,p}(k) + \ell - \left(\rho_{s,r,p}(k-1) + \nabla\rho_{s,r,p}(k-1) + \phi_{r,p}(k) + \theta - \phi_{r,p}(k-1) - \theta\right) \\
&= Inno_{r,p}(k) + \left(\ell - \nabla\rho_{s,r,p}(k-1)\right) \\
&= Inno_{r,p}(k) + \left(\frac{N_s - 1}{N_s}\right)^{k-k_0} \left(\ell - \theta\right)
\end{aligned} \tag{3-13}$$

Equations (3-12) and (3-13) can be combined as:

$$\begin{aligned}
Inno^{\times}_{r,p}(k) &= Inno_{r,p}(k) + \nabla Inno_{r,p}(k) \\
&= Inno_{r,p}(k) + \left(\frac{N_s - 1}{N_s}\right)^{k-k_0} \left(\ell - \theta\right) \cdot u(k-k_0)
\end{aligned} \tag{3-14}$$

Equation (3-14) shows that the magnitude of $\nabla Inno_{r,p}(k)$ has its maximum at k_0 and then decreases exponentially over time. Thus, the innovation test most likely catches channel step errors within the first several epochs after the error occurs.

3.2.3 Acceleration-Ramp-Step Failure Effects

Based on the assumptions made in Section 3.2.1, Equations (2-14)-(2-18) show that acceleration, ramp and step on a receiver are independent of other receivers, and that they do not depend on code measurements. Thus, only the channels on receiver r are of concern. In order to simplify the analysis, it is assumed that the same set of satellites is continuously tracked on receiver r during the period of time that is considered. In other words, $S_r(k)$ is equal to $S_r(k_0 - 1)$ for any $k \geq k_0$. The magnitude of the failure effects will be different only by a scaling factor if $S_r(k)$ changes.

For the failed channel (r, p),

$$\begin{aligned}
\phi^{*\times}_{r,p}(k) &= \phi^{\times}_{c,r,p}(k) - \frac{1}{N_r(k)} \sum_{j \in S_r(k)} \phi^{\times}_{c,r,j}(k) \\
&= \phi_{c,r,p}(k) + \theta \cdot u(k-k_0) - \frac{1}{N_r(k)} \left(\theta \cdot u(k-k_0) + \sum_{j \in S_r(k)} \phi_{c,r,j}(k)\right) \\
&= \phi^{*}_{r,p}(k) + \frac{N_r(k) - 1}{N_r(k)} \theta \cdot u(k-k_0)
\end{aligned} \tag{3-15}$$

For any other channel (r,n) with $n \neq p$,

$$\begin{aligned}\phi^{*\times}_{r,n\neq p}(k) &= \phi^{\times}_{c,r,n}(k) - \frac{1}{N_r(k)}\sum_{j\in S_r(k)}\phi^{\times}_{c,r,j}(k) \\ &= \phi_{c,r,n}(k) - \frac{1}{N_r(k)}\left(\theta\cdot u(k-k_0) + \sum_{j\in S_r(k)}\phi_{c,r,j}(k)\right) \\ &= \phi^{*}_{r,n}(k) - \frac{1}{N_r(k)}\theta\cdot u(k-k_0)\end{aligned} \tag{3-16}$$

At epochs from $k = k_0$ to $k_0 + 8$, the last ten points of $\phi^{*}_{r,n}(k)$, on which an interpolation curve in Equation (2-15) is based, are partially polluted by the failure. Since the interpolation curve is solved by the least-squares method, it is impossible to obtain the failure effects in closed form. However, they can still be determined analytically at other particular epochs. For a step statistic at $k = k_0$,

$$\begin{aligned}Step^{\times}_{r,p}(k_0) &= \phi^{*\times}_{r,p}(k_0) - \phi^{*\times}_{r,p}(k_0-1,t_{k_0}) \\ &= \left(\phi^{*}_{r,p}(k_0) + \frac{N_r(k_0)-1}{N_r(k_0)}\theta\right) - \phi^{*}_{r,p}(k_0-1,t_{k_0}) \\ &= Step_{r,p}(k_0) + \frac{N_r(k_0)-1}{N_r(k_0)}\theta\end{aligned} \tag{3-17}$$

At $k \geq k_0 + 9$, the failure has entered all ten points, so the interpolation curve has the same shape as the one without the failure, except for a bias in the constant term, i.e.,

$$\phi^{*\times}_{0,r,p}(k) = \phi^{*}_{0,r,p}(k) + \frac{N_r(k)-1}{N_r(k)}\theta \tag{3-18}$$

$$\phi^{*\times}_{0,r,n\neq p}(k) = \phi^{*}_{0,r,n}(k) - \frac{1}{N_r(k)}\theta \tag{3-19}$$

and, for example,

$$\begin{aligned}Step^{\times}_{r,p}(k) &= \phi^{*\times}_{r,p}(k) - \phi^{*\times}_{r,p}(k-1,t_k) \\ &= \phi^{*}_{r,p}(k) + \frac{N_r(k)-1}{N_r(k)}\theta - \left(\phi^{*}_{r,p}(k-1,t_k) + \frac{N_r(k)-1}{N_r(k)}\theta\right) \quad \text{for } k \geq k_0 + 10 \\ &= Step_{r,p}(k)\end{aligned} \tag{3-20}$$

More detailed derivations are not given here. The final failure effects for the acceleration-ramp-step test are summarized as follows:

$$
\begin{aligned}
Acc^{\times}_{r,n}(k) = {} & Acc_{r,n}(k)\cdot u(k_0-1-k) \\
& + Acc^{\times}_{r,n}(k)\cdot\big(u(k-k_0)-u(k-k_0-9)\big) \text{ for any } n \\
& + Acc_{r,n}(k)\cdot u(k-k_0-9)
\end{aligned}
\tag{3-21}
$$

$$
\begin{aligned}
Ramp^{\times}_{r,n}(k) = {} & Ramp_{r,n}(k)\cdot u(k_0-1-k) \\
& + Ramp^{\times}_{r,n}(k)\cdot\big(u(k-k_0)-u(k-k_0-9)\big) \text{ for any } n \\
& + Ramp_{r,n}(k)\cdot u(k-k_0-9)
\end{aligned}
\tag{3-22}
$$

$$
\begin{aligned}
Step^{\times}_{r,p}(k) = {} & Step_{r,p}(k)\cdot u(k_0-1-k) \\
& + \left(Step_{r,p}(k_0)+\frac{N_r(k_0)-1}{N_r(k_0)}\theta\right)\cdot\delta(k-k_0) \\
& + Step^{\times}_{r,p}(k)\cdot\big(u(k-k_0-1)-u(k-k_0-10)\big) \\
& + Step_{r,p}(k)\cdot u(k-k_0-10)
\end{aligned}
\tag{3-23}
$$

$$
\begin{aligned}
Step^{\times}_{r,n\neq p}(k) = {} & Step_{r,n}(k)\cdot u(k_0-1-k) \\
& + \left(Step_{r,n}(k_0)-\frac{1}{N_r(k_0)}\theta\right)\cdot\delta(k-k_0) \\
& + Step^{\times}_{r,n}(k)\cdot\big(u(k-k_0-1)-u(k-k_0-10)\big) \\
& + Step_{r,n}(k)\cdot u(k-k_0-10)
\end{aligned}
\tag{3-24}
$$

For example, Equation (3-21) states that the failure has an effect on acceleration at epochs from k_0 to k_0+8 but no effect at $k \le k_0-1$ or $k \ge k_0+9$. The same notation is used in Equations (3-22)-(3-24).

3.2.4 Code-Carrier Divergence Failure Effects

Equation (2-12) shows that divergence in one channel is also independent of other channels. For simplicity, $kT_s \ge \tau_d$ is assumed, which is valid most of the time in practice. Thus, $\tau(k)$ is equal to a fixed constant τ_d in this analysis.

Under the failure, code-minus-carrier becomes:

$$
\begin{aligned}
z^{\times}_{r,p}(k) &= \rho^{\times}_{r,p}(k) - \phi^{\times}_{r,p}(k) \\
&= \rho_{r,p}(k) + \ell \cdot u(k-k_0) - \phi_{r,p}(k) - \theta \cdot u(k-k_0) \\
&= z_{r,p}(k) + (\ell - \theta) \cdot u(k-k_0)
\end{aligned}
\tag{3-25}
$$

and

$$
\begin{aligned}
dz^{\times}_{r,p}(k) &= z^{\times}_{r,p}(k) - z^{\times}_{r,p}(k-1) \\
&= \left(z_{r,p}(k) + (\ell - \theta) \cdot u(k-k_0)\right) - \left(z_{r,p}(k-1) + (\ell - \theta) \cdot u(k-1-k_0)\right) \\
&= dz_{r,p}(k) + (\ell - \theta) \cdot \delta(k-k_0)
\end{aligned}
\tag{3-26}
$$

Divergence can then be calculated with $dz^{\times}_{r,p}$ based on Equation (2-12). At $k = k_0$,

$$
\begin{aligned}
Dvgc^{\times}_{r,p}(k_0) &= \frac{\tau_d - T_s}{\tau_d} Dvgc^{\times}_{r,p}(k_0 - 1) + \frac{1}{\tau_d} dz^{\times}_{r,p}(k_0) \\
&= \frac{\tau_d - T_s}{\tau_d} Dvgc_{r,p}(k_0 - 1) + \frac{1}{\tau_d}\left(dz_{r,p}(k_0) + (\ell - \theta)\right) \\
&= Dvgc_{r,p}(k_0) + \frac{\ell - \theta}{\tau_d}
\end{aligned}
\tag{3-27}
$$

and at $k > k_0$,

$$
\begin{aligned}
Dvgc^{\times}_{r,p}(k) &= \frac{\tau_d - T_s}{\tau_d}\left(Dvgc_{r,p}(k-1) + \nabla Dvgc_{r,p}(k-1)\right) + \frac{1}{\tau_d} dz_{r,p}(k) \\
&= Dvgc_{r,p}(k) + \frac{\tau_d - T_s}{\tau_d} \nabla Dvgc_{r,p}(k-1) \\
&= Dvgc_{r,p}(k) + \left(\frac{\tau_d - T_s}{\tau_d}\right)^{k-k_0} \nabla Dvgc_{r,p}(k_0)
\end{aligned}
\tag{3-28}
$$

Equations (3-27) and (3-28) together lead to:

$$
\begin{aligned}
Dvgc^{\times}_{r,p}(k) &= Dvgc_{r,p}(k) + \nabla Dvgc_{r,p}(k) \\
&= Dvgc_{r,p}(k) + \left(\frac{\tau_d - T_s}{\tau_d}\right)^{k-k_0} \frac{\ell - \theta}{\tau_d} \cdot u(k-k_0)
\end{aligned}
\tag{3-29}
$$

Equation (3-29) is quite similar to (3-14), since the innovation and divergence methods are mathematically equivalent, as will be discussed further in Chapter 4.

3.2.5 MRCC B-Value Failure Effects

As explained in Chapter 2, not every channel can generate B-values. A necessary but insufficient condition for channel (r, p) to have B-values is that $M_p \geq 2$, i.e., satellite p is tracked on at least two receivers. If the channel does not have B-values, its measurements will not enter Equations (2-21)-(2-24). Moreover, satellites that have B-values may or may not be in the common set, S_c. If a satellite is not in S_c, its measurements will not affect the B-values of other satellites.

Satellite p can be tracked on one, two, or three reference receivers. The common set (discussed in Section 2.7), S_c, may or may not contain satellite p. Due to these complicated relationships among the tracking status of channel (r, p), satellite p, and the common set, there exist many possible scenarios. This section analyzes only the most important case in detail, i.e., both satellite p and the failure channel (r, p) are in S_c. In this case, channel (r, p) has B-values.

All channels that have B-values in this most important case can be further divided into the following four groups. For each group, $\rho^{\times}_{sca}$ is calculated using Equation (2-21).

1. The failed channel (r, p):

$$\begin{aligned}
\rho^{\times}_{sca,r,p}(k) &= \rho^{\times}_{sc,r,p}(k) - \frac{1}{N_c(k)} \sum_{j \in S_c(k)} \rho^{\times}_{sc,r,j}(k) \\
&= \rho_{sc,r,p}(k) + \nabla\rho_{s,r,p}(k) - \frac{1}{N_c(k)} \left(\nabla\rho_{s,r,p}(k) + \sum_{j \in S_c(k)} \rho_{sc,r,j}(k) \right) \\
&= \rho_{sca,r,p}(k) + \frac{N_c(k) - 1}{N_c(k)} \nabla\rho_{s,r,p}(k)
\end{aligned} \tag{3-30}$$

2. Non-failed channels (r, n) with $n \neq p$:

$$\begin{aligned}\rho^{\times}_{sca,r,n\neq p}(k) &= \rho^{\times}_{sc,r,n}(k) - \frac{1}{N_c(k)}\sum_{j\in S_c(k)}\rho^{\times}_{sc,r,j}(k) \\ &= \rho_{sc,r,n}(k) - \frac{1}{N_c(k)}\left(\nabla\rho_{s,r,p}(k) + \sum_{j\in S_c(k)}\rho_{sc,r,j}(k)\right) \\ &= \rho_{sca,r,n}(k) - \frac{1}{N_c(k)}\nabla\rho_{s,r,p}(k)\end{aligned} \tag{3-31}$$

3. Non-failed channels (m, p) with $m \neq r$:

$$\rho^{\times}_{sca,m\neq r,p}(k) = \rho_{sca,m,p}(k) \tag{3-32}$$

4. Other non-failed channels (m, n) with $m \neq r$ and $n \neq p$:

$$\rho^{\times}_{sca,m\neq r,n\neq p}(k) = \rho_{sca,m,n}(k) \tag{3-33}$$

Having obtained $\rho^{\times}_{sca}$, pseudorange B-values can be calculated from Equation (2-23). For the failed channel (r, p), for example,

$$\begin{aligned}B^{\times}_{\rho,r,p}(k) &= \frac{1}{M_p(k)}\sum_{i\in S_p(k)}\rho^{\times}_{sca,i,p}(k) - \frac{1}{M_p(k)-1}\sum_{\substack{i\in S_p(k)\\ i\neq r}}\rho^{\times}_{sca,i,p}(k) \\ &= \frac{1}{M_p(k)}\left(\frac{N_c(k)-1}{N_c(k)}\nabla\rho_{s,r,p}(k) + \sum_{i\in S_p(k)}\rho_{sca,i,p}(k)\right) \\ &\quad - \frac{1}{M_p(k)-1}\sum_{\substack{i\in S_p(k)\\ i\neq r}}\rho_{sca,i,p}(k) \\ &= B_{\rho,r,p}(k) + \frac{N_c(k)-1}{M_p(k)\cdot N_c(k)}\nabla\rho_{s,r,p}(k)\end{aligned} \tag{3-34}$$

$B^{\times}_{\rho}$ for the other three groups of channels can be derived in a similar way.

More detailed derivations are not given here. The results can be summarized as follows:

1. If channel (r, p) is in S_c :

$$\rho^{\times}_{sca,r,p}(k) = \rho_{sca,r,p}(k) + \frac{N_c(k)-1}{N_c(k)}\nabla\rho_{s,r,p}(k) \tag{3-35}$$

$$\rho^{\times}_{sca,r,n\neq p}(k) = \rho_{sca,r,n}(k) - \frac{1}{N_c(k)}\nabla\rho_{s,r,p}(k) \tag{3-36}$$

$$\rho^{\times}_{sca,m\neq r,n}(k) = \rho_{sca,m,n}(k) \text{ for any } n \tag{3-37}$$

$$B^{\times}_{\rho,r,p}(k) = B_{\rho,r,p}(k) + \frac{N_c(k)-1}{M_p(k)\cdot N_c(k)} \nabla\rho_{s,r,p}(k) \tag{3-38}$$

$$B^{\times}_{\rho,m\neq r,p}(k) = B_{\rho,m,p}(k) - \frac{N_c(k)-1}{M_p(k)\cdot\left(M_p(k)-1\right)\cdot N_c(k)} \nabla\rho_{s,r,p}(k) \tag{3-39}$$

$$B^{\times}_{\rho,r,n\neq p}(k) = B_{\rho,r,n}(k) - \frac{1}{M_n(k)\cdot N_c(k)} \nabla\rho_{s,r,p}(k) \text{ with } r\in S_n(k) \tag{3-40}$$

$$B^{\times}_{\rho,m\neq r,n\neq p}(k) = \begin{cases} B_{\rho,m,n}(k) + \dfrac{1}{M_n(k)\cdot\left(M_n(k)-1\right)\cdot N_c(k)} \nabla\rho_{s,r,p}(k) & if \quad r\in S_n(k) \\ B_{\rho,m,n} & if \quad r\notin S_n(k) \end{cases} \tag{3-41}$$

2. If channel (r,p) is not in S_c:

$$\rho^{\times}_{sca,m,n}(k) = \begin{cases} \rho_{sca,r,p}(k) + \nabla\rho_{s,r,p}(k) & if \quad m=r \quad and \quad n=p \\ \rho_{sca,m,n}(k) & else \end{cases} \tag{3-42}$$

$$B^{\times}_{\rho,r,p}(k) = B_{\rho,r,p}(k) + \frac{1}{M_p(k)} \nabla\rho_{s,r,p}(k) \tag{3-43}$$

$$B^{\times}_{\rho,m\neq r,p}(k) = B_{\rho,m,p}(k) - \frac{1}{M_p(k)\cdot\left(M_p(k)-1\right)} \nabla\rho_{s,r,p}(k) \tag{3-44}$$

$$B^{\times}_{\rho,m,n\neq p}(k) = B_{\rho,m,n}(k) \text{ for any } m \tag{3-45}$$

The failure effects of $B_{\rho,m,n}(k)$ for any m and n, whether channel (r,p) is in S_c or not, can be summarized as:

$$B^{\times}_{\rho,m,n}(k) = B_{\rho,m,n}(k) + \nabla B_{\rho,m,n}(k,r,p) \tag{3-46}$$

where $\nabla B_{\rho,m,n}(k,r,p)$ is either zero or linearly proportional to $\nabla\rho_{s,r,p}(k)$.

In the same way, $\phi^{\times}_{ca}(k)$ can be calculated from Equation (2-22) and then $B^{\times}_{\phi}(k)$ from Equation (2-24). The results are given in Equations (3-35)-(3-45) with ϕ replacing ρ and $\theta\cdot u(k-k_0)$ replacing $\nabla\rho_{s,r,p}(k)$ due to the difference between Equations (3-10) and (3-11). The failure effects of $B_{\phi,m,n}(k)$ can also be summarized as:

$$B^{\times}_{\phi,m,n}(k) = B_{\phi,m,n}(k) + \nabla B_{\phi,m,n}(k,r,p) \tag{3-47}$$

where $\nabla B_{\phi,m,n}(k,r,p)$ is either zero or linearly proportional to $\theta \cdot u(k-k_0)$.

3.2.6 MFRT Failure Effects

The term ρ_{corr} is calculated from Equation (2-25) – it is actually just the first term in the right side of Equation (2-23). Therefore, a portion of the results in Section 3.2.5 can be utilized directly in this section. $R_{\rho_{corr},n}$ is the change rate of $\rho_{corr,n}$, as shown in Equation (2-26). For simplicity, $S_c(k)$ and $M_n(k)$ are assumed not to change during the period of time that is considered.

The failure effects can be summarized as follows:

1. If channel (r,p) is in S_c:

$$\rho^{\times}_{corr,p}(k) = \rho_{corr,p}(k) + \frac{N_c(k)-1}{M_p(k)\cdot N_c(k)}\nabla\rho_{s,r,p}(k) \tag{3-48}$$

$$\rho^{\times}_{corr,n\neq p}(k) = \begin{cases} \rho_{corr,n}(k) - \dfrac{1}{M_n(k)\cdot N_c(k)}\nabla\rho_{s,r,p}(k) & if \quad r \in S_n(k) \\ \rho_{corr,n} & if \quad r \notin S_n(k) \end{cases} \tag{3-49}$$

$$R^{\times}_{\rho_{corr},p}(k) = R_{\rho_{corr},p}(k) + \frac{N_c(k)-1}{T_s\cdot M_p(k)\cdot N_c(k)}\left(\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1)\right) \tag{3-50}$$

$$R^{\times}_{\rho_{corr},n\neq p}(k) = \begin{cases} R_{\rho_{corr},n}(k) - \dfrac{\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1)}{T_s\cdot M_n(k)\cdot N_c(k)} & if \quad r \in S_n(k) \\ R_{\rho_{corr},n}(k) & if \quad r \notin S_n(k) \end{cases} \tag{3-51}$$

2. If channel (r,p) is not in S_c:

$$\rho^{\times}_{corr,p}(k) = \rho_{corr,p}(k) + \frac{1}{M_p(k)}\nabla\rho_{s,r,p}(k) \tag{3-52}$$

$$\rho^{\times}_{corr,n\neq p}(k) = \rho_{corr,n}(k) \tag{3-53}$$

$$R^{\times}_{\rho_{corr},p}(k) = R_{\rho_{corr},p}(k) + \frac{1}{T_s\cdot M_p(k)}\left(\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1)\right) \tag{3-54}$$

$$R^{\times}_{\rho_{corr},n\neq p}(k) = R_{\rho_{corr},n}(k) \tag{3-55}$$

where

$$\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1)$$
$$= \frac{1}{N_s}\left(\left(\ell + (N_s - 1)\theta\right)\cdot\delta(k-k_0) + \left(\frac{N_s - 1}{N_s}\right)^{k-k_0}(\ell - \theta)\cdot u(k-k_0-1)\right) \tag{3-56}$$

The results for $\phi^{\times}_{corr}$ are nearly the same as Equations (3-48), (3-49), (3-52) and (3-53), except that ρ and $\nabla\rho_{s,r,p}(k)$ need to be replaced with ϕ and $\theta\cdot u(k-k_0)$, respectively.

3.3 Channel Code Step Failures

In this section, a step failure is injected into the code measurements of channel (r, p). This is just a special case of the general failure model with:

$$\theta = 0 \tag{3-57}$$

in Equation (3-5). All of results in Section 3.2 are valid here, and some of them can be further simplified.

The following is the summary of the channel code step failure effects:

$$\nabla\rho_{s,r,p}(k) = \left(1 - \left(\frac{N_s - 1}{N_s}\right)^{k-k_0+1}\right)\ell\cdot u(k-k_0) \tag{3-58}$$

$$\phi^{\times}_{c,r,p}(k) = \phi_{c,r,p}(k) \tag{3-59}$$

$$Inno^{\times}_{r,p}(k) = Inno_{r,p}(k) + \left(\frac{N_s - 1}{N_s}\right)^{k-k_0}\ell\cdot u(k-k_0) \tag{3-60}$$

$$Dvgc^{\times}_{r,p}(k) = Dvgc_{r,p}(k) + \left(\frac{\tau_d - T_s}{\tau_d}\right)^{k-k_0}\frac{\ell}{\tau_d}\cdot u(k-k_0) \tag{3-61}$$

$$\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1) = \frac{1}{N_s}\left(\frac{N_s - 1}{N_s}\right)^{k-k_0}\ell\cdot u(k-k_0) \tag{3-62}$$

The test statistics $B^{\times}_{\rho}$, $\rho^{\times}_{corr}$, and $R^{\times}_{\rho_{corr}}$ are certainly the same as those in Section 3.2 but with simplified $\nabla\rho_{s,r,p}(k)$ and $\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1)$ as expressed in Equations (3-

58) and (3-62), respectively. The acceleration, ramp, step, B_{ϕ}, and ϕ_{corr} test statistics are not affected by any possible code failure, as clarified in Section 3.2.

The analytical results obtained in this section can be used to fill the first column in Table 3-2, but due to limited space in each cell, the results are just qualitatively summarized. The character '√' indicates that a monitor responds to the fault, while '×' shows that the monitor does not directly respond to the fault. The cells corresponding to the acceleration, ramp, step, B_{ϕ}, and ϕ_{corr} test statistics are marked '×' since these test statistics are not affected by any possible code failure.

Failure Modes	**Channel** (r,p)		Satellite p		Receiver r
	Code Meas.	Phase Meas.	Satellite Clock	Ionospheric Divergence	Receiver Clock
	$\nabla\phi_{r,p}=0$				
Acc, $Ramp$ and $Step$	×				
$Inno$	√				
$Dvgc$	√				
B_{ρ}	√				
B_{ϕ}	×				
ρ_{corr}	√				
ϕ_{corr}	×				
$R_{\rho_{corr}}$	√				

Table 3-2 Effects of Channel Code Step Failures on IMT Monitors

3.4 Channel Phase Step Failures

In this section, a step failure is injected into the carrier phase measurements of channel (r, p). This is also a special case of the general failure model with:

$$\ell = 0 \tag{3-63}$$

in Equation (3-4). All of the results in Section 3.2 are valid here, and some of them can be further simplified.

The following is the summary of the channel phase step failure effects:

$$\nabla\rho_{s,r,p}(k) = \left(\frac{N_s - 1}{N_s}\right)^{k-k_0+1} \theta \cdot u(k - k_0) \tag{3-64}$$

$$Inno^{\times}_{r,p}(k) = Inno_{r,p}(k) - \left(\frac{N_s - 1}{N_s}\right)^{k-k_0} \theta \cdot u(k - k_0) \tag{3-65}$$

$$Dvgc^{\times}_{r,p}(k) = Dvgc_{r,p}(k) - \left(\frac{\tau_d - T_s}{\tau_d}\right)^{k-k_0} \frac{\theta}{\tau_d} \cdot u(k - k_0) \tag{3-66}$$

$$\begin{aligned} &\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1) \\ &\quad = \frac{1}{N_s}\left((N_s - 1)\theta \cdot \delta(k - k_0) - \left(\frac{N_s - 1}{N_s}\right)^{k-k_0} \theta \cdot u(k - k_0 - 1)\right) \end{aligned} \tag{3-67}$$

The acceleration, ramp, step, $B^{\times}_{\phi}$, and $\phi^{\times}_{corr}$ test statistics are exactly the same as those in Section 3.2. $B^{\times}_{\rho}$, $\rho^{\times}_{corr}$, and $R^{\times}_{\rho_{corr}}$ are also the same as in Section 3.2 but with simplified $\nabla\rho_{s,r,p}(k)$ and $\nabla\rho_{s,r,p}(k) - \nabla\rho_{s,r,p}(k-1)$ as expressed in Equations (3-64) and (3-67), respectively. The results are summarized in Table 3-3.

Failure Modes	**Channel** (r,p)		Satellite p		Receiver r
	Code Meas.	**Phase Meas.**	Satellite Clock	Ionospheric Divergence	Receiver Clock
	$\nabla\phi_{r,p}=0$	$\nabla\rho_{r,p}=0$			
Acc, $Ramp$ and $Step$	×	√			
$Inno$	√	√			
$Dvgc$	√	√			
B_ρ	√	√			
B_ϕ	×	√			
ρ_{corr}	√	√			
ϕ_{corr}	×	√			
$R_{\rho_{corr}}$	√	√			

Table 3-3 Effects of Channel Phase Step Failures on IMT Monitors

3.5 Satellite Clock Step Failures

If a satellite (PRN p) has a clock error, the error will not only affect both code and carrier phase measurement failures, but will also enter channels (m,p) on all reference receivers that are tracking the satellite, i.e., all $m\in S_p$. So a satellite clock step failure is equivalent to multiple channel step failures happening at the same time. The results in Section 3.2 can be heavily utilized here using the supposition method.

Consider the most general satellite failure that is modeled by:

$$\rho^{\times}_{m,p}(k)=\rho_{m,p}(k)+\ell\cdot u(k-k_0) \text{ for any } m\in S_p(k) \tag{3-68}$$

$$\phi^{\times}_{m,p}(k)=\phi_{m,p}(k)+\theta\cdot u(k-k_0) \text{ for any } m\in S_p(k) \tag{3-69}$$

where ℓ and θ are independent. The failed satellite p is assumed to be in the maximum common set, S_c. The impact on other satellites (PRN n) that are in S_c (i.e.,

$M_n(k) = M_p(k)$) is also assessed here. Equations (3-9) and (3-56) are now valid for any $m \in S_p(k)$, i.e.,

$$\nabla\rho_{s,m,p}(k) = \left(\ell - \left(\frac{N_s - 1}{N_s} \right)^{k-k_0+1} (\ell - \theta) \right) \cdot u(k - k_0) \tag{3-70}$$

$$\begin{aligned} &\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1) \\ &= \frac{1}{N_s} \left(\big(\ell + (N_s - 1)\theta\big) \cdot \delta(k - k_0) + \left(\frac{N_s - 1}{N_s} \right)^{k-k_0} (\ell - \theta) \cdot u(k - k_0 - 1) \right) \end{aligned} \tag{3-71}$$

From Equations (3-38)-(3-41) and with supposition,

$$\begin{aligned} &B^{\times}_{\rho,m,p}(k) = B_{\rho,m,p}(k) + \\ &\left(\frac{N_c(k) - 1}{M_p(k) \cdot N_c(k)} - \big(M_p(k) - 1\big) \frac{N_c(k) - 1}{M_p(k) \cdot \big(M_p(k) - 1\big) \cdot N_c(k)} \right) \nabla\rho_{s,m,p}(k) \\ &= B_{\rho,m,p}(k) \end{aligned} \tag{3-72}$$

$$\begin{aligned} &B^{\times}_{\rho,m,n\neq p}(k) = B_{\rho,m,n}(k) + \\ &\left(\big(M_n(k) - 1\big) \frac{1}{M_n(k) \cdot \big(M_n(k) - 1\big) \cdot N_c(k)} - \frac{1}{M_n(k) \cdot N_c(k)} \right) \nabla\rho_{s,m,p}(k) \\ &= B_{\rho,m,n}(k) \end{aligned} \tag{3-73}$$

From Equations (3-48)-(3-51) and with the supposition method,

$$\begin{aligned} \rho^{\times}_{corr,p}(k) &= \rho_{corr,p}(k) + M_p(k) \frac{N_c(k) - 1}{M_p(k) \cdot N_c(k)} \nabla\rho_{s,m,p}(k) \\ &= \rho_{corr,p}(k) + \frac{N_c(k) - 1}{N_c(k)} \nabla\rho_{s,m,p}(k) \end{aligned} \tag{3-74}$$

$$\begin{aligned} \rho^{\times}_{corr,n\neq p}(k) &= \rho_{corr,n}(k) - M_n(k) \frac{1}{M_n(k) \cdot N_c(k)} \nabla\rho_{s,m,p}(k) \\ &= \rho_{corr,n}(k) - \frac{1}{N_c(k)} \nabla\rho_{s,m,p}(k) \end{aligned} \tag{3-75}$$

$$R^{\times}_{\rho_{corr},p}(k) = R_{\rho_{corr},p}(k) + \frac{N_c(k) - 1}{T_s \cdot N_c(k)} \big(\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1)\big) \tag{3-76}$$

$$R^{\times}_{\rho_{corr},n\neq p}(k) = R_{\rho_{corr},n}(k) - \frac{1}{T_s \cdot N_c(k)} \big(\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1)\big) \tag{3-77}$$

The failure effects on $B_\phi(k)$ and ϕ_{corr} are almost the same as in Equations (3-72)-(3-75), except that ρ and $\nabla\rho_{s,r,p}(k)$ need be replaced with ϕ and $\theta \cdot u(k-k_0)$, respectively. The effects on innovation, divergence, and acceleration-ramp-step are exactly the same as those in Section 3.2, since they are channel-independent.

Now turn to the satellite clock step failure that is modeled by:

$$B^{p\times}(k) = B^p(k) - \nabla B^p \cdot u(k-k_0) \tag{3-78}$$

Note that B here is the satellite clock bias as in Equations (1-1) and (1-2), while B_ρ and B_ϕ represent B-values as in Equations (2-23) and (2-24). Measurement errors due to the satellite clock failure can be described by Equations (3-68) and (3-69) with:

$$\ell = \theta = \nabla B^p \tag{3-79}$$

since the failure has an equal impact on code and carrier measurements.

Equations (3-70)-(3-71) in the previously general case are then simplified to:

$$\nabla\rho_{s,m,p}(k) = \nabla B^p \cdot u(k-k_0) \tag{3-80}$$

$$\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1) = \nabla B^p \cdot \delta(k-k_0) \tag{3-81}$$

Equations (3-72) and (3-73) show that both $\nabla B_\rho(k)$ and $\nabla B_\phi(k)$ are equal to zero. Both $\nabla Inno(k)$ and $\nabla Dvgc(k)$ are also equal to zero because of the equality in Equation (3-79). This means that none of the innovation, divergence, B_ρ, and B_ϕ algorithms are able to detect satellite clock step errors. In fact, they do not respond to any type of satellite clock failures. The effects on test statistics ρ_{corr}, $R_{\rho_{corr}}$, and ϕ_{corr} are summarized as follows:

$$\nabla\rho_{corr,p}(k) = \nabla\phi_{corr,p}(k) = \frac{N_c(k)-1}{N_c(k)} \cdot \nabla B^p \cdot u(k-k_0) \tag{3-82}$$

$$\nabla\rho_{corr,n\neq p}(k) = \nabla\phi_{corr,n}(k) = -\frac{1}{N_c(k)} \cdot \nabla B^p \cdot u(k-k_0) \tag{3-83}$$

$$\nabla R_{\rho_{corr},p}(k) = \frac{N_c(k)-1}{T_s \cdot N_c(k)} \cdot \nabla B^p \cdot \delta(k-k_0) \tag{3-84}$$

$$\nabla R_{\rho_{corr},n\neq p}(k) = -\frac{1}{T_s \cdot N_c(k)} \cdot \nabla B^p \cdot \delta(k-k_0) \quad (3\text{-}85)$$

The failure effects of the acceleration, ramp, and step test statistics are still given by Equations (3-21)-(3-24) but with $\theta = \nabla B^p$.

All these results are summarized in Table 3-4.

Failure Modes	Channel (r, p)		**Satellite** p		Receiver r
	Code Meas.	Phase Meas.	**Satellite Clock**	Ionospheric Divergence	Receiver Clock
	$\nabla\phi_{r,p} = 0$	$\nabla\rho_{r,p} = 0$	$\nabla\phi_{m,p} = \nabla\rho_{m,p}$		
Acc, $Ramp$ and $Step$	×	√	√		
$Inno$	√	√	×		
$Dvgc$	√	√	×		
B_ρ	√	√	×		
B_ϕ	×	√	×		
ρ_{corr}	√	√	√		
ϕ_{corr}	×	√	√		
$R_{\rho_{corr}}$	√	√	√		

Table 3-4 Effects of Satellite Clock Step Failures on IMT Monitors

3.6 Satellite Clock Ramp Failures

Satellite clock drifts have been observed in the form of ramp-like functions [68]. A satellite (PRN p) clock ramp failure is modeled by:

$$B^{p\times}(k) = B^p(k) - (k-k_0)\cdot T_s \cdot \nabla B^p \cdot u(k-k_0) \quad (3\text{-}86)$$

where the clock ramp magnitude, ∇B^p, is in m/s. Measurement errors due to this failure can be described by Equations (3-68) and (3-69) with:

$$\ell = \theta = (k - k_0) \cdot T_s \cdot \nabla B^p \tag{3-87}$$

for all $m \in S_p(k)$. Although this model lets the clock drift run forever, the failure will likely be detected by the GPS Operational Control Segment (OCS) before it becomes too large.

The ramp function can be decomposed into a sequence of delayed step functions, i.e.,

$$\begin{aligned}&(k - k_0) \cdot T_s \cdot \nabla B^p \cdot u(k - k_0) \\ &\quad = T_s \cdot \nabla B^p \left(u(k - k_0 - 1) + u(k - k_0 - 2) + u(k - k_0 - 3) + \ldots \right)\end{aligned} \tag{3-88}$$

where satellite clock step failures have been analyzed in Section 3.5. Using superposition, Equation (3-70) becomes:

$$\nabla\rho_{s,m,p}(k) = (k - k_0) \cdot Ts \cdot \nabla B^p \cdot u(k - k_0) \tag{3-89}$$

and then (3-71) is transformed to:

$$\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1) = T_s \cdot \nabla B^p \cdot u(k - k_0 - 1) \tag{3-90}$$

Finally, the clock ramp failure effects can be summarized as follows:

$$\begin{aligned}Acc^{\times}_{m,n}(k) = {} & Acc_{m,n}(k) \cdot u(k_0 - k) \\ & + Acc^{\times}_{m,n}(k) \cdot \left(u(k - k_0 - 1) - u(k - k_0 - 9) \right) \text{ for any } n \\ & + Acc_{m,n}(k) \cdot u(k - k_0 - 9)\end{aligned} \tag{3-91}$$

$$\begin{aligned}Ramp^{\times}_{m,p}(k) = {} & Ramp_{m,p}(k) \cdot u(k_0 - k) \\ & + Ramp^{\times}_{m,p}(k) \cdot \left(u(k - k_0 - 1) - u(k - k_0 - 9) \right) \\ & + \left(Ramp_{m,p}(k) + \frac{N_m(k) - 1}{N_m(k)} \cdot \nabla B^p \right) \cdot u(k - k_0 - 9)\end{aligned} \tag{3-92}$$

$$\begin{aligned}Ramp^{\times}_{m,n \neq p}(k) = {} & Ramp_{m,n}(k) \cdot u(k_0 - k) \\ & + Ramp^{\times}_{m,n}(k) \cdot \left(u(k - k_0 - 1) - u(k - k_0 - 9) \right) \\ & + \left(Ramp_{m,n}(k) - \frac{1}{N_m(k)} \cdot \nabla B^p \right) \cdot u(k - k_0 - 9)\end{aligned} \tag{3-93}$$

$$
\begin{aligned}
Step_{m,p}^{\times}(k) &= Step_{m,p}(k)\cdot u(k_0-k) \\
&+\left(Step_{m,p}(k)+\frac{N_m(k)-1}{N_m(k)}T_s\cdot\nabla B^p \right)\cdot\delta(k-k_0-1) \\
&+Step_{m,p}^{\times}\cdot\left(u(k-k_0-2)-u(k-k_0-10)\right) \\
&+Step_{m,p}(k)\cdot u(k-k_0-10)
\end{aligned} \tag{3-94}
$$

$$
\begin{aligned}
Step_{m,n\neq p}^{\times}(k) &= Step_{m,n}(k)\cdot u(k_0-k) \\
&+\left(Step_{m,n}(k)-\frac{1}{N_m(k)}T_s\cdot\nabla B^p \right)\cdot\delta(k-k_0-1) \\
&+Step_{m,n}^{\times}\cdot\left(u(k-k_0-2)-u(k-k_0-10)\right) \\
&+Step_{m,n}(k)\cdot u(k-k_0-10)
\end{aligned} \tag{3-95}
$$

$$
\nabla\rho_{corr,p}(k)=\nabla\phi_{corr,p}(k)=\frac{N_c(k)-1}{N_c(k)}(k-k_0)\cdot T_s\cdot\nabla B^p\cdot u(k-k_0) \tag{3-96}
$$

$$
\nabla\rho_{corr,n\neq p}(k)=\nabla\phi_{corr,n\neq p}(k)=-\frac{1}{N_c(k)}(k-k_0)\cdot T_s\cdot\nabla B^p\cdot u(k-k_0) \tag{3-97}
$$

$$
\nabla R_{\rho_{corr},p}(k)=\frac{N_c(k)-1}{N_c(k)}\cdot\nabla B^p\cdot u(k-k_0-1) \tag{3-98}
$$

$$
\nabla R_{\rho_{corr},n\neq p}(k)=-\frac{1}{N_c(k)}\cdot\nabla B^p\cdot u(k-k_0-1) \tag{3-99}
$$

As pointed out in Section 3.5, any satellite clock failures, including satellite clock step and ramp failures, have no effect on the innovation, divergence, B_ρ, and B_ϕ. Note that the satellite clock ramp component, ∇B^p, linearly enters the MQM $Ramp$ test statistic, as shown in Equation (3-92).

The IMT qualitative responses under satellite clock ramp failures are summarized in the same column of Table 3-4 as under satellite clock step failures. Note that their analytical solutions are different.

3.7 Receiver Clock Step Failures

If a reference receiver (with index r) has a clock step failure, the error will appear not only on all its channels, but also on both code and carrier phase measurements output by receiver r . A receiver clock step failure is modeled by:

$$b_r^{\times}(k) = b_r(k) + \nabla b_r \cdot u(k - k_0) \tag{3-100}$$

Measurement errors can then be described by Equations (3-4) and (3-5) with:

$$\ell = \theta = \nabla b_r \tag{3-101}$$

for all $n \in S_r$. The results in Section 3.2 can be used again with superposition since linearity exists between failure inputs and responses for most integrity monitors.

From Equations (3-15) and (3-16) and using the superposition method,

$$\begin{aligned} \phi_{r,n}^{*\times}(k) &= \phi_{r,n}^{*}(k) + \left(\frac{N_r(k)-1}{N_r(k)} + \left(N_r(k)-1\right)\left(-\frac{1}{N_r(k)}\right) \right) \theta \cdot u(k-k_0) \\ &= \phi_{r,n}^{*}(k) \end{aligned} \tag{3-102}$$

This shows that the failure has no effect on ϕ^*, so the acceleration-ramp-step test that is solely based on ϕ^* does not respond to this failure.

From Equations (3-30) and (3-31) and using the supposition method,

$$\begin{aligned} \rho_{sca,r,n}^{\times}(k) &= \rho_{sca,r,n}(k) + \left(\frac{N_c(k)-1}{N_c(k)} + \left(N_c(k)-1\right)\left(-\frac{1}{N_c(k)}\right) \right) \nabla \rho_{s,r,n}(k) \\ &= \rho_{sca,r,n}(k) \end{aligned} \tag{3-103}$$

This means that the failure has no effect on ρ_{sca}; thus B_ρ , $R_{\rho_{corr}}$, and similarly B_ϕ do not respond to the failure.

Though more detailed derivations are not given here, it is straightforward to show that none of the monitors (including innovation, divergence, acceleration-ramp-step, B-values, and MFRT) will detect receiver clock failures. In DGPS, any possible receiver clock drifts are canceled in the corrections; therefore, they actually should not lead to any flags. In other words, receiver clock steps are not hazardous to users and should not be detected nor excluded. The results are summarized in the last column of Table 3-5.

Failure Modes	Channel (r, p)		Satellite p		**Receiver** r
	Code Meas.	Phase Meas.	Satellite Clock	Ionospheric Divergence	**Receiver Clock**
	$\nabla\phi_{r,p}=0$	$\nabla\rho_{r,p}=0$	$\nabla\phi_{m,p}=\nabla\rho_{m,p}$		$\nabla\phi_{r,n}=\nabla\rho_{r,n}$
Acc, $Ramp$ and $Step$	×	√	√		×
$Inno$	√	√	×		×
$Dvgc$	√	√	×		×
B_{ρ}	√	√	×		×
B_{ϕ}	×	√	×		×
ρ_{corr}	√	√	√		×
ϕ_{corr}	×	√	√		×
$R_{\rho_{corr}}$	√	√	√		×

Table 3-5 Effects of Receiver Clock Step Failures on IMT Monitors

3.8 Ionospheric Gradients

3.8.1 Failure Model

The ionosphere, a layer of ionized gases, lies approximately 50 km to 1000 km above the Earth. The ionosphere has a property of code-carrier divergence, i.e., it affects GPS signal propagation by delaying code-phase measurements, ρ, while advancing carrier-phase measurements, ϕ [42]. Thus, opposite signs appear in front of the ionosphere error component, I, in the GPS measurement Equations (1-1) and (1-2).

The physical characteristics of the ionosphere can vary widely. A handful of very large spatial ionospheric gradients have recently been discovered during ionosphere storms. Ionospheric gradients are potentially major threats to the LAAS. Chapter 4 will discuss the potential impact of large spatial ionospheric gradients on the DGPS in detail.

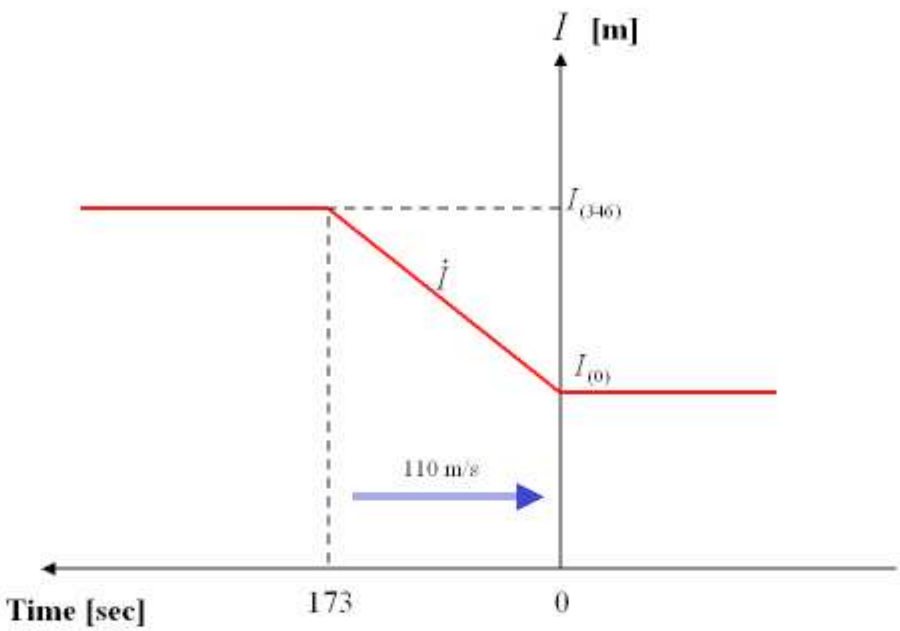

Figure 3-3 Ionospheric Gradient Failure Model

Figure 3-3 is the diagram of the ionospheric gradient failure model based on the Wide Area Augmentation System (WAAS) supertruth data that is used in this analysis and in later chapters [12,31]. In Figure 3-3, an ionospheric gradient with a magnitude of $\dot{I}$ passes over the LGF with a duration of 173 seconds. Our analysis assumes that, at any given time, all reference receivers are impacted with the same ionospheric delays in the measurements from only one satellite (PRN p) [31].

3.8.2 Failure Analyses

Based on the ionospheric gradient failure model in Figure 3-3, measurement errors due to the gradient can be described by Equations (3-68) and (3-69) with:

$$\ell = -\theta = (k - k_0) \cdot T_s \cdot \dot{I}^p \tag{3-104}$$

where the ionospheric gradient, $\dot{I}^p$, is in meters per second. Note that Equation (3-104) models the ionospheric gradient with an infinite duration for simplicity in this failure analysis. In practice, some integrity monitors will eventually detect this gradient before the ionospheric delay becomes very large. This failure model is quite similar to that of the satellite clock ramp failures in Section 3.6, except that the errors in code and carrier measurements have opposite signs under ionospheric gradients due to the ionosphere divergence property discussed in Section 1.1.4.

The gradient failure can be a sum of step functions as well, i.e.,

$$\begin{aligned}&(k - k_0) \cdot T_s \cdot \dot{I}^p \cdot u(k - k_0) \\ &\quad = T_s \cdot \dot{I}^p \left(u(k - k_0 - 1) + u(k - k_0 - 2) + u(k - k_0 - 3) + \ldots \right)\end{aligned} \tag{3-105}$$

Using superposition, Equations (3-70) and (3-71) become:

$$\nabla\rho_{s,m,p}(k) = (k - k_0) \cdot Ts \cdot \dot{I}^p \cdot \left(1 - 2\left(\frac{N_s - 1}{N_s} \right)^{k-k_0+1} \right) \cdot u(k - k_0) \tag{3-106}$$

and

$$\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1) = \left(1 - 2\left(\frac{N_s - 1}{N_s} \right)^{k-k_0} \left(1 - \frac{k - k_0}{N_s} \right) \right) \cdot T_s \cdot \dot{I}^p \cdot u(k - k_0 - 1) \tag{3-107}$$

Ionospheric gradient failure effects are summarized here for any $m \in S_p(k)$,

$$Inno^{\times}_{m,p}(k) = Inno_{m,p}(k) + 2 \cdot N_s \cdot T_s \cdot \dot{I}^p \cdot \left(1 - \left(\frac{N_s - 1}{N_s} \right)^{k-k_0} \right) \cdot u(k - k_0) \tag{3-108}$$

$$Dvgc^{\times}_{m,p}(k) = Dvgc_{m,p}(k) + 2 \cdot \dot{I}^p \cdot \left(1 - \left(\frac{\tau_d - T_s}{\tau_d} \right)^{k-k_0} \right) \cdot u(k - k_0) \tag{3-109}$$

The acceleration, ramp, and step test statistics are the same as those in Equations (3-91)-(3-95), except that ∇B^p needs to be replaced with $-\dot{I}^p$. The effects on ρ_{corr}, $R_{\rho_{corr}}$, and ϕ_{corr} are the same as in Equations (3-74)-(3-77) but with $\nabla\rho_{s,m,p}(k)$ and $\nabla\rho_{s,m,p}(k) - \nabla\rho_{s,m,p}(k-1)$ from Equations (3-106) and (3-107), respectively. B_ρ and

B_ϕ never detect any ionospheric storms, as shown by Equations (3-72) and (3-73). These results are also summarized in Table 3-6.

Failure Modes	Channel (r, p)		**Satellite** p		Receiver r
	Code Meas.	Phase Meas.	Satellite Clock	**Ionospheric Divergence**	Receiver Clock
	$\nabla\phi_{r,p} = 0$	$\nabla\rho_{r,p} = 0$	$\nabla\phi_{m,p} = \nabla\rho_{m,p}$	$\nabla\phi_{m,p} = -\nabla\rho_{m,p}$	$\nabla\phi_{r,n} = \nabla\rho_{r,n}$
Acc, *Ramp* and *Step*	×	√	√	√	×
Inno	√	√	×	√	×
Dvgc	√	√	×	√	×
B_ρ	√	√	×	×	×
B_ϕ	×	√	×	×	×
ρ_{corr}	√	√	√	√	×
ϕ_{corr}	×	√	√	√	×
$R_{\rho_{corr}}$	√	√	√	√	×

Table 3-6 Effects of Ionospheric Gradients on IMT Monitors

3.9 Summary

3.9.1 Qualitative Results

This chapter started with the effects analysis of the fundamental channel step failures in Section 3.2, followed by the analysis of each individual fault mode in Sections 3.3-3.8. Table 3-7 qualitatively summarizes the results that have been obtained in these sections. The character '√' indicates that a monitor responds to the corresponding fault, while '×' shows that the monitor does not directly respond to the fault. None of monitors detect receiver clock failures, and this is acceptable because any possible receiver clock drifts are canceled in the DGPS differential corrections, as discussed in Section 3.7. The acceleration-ramp-step test detects only carrier phase measurement failures, while it does not respond to any code failure. Table 3-7 clearly shows that the innovation method never detects satellite clock failures, as discussed at the beginning of this chapter. The table also demonstrates that the performance of the innovation and divergence tests is similar. The B-value test can detect channel failures; however, it cannot detect satellite clock failures and ionospheric gradients, since the failure analyses assume that these two kinds of failures have the same impact on all reference receivers at any given time.

Failure Modes	Channel (r, p)		Satellite p		Receiver r
	Code Meas.	Phase Meas.	Satellite Clock	Ionospheric Divergence	Receiver Clock
	$\nabla\phi_{r,p} = 0$	$\nabla\rho_{r,p} = 0$	$\nabla\phi_{m,p} = \nabla\rho_{m,p}$	$\nabla\phi_{m,p} = -\nabla\rho_{m,p}$	$\nabla\phi_{r,n} = \nabla\rho_{r,n}$
Acc, $Ramp$ and $Step$	×	√	√	√	×
$Inno$	√	√	×	√	×
$Dvgc$	√	√	×	√	×
B_ρ	√	√	×	×	×
B_ϕ	×	√	×	×	×
ρ_{corr}	√	√	√	√	×
ϕ_{corr}	×	√	√	√	×
$R_{\rho_{corr}}$	√	√	√	√	×

Table 3-7 Effects of Known Failure Modes on IMT Monitors

The IMT has more monitors than those listed in the table, and other failure modes certainly also exist. This chapter does not attempt to cover all monitors and all failure modes. One reason is because of complexity, i.e., the responses of a test statistic may be nonlinear, may not have a closed form solution, or it may even be impossible to determine the magnitude of the responses without having actual GPS measurements. A typical example is the acceleration-ramp-step test. The same failure with the same magnitude can have different acceleration, ramp, and step test responses, depending on the actual value of the ten points, ϕ^*, from Equation (2-14). Without having the actual values of these ten points, ϕ^*, even a numerical solution to the least-squares problem is impractical. The analysis concludes that the IMT detects the entire set of threats analyzed in this work.

3.9.2 Quantitative Results on Minimum Detectable Errors

In addition to providing a clear overall picture of each monitor's function, the failure analyses in this chapter can be utilized to quantitatively evaluate and compare the performance of the IMT integrity monitors. In this section, the size of minimum detectable errors (MDEs) achieved by different monitors will be calculated. The MDEs are the smallest magnitudes of threatening failures that can be detected within a certain missed-detection probability, P_{MD}, and are an important metric for LAAS integrity analysis [64].

As shown in Equation (2-1), the detection threshold, D_t, is set at $6f\sigma$ for a zero-mean Gaussian test statistic to protect continuity based on the required probability of false alarm (2×10^{-9}). If the dashed curve in Figure 3-4 represents the PDF of the test statistic after a failure is injected, then the shaded area will be below the threshold and will not be detected. Integrity monitors are required to detect threatening failures within three seconds at least 99.9% of the time [39], and thus P_{MD} is usually set at 0.001 [64,72]. For the same Gaussian distribution, the point with its CDF at 0.001 is $3.1f\sigma$ from the mean. Thus, the size of MDEs is $(6+3.1)f\sigma = 9.1f\sigma$.

The MDEs for any given monitor do not directly translate into range error bounds. After all, some monitors are detecting rates or accelerations. For example, the code-carrier divergence test detects channel code step errors, as shown in Table 3-7, and the MDEs of this test are expressed in meters/second. The MDEs of the test must be transformed from m/s (rate) to meters (offset). The analysis developed in this chapter provides this transformation and makes the calculations of the range MDEs directly in terms of actual failures possible.

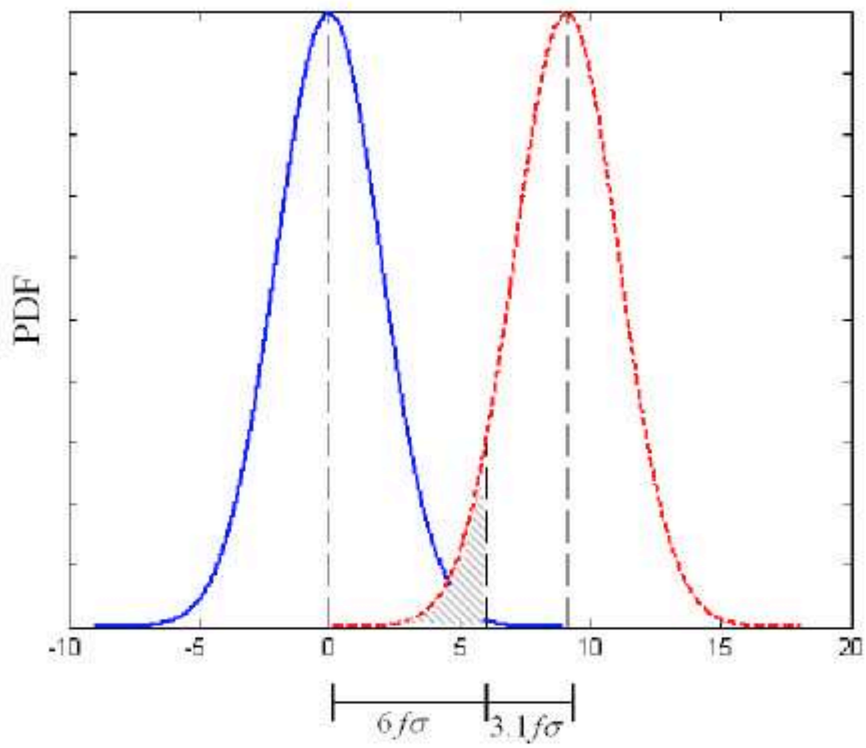

Figure 3-4 Minimum Detectable Errors and P_{MD}

The first example happens to calculate and compare the MDEs of the divergence, innovation, and B-value monitors under the channel code step failure mode that was analyzed in Section 3.3. Equations (3-60), (3-61), and (3-38) are repeated here:

$$\begin{aligned} Inno^{\times}_{r,p}(k) &= Inno_{r,p}(k) + \nabla Inno_{r,p}(k) \\ &= Inno_{r,p}(k) + \left(\frac{N_s - 1}{N_s}\right)^{k-k_0} \ell \cdot u(k-k_0) \end{aligned} \tag{3-110}$$

$$\begin{aligned} Dvgc^{\times}_{r,p}(k) &= Dvgc_{r,p}(k) + \nabla Dvgc_{r,p}(k) \\ &= Dvgc_{r,p}(k) + \left(\frac{\tau_d - T_s}{\tau_d}\right)^{k-k_0} \frac{\ell}{\tau_d} \cdot u(k-k_0) \end{aligned} \tag{3-111}$$

$$
\begin{aligned}
B^{\times}_{\rho,r,p}(k) &= B_{\rho,r,p}(k) + \nabla B_{\rho,r,p}(k) \\
&= B_{\rho,r,p}(k) + \frac{N_c(k)-1}{M_p(k)\cdot N_c(k)}\left(1-\left(\frac{N_s-1}{N_s}\right)^{k-k_0+1}\right)\ell\cdot u(k-k_0)
\end{aligned}
\tag{3-112}
$$

These three equations show that the responses of the innovation, divergence, and B_ρ test statistics under a channel code step error with a magnitude of ℓ are $\nabla Inno_{r,p}(k)$, $\nabla Dvgc_{r,p}(k)$, and $\nabla B_{\rho,r,p}(k)$, respectively. Therefore, if $\nabla Inno_{r,p}(k)$, $\nabla Dvgc_{r,p}(k)$, and $\nabla B_{\rho,r,p}(k)$ are set at $9.1 f\sigma$, where the inflated standard deviations, $f\sigma$, of the corresponding test statistics were derived in Chapter 2, then ℓ values representing the magnitude of minimum detectable code step errors can be calculated.

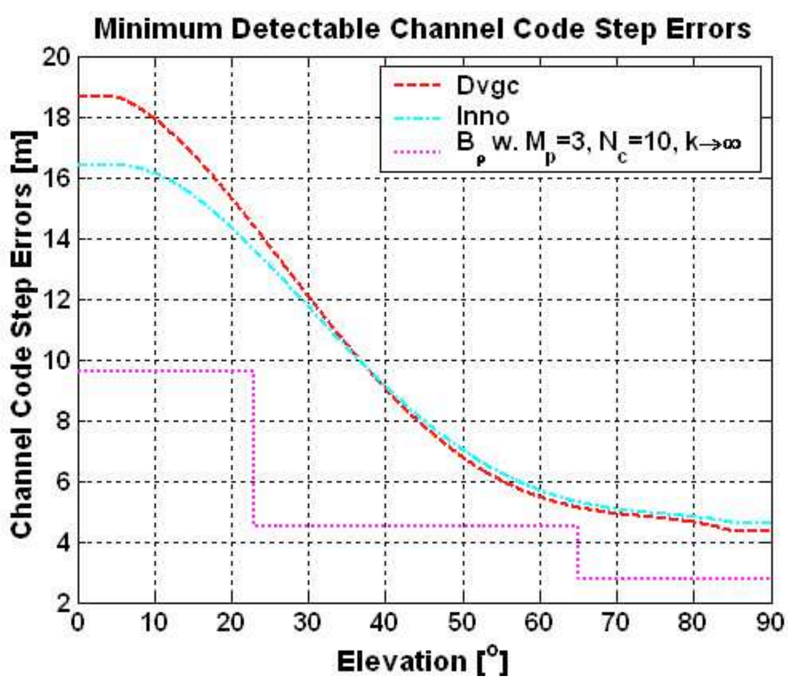

Figure 3-5 Minimum Detectable Channel Code Step Errors

The calculated minimum detectable code step errors are shown in Figure 3-5. During the calculation, $k-k_0$ in both Equations (3-110) and (3-111) is set at 0, so the innovation and divergence methods will likely detect their MDEs immediately. Figure 3-5 shows that the MDEs of the innovation and divergence test statistics are very close to each other. The standard deviation, $f\sigma$, of B_ρ is a three-step function versus satellite elevation (as derived in Section 2.8), and so is its MDE. Though B_ρ has smaller MDEs than those of the innovation and divergence test statistics, its MDE curve is only a low bound since $k-k_0$ in Equation (3-112) is set at ∞ in the calculation. Unlike the innovation and divergence tests that tend to detect channel code step errors immediately, the B_ρ test responds to channel code step errors slowly.

As a second example, it is possible to calculate and compare the size of minimum detectable ionospheric gradients based on the innovation, divergence, and ramp tests. The failure effects of ionospheric gradients were analyzed in Section 3.8. Equations (3-108), (3-109), and (3-92) are repeated here:

$$\begin{aligned} Inno^{\times}_{m,p}(k) &= Inno_{m,p}(k) + \nabla Inno_{m,p}(k) \\ &= Inno_{m,p}(k) + 2 \cdot N_s \cdot T_s \cdot \dot{I}^p \cdot \left(1 - \left(\frac{N_s - 1}{N_s}\right)^{k-k_0}\right) \cdot u(k-k_0) \end{aligned} \tag{3-113}$$

$$\begin{aligned} Dvgc^{\times}_{m,p}(k) &= Dvgc_{m,p}(k) + \nabla Dvgc_{m,p}(k) \\ &= Dvgc_{m,p}(k) + 2 \cdot \dot{I}^p \cdot \left(1 - \left(\frac{\tau_d - T_s}{\tau_d}\right)^{k-k_0}\right) \cdot u(k-k_0) \end{aligned} \tag{3-114}$$

$$\begin{aligned} Ramp^{\times}_{m,p}(k) &= Ramp_{m,p}(k) + \nabla Ramp_{m,p}(k) \\ &= Ramp_{m,p}(k) \cdot u(k_0 - k) \\ &+ Ramp^{\times}_{m,p}(k) \cdot \big(u(k-k_0-1) - u(k-k_0-9)\big) \\ &+ \left(Ramp_{m,p}(k) - \frac{N_m(k)-1}{N_m(k)} \cdot \dot{I}^p\right) \cdot u(k-k_0-9) \end{aligned} \tag{3-115}$$

For the ramp test statistic, only the last term in Equation (3-115) can be used in this example, since the ramp response for $k_0 + 1 \le k \le k_0 + 8$ is not solvable analytically. Similarly, if $\nabla Inno_{m,p}(k)$, $\nabla Dvgc_{m,p}(k)$, and $\frac{N_m(k)-1}{N_m(k)} \cdot \dot{I}^p$ are set at $9.1 f\sigma$, where the inflated standard deviations, $f\sigma$, of the corresponding test statistics were derived in Chapter 2, then $\dot{I}^p$ values, which represent the magnitude of minimum detectable ionospheric gradients, can be calculated. This calculated $\dot{I}^p$ can further be converted to minimum detectable vertical ionospheric gradients based on the obliquity factor. The obliquity factor is used to account for a longer path (and ionosphere) delay for a lower elevation satellite [42], and its formula will be given in Chapter 4.

Figure 3-6 shows the MDEs of these three monitors in terms of vertical ionospheric gradient magnitudes. The dashdot and dotted lines are the MDEs for the innovation and divergence methods, respectively. They appear with the same shape, and both require 173 seconds for detection since the value of $k - k_0$ in both Equations (3-113) and (3-114) is set at $173 \times 2 = 346$ epochs in the calculation. Because the code-carrier divergence test is able to detect smaller MDEs than the innovation method with the same probability of false alarm, the GMA divergence method is more effective in detecting ionospheric gradients. The dashed lines are those for the ramp test, assuming $N_m(k) = 10$ in Equation (3-115). The ramp test can detect its MDEs in about 5 seconds.

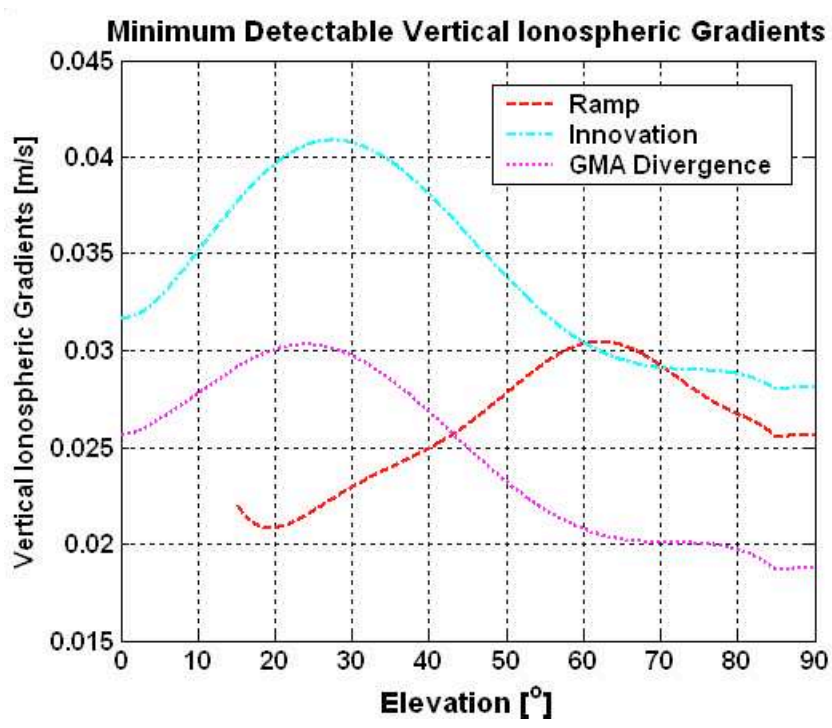

Figure 3-6 Minimum Detectable Ionospheric Gradients

Note that Figure 3-6 does not show ramp test results for low elevation satellites. At low elevations, it seems improper to calculate MDEs using only the last term in Equation (3-

115). The second term in Equation (3-92) can largely determine the ramp test performance in this case, though its analytical solution is not available. Also, in the real operation of a future LAAS system, ionospheric gradients below 0.01 m/s probably should not be detected.

Chapter 4 Cumulative Sum (CUSUM) Method to Detect Ionospheric Gradients

4.1 Introduction

Section 3.8.1 introduced the ionosphere and the hazard associated with ionospheric gradients. Figure 3-3 on page 95 showed the diagram of the ionospheric gradient model based on WAAS supertruth data. In accord with the model given in Section 3.8.1, an ionospheric gradient with a magnitude of $\dot{I}$ passes over the LGF with a duration of 173 seconds.

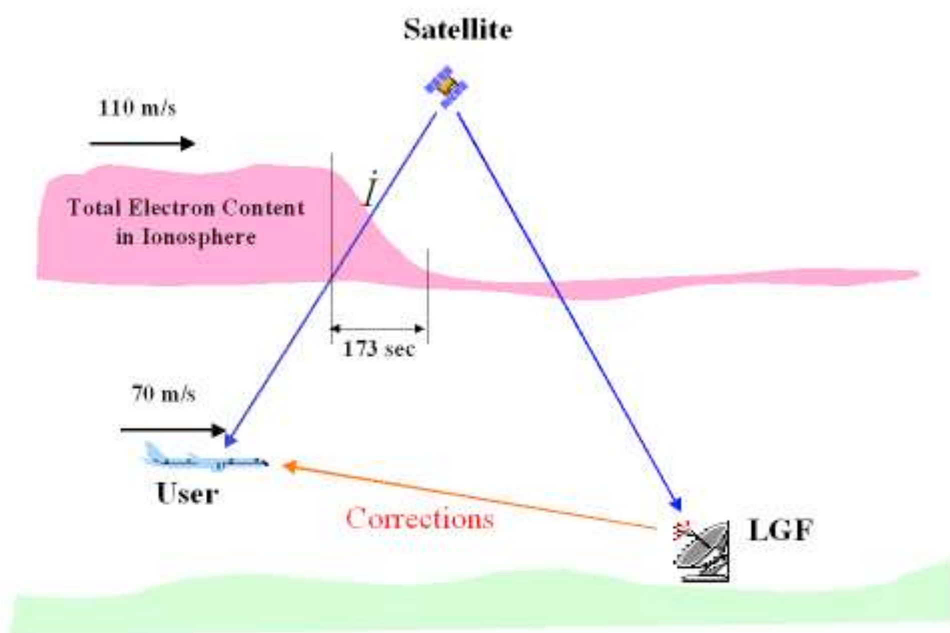

Figure 4-1 Different Ionospheric Delays in Two Different Paths Due to an Ionospheric Gradient

The potential impact of large spatial ionospheric gradients on the DGPS is of great concern. Since anomalous code-carrier divergence undermines the accuracy and integrity of carrier-smoothed code and may not be common to both the LGF and aircraft, the ionospheric gradients could potentially create hazardous errors for LAAS users [9,31,32]. Figure 4-1 shows the diagram of different ionospheric delays in two different paths from a satellite to the LGF and to an aircraft due to an ionospheric spatial gradient, $\dot{I}$. Calculations in [32] show that the worst vertical positioning error due to ionospheric gradients can be as large as 19 meters if no integrity monitors are implemented to detect the gradients. To support aircraft precision approach, LGF must detect the ionospheric gradients as quickly as possible subject to the required low probability of false alarm.

The current IMT employs several monitors that can detect ionospheric gradients, as already analyzed in Chapters 2 and 3. The code-carrier divergence test (discussed in Section 2.4.3) most-directly targets ionosphere gradients. It uses a geometric moving averaging (GMA) method to estimate the code-minus-carrier divergence. A fixed time constant, τ_d, is used in Equation (2-8). The GMA method gives low-noise divergence estimates by averaging multipath differences, $M(k) - M(k-1)$. Though this method is very stable, generally speaking, it is slow to detect ionospheric gradients, $\dot{I}$, since changes in $\dot{I}$ are also averaged. For example, another IMT integrity monitor, the acceleration-ramp-step test, can often detect a relatively large ionospheric gradient within several seconds, while the GMA divergence test may need more than 100 seconds to detect it.

Equation (2-12) is a slightly improved version of the GMA method. It uses a variable time constant, $\tau(k)$, instead of a fixed value when a new satellite rises in view or a filter reset occurs after a satellite is temporarily excluded [53]. With a reduced time constant, $\tau(k)$, the divergence test statistic, $Dvgc$, should converge faster. However, in practice, it offers little help for two reasons. First, since the reduced $\tau(k)$ is used only during the first 200 seconds, while ionospheric storms can occur at any time, the speed of convergence will be the same as in (2-8) after 200 seconds. Second, even during the first 200 seconds, the calculated divergence results are quite noisy and therefore often ignored in the IMT.

Running in parallel with the GMA divergence monitor, the carrier-smoothed code (CSC) innovation test using Equation (2-19) on page 50 is another monitor that can respond to ionospheric gradients. Together with Equations (2-3) and (2-9), the innovation test statistic can be rewritten as:

$$Inno_{m,n}(k) = \frac{N_s - 1}{N_s} Inno_{m,n}(k-1) + dz_{m,n}(k) \tag{4-1}$$

Comparing Equation (2-8) on page 37 with (4-1) leads to the conclusion that the innovation method is mathematically equivalent to the GMA divergence method except for a different smoothing time constant and scaling factor. The time coefficient $\frac{N_s - 1}{N_s} = 199/200$ in the innovation method is very close to $\frac{\tau_d - T_s}{\tau_d} = 199.5/200$ in the GMA divergence method, so these two monitors are expected to perform similarly. Failure test results in Section 4.4 will show that the innovation monitor detects ionospheric gradients slower than the GMA divergence monitor. The smaller the smoothing time constant, the faster the filter converges. However, a smaller time constant results in a larger standard deviation inflation factor when its threshold is derived, and therefore it decreases the detection speed. This is the fundamental tradeoff in setting the time constant. Recent studies have discovered that the time constant used in the GMA divergence test is nearly optimal in terms of detection speed [86].

The acceleration-ramp-step test (discussed in Section 2.6.2) is the third monitor that can rapidly detect large ionospheric gradients often within several seconds. As pointed out in Section 3.2.3, it is difficult and perhaps even impossible to evaluate analytically the performance of this test. Failure tests conducted in Section 4.4.2 reveal that the acceleration and ramp test statistics are usually more effective in detecting ionospheric gradients than the step test statistic. However, the failure tests also disclose that the acceleration-ramp-step test is incapable of detecting small gradients if, for example, their magnitude is less than 0.012 m/s at 50^{o} elevation, as will be shown in Figure 4-13 on page 131. The reason why the acceleration-ramp-step test detects relatively large ionospheric gradients within several seconds but fails to detect small ones is that this test does not use the filtering method. A large failure can immediately result in test statistics that are large enough to exceed the corresponding thresholds, and the failure is detected. If a failure is too small to result in large enough test statistics, the failure will not be detected.

Other monitors in the IMT can also respond to ionospheric gradients, such as the MRCC B-values, but they are not specifically designed to detect ionospheric gradients. In [67], an adaptive filtering of divergence was proposed. This method could have potential for early detection of ionospheric gradients embedded in small multipath differences, $M(k)-M(k-1)$, when multipath limiting antennas are used in the future.

Thus, designing a fast, non-GMA-based divergence monitor that can quickly detect relatively small but hazardous gradients (those that the acceleration-ramp-step monitor fails to detect) is of importance. In this chapter, a new method, the divergence cumulative sum (CUSUM) method, which is designed to quickly and reliably detect marginal ionospheric gradients, will be explored. The CUSUM method is very effective in detecting small persistent changes [21]. It has already been successfully used for LGF error sigma-mean monitoring [28].

This chapter is organized as follows. Section 4.2 explains the general CUSUM algorithm. Its application to code-carrier divergence is discussed in Section 4.3. Section 4.4 presents CUSUM nominal and failure test results and then compares the CUSUM method with other existing IMT monitors. Conclusions are given in Section 4.5.

4.2 CUSUM Algorithm

The CUSUM method achieves a theoretical minimum time-to-detection of a change in the parameters of a random process under two assumptions [21]. First, the input to the CUSUM follows a Gaussian distribution, and different epochs are statistically independent from each other. Second, the true mean and standard deviation (under nominal conditions) are known. The method has been widely applied in continuous inspection for quality control. In this section, the CUSUM algorithm used to detect changes in the process mean, μ , is explained. Its application to code-carrier divergence is explored in Section 4.3.

The formula of the CUSUM mean method can be derived by using a statistical approach [2,21]. Consider a sequence of independent Gaussian random variables, $x(k)$. The Probability Density Function (PDF) of any one of these random variables is

$$p_{\mu}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{4-2}$$

The PDF is also a function of the process mean, μ . As shown in Figure 4-2, the mean in the nominal case is equal to μ_0, called the in-control mean. When the mean changes to a given out-of-control mean, μ_1, then some failure has happened in the random process. In this mode, the variance, σ^2, is a known constant and does not change before and after the failure.

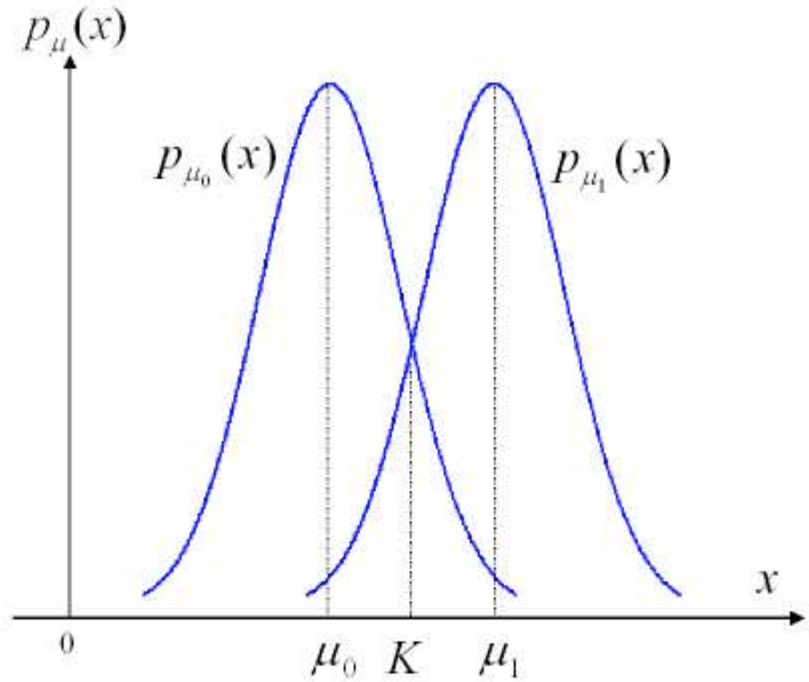

Figure 4-2 The PDF of a Gaussian Random Process with Means μ_0 and μ_1

In the analysis of the CUSUM mean method, a log-likelihood ratio, s, is first defined [2]:

$$s(k) = \ln \frac{p_{\mu_1}\left(x = x(k)\right)}{p_{\mu_0}\left(x = x(k)\right)} \tag{4-3}$$

It is the ratio of the probability of x at $x(k)$ if $\mu = \mu_1$ to the same probability if $\mu = \mu_0$. If the probability distributions in (4-2) are applied, Equation (4-3) can be changed into [2]:

$$s(k) = \frac{\mu_1 - \mu_0}{\sigma^2}\left(x(k) - \frac{\mu_0 + \mu_1}{2}\right) = \frac{\mu_1 - \mu_0}{\sigma^2}\left(x(k) - K\right) \tag{4-4}$$

where a constant value K, also called the windowing factor, is the middle point between two means as shown in Figure 4-2, i.e.,

$$K = \frac{\mu_0 + \mu_1}{2} \tag{4-5}$$

The CUSUM method then accumulates information from past observations by summing up $s(k)$ as follows [2]:

$$C(k) = \sum_{i=1}^{k} s(i) = C(k-1) + s(k) \quad (4\text{-}6)$$

where $C(k)$ is the so-called cumulative sum.

Let us consider the case when $\mu_1 > \mu_0$. In the nominal situation, when the actual mean of a process is μ_0, Figure 4-2 shows that the value of $x(k)$ is less than K with a high probability. In other words, most probably $p_{\mu_0}(x = x(k))$ is larger than $p_{\mu_1}(x = x(k))$; thus, $s(k)$ is usually negative in both Equations (4-3) and (4-4), and $C(k)$ has a negative drift. The CUSUM method resets $C(k)$ when its value goes below zero. Because of the windowing factor, K, resets are common under nominal conditions. For this test, the $\frac{h}{2}$ Fast Initial Response (FIR) CUSUM, in which the value of $C(k)$ is reset to $\frac{h}{2}$ if $C(k) < 0$, is used. The value of the first point CUSUM, $C(0)$, is also initialized to $\frac{h}{2}$. Here, h is the CUSUM threshold, and its derivation is explained below. FIR CUSUMs are recommended for the case where there may be multiple causes for an out-of-control signal and the failure needs to be detected as quickly as possible if the process is still out of control after the process is restarted. Using a start value of $\frac{h}{2}$ is a common recommendation [21].

On the other hand, when the actual mean is changed to μ_1, the value of $x(k)$ is larger than K with a high probability. In other words, most probably $p_{\mu_1}(x = x(k))$ is larger than $p_{\mu_0}(x = x(k))$; thus, $s(k)$ is usually positive in both Equations (4-3) and (4-4), and $C(k)$ shows a positive drift after the change. The value of $C(k)$ is compared with the threshold, h, at every epoch. If $C(k) < h$, the hypothesis of $\mu = \mu_0$ is true; otherwise, the hypothesis of $\mu = \mu_1$ becomes valid and the failure is detected. Examples of positive drift, negative drift and resets can be observed in Figure 4-12 on page 130.

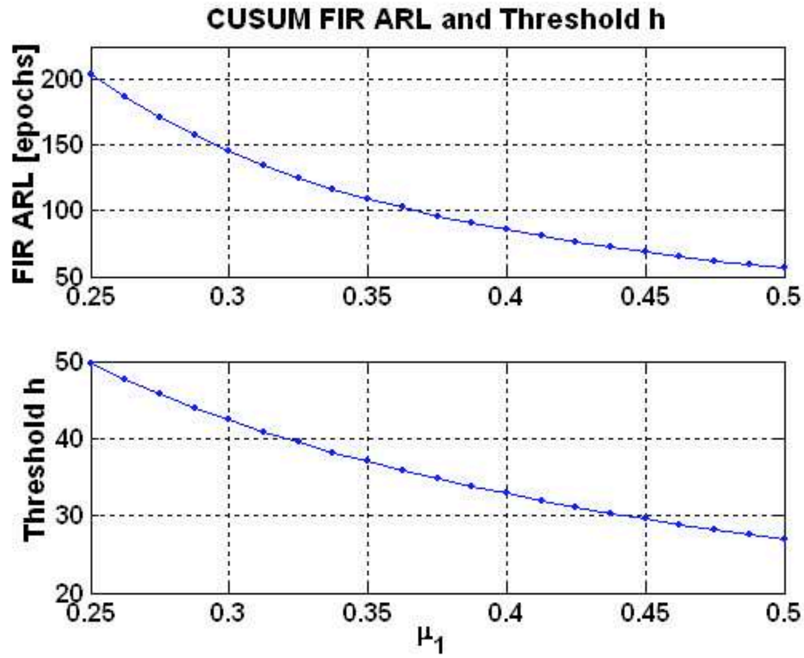

Figure 4-3 FIR ARL and h in the $\frac{h}{2}$ FIR CUSUM with In-Control ARL of 10^7

The average run length (ARL) is one standard measure of CUSUM performance. It is the expected number of independent epochs required for $C(k)$ to exceed h when starting from a given initial value. The in-control ARL should be sufficiently long in order to result in a low probability of false alarm. Based on a sub-allocation of the specified Category I continuity risk allowed per 15-second interval under nominal conditions (in Table 1-2), ARL is set to 10^7 in this application [66,81]. On the contrary, a short out-of-control ARL is desired to detect failures as quickly as possible. The top plot in Figure 4-3 is the overall FIR ARL versus μ_1 when the input, $x(k)$, has a standard normal distribution (i.e., $\mu_0 = 0$ and $\sigma^2 = 1$). The plot shows that the CUSUM takes less time to detect large mean shifts than to detect small ones on average, since fewer data points need to be accumulated for the CUSUM value to exceed the CUSUM threshold.

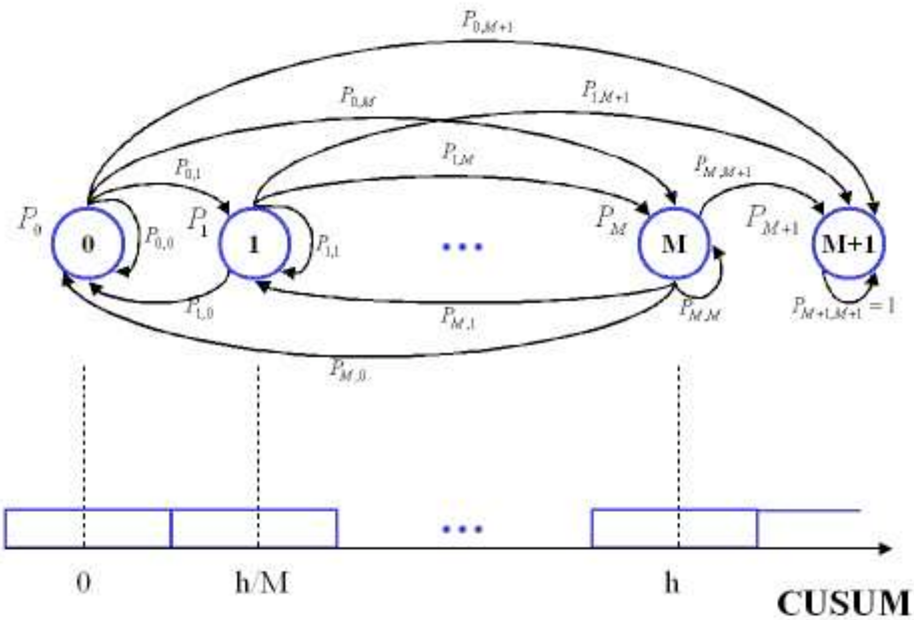

Figure 4-4 Discretized Markov Chain Model

The CUSUM threshold, h, is a function of μ_1 (or, equivalently, K) and the specified in-control ARL for standardized normal observations. It can be computed by performing a numerical search of the Markov Chain model of the CUSUM [21,66]. Figure 4-4 shows a discretized Markov Chain model. Given an h, the range of the CUSUM value from 0 to h is divided into M segments that result in $M+1$ states. Adding in State $M+1$ which corresponds to all possible CUSUM values beyond h, there are $M+2$ states total. Since the CUSUM value is incremented every epoch, the distribution of the CUSUM state at epoch k depends only on its state at the previous epoch $k-1$ and the distribution of the incrementing value at epoch k. With a standard normal distribution of x under nominal conditions, a "one-step" $(M+2)\times(M+2)$ transition matrix, $\mathbf{P}$, can be derived from Equations (4-4)-(4-6) (given a μ_1). Its element, $P_{i,j}$, is the transition probability from state i to state j.

For example, $P_{0,\frac{M}{2}}$ is the transition probability from state 0 to state $\frac{M}{2}$ (typically M is a very large even integer). The CUSUM value, $C(k-1)$, falls between $-\frac{h}{2M}$ and $\frac{h}{2M}$ in State 0. Since M is a very large number, both boundaries approach 0. Thus, the CUSUM value in State 0 can be quantized to 0. Similarly, $C(k-1)$, that falls between $\frac{(M-1)h}{2M}$ and $\frac{(M+1)h}{2M}$ in State $\frac{M}{2}$, is quantized to $\frac{h}{2}$. In the $\frac{h}{2}$ FIR CUSUM method, the CUSUM can transit from State 0 (i.e., $C(k-1)=0$) to State $\frac{M}{2}$ (i.e., $C(k)=\frac{h}{2}$) if one of the following conditions is met:

1. $C(k)=C(k-1)+s(k)=s(k)$ is in the range of State $\frac{M}{2}$. In other words,

$$\frac{(M-1)h}{2M} < C(k) = s(k) < \frac{(M+1)h}{2M} \tag{4-7}$$

   With Equation (4-4), (4-7) can be rewritten into:

$$\frac{\sigma^2}{\mu_1-\mu_0}\frac{(M-1)h}{2M} + K < x(k) < \frac{\sigma^2}{\mu_1-\mu_0}\frac{(M+1)h}{2M} + K \tag{4-8}$$

   which corresponds to a probability of

$$P_{\mu_0}\left(\frac{\sigma^2}{\mu_1-\mu_0}\frac{(M-1)h}{2M} + K < x(k) < \frac{\sigma^2}{\mu_1-\mu_0}\frac{(M+1)h}{2M} + K\right) \tag{4-9}$$

2. $C(k)=C(k-1)+s(k)=s(k)$ is below $-\frac{h}{2M}$. In this case the CUSUM will also be in State $\frac{M}{2}$ after a reset. This portion corresponds to a probability of

$$P_{\mu_0}\left(x(k) < -\frac{\sigma^2}{\mu_1-\mu_0}\frac{h}{2M} + K\right) \tag{4-10}$$

With μ_1, K (from Equation (4-5)), and the distribution of input (i.e., $X \sim (\mu_0, \sigma^2)$ in the nominal situation), the two probabilities in (4-8) and (4-9) can be calculated. The transition probability, $P_{0,\frac{M}{2}}$, is the sum of these two probabilities.

After $\mathbf{P}$ is derived, the following formula, which is given in [21], can be used to find ARLs:

$$(\mathbf{I} - \mathbf{P}_{\mathbf{red}})\mathbf{?} = \mathbf{1} \tag{4-11}$$

where the "reduced" transition matrix, $\mathbf{P}_{\mathbf{red}}$, is $\mathbf{P}$ without the last row and column. In other words, $\mathbf{P}_{\mathbf{red}}$ disregards the transitions to and from state $M+1$. $\mathbf{I}$ is the $(M+1)\times(M+1)$ identity matrix, $\mathbf{1}$ is a vector of length $M+1$ all of whose elements are 1, and $\mathbf{?}$ is the vector of ARL values for CUSUMs starting in the corresponding States 0, 1, …, M. The middle element in $\mathbf{?}$ is the ARL value for the CUSUM value starting from $\frac{h}{2}$, i.e., the $\frac{h}{2}$ FIR ARL. This process of calculating ARL value given an h, can be iterated on different h until the calculated nominal ARL is equal to 10^7**.** This h is the threshold that has been numerically searched for. The bottom plot of Figure 4-3 is the threshold, h, as a function of μ_1 with the specified in-control ARL.

The CUSUM formula for $X \sim N(\mu,\sigma^2)$ is as follows [2]:

$$C^+(k) = \left[C^+(k-1) + \frac{\mu_1 - \mu_0}{\sigma^2}\big(x(k) - K\big)\right]^+ \tag{4-12}$$

where the superscript "+" outside of the rectangle bracket represents the reset process:

$$C^+(0) = C^+(k) = \frac{h^+}{2} \text{ if } C^+(k) < 0 \tag{4-13}$$

The CUSUM, $C(k)$, from Equation (4-6) is also shown with a superscript "+" in (4-12), since its only function is to detect positive mean changes and its value is positive. A parallel CUSUM C^-, used to detect negative mean jumps (i.e., $\mu_1 < \mu_0$), is completely analogous to C^+, i.e.:

$$C^-(k) = \left[C^-(k-1) - \frac{\mu_1 - \mu_0}{\sigma^2}\big(x(k) - K\big)\right]^- \tag{4-14}$$

C^- is similarly reset to $\frac{h^-}{2}$ when its value is larger than zero. In practice, the two-sided CUSUM algorithms used to detect both positive and negative mean shifts are implemented in parallel. This chapter will focus on C^+ for illustration without losing generality.

4.3 Divergence CUSUM Algorithm

Now the CUSUM formulas from the previous section are applied to the code-carrier divergence problem. The algorithm in this section is based on the generic CUSUM algorithm described in the last section, but it also adds a real-time divergence mean estimation and a delayed version of raw divergence. As will be seen, these embellishments improve overall performance. Since each channel has an independent CUSUM, the channel index – (m, n) – will be omitted for brevity.

4.3.1 Delayed Version of Raw Divergence as Input

Since this method is aimed at detecting mean changes in raw divergence, $dz(k)$, from Equation (2-10), $dz(k)$ is naturally selected as the input to the CUSUM. The more general form of $dz(k)$ is defined as:

$$
\begin{aligned}
dz(k) &= \frac{1}{2T_s k_0}\left(z(k) - z(k-k_0)\right) \\
&= \frac{I(k) - I(k-k_0)}{T_s k_0} + \frac{1}{2T_s k_0}\left(M(k) - M(k-k_0)\right) \\
&= \dot{I}(k) + \frac{1}{2T_s k_0}\left(M(k) - M(k-k_0)\right)
\end{aligned}
\tag{4-15}
$$

where k_0 is not necessarily equal to 1. The magnitude of $M(k) - M(k-k_0)$ is several meters and does not change much with k_0, while the magnitude of ionospheric gradients, $\dot{I}(k)$, is approximately several centimeters per second. Therefore, dz can have a higher SNR with a larger k_0, although a larger k_0 implies a longer monitor lag. Some delay is acceptable for small anomalies since the smaller the gradient, the less imminent the threat (if a threat exists at all). On the other hand, if the gradient is large, it is quickly detected by the CUSUM test (if not detected even sooner by the separate acceleration-ramp-step test). Based on these considerations and the optimization method as will be discussed in Section 4.3.6, k_0 is set at 40 epochs, which is equivalent to 20 seconds at the 2 Hz IMT update rate.

Plots (A) and (B) in Figure 4-5 show $dz(k)$ with $k_0 = 1$ and $k_0 = 40$, respectively. With the same small amount of $\dot{I}(k)$ embedded in $dz(k)$, the noise component in Plot (B) is nearly 15 times smaller than in Plot (A). Smaller noise in dz results in smaller standard deviations of dz and effectively a larger out-of-control mean, μ_1, after normalization (discussed in Section 4.3.5). The top plot in Figure 4-3 shows that a short ARL (i.e., a shorter detection time) is needed to detect a larger mean change. Thus, with the delayed version of $dz(k)$ as the input, the CUSUM should more rapidly detect small gradients.

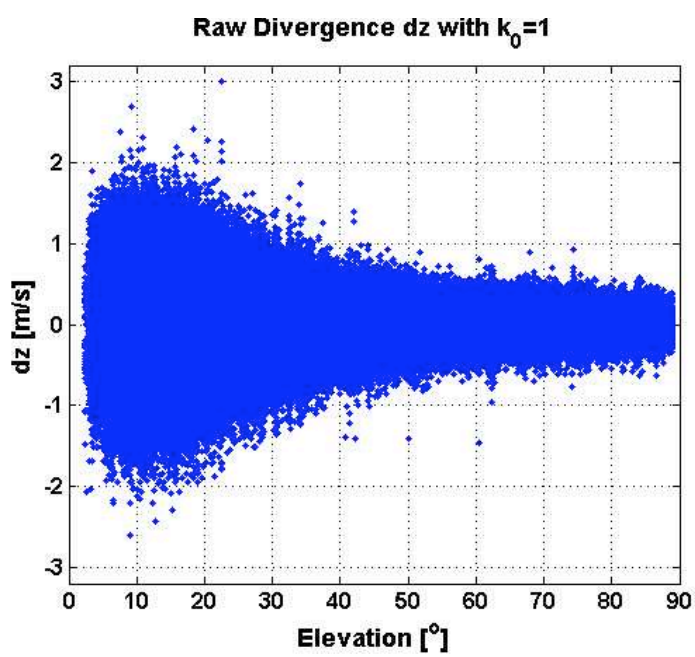

Figure 4-5-A: $k_0 = 1$

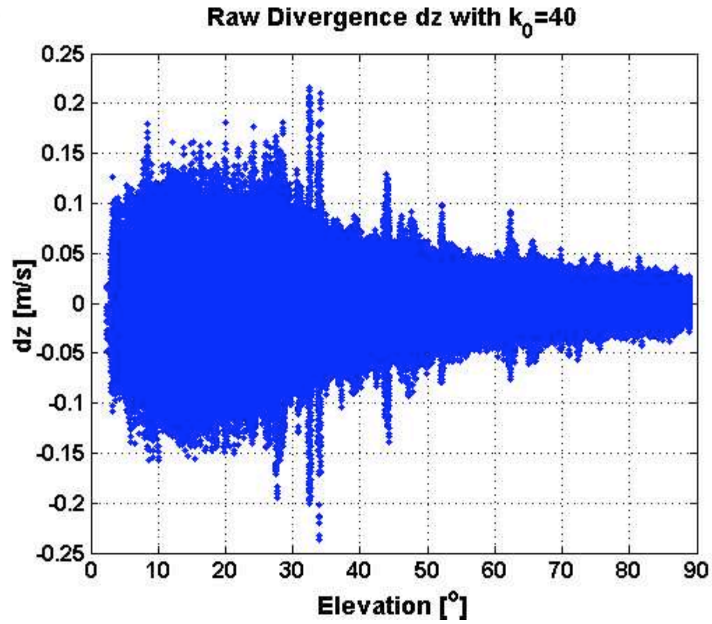

Figure 4-5-B: $k_0 = 40$

Figure 4-5 Raw Divergence dz from All Channels (Note the Difference in Scale on Plots A and B)

Multipath error, $M(k)$, is a temporally correlated random variable. Correlation time in multipath is about 46 – 194 seconds as reported in [45]. Therefore, with $k_0 = 40$ epochs (or, equivalently, 20 seconds), $M(k)$ and $M(k-k_0)$ can be highly correlated with each other, and their difference, $M(k)-M(k-k_0)$, can be assumed to be independent of $M(k+i)-M(k+i-k_0)$ for $i \neq 0$. In fact, $M(k)-M(k-k_0)$ dependence is less important in the CUSUM method than in the GMA divergence and innovation tests, since the CUSUM reset removes its previous "history." The residual dependence is accommodated by inflating the standard deviation obtained from nominal testing in each of these three methods. The standard deviation of the raw divergence will be derived in Section 4.3.4.

4.3.2 Real-Time In-Control Mean Estimation

The true mean and standard deviation of $dz(k)$ should be (approximately) known, as required by the CUSUM algorithm. Because the mean, μ_0, is non-zero and changes with elevation angle and ionosphere state, it is estimated in real-time by the following GMA method (the Geometric Moving Averaging method was discussed in Section 2.4.3):

$$\mu_0(k) = \frac{\tau(k) - T_s}{\tau(k)} \mu_0(k-1) + \frac{T_s}{\tau(k)} dz(k) \tag{4-16}$$

with

$$\tau(k) = \begin{cases} kT_s, & k < \tau_{\max}/T_s \\ \tau_{\max}, & else \end{cases} \tag{4-17}$$

The CUSUM starts to be updated after $2\tau_{\max}$, when $\mu_0(k)$ has nearly converged. A large $\tau_{\max}$ and resulting smooth estimates of $\mu_0(k)$ are preferred, since the mean estimator in Equation (4-16) is not used as an integrity monitor. However, the mean estimation cannot converge the actual $\mu_0(k)$ immediately, which leads to a delay. The larger $\tau_{\max}$, the longer the delay. The value of $\tau_{\max}$ is set at 400 seconds, doubling the time constant used in the GMA divergence test.

The in-control mean estimation can converge to the out-of-control mean, $\mu_1(k)$, in case that an ionospheric gradient occurs. In order not to be "polluted" by the gradient, the in-control mean, $\mu_0(k)$, is assigned the value that is k_1 epochs earlier than the observed value:

$$\mu_0(k) \leftarrow \mu_0(k - k_1) \tag{4-18}$$

The value of k_1 should not be too small; otherwise, $\mu_0(k-k_1)$ can still possibly be polluted by $\mu_1(k)$. If k_1 is too large, $\mu_0(k-k_1)$ may not approximate the current in-control mean level. A large k_1 also requires a large memory to store $\mu_0(k)$ during the last k_1 epochs. The parameter k_1 is set at 500 epochs (equivalent to 250 seconds) for the following reasons. First, the duration of ionospheric gradients is about 173 seconds. Second, CUSUM detection time is typically around 50 – 150 seconds, as will be seen in

Section 4.4.2. Third, the mean estimation delay in Equation (4-16) should also be considered. In reality, the in-control mean changes little in 250 seconds, and this explains why the CUSUM performance in the failure tests conducted in Section 4.4.2 is not very sensitive to the value of $\tau_{\max}$ and k_1.

The solid line in Figure 4-6 is an example of $\mu_0(k)$ from a single channel. While the in-control-mean, $\mu_0(k)$, changes little in 250 seconds, it does change significantly over the course of a satellite pass, as shown in this figure. This illustrates the need for real-time mean estimation.

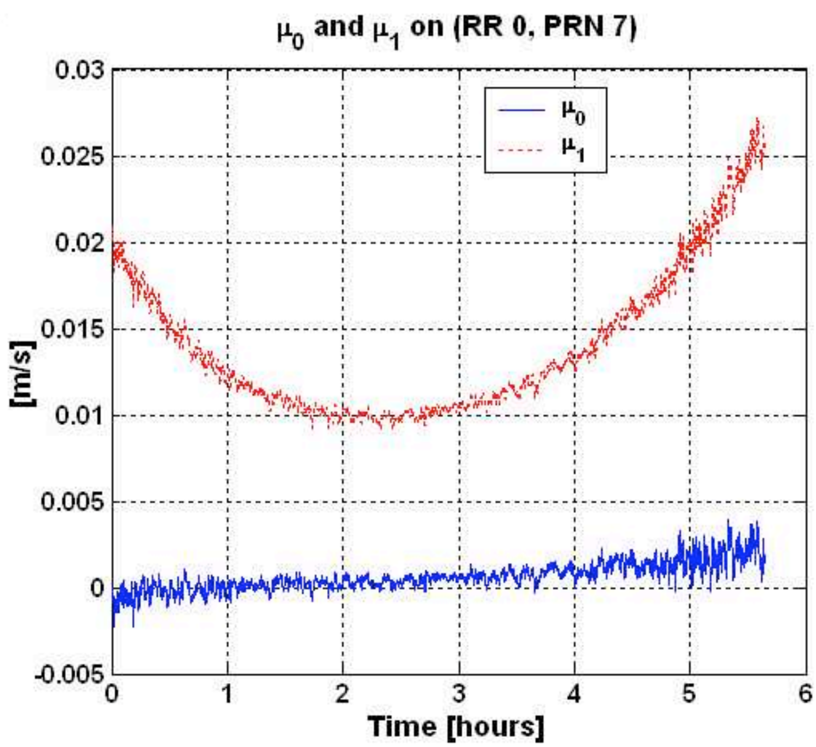

Figure 4-6 $\mu_0(k)$ and $\mu_1(k)$ on Channel (0, 7) with $\tau_{\max}$ = 400 Seconds

4.3.3 Construction of the Out-of-Control Mean

The out-of-control mean, $\mu_1(k)$, is constructed by:

$$\mu_1(k) = \mu_0(k) + v(k) \tag{4-19}$$

where $v(k)$ can be regarded as the "target" (out-of-control) magnitude of the ionospheric gradients that are to detected. It is designed as:

$$v(k) = \dot{I}_{90} \cdot OF(k) \tag{4-20}$$

where $\dot{I}_{90}$ is a fixed value and represents the vertical ionospheric rate at an elevation of 90^o. At other elevation angles, $v(k)$ is magnified by the obliquity factor $OF(k)$ that equals [42]:

$$OF(k) = \left(1 - \left(\frac{R_E \cos(\theta_{elev}(k))}{R_E + h_l}\right)^2\right)^{-1/2} \tag{4-21}$$

where $\theta_{elev}(k)$ is the satellite elevation angle. The value of $v(k)$ is only satellite elevation dependent. With this construction of $\mu_1(k)$, the windowing factor, K, in Equation (4-5) becomes:

$$K(k) = \mu_0(k) + \frac{v(k)}{2} \tag{4-22}$$

K is no longer a constant in our application.

The value of $\dot{I}_{90}$, which represents the "target" (vertical) gradient to be detected, is clearly important since it represents the out-of-control value for which the CUSUM is theoretically optimal [21]. The CUSUM method is designed to detect effectively ionospheric gradients with a magnitude of $\dot{I}_{90}$ which the existing acceleration-ramp-step test fails to promptly detect. Failure tests conducted in Section 4.4.2 show that the acceleration-ramp-step test is fast and reliable in detecting vertical ionospheric gradients whose magnitudes are larger than about 0.011 m/s. Therefore, $\dot{I}_{90}$ is set at 0.0095 m/s so that the CUSUM method can most effectively detect ionospheric gradients of this magnitude, which cannot be reliably detected by the acceleration-ramp-step test. This is the design goal of the divergence CUSUM method. Actually this heuristic choice of $\dot{I}_{90}$ can also be based on a more formal optimization, as will be discussed in Section 4.3.6.

The dashed line in Figure 4-6 is an example of $\mu_1(k)$. The gap between the red and blue lines is $v(k)$. This figure also shows that variations in $\mu_0(k)$ are very small compared to $v(k)$, which indicates that the real-time mean estimation in the CUSUM method achieves its purpose.

4.3.4 Off-Line Calculated Inflated Standard Deviation

To protect continuity, the variance, $\sigma^2(k)$, of the input, $dz(k)$, is pre-computed based on the statistical analysis of nominal data sets. From the raw divergence, $dz(k)$, in Figure 4-5-B, its standard deviation and inflation factor were calculated with the Gaussian overbounding method described in Section 2.2. Normalization in this overbounding method allows us to combine data for all elevation angles. Figure 4-7 shows the observed data distribution as the solid line and the PDF of a Gaussian with an inflated standard deviation as the red line. Its standard deviation inflation factor equals 2.4502. The dashed line in Figure 4-8 is the inflated standard deviation of the CUSUM input versus elevation angle. The inflated standard deviation for elevations below 5^o may not be properly derived in Figure 4-8 since there is no "constraint" at 0^o elevation in the high-order polynomial curve used to derive the standard deviation. However, this does not cause any serious problem because, the "mask" angle in the IMT is set at 5^o, i.e., the IMT does not generate differential corrections for any GPS satellite whose elevation is below this mask angle.

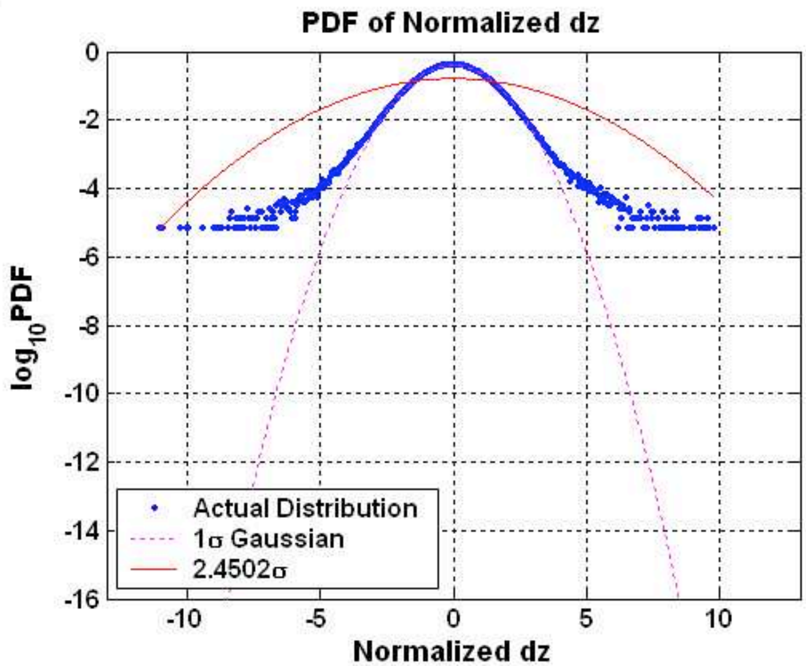

Figure 4-7 PDF of Normalized Raw Divergence, dz

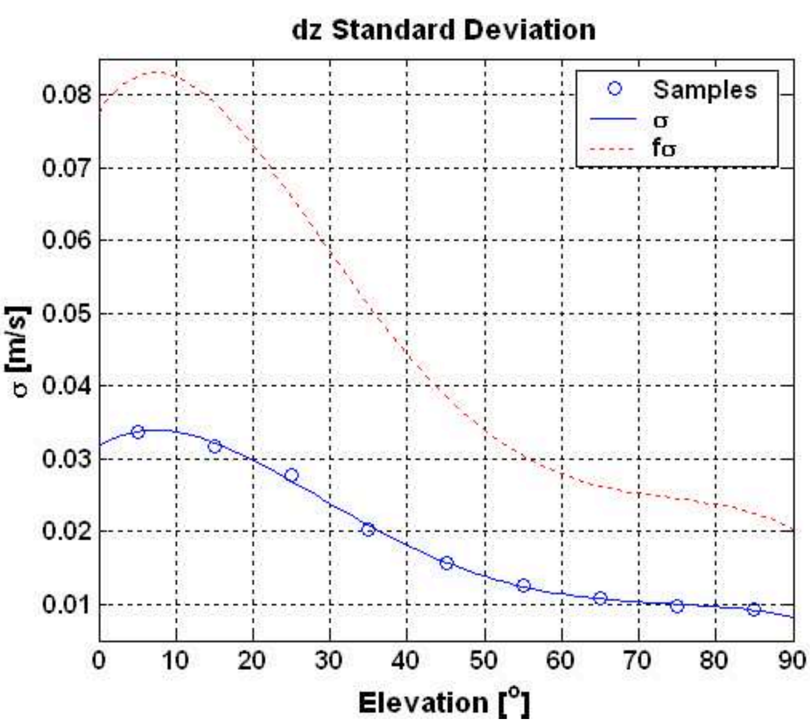

Figure 4-8 Sample and Inflated Standard Deviation, σ, of Raw Divergence, dz

4.3.5 Normalization

Figure 4-8 also illustrates that the standard deviation, σ, of dz is elevation dependent. In order to keep the standard deviation of the CUSUM input constant, a normalization process is applied to $dz(k)$ as follows:

$$\begin{cases} X(k) = \dfrac{dz(k) - \mu_0(k)}{\sigma(k)} \\ V(k) = \dfrac{v(k)}{\sigma(k)} \end{cases} \tag{4-23}$$

After this normalization, the distribution of the actual CUSUM input, $X(k)$, is overbounded by a standard Gaussian. The variance, σ^2, of the raw divergence, dz, is largely due to the multipath noise difference. Therefore, after this normalization process, the variance of $X(k)$ can be bounded by a constant (which is equal to 1) over time even when spatial ionospheric gradients occur. This feature is required for CUSUM optimality, and it is thus reasonable to apply the CUSUM mean method to the ionosphere divergence problem.

The following two formulas can be derived from Equations (4-19), (4-22) and (4-23):

$$\mu_1(k) = V(k) \tag{4-24}$$

and

$$K(k) = \frac{V(k)}{2} \tag{4-25}$$

The CUSUM formula (4-12) in the divergence application thus becomes:

$$C^+(k) = \left[C^+(k-1) + V(k)\left(X(k) - \frac{V(k)}{2} \right) \right]^+ \tag{4-26}$$

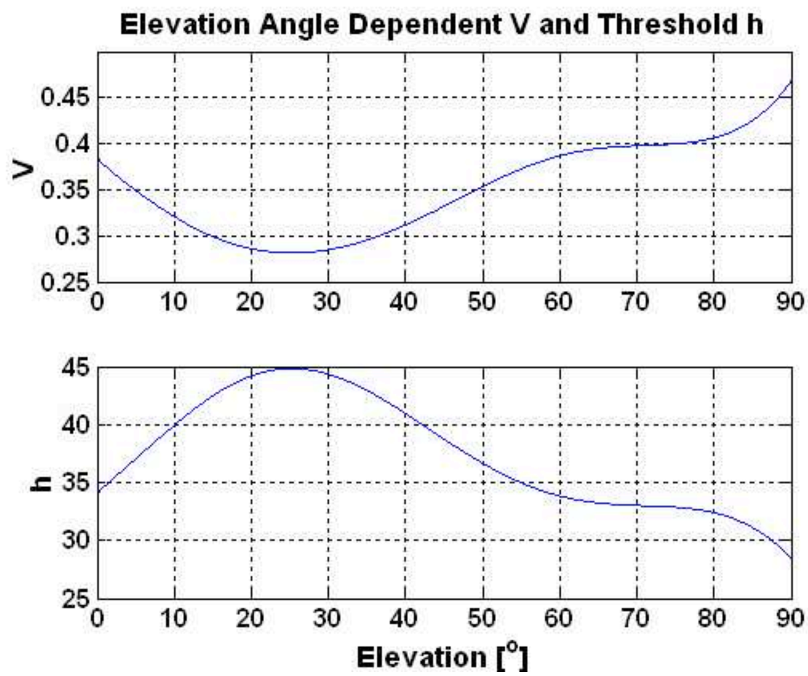

Figure 4-9 V and CUSUM Threshold, h, as a Function of Satellite Elevation

Since both $v(k)$ from Equation (4-20) and $\sigma(k)$ from Figure 4-8 are elevation dependent, $V(k)$ is also elevation dependent, as shown in the top plot of Figure 4-9. Note that $V(k)$, which is equal to $\frac{v(k)}{\sigma(k)}$, is completely determined by $\sigma(k)$ from Figure 4-8 and $v(k)$ from Equation (4-20). No polynomial curve is used to derive $V(k)$. Therefore, the irregular behavior in $V(k)$ comes from the obliquity factor, $OF(k)$, and the high-order polynomial curve for $\sigma(k)$.

With $X(k)$ as the input to the CUSUM after normalization, the CUSUM threshold, h, a function of K, also becomes elevation dependent. Given an elevation (for example, 50^o), V at this elevation from the top plot of Figure 4-9 can be found (V equals 0.353 at 50^o), and the threshold h can then be interpolated from the bottom plot of Figure 4-3 (h equals 36.7 at $\mu_1 = V = 0.353$). The bottom plot of Figure 4-9 shows the resulting threshold, h, as a function of elevation (h equals 36.7 at 50^o). The irregular behavior in

h comes from the obliquity factor, the high-order polynomial curve for $\sigma(k)$, and the CUSUM threshold curve from the bottom plot of Figure 4-3.

4.3.6 Formal Optimization of Parameters

The divergence CUSUM algorithm must determine the value of parameters k_0, k_1, $\tau_{\max}$, and $\dot{I}_{90}$. These parameters were discussed in Section 4.3 and set at $k_0 = 40$, $k_1 = 500$ epochs, $\tau_{\max} = 400$ seconds, and $\dot{I}_{90} = 0.0095$ m/s. In fact, the values of these parameters were not selected independently. Instead, they affect each other and should be optimized at the system level. For example, k_0, k_1, and $\tau_{\max}$ all have some delay effect on the CUSUM method. Since ionospheric gradients have a limited duration, the delay effect needs to be considered when deciding the value of the "target" magnitude of ionospheric gradients, $\dot{I}_{90}$.

These four parameters are optimized primarily from the nominal and failure test performances (failure tests are discussed in Section 4.4.2). In order to simplify the optimization process, the parameters k_1 and $\tau_{\max}$ are determined based on the discussion in Section 4.3.2. This may be done because the CUSUM performance is not very sensitive to these two values. The remaining two parameters, k_0 and $\dot{I}_{90}$ are selected based on a series of failure tests. The best detection performance (in terms of averaged detection time) is searched for with k_0 ranging from 2 to 80 and $\dot{I}_{90}$ ranging from 0.008 m/s to 0.012 m/s. Step sizes in k_0 and $\dot{I}_{90}$ can be coarse first and then adjusted down to 2 epochs and 0.0005 m/s. Note that the standard deviations of raw divergence must be computed every time a different k_0 is used. The CUSUM method achieves the fastest detection speed on average with $k_0 = 40$ and $\dot{I}_{90} = 0.0095$ m/s.

4.4 Test Results and Performance Comparison

The divergence CUSUM algorithm (4-26) has been implemented in the IMT in order to evaluate CUSUM performance and make a fair comparison with other monitors in the same environment. The tests took various forms and are presented below.

4.4.1 Nominal Testing

Figure 4-10-A shows nominal $C^{+}(k)$ test results from all channels versus satellite elevation. As expected, all points are below the corresponding threshold shown as a solid line. Figure 4-10-B is $C^{+}(k)$ from Channel (RR 0, PRN 7). This satellite (PRN 7) will be heavily used in subsequent failure testing, since it makes a lengthy pass in view of the IMT and reaches an elevation of greater than 80 degrees. Figure 4-11 shows the sky plot of PRN 7 in the nominal data set.

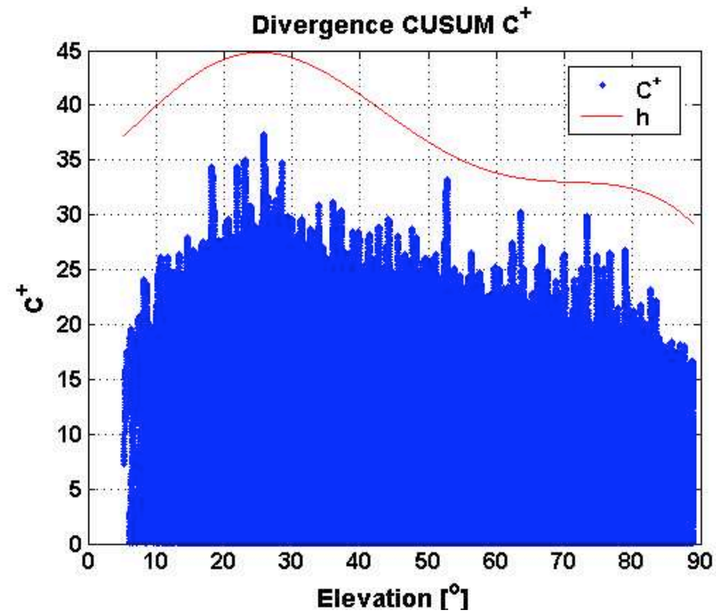

Figure 4-10-A

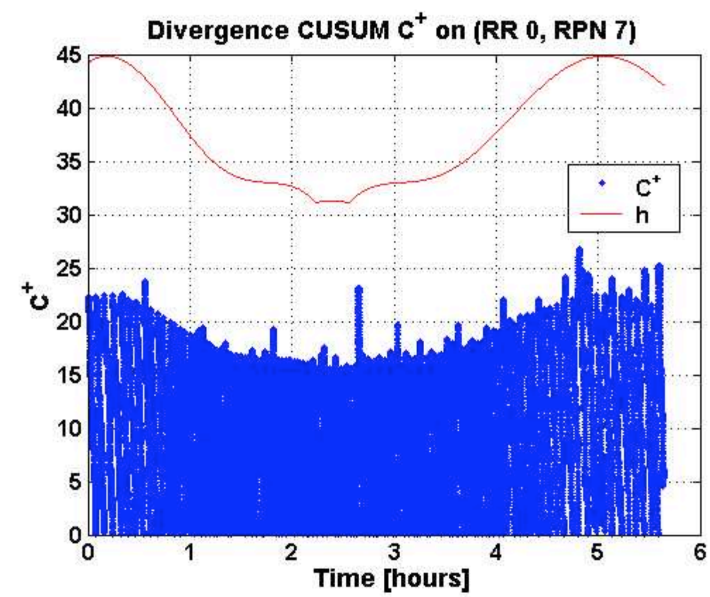

Figure 4-10-B

Figure 4-10 Nominal CUSUM Test Results

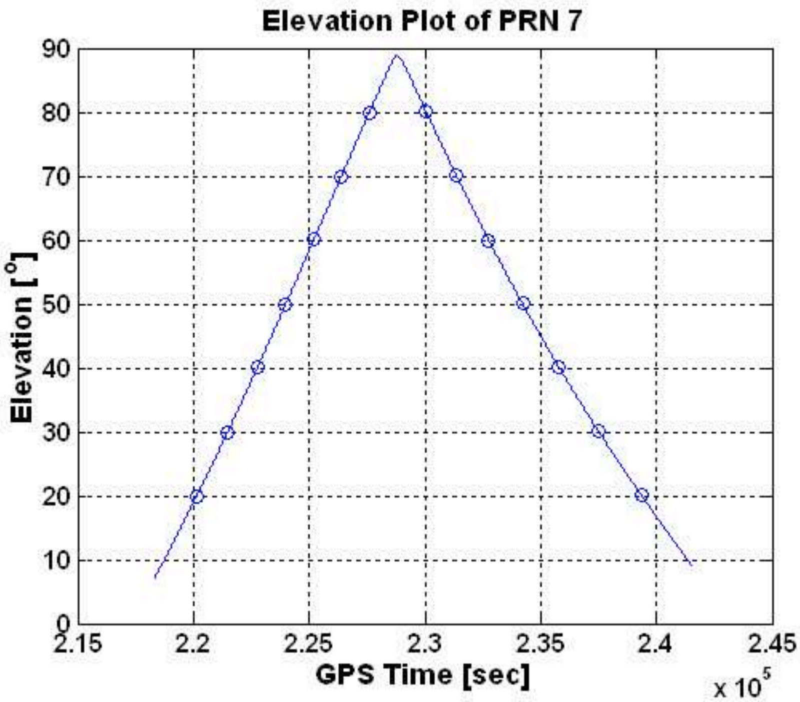

Figure 4-11 Satellite Elevation Plot of PRN 7

4.4.2 Failure Testing

Failure testing is necessary in order to compare the performance of the CUSUM to other integrity monitors and to the theoretical predictions of Chapter 3. Figure 3-3 (on page 95) shows the failure model for ionospheric gradients. The duration of ionospheric gradients is set at 173 seconds based on the WAAS supertruth data [12,32]. In Figure 4-11, the circles on the plot are the points where gradient anomalies start to be injected on all three channels of PRN 7. Each failure test case is independent of the others, as it acts upon a different "swath" of nominal data. During the failure tests, EXM is partially disabled as needed so that the detection times of all relevant monitors can be measured. Note that ionospheric gradients may be simulated with an infinite duration in order to fully measure and compare the performance among different monitors.

Elev. [°]	Time to Flag [sec]							
	Acc-Ramp-Step		Innovation		GMA Divergence		Divergence CUSUM	
20	15.5	32.5	>500	>500	373.5	462.0	200.0	220.0
30	2.0	2.5	>500	>500	303.0	455.5	198.5	310.5
40	248.0	259.0	>500	>500	276.5	309.0	151.0	179.5
50	>500	>500	>500	>500	204.0	205.5	91.0	135.5
60	>500	>500	>500	>500	167.0	174.5	116.5	122.5
70	2.0	2.0	381.0	>500	188.5	195.5	61.5	78.5
80	126.5	215.5	478.5	>500	153.5	166.5	55.5	119.0
80	1.5	1.5	>500	>500	154.5	171.5	111.5	124.0
70	115.0	134.0	>500	>500	170.5	191.5	94.0	97.5
60	>500	∞	∞	∞	154.0	202.5	154.5	175.0
50	2.0	∞	∞	∞	192.5	203.0	126.5	137.5
40	50.0	∞	>500	∞	249.0	275.0	166.0	178.5
30	199.0	437.5	265.0	∞	239.0	348.5	178.5	220.5
20	70.0	149.0	∞	∞	356.5	358.0	314.0	∞

Table 4-1 Failure Test Results by Injecting Ionospheric Gradients of $0.011 \times OF$ m/s at Different Elevations (All Monitors Considered Here Have the Same False Alarm Rate)

The first set of failure tests checks the performance of the CUSUM method in the presence of ionospheric gradients that cannot be reliably detected by the acceleration-ramp-step test. Table 4-1 shows the time needed for the acceleration-ramp-step, innovation, GMA divergence, and divergence CUSUM monitors to detect an ionospheric gradient that is equal to 0.011 m/s multiplied by the corresponding obliquity factor in each test case. The symbol "∞" indicates that the corresponding monitor fails to detect the gradient during the interval simulated. For each monitor, the two earliest detection times among the three IMT receivers tracking PRN 7 are listed. According to the EXM logic detailed in Section 2.7, when one channel is detected as failed, the EXM will exclude this channel and its erroneous measurements will not propagate to other channels. If two or more channels have failed, the corresponding satellite will be excluded on all reference receivers. The cells marked in gray in Table 4-1 reflect the monitor that first excludes the satellite in that test case. The table shows that the CUSUM method excludes the satellite first in nine out of the fourteen cases, while the acc-ramp-step method detects the failure first in the remaining five cases. The CUSUM method is a reliable method and its performance changes gradually, while the performance of the acceleration-ramp-step method is dramatically different in different cases, since random measurement noise plays a decisive role when a statistic is close to its threshold. The CUSUM method is always better than the GMA method except for the last test case, and the innovation method is the worst in this test set.

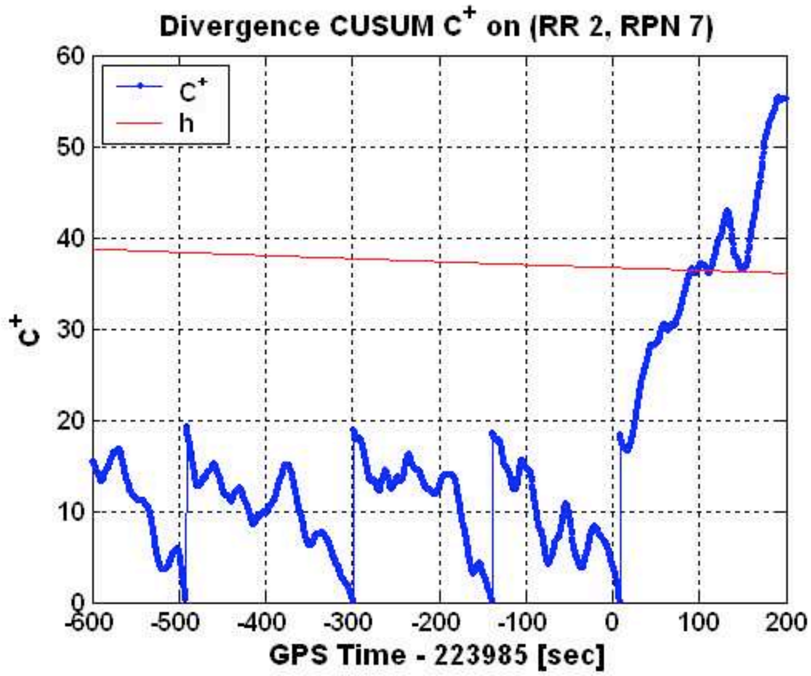

Figure 4-12 Failure Testing CUSUM Results with an Ionospheric Gradient of $0.011 \times OF$ m/s Injected at 50^o Elevation

Figure 4-12 shows the CUSUM response over time for the rising 50^o failure test case in Table 4-1. Before the fault affects IMT measurements, $C^+(k)$ is repeatedly reset to

$\frac{h^+(k)}{2}$ as its value becomes negative. Once the anomaly is injected at the GPS time of 223,985 seconds, the CUSUM statistic rapidly rises and crosses the threshold after 91.0 seconds.

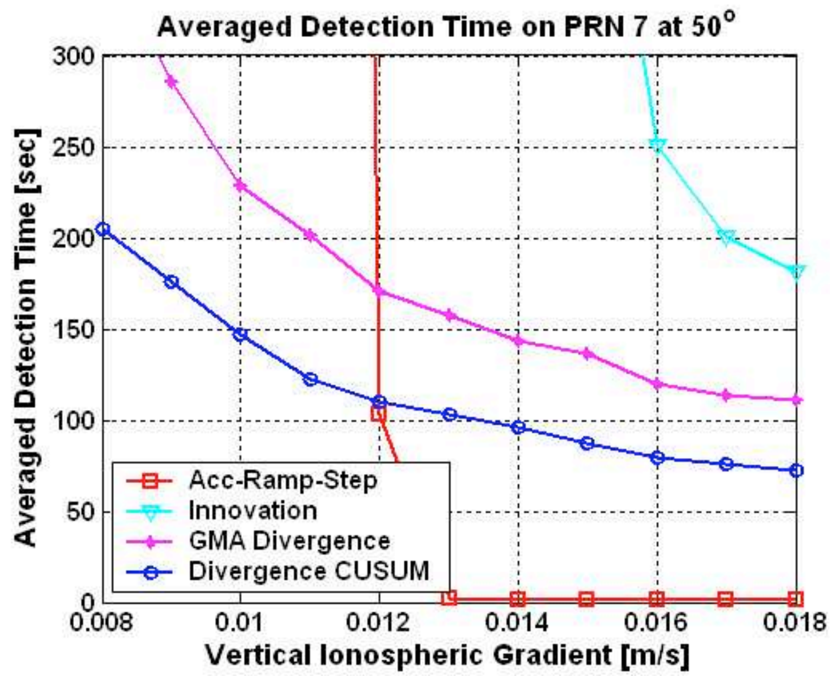

Figure 4-13 Failure Test Result by Injecting Different Magnitudes of Ionospheric Gradients at 50^o (All Monitors Considered Here Have the Same False Alarm Rate)

The second set of failure tests injects ionospheric gradients with different magnitudes into the measurements of PRN 7 when it is at 50^o elevation (both rising and setting). The injected vertical gradient varies from 0.008 m/s to 0.018 m/s, with a 0.001 m/s increase at each step. Figure 4-13 is a plot of the averaged detection time for the four monitors. The averaged detection time at each elevation angle is the value obtained by averaging the two earliest detection times from three channels in both the rising and setting test cases. From Table 4-1, for example, the averaged time to detect a vertical ionospheric gradient of 0.011 m/s at 50^o by the CUSUM method is equal to:

$$\frac{1}{2}\left(\frac{91.0+135.5}{2}+\frac{126.5+137.5}{2}\right)=122.6 \text{ sec} \tag{4-27}$$

The figure clearly shows that the acceleration-ramp-step test can detect relatively large gradients quickly. The CUSUM method does not detect large failures faster than the acceleration-ramp-step monitor, but it is usually better than the GMA divergence and innovation tests. It is also the fastest in detecting small gradients, as expected. The acceleration-ramp-step and CUSUM methods together give the lowest bounds on detection time.

The third set of failure tests repeats the second set but at 80^o elevation (rising and setting). Figure 4-14 shows the performance of the four monitors. It shows the same phenomena as Figure 4-13. The acceleration-ramp-step and CUSUM methods together again give the lowest bounds on detection time. Their performance curves intersect at approximately $\dot{I} = 0.01$ m/s at 80^o instead of 0.012 m/s at 50^o.

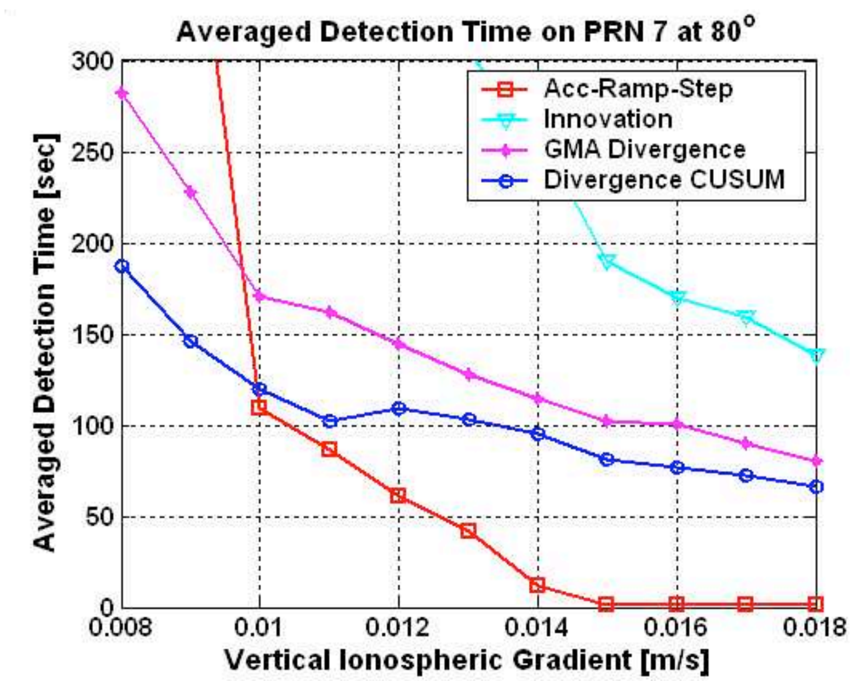

Figure 4-14 Failure Test Result by Injecting Different Magnitudes of Ionospheric Gradients at 80^o (All Monitors Considered Here Have the Same False Alarm Rate)

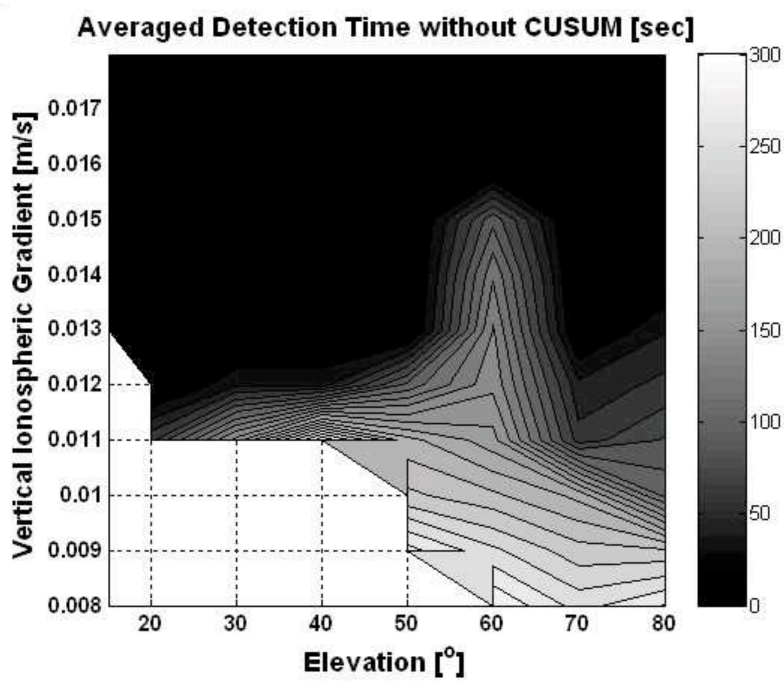

Figure 4-15-A

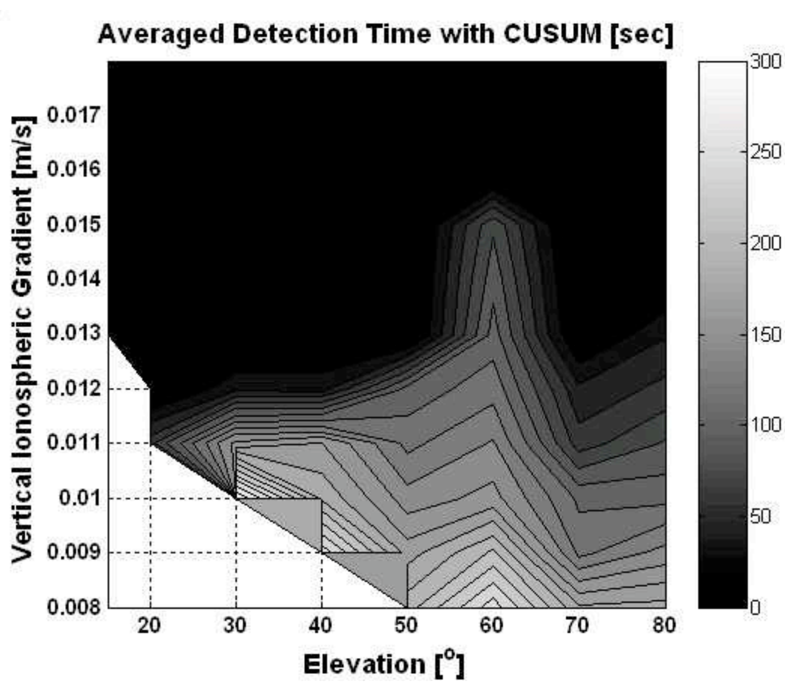

Figure 4-15-B

Figure 4-15 Averaged Detection Time by the IMT (A) without and (B) with the Divergence CUSUM Method

Figures 4-13 and 4-14 compare the performance of the four integrity monitors in terms of averaged detection time at elevations of 50^o and 80^o, respectively. The next step is to compare them across the whole failure space of ionospheric gradients, i.e., with ionospheric gradients of any magnitude and at any elevation. Failure tests similar to the second and third failure test sets were performed at elevations of 15^o, 20^o, 30^o, 40^o, 60^o, and 70^o. Figure 4-15-A shows the fastest detection speeds achieved by the acceleration-ramp-step, innovation, and GMA divergence tests in the IMT. The gray scheme indicates average detection time, with black representing a fast detection speed and white representing a slow detection speed. If the average detection time is larger than 300 seconds or a monitor fails to detect a failure, it is identified by a blank area in the figure. The short detection time in the top half of the figure is largely accomplished by the acceleration-ramp-step test. Figure 4-15-B shows the comparison after the divergence CUSUM method is integrated into the IMT. Compared with Figure 4-15-A, this figure clearly demonstrates that the overall performance is improved by the CUSUM method, especially in the bottom half, since the CUSUM method is fastest (in relative terms) at detecting small gradients. Moreover, the gradients can be detected earlier at low elevations, and this is crucial to the LAAS operation, since moving ionospheric gradients will generally affect low-elevation satellites first [32].

Overall, these failure tests demonstrate that the divergence CUSUM method can most quickly detect hazardous ionospheric gradients that the acceleration-ramp-step test fails to detect first. The CUSUM method reduces the average detection time of the previous GMA divergence method by about 30%.

4.4.3 Minimum Detectable Vertical Ionospheric Gradients

It is important to determine the minimum detectable vertical ionospheric gradients with the required missed-detection probability (e.g., $P_{MD} = 0.001$) in the presence of the integrity monitor. As defined in Section 3.9.2, the MDEs are the smallest magnitudes of threatening failures that can be detected within a certain missed-detection probability,

P_{MD}. Figure 3-6 on page 103 showed the minimum detectable ionospheric gradients achieved by the innovation, divergence, and ramp tests. This section will use the discretized Markov Chain model to calculate theoretically the MDEs achieved by the divergence CUSUM method. Calculated MDEs are a function of elevation angle.

To evaluate the MDEs using the CUSUM method, the out-of-control ARL needs to be calculated first based on a discretized Markov Chain model [63]. The example here is to calculate the ARL at 50^o with a vertical ionospheric gradient, $\dot{I}$. From the algorithm in Section 4-3, $f\sigma$, h, and V at this elevation can be determined. They are equal to 0.0339 m/s, 36.7, and 0.353, respectively. The input to the CUSUM after normalization, X, has a normal distribution with the actual mean at:

$$\mu_{1,act} = \frac{\dot{I} \cdot OF_{\theta_{elev}}}{f\sigma} \tag{4-28}$$

and standard deviation at 1. In addition, $\mu_0 = 0$ and $\mu_1 = V$, as given in Section 4.3.5. With all the above information and Equation (4-26), the $(M+2)\times(M+2)$ transition matrix, $\mathbf{P}$, can be derived. Section 4.2 explained how to derive the matrix, $\mathbf{P}$, in detail.

The CUSUM has an initial value at $\frac{h}{2}$ according to the definition of the $\frac{h}{2}$ FIR ARL [21]. In other words, the CUSUM is initially in State $\frac{M}{2}$ with a probability of 1. The corresponding vector of state probabilities, $\varphi(0)$, at epoch 0 is [0, …, 0, 1, 0, …, 0]. The elements in φ are the probabilities of the CUSUM in States 0, 1, …, $M+1$. The probability of the CUSUM in different states evolves once per epoch based on the transition probabilities, i.e.:

$$\varphi(k) = \varphi(k-1) \cdot \mathbf{P} = \varphi(0) \cdot \mathbf{P}^k \tag{4-29}$$

Note that the last element in $\varphi(k)$ is the probability of the CUSUM in State $M+1$ at epoch k, or, equivalently, the probability that the CUSUM value exceeds the detection threshold, h. The top plot of Figure 4-16 shows the number of epochs needed for CUSUM to transit to State $M+1$ with a probability of 0.999 or greater for different $\mu_{1,act}$

at 50^o. Based on the IMT GPS measurement update rate, $T_s = 0.5$ seconds, and Equation (4-23), this result can be mapped one-to-one to the bottom plot of Figure 4-16, which is the time needed to detect minimum detectable vertical ionospheric gradients 99.9% of the time. Dots in Figure 4-16, for example, show that the ARL is 346 epochs (or, equivalently, 173 seconds) in detecting an actual out-of-control mean failure, $\mu_{1,act}$, of 0.4953 (or, equivalently, a vertical ionospheric gradient of 0.0133 m/s). Therefore, the minimum detectable vertical ionospheric gradient that can be detected by the CUSUM method within 173 second is 0.0133 m/s, as also identified by a dot in Figure 4-18. Figure 4-17 shows how the probability of State $M+1$ changes with time under a vertical ionospheric gradient of 0.0133 m/s at 50^o. It is obviously a monotonic increasing function since there is no chance for the CUSUM to transit from State $M+1$ to any other states once it exceeds the threshold, as also indicated in Figure 4-4. It takes 346 epochs to reach the probability of 99.9%.

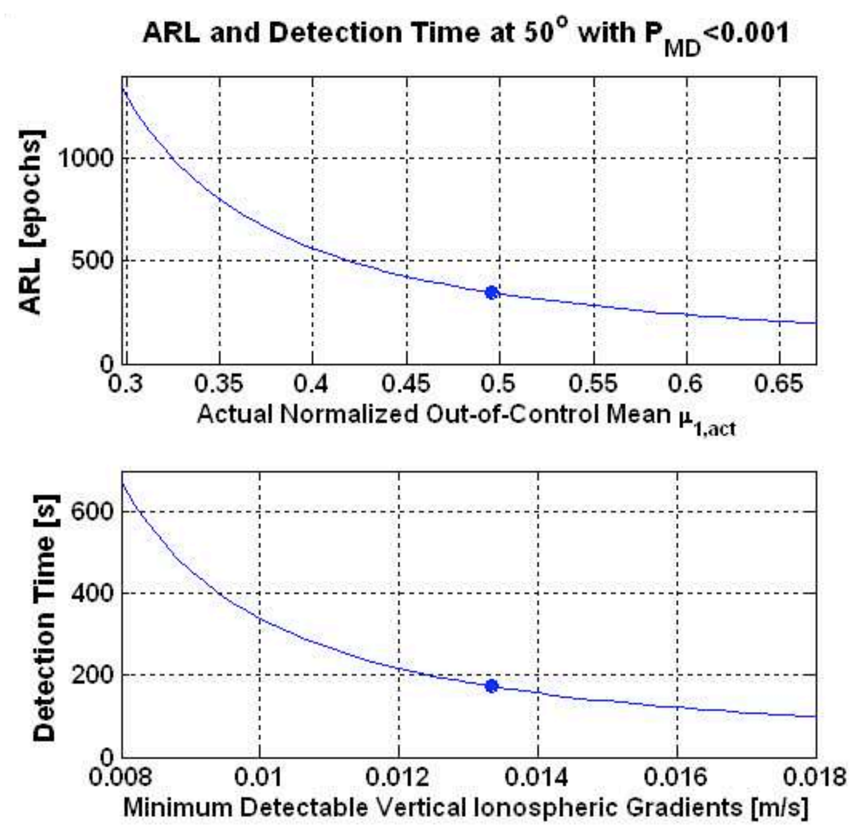

Figure 4-16 Divergence CUSUM ARL at 50^o

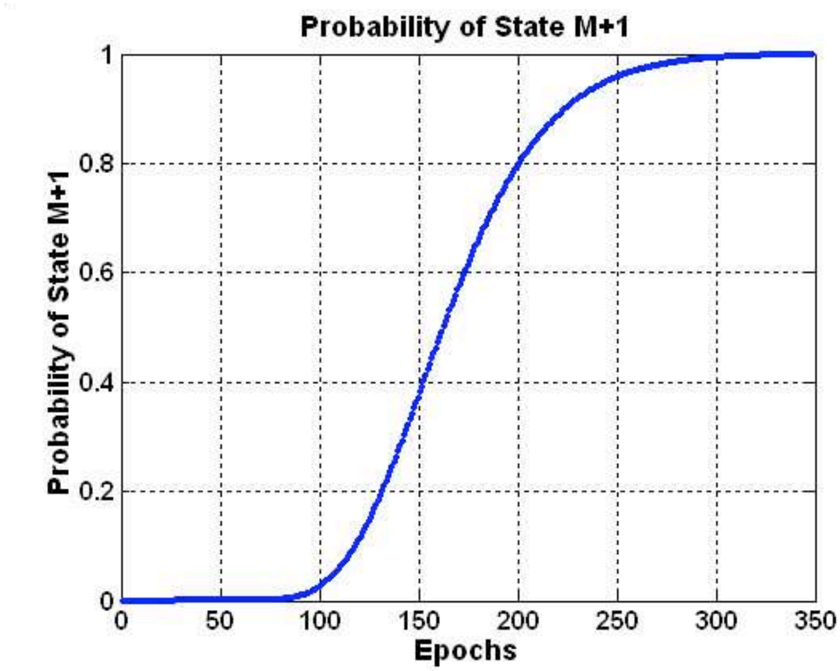

Figure 4-17 The Probability of the CUSUM in State $M+1$ versus Time under a Vertical Ionospheric Gradient of 0.0133 m/s at 50^o

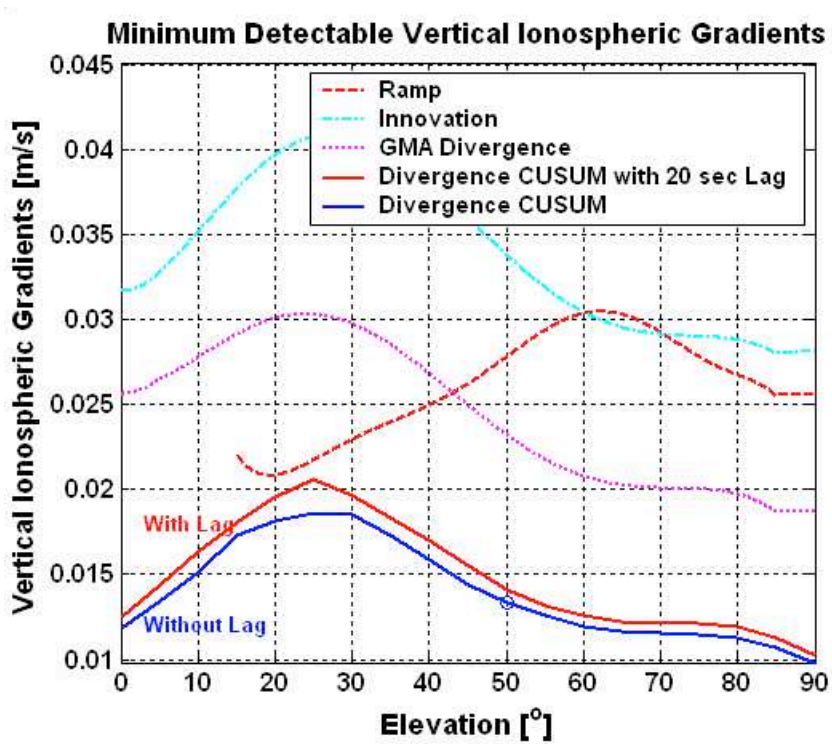

Figure 4-18 Minimum Detectable Vertical Ionospheric Gradients

The MDEs at other elevations can be obtained with the same method as just described. The solid (bottom) line in Figure 4-18 shows the MDEs for the divergence CUSUM method. In the calculation of the CUSUM MDEs on this line, the delay effect from Equation (4-15) is ignored. If the delay is assumed to be $k_0 = 40$ epochs (or, equivalently, 20 seconds), then the CUSUM MDEs are also calculated and shown in the figure. The MDEs with and without consideration of the delay are very close.

For convenience, the MDEs from the other three IMT monitors (as already shown in Figure 3-6) are also plotted in Figure 4-18. In this figure, the innovation, GMA, and divergence CUSUM monitors all detect their MDEs within 173 seconds, while the ramp test can detect its MDEs in about 5 seconds. The figure shows that the divergence CUSUM method can detect smaller ionospheric gradients and thus detect the same ionospheric gradients faster than the GMA and innovation tests. As already pointed out in Section 3.9.2, ionospheric gradients above 0.01 m/s can be hazardous for a future LAAS system. Figure 4-18 demonstrates that the CUSUM method can quickly and reliably detect marginal ionospheric gradients. It achieves the design goal.

4.5 Conclusion

In this chapter, the CUSUM algorithm was explained and applied to the code-carrier ionosphere divergence problem, for which pre-existing IMT monitors are slow to detect. The divergence CUSUM monitor is designed to detect more rapidly relatively small (but still potentially hazardous) gradients that the acceleration-ramp-step monitor fails to detect.

Our nominal and failure test results have demonstrated that the CUSUM method successfully achieves this goal. The acceleration-ramp-step test detects large ionospheric gradients in seconds, while the CUSUM method is fastest in detecting the small ionospheric gradients that the acceleration-ramp-step test completely fails to detect. The CUSUM method improves upon the detection speed of the previous GMA estimation method by about 30%. Thus, the divergence CUSUM and acceleration-ramp-step monitors combined offer the lowest bounds on detection time. This improvement is significant in terms of reducing the risk posed by ionosphere spatial gradient anomalies. Calculations in [32] demonstrate that the worst vertical positioning error can be reduced from 19 meters to 5 meters when the CUSUM and acceleration-ramp-step methods are implemented together.

This method is suggested for future Category II/III precision landings supported by LAAS. Because of its simple computation and excellent performance, applying variants of the CUSUM method to detect other kinds of failures is of great interest for future study.

Chapter 5 Failure-Specific Tests

5.1 Introduction

The analysis in Chapter 3 described how several different integrity monitors would respond to the same failures. Figure 5-1 demonstrates this fact again based on the more general failure space of GPS measurements. In this figure, the *x*-axis is the code measurement error, $\nabla\rho$, and the *y*-axis is the carrier phase measurement error, $\nabla\phi$. For example, the MQM acceleration, ramp, and step test statistics all respond to satellite clock failures (characterized by $\nabla\rho = \nabla\phi$). Similarly, different monitors, such as the acceleration-ramp-step, innovation, and GMA divergence monitors, all can detect ionospheric gradients (characterized by $\nabla\rho = -\nabla\phi$) but with different detection speeds. If several different test statistics that can respond to the same failures are combined together into a new test statistic, this new test may be able to detect the failures faster than each of these individual monitors. Here, this new test statistic is named the "failure-specific test (FST) statistic" and the individual monitors are called "component monitors." The idea of FSTs was pointed out by Pullen in [70].

This chapter demonstrates theoretically the concept of failure-specific testing in Section 5.2. Two practical FSTs are given in Sections 5.3 and 5.4. Also, in Section 5.3, the overbounding method used to determine the thresholds of FST statistics is discussed in detail.

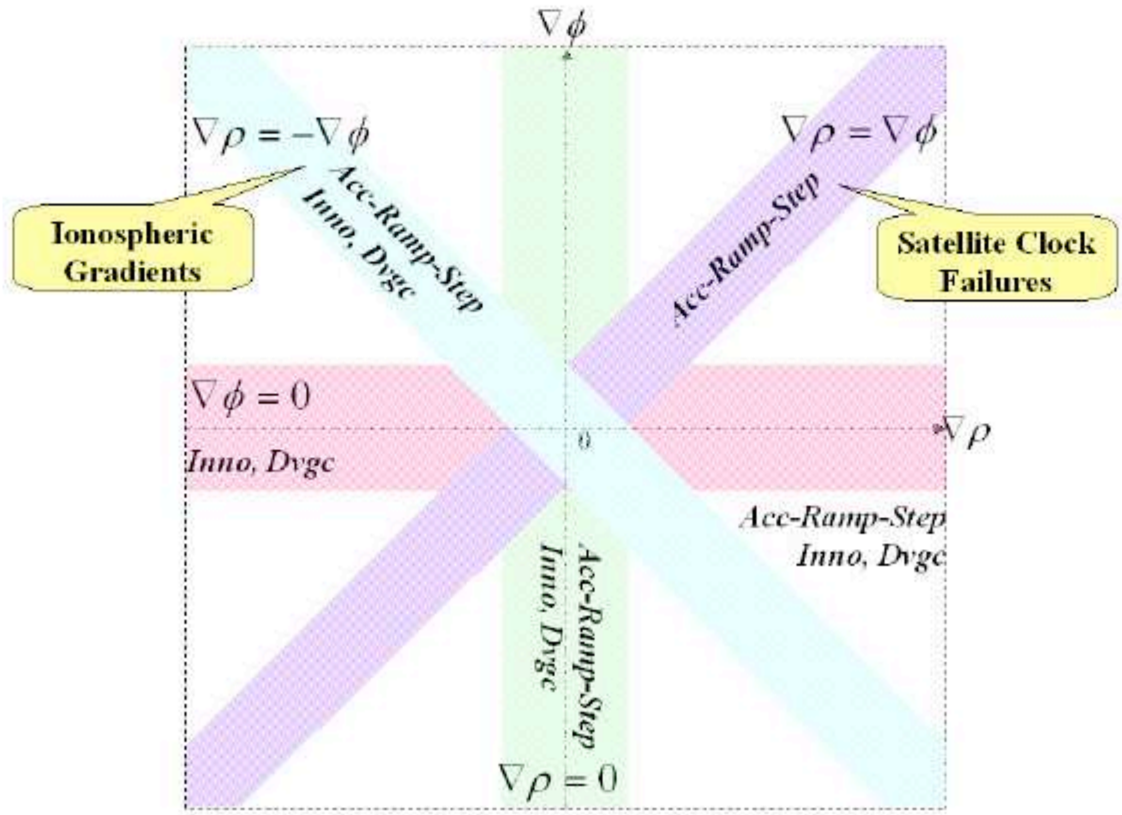

Figure 5-1 EXM-I Integrity Monitors in the Failure Space of Code and Carrier Measurements

5.2 Illustrative Example

IMT test statistics are modeled as Gaussian variables. When they are combined, the distribution of the combined statistic needs to be known. Assume that X_1, X_2, …, X_K are K independent Gaussian random variables with mean, μ_i, and variance, σ_i^2, for $i = 1$, …, K, written as

$$X_i \sim N(\mu_i, \sigma_i^2) \tag{5-1}$$

A new random variable, Y, models the FST statistic, and is defined as the sum of these K squared independent standard normal variables:

$$Y = \sum_{i=1}^{K} \left(\frac{x_i - \mu_i}{\sigma} \right)^2 \tag{5-2}$$

In this case, Y has a χ^2-distribution with K degrees of freedom [8,38,71]. It is expressed as:

$$Y \sim \chi_K^2 \tag{5-3}$$

The mean and variance of a χ_K^2-distribution are [38]:

$$E(Y) = K \tag{5-4}$$

and

$$Var(Y) = 2K \tag{5-5}$$

A χ^2-distribution is a special case of the gamma distribution, $\gamma(\alpha, \beta)$ [38]. These two distributions have the following relation:

$$\chi_K^2 = \gamma\left(\frac{K}{2}, 2\right) \tag{5-6}$$

i.e., $\alpha = \frac{K}{2}$ and $\beta = 2$. The definitions of the parameters for the χ^2 and γ distributions differ among references. The ones defined in MATLAB are used here [35].

Each Gaussian random variable, X_i, can be regarded as a test statistic in the IMT, such as the outputs of the innovation or ramp monitors. If the threshold for X_i is set at $\mu_i \pm 6\sigma_i$ (for briefness, σ_i actually represents inflated standard deviation; the inflation factor is derived using the Gaussian overbounding method discussed in Section 2.2), then the probability of false alarm is equal to:

$$\Pr\left(\left|x_i - \mu_i\right| > 6\sigma_i\right) = 2 \cdot \Pr\left(\frac{x_i - \mu_i}{\sigma_i} > 6\right) = 2 \cdot Q(6) = 1.9732 \times 10^{-9} \tag{5-7}$$

A threshold for the new test statistic, Y, needs to be determined before its performance is evaluated. In order to make a fair comparison between Y and X_i, the threshold of Y is set so that the probability of false alarm is also equal to 1.9732×10^{-9}, the same as that of each separate monitor, X_i.

Since the IMT has three reference receivers, this section takes the case of $K = 3$ for illustration, i.e., $Y = \sum_{i=1}^{3}\left(\frac{x_i - \mu_i}{\sigma}\right)^2$. Y is χ_3^2-distributed, or, equivalently, $\gamma(\frac{3}{2},2)$-distributed. Since

$$\Pr(Y > 43.4518) = 1.9732\times10^{-9} \tag{5-8}$$

the threshold of Y is set at 43.4518.

After the threshold of Y has been determined, its detection speed can then be evaluated. Continuing with the example of $K = 3$, this section considers the following three cases:

1. All of X_1, X_2, and X_3 are precisely at their detection thresholds, and thus, each of these three X_i generates a flag:

   $\begin{cases} x_1 = \mu_1 \pm 6\sigma_1 \\ x_2 = \mu_2 \pm 6\sigma_2 \\ x_3 = \mu_3 \pm 6\sigma_3 \end{cases}$, then $y = 6^2 + 6^2 + 6^2 = 108 > 43.4518$ from Equation (5-2).

   This is the worst case for Y with three X_i flags, since the value of Y is now at its minimum.

2. Two of the X_i values are precisely at their thresholds, and the third one is at its mean:

   $\begin{cases} x_1 = \mu_1 \pm 6\sigma_1 \\ x_2 = \mu_2 \pm 6\sigma_2 \\ x_3 = \mu_3 \end{cases}$, then $y = 6^2 + 6^2 + 0 = 72 > 43.4518$.

   This is the worst case for Y with two X_i flags.

3. Only one of X_i is precisely at its threshold, and the others are at their means:

   $\begin{cases} x_1 = \mu_1 \pm 6\sigma_1 \\ x_2 = \mu_2 \\ x_3 = \mu_3 \end{cases}$, then $y = 6^2 + 0 + 0 = 36 < 43.4518$.

   This is the worst case for Y with one X_i flag.

These three scenarios provide desirable results for Y. The first two cases indicate that Y is capable of detecting failures earlier than each individual X_i. In Case 3, Y does not flag, and in fact it should not. The reason is as follows. X_1, X_2, and X_3 are typically values from three channels tracking the same satellite. Thus, Y, the squared sum of the three values, represents the status of the satellite. In other words, X_i is used to detect channel failures, while Y is designed to detect satellite failures. If Y is larger than its threshold, the whole satellite, including all three of its channels, should be excluded. In Cases 1 and 2, the satellite will be excluded since Y is larger than its threshold. According to the EXM discussed in Section 2.7, the satellite should also be excluded, since there are multiple flags on the same satellite. In Case 3, since there is only one flagged channel, the EXM will not exclude the satellite. Neither will the FST Y. Therefore, Y not only follows the EXM failure exclusion rules, but also is able to detect failures more quickly than X_i. However, there is some likelihood that the results from Y and from the EXM will be inconsistent. This is because their false alarm scenarios do not completely overlap, though they both have approximately the same false alarm probabilities.

The example of $K = 3$ illustrates that the new FST statistic, Y, the sum of squared Gaussian random variables, can detect failures faster than the component test statistics. In the following sections, two practical IMT FSTs are designed and analyzed.

5.3 Failure-Specific Test for Satellite Clock Failures

5.3.1 Construction

Figure 5-1 shows that acceleration, ramp, and step test statistics all respond to satellite clock failures. In our first example, these three test statistics from three channels corresponding to the same satellite are squared and summed to construct a new FST statistic named $H_{sv,ars}$, which is designed to quickly detect satellite failures. Mathematically, it is constructed as follows:

$$H_{sv,ars,n}(k) = \frac{1}{9}\sum_{i=0}^{2}\left(x_{acc,i,n}^{2}(k) + x_{ramp,i,n}^{2}(k) + x_{step,i,n}^{2}\right) \tag{5-9}$$

where

$$x_{acc,i,n}(k) = \frac{Acc_{i,n}(k)}{f_{acc}\sigma_{acc}(k)} \tag{5-10}$$

$$x_{ramp,i,n}(k) = \frac{Ramp_{i,n}(k)}{f_{ramp}\sigma_{ramp}(k)} \tag{5-11}$$

$$x_{step,i,n}(k) = \frac{Step_{i,n}(k)}{f_{step}\sigma_{step}(k)} \tag{5-12}$$

for receiver indexes, $i = 0$, 1, and 2. Satellites are indexed by PRN n. As can be seen from Equations (5-10)-(5-12), acceleration, ramp, and step test statistics are normalized by their inflated standard deviations (the inflation factors and standard deviations were discussed in detail in Chapter 2). Thus, $H_{sv,ars}$ integrates nine unit normal random variables (in the overbounding sense) at each epoch, k, and ideally $9H_{sv,ars}$ should be χ_9^2-distributed. For a probability of false alarm equal to 1.9732×10^{-9}, the threshold for $9H_{sv,ars}$ is 59.1281, or, equivalently, ${59.1281}/{9} = 6.5698$ is the threshold on $H_{sv,ars}$.

Figure 5-2 shows IMT nominal test results for $H_{sv,ars}$ from all satellites, with a maximum of 8.4697. Since the acceleration, ramp, and step statistics are normalized, $H_{sv,ars}$ appears to be roughly, but not entirely, independent of satellite elevation. In other words, the normalizations are approximately correct but are not perfect.

Figure 5-3-A shows the actual distribution of $9H_{sv,ars}$, compared with the PDF of χ_9^2. The plot shows that they look quite different and that the theoretical threshold is not large enough to cover all nominal test points. One of the reasons why $9H_{sv,ars}$ is not χ_9^2-distributed is that the three component test statistics are not exactly Gaussian distributed which violates the Gaussian assumption on the input random variables. If the averaged value of three accelerations is regarded as one random variable, and ramps and steps are also averaged, then $H_{sv,ars}$ from Equation (5-9) actually combines only three independent unit normal variables. In Figure 5-3-B, the actual distribution of $3H_{sv,ars}$ is compared with the PDF of χ_3^2. The tail of the actual distribution of $H_{sv,ars}$ is clearly much more overbounded by $\frac{1}{3}\chi_3^2$-distribution than by $\frac{1}{9}\chi_3^2$-distribution, in terms of overbounding the tail of the actual distribution. However, the new threshold of $H_{sv,ars}$ at $\frac{43.4518}{3} = 14.4839$ obtained from Section 5.2, becomes slightly larger than necessary, as will be seen from our nominal and failure tests later on. It is clear that $3H_{sv,ars}$ is not χ_3^2-distributed, either.

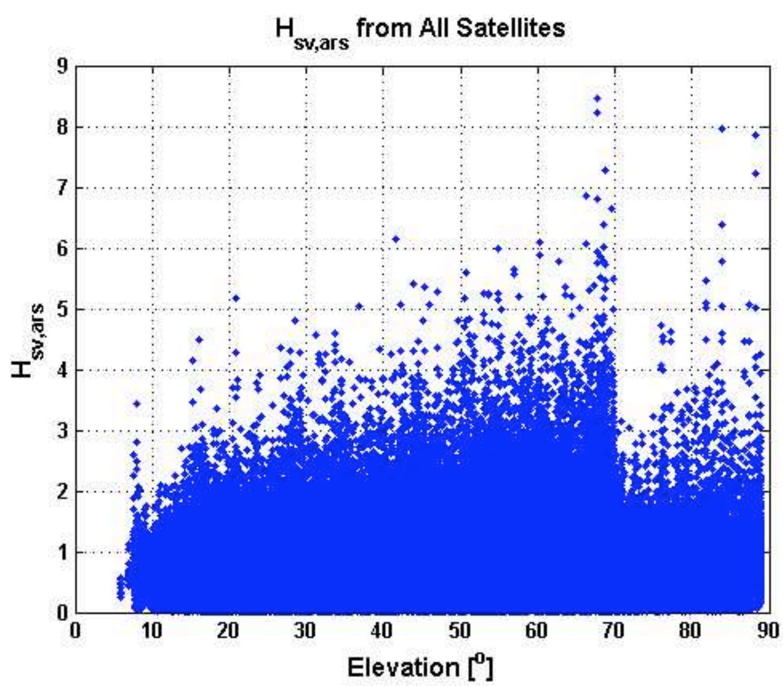

Figure 5-2 Nominal $H_{sv,ars}$ Test Results

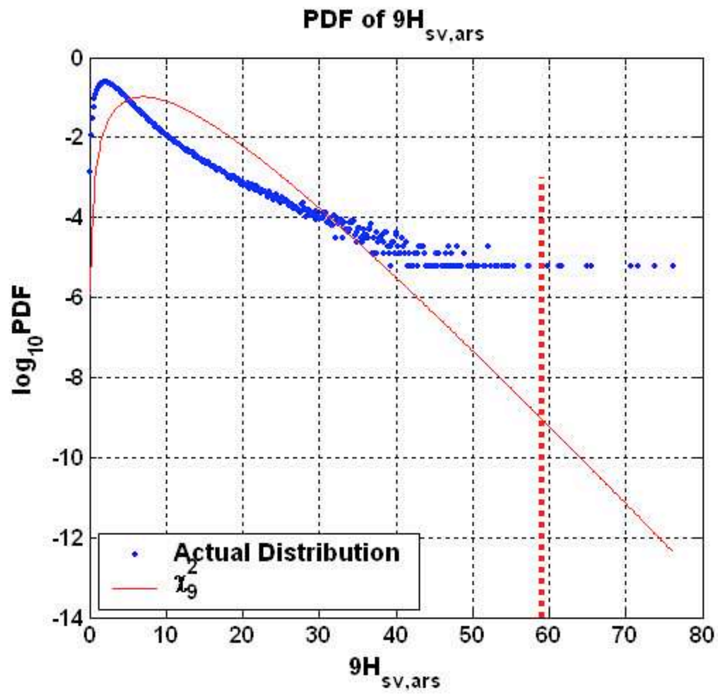

Figure 5-3-A

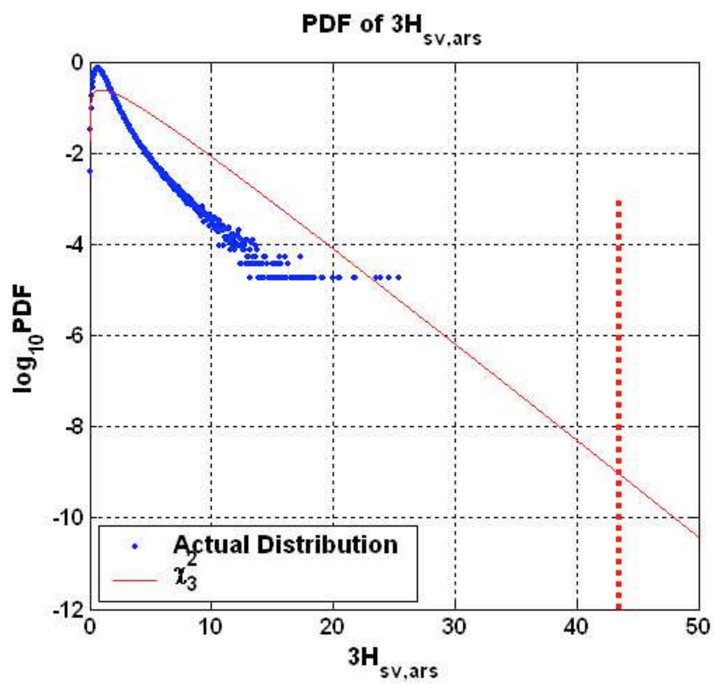

Figure 5-3-B

Figure 5-3 The Distribution of $H_{sv,ars}$

The reasons why neither the χ_9^2- nor the χ_3^2-distribution can be used to approximate the distribution of $H_{sv,ars}$ can be summarized as follows:

1. X_i are not perfectly normalized. The standard deviation, σ_i, cannot be precisely estimated and may not be equally well estimated in different elevation bins. They can be further away from actual values after being multiplied by an inflation factor, f_i. This can be the primary reason.
2. X_i are not exactly Gaussian distributed, although a Gaussian distribution is used to overbound the two tails of the actual distribution of X_i, in order to derive a conservative threshold.
3. X_i can be correlated. For example, divergence estimates through different IMT reference receivers should be highly correlated. Similarly, the acceleration, ramp, and step statistics from three channels can be somewhat correlated with each other.
4. Nominal data sets may contain small but significant abnormalities. It is not guaranteed that the nominal data sets that are used never contain any abnormal multipath or ionospheric gradients.

Therefore, it is not surprising to see that the actual $H_{sv,ars}$ is not χ^2-distributed. It is necessary to find a practical method to determine the threshold for $H_{sv,ars}$.

5.3.2 Practical Overbounding Method

As shown above, $H_{sv,ars}$ is not χ^2-distributed, and its threshold cannot realistically be derived from the χ^2-distribution. The objective in this section is to find a tractable distribution that can overbound the tail part of the actual distribution and then to derive the threshold based on this distribution while maintaining the same probability of false alarm. A proper threshold should not only be conservative in nominal testing (to meet the continuity allocation), but should also still lead to short detection times in failure testing. This section develops a practical method to build FST thresholds: carry out nominal and failure tests first, then recognize the range for a desirable threshold, and finally search for a reasonable and tractable distribution that yields a threshold in this range, if possible.

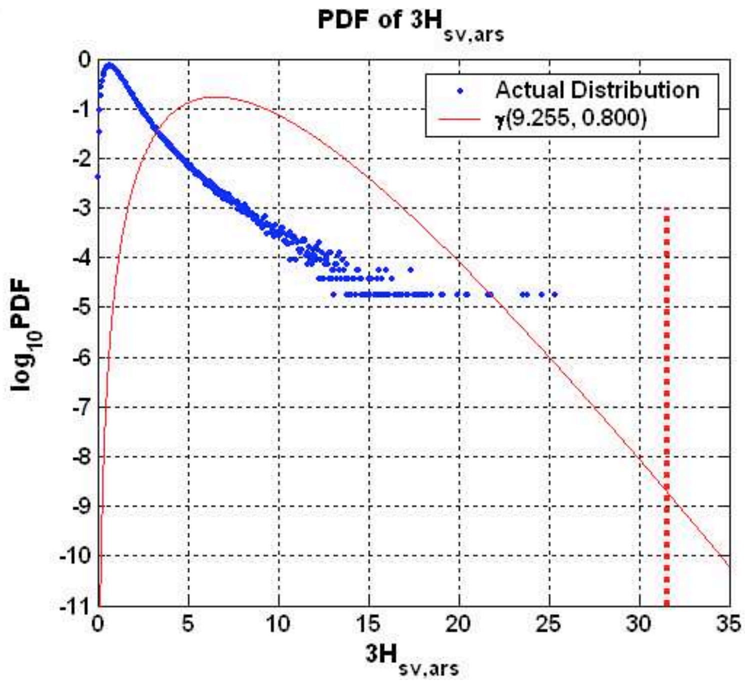

Figure 5-4 γ (9.255, 0.8)-distribution after Deleting Five Largest Tail Points

The χ^2-distribution is a special case of the gamma distribution as illustrated in Equation (5-6). The γ -distribution has one more parameter to be tuned than the χ^2-distribution. Since $3H_{sv,ars}$ is reasonably close to χ_3^2, or equivalently $\gamma(\frac{3}{2},2)$ (see Figure 5-3-B), the $\gamma(\alpha,\beta)$-distribution was chosen to overbound the tail of the actual distribution of $3H_{sv,ars}$. The parameters α and β are determined by using a free numerical search in the ranges $1 \le \alpha \le 10$ and $0.2 \le \beta \le 3$. The object of this search is to minimize the resulting threshold subject to the requirement to overbound the tail of the actual distribution of $3H_{sv,ars}$. The solid line in Figure 5-4 shows the PDF of γ (9.255, 0.8) that results from the numerical search. It overbounds the tail of the distribution of $3H_{sv,ars}$ except for the five largest tail points. With the same probability of false alarm from this

γ -distribution, the threshold of $H_{sv,ars}$ is $\frac{31.5636}{3} = 10.5212$, shown as the dashed line in Figure 5-4. This resulting threshold bounds all points, including the five largest tail points.

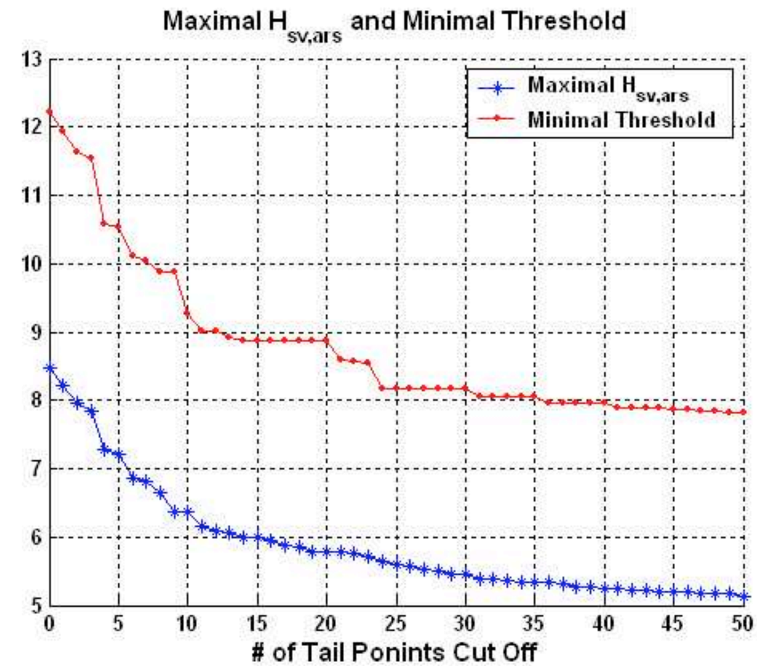

Figure 5-5 Maximum $H_{sv,ars}$ and Minimum Threshold after Deleting Largest Tail Points

The reason why the resulting γ -distribution was not used to overbound the five largest tail points is that these tail points are most likely driven by small non-hazardous anomalies. These small anomalies, which pass undetected by the existing IMT integrity monitors, can result in a large value in the FST statistic because of the FST's ability to detect small failures. They should not contribute to the distribution of $3H_{sv,ars}$ but will still be bounded by the threshold being derived. The star line in Figure 5-5 is the maximum value of the remaining $H_{sv,ars}$ after a certain number of the largest $H_{sv,ars}$ observations (shown on the *x*-axis) are deleted. Each time a tail point is deleted, an optimal γ -distribution that overbounds the actual distribution and also leads to a minimum threshold is found. The dotted line in Figure 5-5 is the minimum threshold versus the number of tail points deleted. Note that both the star and dotted lines have an "L" shape, and both decrease very quickly as the worst "tail" points are deleted.

The overbounding method determines the number of tail points (five points in this case) that need to be deleted based on nominal and failure test results. The threshold should be larger than the maximum value of $H_{sv,ars}$, 8.4697, from nominal testing so that no false alarm is generated in nominal tests. On the other hand, it should not be so large that the FST becomes incapable of detecting failures. Based on Figure 5-5, the choice was to delete the five largest tail points out of about one million points total, since the two curves decrease with a slower rate after that.

In brief, this practical overbounding method is used to search for an optimal γ-distribution that can yield a proper threshold within a desirable range predetermined from nominal and failure tests. The optimal γ-distribution does not overbound the whole tail of the data distribution, though tail points are bounded by the resulting threshold. More nominal and failure tests are needed to validate that the thresholds obtained from this method are conservative. There is also sufficient margin for raising the thresholds if necessary to protect continuity.

5.3.3 Failure Test Results

After the threshold of $H_{sv,ars}$ is set at 10.5212, failure testing can be used to make comparisons between $H_{sv,ars}$ and the three individual IMT integrity monitors from which it is generated. In order to do this, a satellite clock failure that caused the following carrier phase measurement error was designed:

$$\nabla\phi = 0 \cdot \frac{t^2}{2} + 0.014 \cdot t + rand\left[-0.006 : +0.006\right] \tag{5-13}$$

where the third term "$rand\left[-0.006 : +0.006\right]$" represents a uniformly distributed random noise in the range of ± 0.006 meters. The failure is injected into the carrier-phase measurements of PRN 7 with a duration of 173 seconds. Note that the acceleration-ramp-step method cannot distinguish whether the carrier phase measurement error is due to a satellite clock failure or due to an ionospheric storm.

Figure 5-6 shows the failure test results, where the failure occurs at GPS time 223985 seconds. The second term in Equation (5-13) is a ramp error, and it causes a sharp mean increase in the ramp test results shown in Figure 5-6-B. Meanwhile, all the acceleration, ramp, and step test results clearly appear to have larger standard deviations during the failure condition period due to the uniform random noise component (i.e., $rand\left[-0.006:+0.006\right]$), as can be observed in Plots (A), (B), and (C), respectively. The acceleration and ramp tests flag within 174.5 seconds after the failure is injected, and the step test does not flag. Conversely, $H_{sv,ars}$ detects the failure after only 35.5 seconds, which is much more rapid than the individual acceleration, ramp, and step tests. This advantage also holds for faults other than the one described by Equation (5-13).

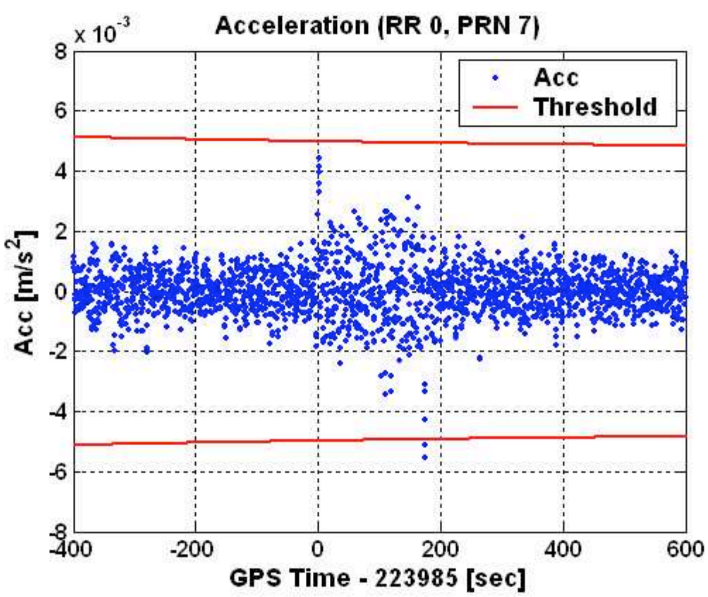

Figure 5-6-A

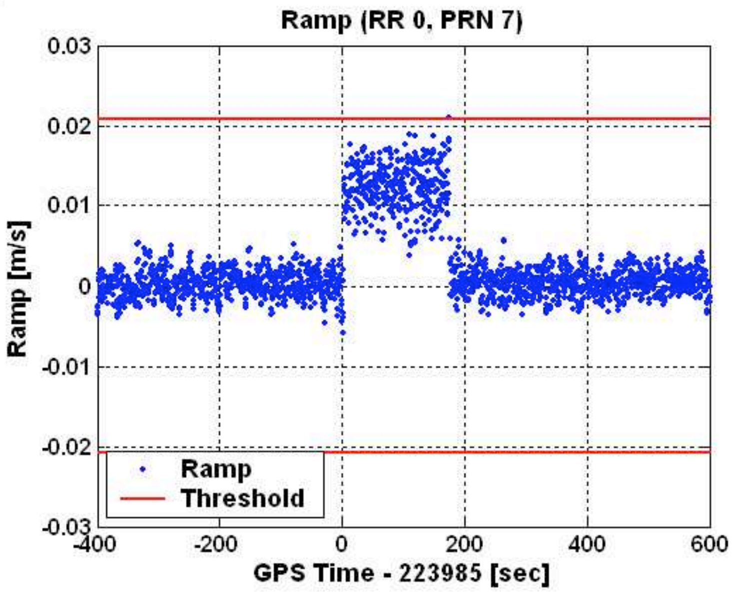

Figure 5-6-B

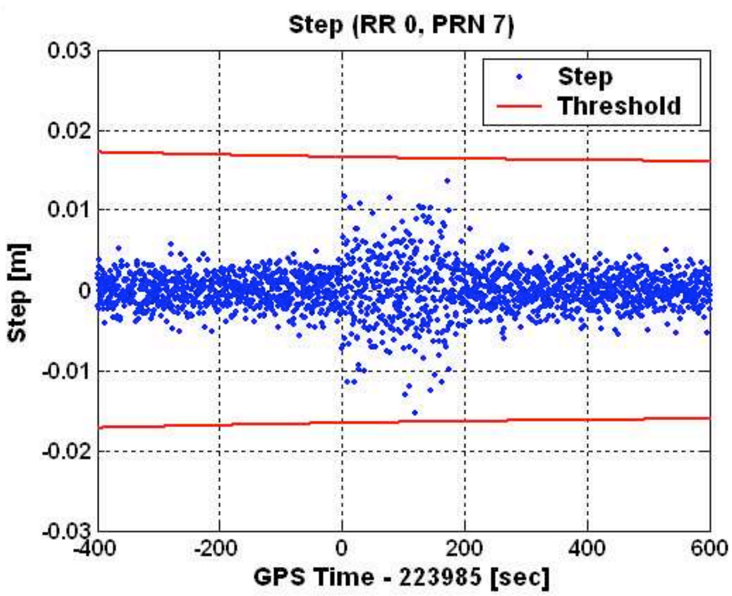

Figure 5-6-C

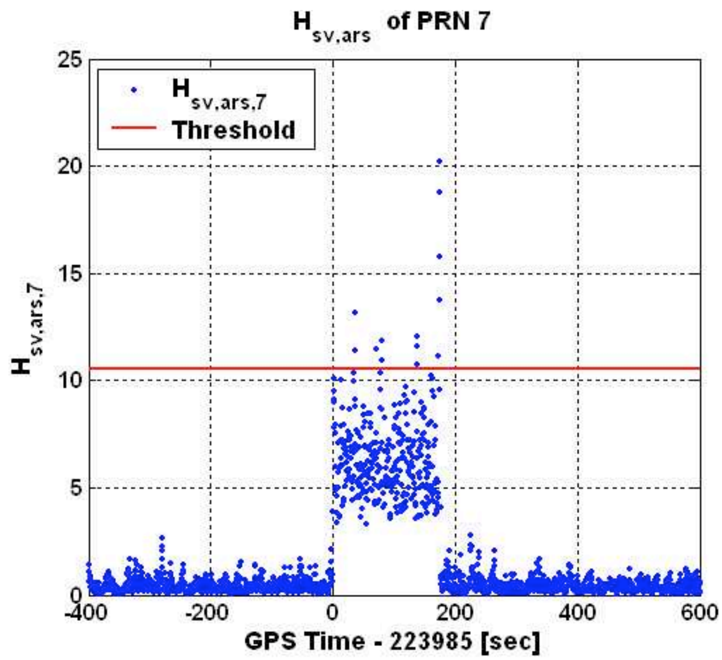

Figure 5-6-D

Figure 5-6 Failure Test Results of (A) Acceleration, (B) Ramp, (C) Step, and (D) $H_{sv,ars}$

5.4 Failure-Specific Test for Ionospheric Gradients

The FST $H_{sv,ars}$ in the previous section combined the acceleration, ramp, and step test statistics generated from the same MQM monitor. A second practical FST, named $H_{sv,dir}$, is analyzed in this section. $H_{sv,dir}$ is the sum of squared and normalized divergence, innovation, and ramp test statistics from three different monitors. As illustrated in Figure 5-1, these three monitors all respond to ionospheric gradients, so $H_{sv,dir}$ is designed to more quickly detect ionospheric gradients than any of the individual component monitors. Mathematically, it is constructed as follows:

$$H_{sv,dir,n}(k) = \frac{1}{9}\sum_{i=0}^{2}\left(x_{dvgc,i,n}^{2}(k) + x_{inno,i,n}^{2}(k) + x_{ramp,i,n}^{2}\right) \qquad (5\text{-}14)$$

where

$$x_{dvgc,i,n}(k) = \frac{Dvgc_{i,n}(k)}{f_{dvgc}\sigma_{dvgc}(k)} \qquad (5\text{-}15)$$

$$x_{inno,i,n}(k) = \frac{Inno_{i,n}(k)}{f_{inno}\sigma_{inno}(k)} \quad (5\text{-}16)$$

$$x_{ramp,i,n}(k) = \frac{Ramp_{i,n}(k)}{f_{ramp}\sigma_{ramp}(k)} \quad (5\text{-}17)$$

Figure 5-7 shows the actual distribution of $3H_{sv,dir}$, together with the PDF of χ_3^2. Unlike $3H_{sv,ars}$, the tail part of $3H_{sv,dir}$'s distribution is overbounded by the PDF of χ_3^2. This overbounding occurs because the standard deviations and inflation factors might be overestimated in the elevation bins to which the largest tail points belong.

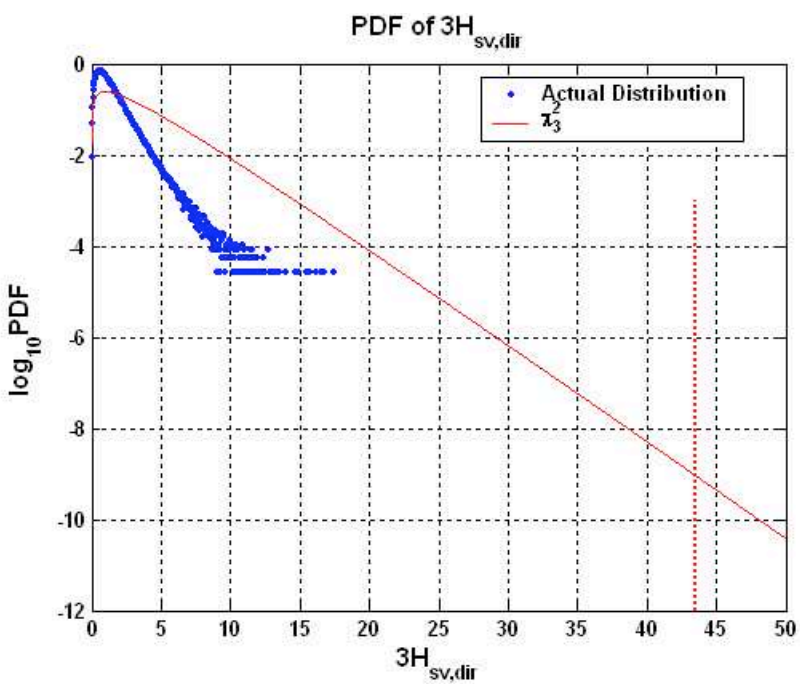

Figure 5-7 The Distribution of $H_{sv,dir}$

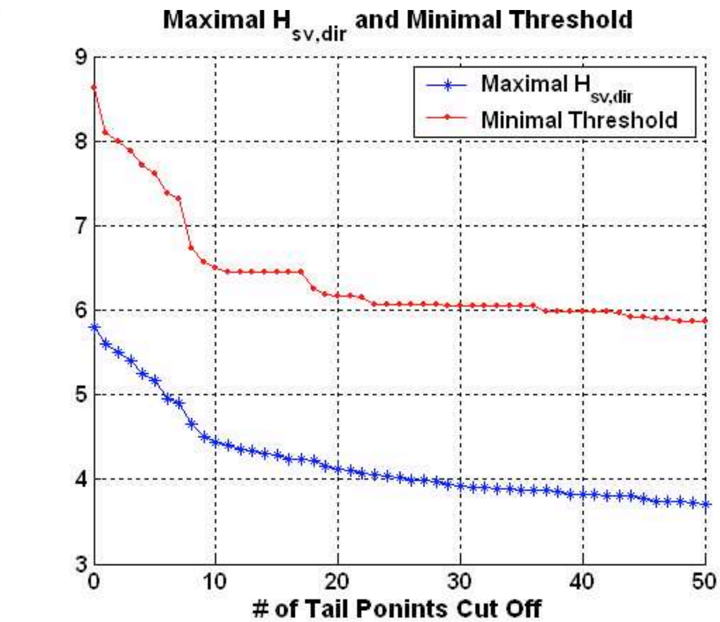

Figure 5-8-A

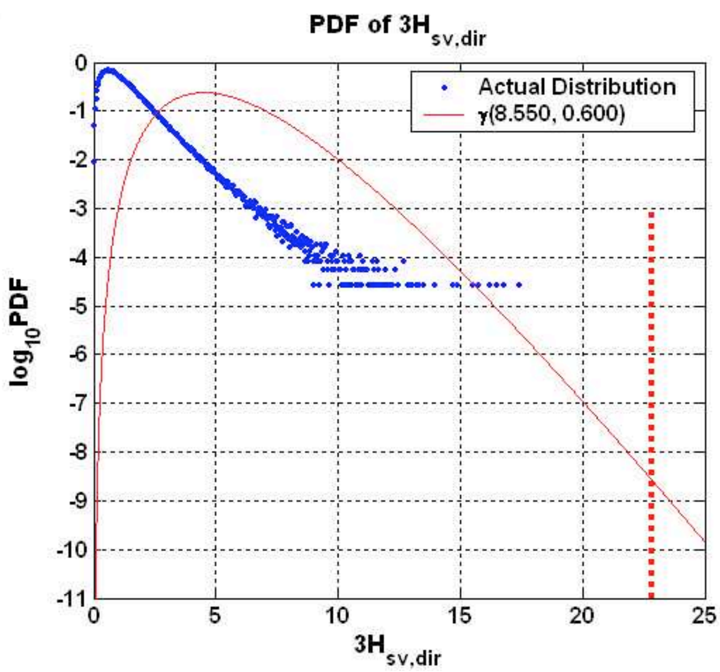

Figure 5-8-B

Figure 5-8 $H_{sv,dir}$ Threshold Determined with the Practical Overbounding Method

The same practical γ -distribution overbounding method developed in Section 5.3.2 is again applied here to determine the threshold of $H_{sv,dir}$. Figure 5-8-A shows that both the maximum $H_{sv,dir}$ and the minimum threshold dramatically decrease if the largest tail points are deleted. Again, the five largest points are removed, and the threshold for $H_{sv,dir}$ is $\frac{22.8564}{3} = 7.6188$ from the optimal γ (8.550, 0.600)-distribution. This γ -distribution, together with the threshold, is shown in Figure 5-8-B.

To compare the performance of $H_{sv,dir}$ with that of the other three monitors, an ionospheric gradient with a magnitude of 0.014 m/s at 50^o (or, equivalently, 0.0111 m/s vertically) is simulated as follows:

$$\begin{cases} \nabla\rho = +0.014 \cdot t \\ \nabla\phi = -0.014 \cdot t \end{cases} \tag{5-18}$$

and then it is injected into the measurements of PRN 7 with a duration of 173 seconds. The resulting failure test outputs are shown in Figure 5-9. The GMA divergence and innovation tests do not reach their thresholds before the gradient ends, as shown in Figures 5-9-A and 5-9-B, respectively. The ramp test statistic jumps to a high level but is still below its threshold, as shown in Figure 5-9-C. None of these three monitors detect this particular failure, except for an acceleration flag generated 175 seconds after the failure is injected. This acceleration flag is a “lucky” one, since random measurement noise plays a decisive role when the acceleration test statistic is close to its threshold. In contrast, the FST $H_{sv,dir}$ is able to detect the failure in only 15.5 seconds, as shown in Figure 5-9-D. It not only detects this small ionospheric gradient much more quickly, but there is also significant margin for us to raise the threshold if needed to protect continuity, as demonstrated in Figure 5-9-D.

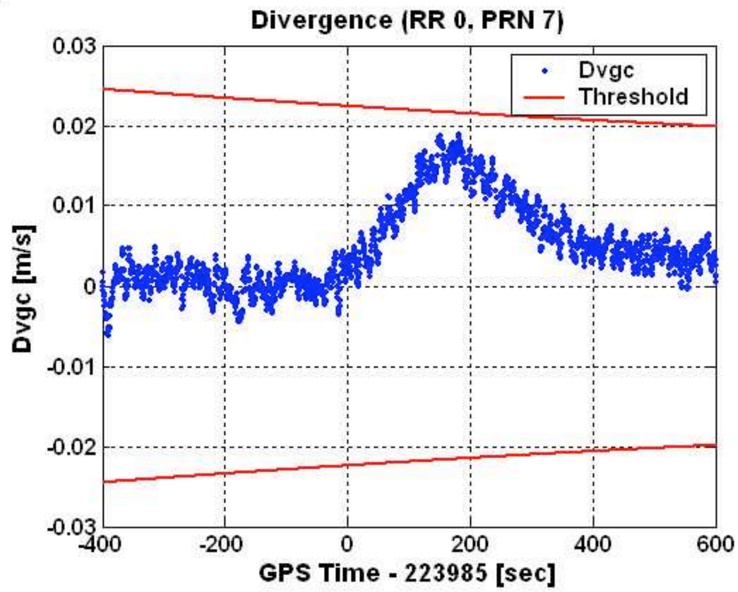

Figure 5-9-A

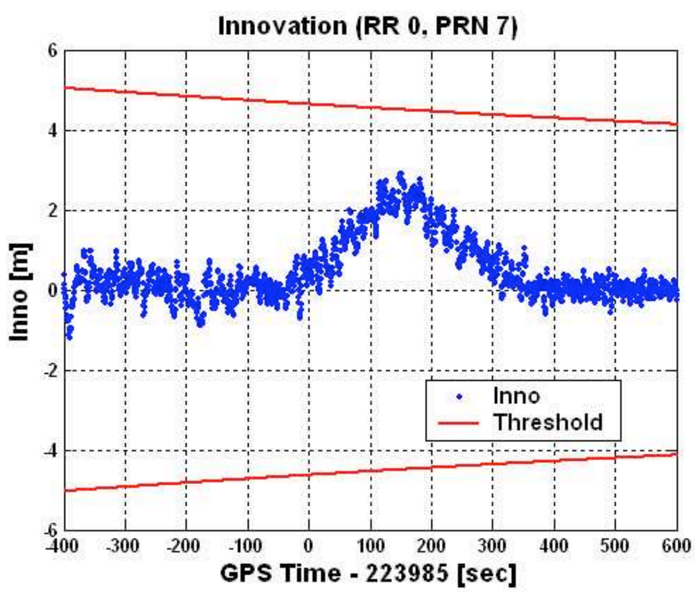

Figure 5-9-B

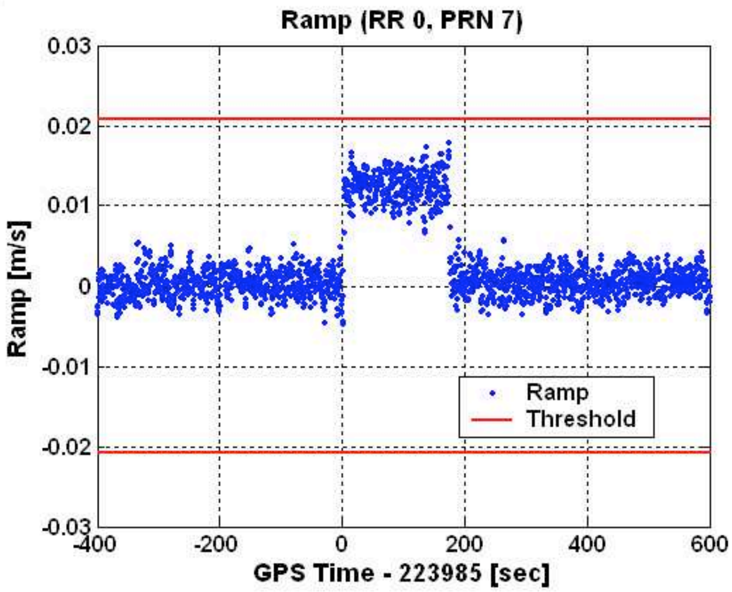

Figure 5-9-C

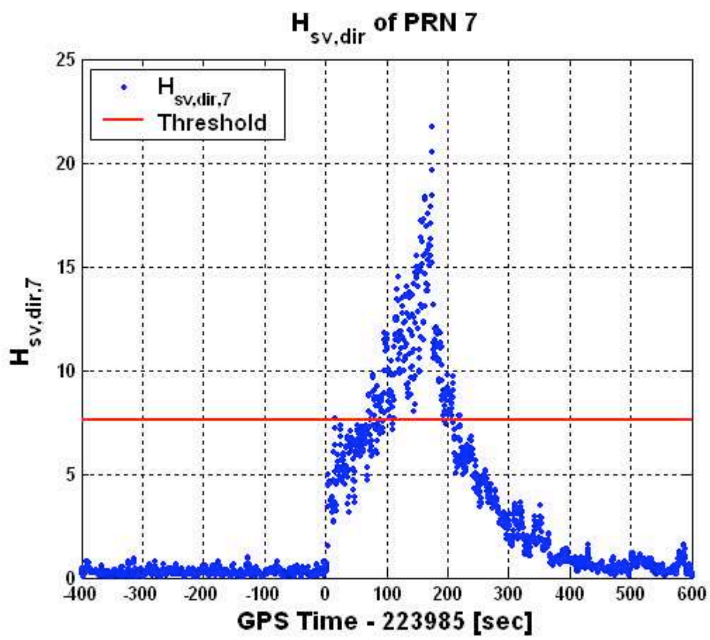

Figure 5-9-D

Figure 5-9 Failure Test Results of (A) GMA Divergence, (B) Innovation, (C) Ramp, and (D) $H_{sv,dir}$

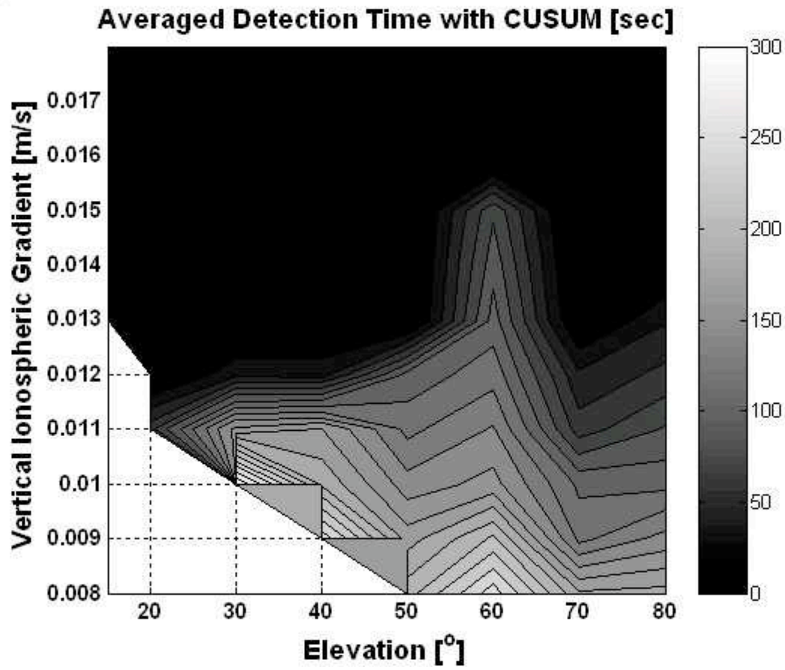

Figure 5-10-A

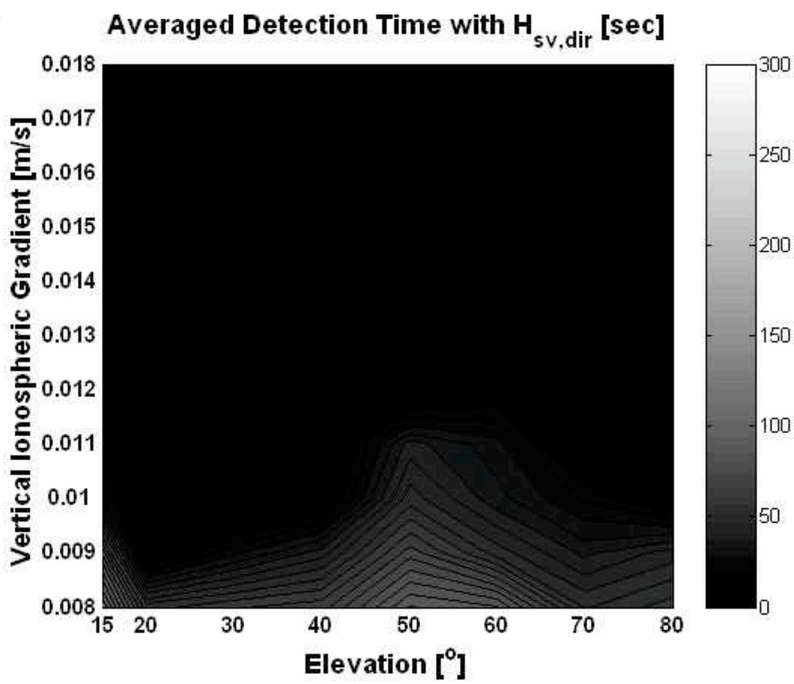

Figure 5-10-B

Figure 5-10 Averaged Detection Time by the IMT (A) without and (B) with the FST $H_{sv,dir}$

With this definition of $H_{sv,dir}$, our failure tests over the entire failure space of ionospheric gradients described in Chapter 4 can be repeated. Figure 5-10-A, copied from Figure 4-15-B, shows the averaged detection time after the divergence CUSUM method was integrated into the IMT. If the FST $H_{sv,dir}$ is also implemented in the IMT, then the ionosphere gradient detection performance is significantly improved, as shown in Figure 5-10-B. Note that these results do not necessarily mean that the divergence CUSUM method is not needed once $H_{sv,dir}$ is implemented. Most likely future LAAS systems will employ FSTs to "soften" EXM logic rather than as additional integrity monitor statistics. This will be discussed in Section 5.5.

5.5 Discussion and Conclusions

This chapter first demonstrated theoretically that an ideal FST, which is the sum of squared independent standard normal random variables, could detect failures more quickly than the component monitors. It then demonstrated that the satellite clock FST, $H_{sv,ars}$, and ionospheric gradient FST, $H_{sv,dir}$, could significantly increase the speed of detection of satellite clock failures and ionospheric gradients, respectively.

In fact, the potential advantages of FSTs can be explored further. First, if a search of all possible failure modes were conducted, one would find that FSTs detect some failure scenarios that are not detected by the existing individual integrity monitors. Second, more FSTs can be constructed, with each one designed to target a specific failure source, such as $H_{sv,ars}$ targeting satellite clock failures and $H_{sv,dir}$ targeting ionospheric gradients.

Meanwhile, FSTs can be constructed in various ways. An FST can consist of the same statistic from three channels to indicate the status of that statistic. Here is an example for the acceleration statistic:

$$H_{sv,acc} = \frac{1}{3}\sum_{i=0}^{2} x_{acc,i,n}^{2}(k) \tag{5-19}$$

An FST can also be composed of all statistics from a channel to detect channel failures such as:

$$H_{channel,m,n} = \frac{1}{N} \sum \left(x^2_{acc,m,n}(k) + x^2_{ramp,m,n}(k) + \ldots + x^2_{inno,m,n}(k) + x^2_{dvgc,m,n}(k) \right) \tag{5-20}$$

where a total of N test statistics are combined. Based on channel FST statistics, satellite and receiver failure-specific tests can be further constructed. For example,

$$H_{sv,n} = \frac{1}{3} \sum_{i=0}^{2} H_{channel,i,n} \tag{5-21}$$

and

$$H_{rr,m} = \frac{1}{N_m} \sum_{j} H_{channel,m,j} \tag{5-22}$$

are designed to detect satellite and receiver failures, respectively.

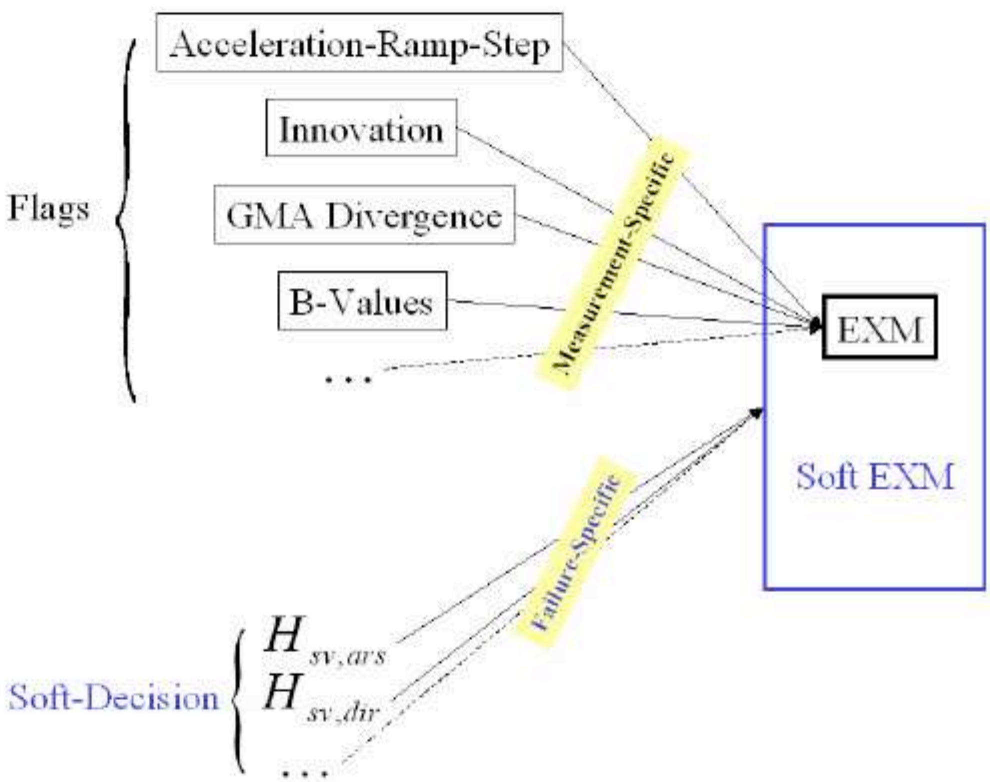

Figure 5-11 Soft EXM with Failure-Specific Tests

Finally and more importantly, FSTs can be used to "soften" the current EXM logic so that LAAS can meet tighter continuity requirements in the future, as pointed out in [70]. The current EXM excludes failed measurements based on flags generated by integrity monitors. If there are multiple flags on multiple reference receivers, then these multiple receivers will be excluded and LAAS corrections must then cease (and no further service can be provided until the system is manually reset). This severe consequence may result from the combination of an actual failure with one or more false alerts, since the chance that two different failures occur at the same time is very low. If only one failure occurs, most likely there will also be only one flag generated from one of the FSTs that targets this failure mode. Therefore, FSTs help EXM differentiate real flags from false alarms and the LGF can thus delay or prevent LAAS corrections from being mistakenly shut down.

In future work, more FST statistics, like $H_{sv,ars}$ and $H_{sv,dir}$, need to be constructed. One and only one of these FST statistics should respond to a failure that has a specific FST statistic associated with it. Figure 5-11 shows a potential structure of soft EXM with these FST statistics. Measurement-specific integrity monitors generate Boolean (0/1) outputs ("1" represents a flag), with one output for each channel. In order to identify real flags from the outputs, soft EXM can utilize FST flags, with one FST flag corresponding to one failure. A real channel flag should accompany one FST flag; otherwise, a channel flag without a corresponding FST flag can be regarded as a false alert. Thus continuity is protected. FSTs need Boolean outputs as well, since FSTs do not know which channel (or channels) a failure comes from. Soft EXM based on the measurement-specific and failure-specific flags provides a clearer picture of what has happened than do Boolean outputs. It will be easier and quicker to identify real failures and then to recover failed channels. Soft EXM logic also can be designed to handle all possible interactions between integrity monitor flags and FST statistics in the future.

Also, more nominal and failure testing is needed to confirm that the thresholds of $H_{sv,ars}$ and $H_{sv,dir}$ selected here are indeed conservative such that there is margin for threshold relaxation in a practical LGF implementation.

Chapter 6 Conclusions

6.1 Summary of Contributions

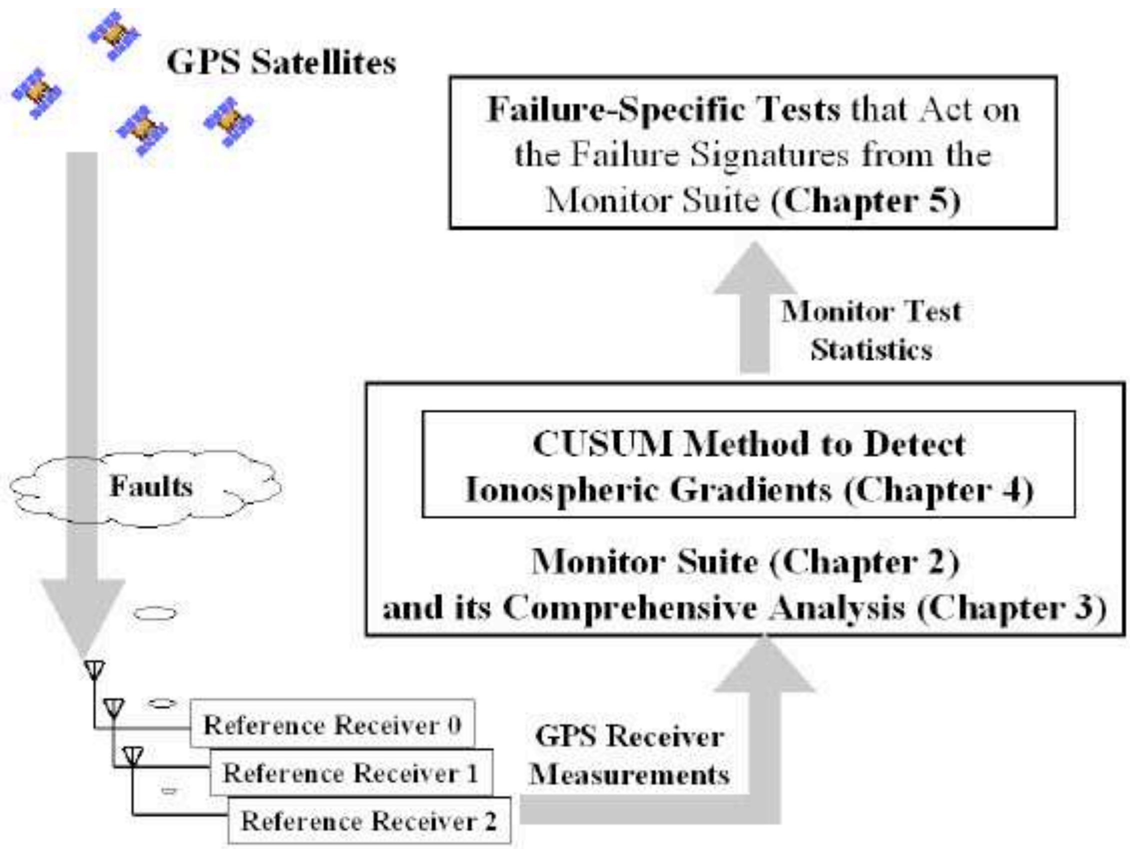

Figure 6-1 Relationship among Four Contributions

My contributions are summarized in this section. Figure 6-1 describes the relationship among these contributions. An existing set of LAAS integrity monitors are designed to detect a variety of possible failures. The first contribution, which was discussed in Chapter 3, is the first comprehensive analysis of this LAAS monitor suite. These analysis and failure tests uncover the limitations of the existing monitors in detecting small but hazardous ionospheric gradients. Thus the Cumulative Sum (CUSUM) method is applied to more quickly detect marginal ionospheric gradients. The CUSUM method is the second contribution in this thesis and was explored in Chapter 4. The failure effect analysis also provides valuable information on how to combine monitor test statistics to construct effective failure-specific test (FST) statistics. This third contribution, FST, was investigated in Chapter 5. All these contributions are based on my correlated work on the Integrity Monitor Testbed (IMT) design, development, and verification. IMT was explained in Chapter 2.

6.1.1 First Comprehensive Analysis of the Existing LAAS Monitor Suite

LAAS integrity monitors are designed to detect a collection of hazardous failures. This thesis creates a generic failure model, in which a step failure is injected into both or either the code and carrier measurements of a GPS receiver channel. This channel step failure model is simple, but it is very important. First, step or step-like failures do occur in practice. Second, any arbitrary error can be mathematically constructed as the sum of a sequence of delayed step functions. Third, satellite and receiver failures can be decomposed into failures on multiple channels. Linearity exists between failure inputs and responses for most integrity monitors. Therefore, superposition can be used to analyze other failure modes based on this generic failure model.

The thesis derives analytically the step responses of most of the LAAS integrity monitors using this step failure model. In this analysis, measurements from an individual channel are assumed not to interfere with those from other channels in the same GPS receiver. The analysis of ionospheric gradients assumes that all reference receivers are impacted with the same ionospheric delays in the measurements from one satellite [31].

The thesis then analyzes several other known fault modes based on the step failure model using the superposition method. These fault modes include channel code step failures, channel phase step failures, satellite clock step failures, satellite clock ramp failures, receiver clock step failures, and ionospheric gradients.

Table 6-1, which is a duplicate of Table 3-7, summarizes qualitatively the results of the failure analyses. It shows which integrity monitor can detect which failure modes. Here are the key findings:

Failure Modes	Channel (r, p)		Satellite p		Receiver r
	Code Meas.	Phase Meas.	Satellite Clock	Ionospheric Divergence	Receiver Clock
	$\nabla\phi_{r,p}=0$	$\nabla\rho_{r,p}=0$	$\nabla\phi_{m,p}=\nabla\rho_{m,p}$	$\nabla\phi_{m,p}=-\nabla\rho_{m,p}$	$\nabla\phi_{r,n}=\nabla\rho_{r,n}$
Acc, $Ramp$ and $Step$	×	√	√	√	×
$Inno$	√	√	×	√	×
$Dvgc$	√	√	×	√	×
B_ρ	√	√	×	×	×
B_ϕ	×	√	×	×	×
ρ_{corr}	√	√	√	√	×
ϕ_{corr}	×	√	√	√	×
$R_{\rho_{corr}}$	√	√	√	√	×

Table 6-1 Effects of Known Failure Modes on IMT Monitors

1. The acceleration-ramp-step test detects only carrier phase measurement failures. It does not respond to any code failure, as designed.
2. The carrier-smoothed code (CSC) innovation test never detects satellite clock failures.
3. The CSC innovation method is mathematically equivalent to the geometric moving averaging (GMA) code-carrier divergence method except for a different smoothing time constant and scaling factor.
4. The B-value test can detect channel failures; however, it cannot detect satellite clock failures and ionospheric gradients since they affect three reference receivers at the same time.

5. None of the IMT monitors detect receiver clock failures, and they should not, since in DGPS any receiver clock drifts are removed by the corrections.
6. One monitor can respond to several different failure modes, and several different monitors can detect the same type of failure.
7. The LAAS monitor suite detects the entire set of possible threats analyzed in this work

Importantly, we find that noise causes the alarm for ionospheric gradients to be later than we would like. This finding motivates my work on the CUSUM method to detect ionospheric gradients and my work on failure-specific tests. My failure analyses extend to evaluate and compare quantitatively the noise performance of the IMT integrity monitors and potential new monitors. Two examples of performance comparison based on the Minimum Detectable Errors (MDEs) are given for illustration in Section 3.9.2.

6.1.2 Cumulative Sum (CUSUM) Method to Detect Ionospheric Gradients

Severe ionospheric gradients are the most dangerous threats to LAAS. Potentially they could lead to vertical positioning errors as large as 19 meters for LAAS users if no integrity monitors are used to detect the gradients [32]. A major focus of this thesis is exploring methods to detect quickly anomalous ionospheric gradients subject to the required low probability of false alarm.

In this thesis, failure analyses and tests with (simulated) ionospheric gradient failures have been used to quantify the limitations of the existing monitors in detecting small but hazardous ionospheric gradients. This thesis applies the generic CUSUM quality-assurance method to the detection of unusual ionosphere divergence and develops the divergence CUSUM method that is able to more quickly detect marginal ionospheric gradients.

This thesis compares the performance of the CUSUM to the other relevant IMT integrity monitors and analyzes the minimum detectable ionospheric gradients achieved by these monitors. Figure 6-2 shows a set of failure tests conducted in Chapter 4. These failure tests demonstrate that the divergence CUSUM method is fastest in detecting the small ionospheric gradients that the acceleration-ramp-step test fails to detect. The CUSUM method improves upon the detection speed of the existing GMA method by about 30%. When combined in the IMT, the divergence CUSUM and acceleration-ramp-step monitors offer the lowest bounds on detection time. The divergence CUSUM method has achieved its design goal.

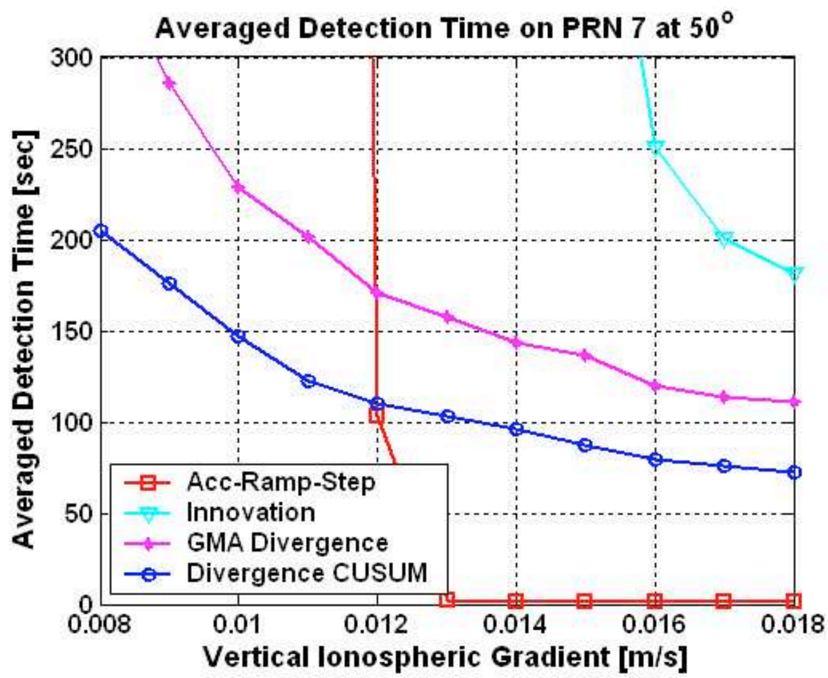

Figure 6-2 Failure Test Result by Injecting Different Magnitudes of Ionospheric Gradients at 50^o (All Monitors Considered Here Have the Same False Alarm Rate)

The CUSUM method speeds up detection of anomalies that take time to build up in smoothed pseudoranges before they threaten users. This improvement is significant in terms of reducing the risk posed by ionosphere spatial gradient anomalies. Calculations in [32] demonstrate that the worst vertical positioning error can be reduced from 19 meters to 5 meters when the CUSUM and acceleration-ramp-step methods are implemented. This method can and should be utilized in future precision approaches and landings.

6.1.3 Combining Monitors for Faster Detection of Specific Failures

Some failures excite multiple monitors, and these failures are best detected by combining the monitors. Monitor combinations detect certain failures more quickly than any individual monitor [70]. These combinations yield powerful failure-specific tests (FSTs), which are pioneered in this dissertation for LAAS.

These failure-specific test statistics are the sum of K squared independent standard normal variables, and ideally they have a χ_K^2-distribution. This thesis first gives a mathematical derivation to prove that the FST constructed in an ideal case is able to detect failures more quickly than individual monitors with the same probability of false alarm.

Unfortunately, the sum of squared normalized test statistics is not χ^2-distributed in practice. This thesis develops an overbounding method to determine the threshold for FSTs. It uses a $\gamma(\alpha, \beta)$-distribution to overbound the tail of the apparent distribution observed from nominal testing. The several largest tail points are excluded from the overbounding. The parameters α and β are determined by using a numerical search method in the range of $1 \le \alpha \le 10$ and $0.2 \le \beta \le 3$, so that not only can the selected optimal $\gamma(\alpha, \beta)$-distribution overbound the tail of the apparent distribution, but also the derived threshold is not overly conservative.

The failure analyses in Chapter 3 offer valuable information on how to construct effective FST statistics targeting specific failure modes. This thesis constructed two FST statistics and tested them with IMT data. The FST, $H_{sv,ars}$, which combines the acceleration, ramp, and step test statistics, detects satellite clock failures more quickly than the acceleration-ramp-step monitor. The FST, $H_{sv,dir}$, which combines the divergence, innovation, and ramp test statistics, significantly reduces the time required to detect ionospheric gradients, even compared to the CUSUM method.

In principle, FST statistics can be constructed to target each failure mode, each satellite, and each reference receiver, although only a subset of these are likely to significantly improve IMT detection times. As pointed out in [70], these fault-specific FSTs can soften the current EXM failure exclusion logic so that the LAAS can meet the tighter continuity requirements that apply to Category III precision landings.

6.2 Related Work on IMT Design, Development, and Verification

As described in detail in Chapter 2, the Integrity Monitor Testbed (IMT) is a prototype of the LAAS ground facility (LGF) developed at Stanford University. It is used to evaluate whether the LGF can meet the integrity requirements of the LAAS-supported precision approaches. The IMT consists of a variety of integrity monitors that are designed to detect all possible failures as well as Executive Monitoring (EXM), which includes a comprehensive failure-handling logic.

In this thesis, the IMT software development, including all integrity monitoring algorithms and the EXM logic, have been finished. During the IMT software development, many data structures and functions from previous Stanford LAAS flight-test prototypes have been redesigned. The details of the IMT software development are not covered in this thesis due to the complexity of the software but can be found in [87].

IMT software development has also improved the monitoring algorithms in several places. For example, the IMT implements the receiver lock time check (discussed in Section 2.6.1) in such a way that EXM will not reject measurements from a channel as hazardous to protect continuity even if this check fails. Instead, the IMT initializes and resets several memory buffers properly before using it, including re-initializing its carrier smoothing filter. In addition, the ten previous points of $\phi^*_{m,n}(k)$ in Equation (2-14) are now recomputed when the ephemeris of a satellite has just been validated by DQM to smoothly transition between old and new ephemerides. Also, the fault-determination logic shown in Figure 2-11 has been added inside the acceleration-ramp-step test algorithm to avoid possible false alarm (i.e., alarm generated on non-failed channels).

The IMT has become an indispensable tool for design, development, and testing of new integrity monitors such as the divergence CUSUM method and FST developed in this thesis. Both the σμ-monitor (discussed in Section 2.9) and position domain monitor (discussed in [27]) were also developed and tested using this tool.

Of all contributions, this part is not only the most time and energy consuming, but also the most rewarding. Deeply understanding the workings of the IMT is the key stepping stone to the three fundamental contributions described above.

6.3 Future Work

The following areas require further attention:

1. The CUSUM method can be applied to detect other kinds of failures because of its simplicity and excellent performance. For example, the CUSUM method can possibly be used to detect sudden and large mean and standard deviation changes in reported receiver carrier-to-noise ratio, C/N_0. These changes, probably due to shadowing and strong multipath fading in GPS signal propagation, can cause code error and result in integrity risk. Using the CUSUM method to provide a real-time posterior distribution of sigma based on the current CUSUM state has been investigated in [63].
2. Besides the failure-specific tests (FSTs) for satellite clock failures and ionospheric gradients (i.e., $H_{sv,ars}$ and $H_{sv,dir}$), more FST statistics in various forms can be constructed. For example, based on Figure 5-1, FST statistics could target code and carrier-phase measurement errors. FST statistics could target channel, satellite, and reference receiver failures by using Equations (5-20)-(5-22).
3. As discussed in Section 5.5, "soft" EXM logic can be designed with the help of failure-specific tests so that the LAAS can meet the tighter continuity requirements that apply to Category III precision landings.

Stanford's LAAS IMT can detect and exclude failures such that it meets the LGF integrity requirements for Category I landings [89,69]. Categories (CAT) II and III have much stricter requirements both on integrity and continuity than CAT I, as shown in Table 1-2. To meet the requirements for CAT II/III approaches, the existing LGF system requires additional system elements, improved algorithms, and justifications for relaxing over-conservative assumptions built into integrity assessment [69].

Category I LAAS is a single-frequency (L1 at 1575.42 MHz) differential GPS system. As a new GPS civil signal (L5 at 1176.45 MHz) will join the current civil signal, L1, over the next ten years [15], this frequency diversity will definitely be utilized to improve the performance of GPS itself and almost all GPS applications, including LAAS. L1 LAAS will eventually transition to L1/L5 LAAS. The three fundamental contributions developed in my thesis will become more important for this future L1/L5 LAAS.

First, the methodology used in the comprehensive analysis of the LAAS monitor suite is valid not only for L1-only LAAS but also for both L5-only and L1/L5 integrated LAAS systems. Moreover, a comprehensive analysis, similar to the one conducted in Chapter 3, for the L1/L5 LAAS monitor suite can evaluate and compare the performance of different monitors under each failure mode with each frequency. The analysis can then identify which monitor can detect which failure mode more effectively using which frequency. Therefore, with the help of this analysis, L1/L5 LAAS can employ the most effective monitors only. This efficiency should simplify the monitor architecture that might otherwise become very complicated due to the introduction of L5.

Second, the divergence CUSUM method will also retain its importance in the future. Although dual frequency measurements can be used to estimate the ionospheric delay in avionics [42], these airborne estimates will be no longer available if LAAS loses one frequency due to radio frequency interference. Therefore, using single-frequency LAAS, powered with the divergence CUSUM method as a backup, future LAAS can maintain a higher availability and continuity, and is able to smoothly transition from L1/L5 LAAS to a single-frequency LAAS.

Third, failure-specific tests (FSTs) will also retain their importance in a future L1/L5 LAAS. With additional monitor test statistics from L5 LAAS, it will be easier to construct more effective FST statistics.

Appendix A Keplerian Orbit Elements and the GPS Navigation Message

As described below, five Keplerian parameters can completely describe an ideal orbital plane and an ideal elliptical orbit. However, in the presence of perturbing forces on GPS satellites, their orbits need to be described by an expanded set of 16 quasi-Keplerian parameters in order to provide acceptable orbit accuracy to GPS users. These parameters, also called ephemeris parameters, are contained in the navigation message that is broadcast by a GPS satellite. The position and velocity vectors of a satellite can be calculated based on its ephemeris. This appendix gives a brief summary of Keplerian parameters and navigation messages that are fully explained in [42].

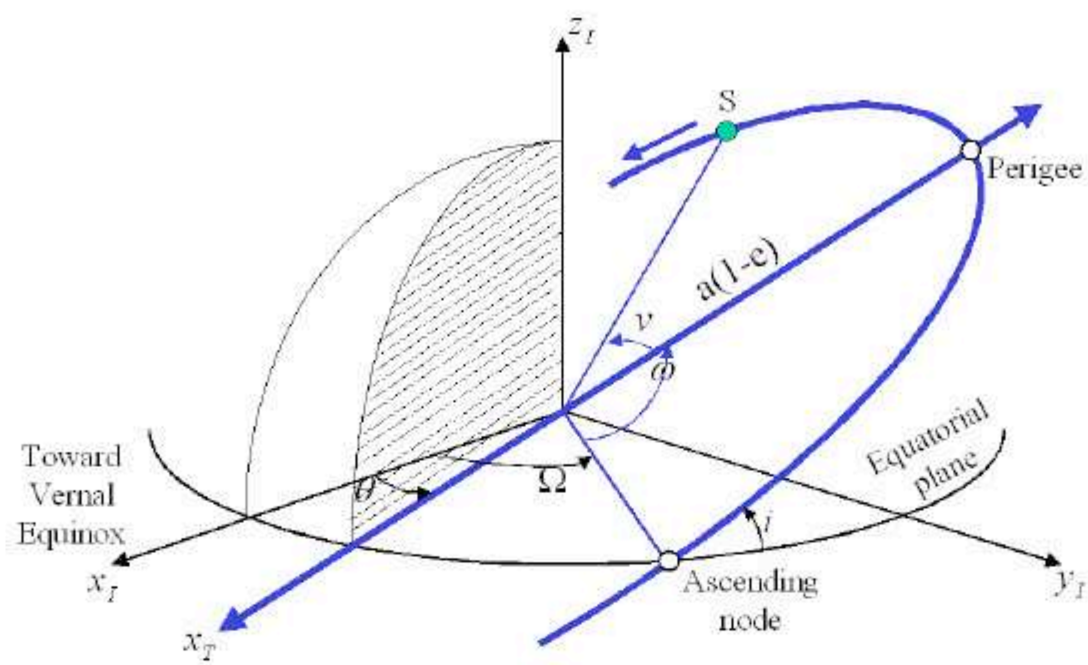

Figure A-1 Six Keplerian Elements [42]

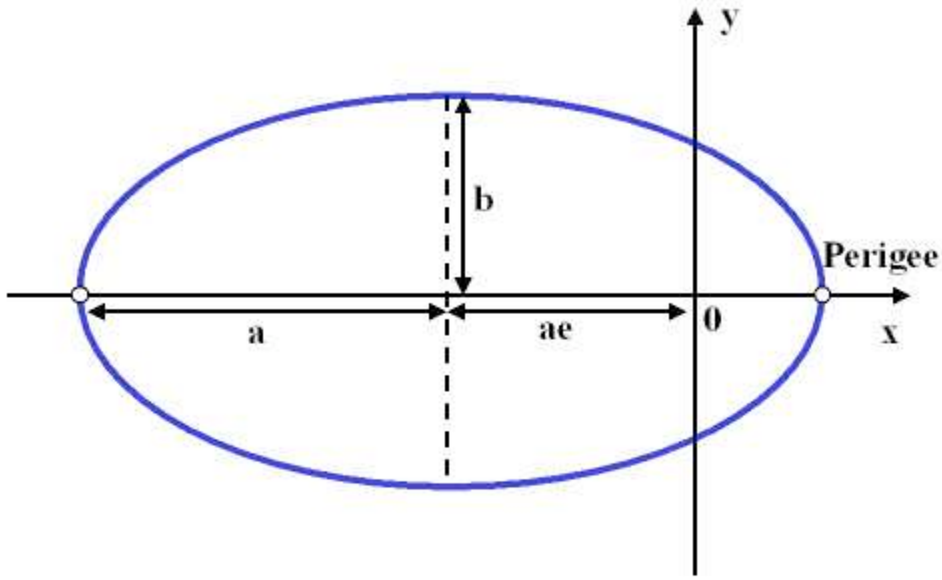

Figure A-2 Ellipse

A GPS satellite follows Kepler's three laws of planetary motion and Newton's laws of motion under the assumption that both the Earth and the satellite are spheres of uniform composition and that there are no other external forces beyond the gravitation of these two bodies. Under these idealized conditions, the motion of a GPS satellite, its position and velocity, can be determined by six Keplerian elements, i.e., $\{a, e, i, \Omega, \omega$, and $v\}$, as shown in Figure A-1 [3,42]. x_I, y_I, and z_I are three axes of an inertial and terrestrial Cartesian coordinate system. These orbit elements are defined as follows:

1. Semi-major axis, a: The motion of a GPS satellite is an elliptical orbit with the Earth at one of the foci. a is the semi-major axis length of the ellipse. Figure A-2 shows the diagram of an ellipse.
2. Eccentricity, e: If c is the distance from the center of the ellipse to a focus, then the eccentricity e is defined by:

$$e = \frac{c}{a} \tag{A-1}$$

Note that $e = 0$ for a circular orbit. With a and c, the semi-minor axis length, b, is determined by the following formula:

$$b = \sqrt{a^2 - c^2} \tag{A-2}$$

These two parameters a and e together characterize the size and shape of the orbital ellipse in two dimensions.

3. Inclination, i: This is defined as the angle between the orbital plane of the satellite and Earth's equatorial plane.
4. Right ascension of the ascending node (RAAN), Ω: The vernal equinox is the direction determined by the intersection of the equatorial plane of the Earth with the plane of the Earth's orbit around the sun, and the ascending node is the point on the satellite's orbit that moves in the northerly direction and intersects with the equatorial plane. The parameter, Ω, is the angle between the direction pointing to the vernal equinox and the ascending node measured in the Earth's equatorial plane. The angles i and Ω together specify the orientation of the orbital plane of the GPS satellite relative to the inertial and terrestrial reference frame.
5. Argument of perigee, ω: Perigee is the point in the satellite orbit where the satellite is closest to the center of the Earth. The parameter, ω, is the angle between the ascending node and the perigee measured in the orbital plane. It determines the orientation of the ellipse in the orbital plane.
6. True anomaly, v: After the orbital ellipse has been specified, v is used to locate the satellite in the orbit at a given instant. The parameter $v(t)$ is the angle between the perigee and the satellite position (at the given time t) measured in the orbital plane.

The first five Keplerian parameters {a, e, i, Ω, and ω} specify the ideal elliptical orbit [42]. Given v at an epoch, the satellite position can be easily calculated in an orbital coordinate system and then transformed into the WGS 84 Earth-Centered, Earth-Fixed (ECEF) coordinate system. The satellite velocity vector can also be computed based on Kepler's laws.

However, these five Keplerian parameters are not enough to accurately describe the actual motion of a GPS satellite, since the Earth is not a uniform sphere, and there are other forces besides the central gravitational force at play. The main forces perturbing the motion of a GPS satellite include the gravitational effects of the equatorial "bulge," lunar and solar gravity effects, and solar radiation pressure [42].

In the presence of the perturbing forces, the orbit of a GPS satellite can still be regarded as Keplerian but with an expanded orbital parameter set in order to account for these perturbations. The expanded set of quasi-Keplerian parameters consists of 15 elements. Together with an identifier parameter, called Issue of Data–Ephemeris (IODE), there are 16 parameters in each ephemeris broadcast by a GPS satellite. They are [42,83]:

1.	t_{0e}	ephemeris reference time
2.	$\sqrt{a}$	square root of the semi-major axis
3.	e	eccentricity
4.	i_0	inclination angle at the reference time
5.	Ω_0	longitude of the ascending node at the beginning of the GPS week
6.	ω	argument of perigee
7.	M_0	mean anomaly at the reference time
8.	Δn	correction to the computed mean motion
9.	$\dot{i}$	rate of change of inclination with time
10.	$\dot{\Omega}$	rate of change of RAAN with time
11-12.	C_{uc}, C_{us}	amplitudes of harmonic correction terms for the computed argument of latitude
13-14.	C_{rc}, C_{rs}	amplitudes of harmonic correction terms for the computed orbit radius
15-16.	C_{ic}, C_{is}	amplitudes of harmonic correction terms for the computed inclination angle

These orbital parameters are calculated by the Master Control Station (located in Colorado Springs, Colorado) based on code and carrier phase measurements at six monitor stations around the world. The orbital data are uploaded to the satellites typically once a day, and the ephemeris parameters broadcast by a satellite change every two hours [42].

Each satellite broadcasts its own ephemeris. Meanwhile, each satellite also transmits an almanac in its navigation message. The almanac is a subset of the clock and ephemeris data that covers all satellites in the GPS constellation but with reduced precision. The almanac enables a receiver to determine approximately when a satellite will rise above the horizon if an approximate user position is given. This assists in user receiver signal acquisition [42].

In addition to the ephemeris and almanac, a GPS satellite also broadcasts other information in its navigation message. The navigation data message is transmitted at 50 bits per second and so each bit is 20 ms long. It takes a total of 12.5 minutes to transmit the complete navigation message that is composed of 25 frames. Each frame is 1500 bits or, equivalently, 30 seconds long. A frame is further formatted into five subframes. Each subframe is six seconds long and contains ten 30-bit words. Figure A-3 diagrams the organization of a GPS navigation message [42]. Typically, subframes 1-3 repeat the same information from frame to frame. Subframes 4-5 of the consecutive frames contain different "pages" of the navigation message. The summary of the subframe information content is as follows [42,83]:

Subframe 1:	satellite clock corrections, health indicators, age of data
Subframe 2-3:	satellite ephemeris parameters
Subframe 4:	ionosphere model parameters, UTC data, almanac and health status data for satellites numbered 25 and higher
Subframe 5:	almanac and health status data for satellites numbered 1-24

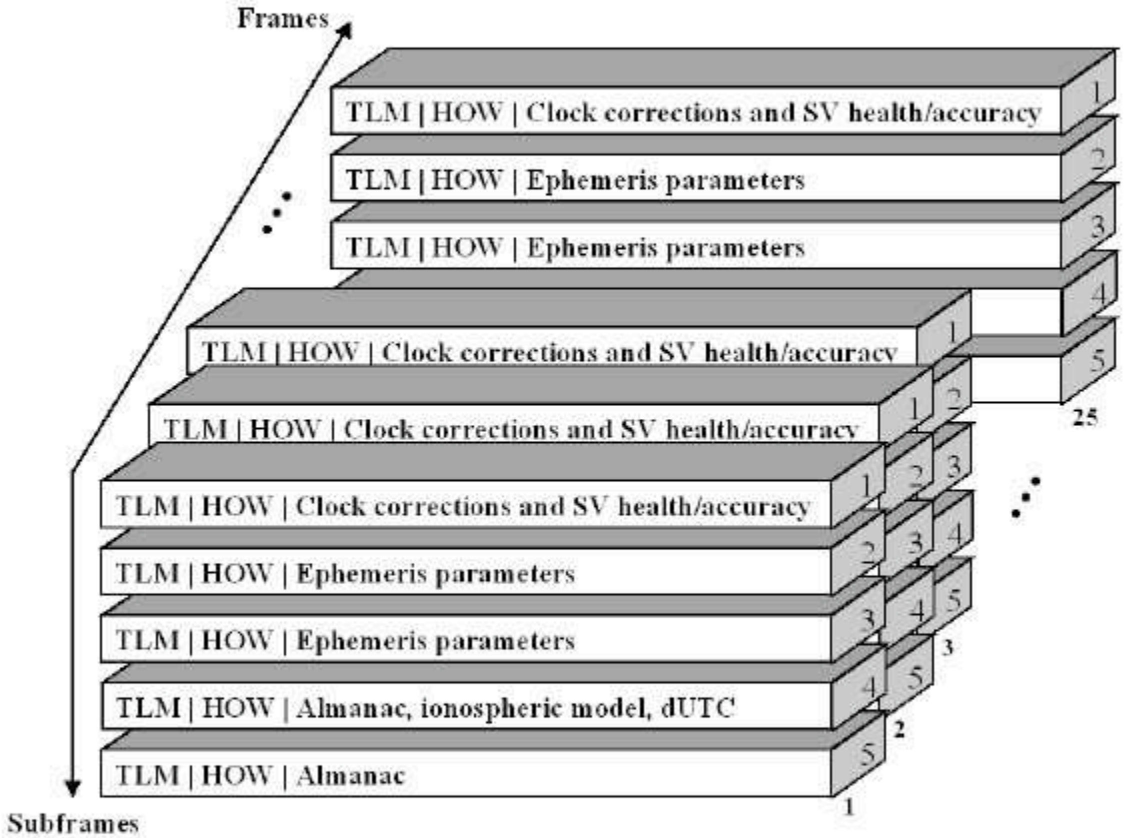

Figure A-3 GPS Navigation Message Organization [42,84]

The first word of each subframe is the Telemetry (TLM) that contains a fixed 8-bit synchronization pattern and a 14-bit message. The second word is the Hand-over word (HOW) that provides time information in order for a receiver to acquire the P(Y)-code segment [42].

Subframe 4 provides the time difference between the GPS time and the Coordinated Universal Time (UTC) in whole seconds and the rate and time difference estimate between them modulo one second. This allows a receiver to calculate an accurate estimate of UTC from GPS.

Bibliography

[1] D. Akos, R.E. Phelts, S. Pullen, P. Enge, "GPS Signal Quality Monitoring: Test Results," *Proceedings of the ION 2000 National Technical Meeting*. Anaheim, CA., Jan. 26-28, 2000, pp. 536-541.

[2] M. Basseville, I. Nikiforov, *Detection of Abrupt Changes – Theory and Application*. Prentice-Hall, ISBN 0-13-126780-9, 1993.

[3] R. Bate, D. Mueller, J. White, *Fundamentals of Astrodynamics*. Dover Publications, 1971.

[4] G. Beutler, "GPS Satellite Orbits," *GPS for Geodesy*, A. Kleusberg, and P.J.G. Teunissen (Eds.), Springer, Lecture Notes on Earth Sciences, pp. 37-101, 1996.

[5] M. Braasch, A.J. Van Dierendonck, "GPS Receiver Architectures and Measurements," *Proceedings of the IEEE*, Vol. 87, No. 1, Jan. 1999.

[6] R. Braff, "Description of the FAA's Local Area Augmentation System (LAAS)," *Journal of the Institute of Navigation*. Vol. 44, No. 4, Winter 1997-1998, pp. 411-424.

[7] R. Braff, C. Shively, "An Overbound Concept for Pseudorange Error from the LAAS Ground Facility," *Proceedings of IAIN World Congress/ION 56th Annual Meeting*. San Diego, CA., June 26-28, 2000.

[8] G. Casella, R. Berger, *Statistical Inference*. Pacific Grove, CA.: Wadsworth & Brooks/Cole Statistics/Probability Series, 1990.

[9] J. Christie, P.Y. Ko, A. Hansen, D.Dai, S. Pullen, B. Pervan, B. Parkinson, "The Effects of Local Ionospheric Decorrelation on LAAS: Theory and Experimental Results," ION NTM, San Diego, CA, Jan. 1999.

[10] C. Cohen, B. Parkinson, "Integer Ambiguity Resolution of the GPS Carrier for Spacecraft Attitude Determination," *Proceedings of the 15th Annual AAS Guidance and Control Conference*, Keystone, CO., Feb. 8-12, 1992, pp. 107-118.

[11] D. Cox, *Introduction to Wireless Personal Communications*, EE276 Course Reader, Stanford University, 2003.

[12] S. Datta-Barua, T. Walter, S. Pullen, M. Luo, J. Blanch, P. Enge, "Using WAAS Ionospheric Data to Estimate LAAS Short Baseline Gradients," Proceedings of ION 2002 National Technical Meeting, Anaheim, CA, Jan. 28-30, 2002, pp. 523-530.

[13] T. Dehel, "Ionospheric Wall Observations," Atlantic City, N.J., William J. Hughes FAA Technical Center, FAA ACT-360, Feb. 24, 2003.

[14] P. DeLeon, C. Screenan, "An Adaptive Predictor for Media Playout Buffering," Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. 6, March 1999, pp. 3097-3100.

[15] P. Enge, "GPS Modernization: Capabilities of the New Civil Signals," *Australian International Aerospace Congress*, Brisbane, Australia, July 29-Aug. 1, 2003.

[16] P. Enge, "Local Area Augmentation of GPS for the Precision Approach of Aircraft," Proceedings of the IEEE, Vol. 87, No. 1, January 1999.

[17] P. Enge, E. Phelts, D, Akos, "Detecting Anomalous Signals from GPS Satellites," Stanford University, Draft Manuscript, Oct. 12, 1999.

[18] W. Feess, S. Stephens, "Evaluation of GPS Ionospheric Model," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-23, No. 3, pp. 332-338, 1987.

[19] *GPS Risk Assessment Study: Final Report*. Johns Hopkins University Applied Physics Laboratory, VS-99-007, January 1999.

[20] F. van Graas, "Detection of Satellite Low Signal Power," Ohio University, Revised Draft, April 30, 2001.

[21] D.M. Hawkins, D.H. Olwell, *Cumulative Sum Charts and Charting for Quality Improvement*. New York: Springer-Verlag, Inc., 1998.

[22] J. Jung, "High Integrity Carrier Phase Navigation for Future LAAS Using Multiple Civilian GPS Signals," *ION GPS 1999*, Nashville, TN., Sept. 14-17, 1999.

[23] E. Kaplan, *Understanding GPS: Principles and Applications*. Norwood, MA.: Artech House, Inc., 1996.

[24] R. Key, et.al., "LAAS Verification Methodologies and Cases," FAA AOS-240, Oklahoma City, OK., Unpublished Draft, May 20, 2000.

[25] D. Klein, B. Parkinson, "The Use of Pseudo-Satellites for Improving GPS Performance," *Journal of the Institute of Navigation*, Vol. 31, No., 4, Winter 1984-1985.

[26] LAAS KTA Group, *FAA LAAS Ground Facility (LGF) Functions*, Version 2.4, Unpublished Manuscript, Sept. 9, 1998.

[27] J. Lee, S. Pullen, G. Xie, P. Enge, "LAAS Position Domain Monitor Analysis and Failure-Test Verification," *AIAA 21st International Communications Satellite Systems Conference*, Apr. 15-19, 2003, Yokohama, Japan.

[28] J. Lee, S. Pullen, G. Xie, P. Enge, "LAAS Sigma Monitor Analysis and Failure-Test Verification," *Proceedings of the ION 57th Annual Meeting*, Albuquerque, NM., June 11-13, 2001, pp. 694-704.

[29] E. G. Lightsey, J. Grassidis, F. Markley, "Fast Integer Ambiguity Resolution for GPS Attitude Determination," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR, Aug. 9-11, 1999.

[30] M. Luo, S. Pullen, "LAAS Reference Receiver Correlation Analysis," Stanford University, Unpublished Manuscript, Feb. 17, 1999.

[31] M. Luo, S. Pullen, D. Akos, G. Xie, S. Datta-Barua, T. Walter, P. Enge, "Assessment of Ionospheric Impact on LAAS Using WAAS Supertruth Data," *Proceedings of The ION 58th Annual Meeting*, Albuquerque, NM., June 24-26, 2002, pp. 175-186.

[32] M. Luo, S. Pullen, J. Dennis, H. Konno, G. Xie, T. Walter, P. Enge, S. Datta-Barua, T. Dehel, "LAAS Ionosphere Spatial Gradient Threat Model and Impact of LGF and Airborne Monitoring," ION GPS/GNSS 2003, Sept. 9-12, 2003, Portland, OR.

[33] M. Luo, S. Pullen, J. Zhang, S. Gleason, G. Xie, J. Yang, D. Akos, P. Enge, B. Pervan, "Development and Testing of the Stanford LAAS Ground Facility Prototype," *Proceedings of the ION 2000 National Technical Meeting*. Anaheim, CA., Jan. 26-28, 2000, pp. 210-219.

[34] M. Luo, G. Xie, D. Akos, S. Pullen, P. Enge, "Radio Frequency Interference Validation Testing for LAAS Using the Stanford Integrity Monitor Testbed," *2003 ION National Technical Meeting*, Jan. 22-24, 2003, Anaheim, CA.

[35] *Matlab Statistics Toolbox User's Guide*, Version 4.0 (R13), The MathWorks, Inc., June 20, 2002.

[36] S. Matsumoto, S. Pullen, M. Rotkowitz, B. Pervan, "GPS Ephemeris Verification for Local Area Augmentation System (LAAS) Ground Stations," *Proceedings of ION GPS 1999*. Nashville, TN., Sept. 14-17, 2000, pp. 691-704.

[37] G. McGraw, T. Murphy, et.al., "Development of the LAAS Accuracy Models," Proceedings of ION GPS 2000, Salt Lake City, UT., Sept. 19-22, 2001, pp. 1212-1223.

[38] P. Meyer, *Introductory Probability and Statistical Applications*, Addison-Wesley, 1970, ISBN 0-201-04710-1.

[39] *Minimum Aviation System Performance Standards for Local Area Augmentation System (LAAS)*. Washington, D.C., RTCA SC-159, WG-4A, DO-245, Sept. 28. 1998.

[40] *Minimum Operational Performance Standards for GPS/Local Area Augmentation System Airborne Equipment*. Washington, D.C.: RTCA SC-159, WG-4A, DO-253A, Nov. 28, 2001.

[41] *Minimum Operational Performance Standards for GPS/Wide Area Augmentation System Airborne Equipment*. Washington, D.C.: RTCA SC-159, WG-2, DO-229C, Nov. 28, 2001.

[42] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements, and Performance*. Lincoln, MA.: Ganga-Jamuna Press, 2001.

[43] A. Mitelman, R.E. Phelts, D. Akos, S. Pullen, P. Enge, "A Real-Time Signal Quality Monitor for GPS Augmentation Systems," *Proceedings of ION GPS 2000*. Salt Lake City, UT., Sept. 19-22, 2000, pp. 862-871.

[44] T. Murphy, "Development of Signal in Space Performance Requirements for GBAS to Support CAT II/III Landing Operations," *Proceedings of ION GPS 2002, Portland*, OR., Sept. 24-27, 2002.

[45] H. Nahavandchi, "Temporal Characteristics of Multipath Effects in GPS-Code and Carrier Phase Observations," Norwegian University of Science and Technology, Division of Geomatics, June 11, 2003.

[46] Navstar GPS Joint Program Office, *Global Positioning System Standard Positioning Service Signal Specification*, 2nd Edition, Jun. 2, 1995.

[47] Navstar GPS Joint Program Office, *Interface Control Document (ICD)-GPS-200*. Internet URL: http://www.navcen.uscg.gov/pubs/gps/icd200/default.htm

[48] P.L. Normark, D. Akos, S. Pullen, G. Xie, J. Lee, M. Luo, P. Enge, B. Pervan, "Integration of the Multipath Limiting Antenna with the LAAS Integrity Monitor Testbed," *ION GPS 2002*, Sept. 24-27, Portland, OR.

[49] P.L. Normark, G. Xie, D. Akos, S. Pullen, M. Luo, J. Lee, P. Enge, B. Pervan, "The Next Generation Integrity Monitor Testbed (IMT) for Ground System Development and Validation Testing," *ION GPS 2001*, Sept. 11-14, Salt Lake City, UT.

[50] E.S. Page, "Continuous Inspection Schemes," *Biometrika*, Vol. 41, pp. 100-115, 1954.

[51] B. Parkinson, K. Fitzgibbon, "Optimal Locations of Pseudolites for Differential GPS," *Journal of the Institute of Navigation*, Vol. 33, No. 4, Winter 1986-1987.

[52] B. Parkinson, J. Spilker, P. Axelrad, P. Enge eds., *Global Positioning System: Theory and Applications*, Washington, D.C.: American Institute of Aeronautics and Astronautics, 1996.

[53] B. Pervan, "A Review of LGF Code-Carrier Divergence Issues," Illinois Institute of Technology, MMAE Dept., Draft Revision 2, May 29, 2001.

[54] B. Pervan, F. Chang, "Detecting Global Positioning Satellite Orbit Errors Using Short-Baseline Carrier Phase Measurements," *AIAA Journal of Guidance, Control, and Dynamics*. Vol. 25, No. 6, Nov.-Dec. 2002.

[55] B. Pervan, S. Pullen, J. Andreacchi, P. Enge, "Characterizing the Effects of Ionospheric Divergence and Decorrelation on LAAS," Proceedings of ION GPS 2000. Salt Lake City, UT., Sept. 19-22, 2000, pp. 653-661.

[56] B. Pervan, S. Pullen, J. Christie, "A Multiple Hypothesis Approach to Satellite Navigation Integrity," *Journal of the Institute of Navigation*. Vol. 45, No. 1, Spring 1998, pp. 61-71.

[57] B. Pervan, S. Pullen, M. Luo, "Navigation Integrity Monitoring for DGPS-Based Precision Landing of Aircraft," *Proceedings of the 1999 American Control Conference*. San Diego, CA., June 2-4, 1999.

[58] B. Pervan, I. Sayim, "Issues and Results Concerning the LAAS σ_{pr_gnd} Overbound," *IEEE PLANS 2000*, San Diego, CA., Mar. 13-16, 2000.

[59] B. Pervan, I. Sayim, S. Pullen, "Sigma Estimation, Inflation, and Monitoring in the LAAS Ground System," *Proceedings of ION GPS 2000*. Salt Lake City, UT., Sept. 19-22, 2000, pp. 1234-1244.

[60] R.E. Phelts, D. Akos, P. Enge, "Robust Signal Quality Monitoring and Detection of Evil Waveforms," *Proceedings of ION GPS 2000*, Salt Lake City, UT., Sept. 19-22, 2000, pp. 1180-1190.

[61] R.E. Phelts, D. Akos, P. Enge, "SQM Validation Report," ICAO GNSSP WG-B Meeting, Seattle, WA., May 30, 2000.

[62] R.E. Phelts, A. Mitelman, S. Pullen, D. Akos, P. Enge, "Transient Performance Analysis of a Multicorrelator Signal Quality Monitor," *Proceedings of ION GPS 2001*, Salt Lake City, UT., Sept. 11-14, 2001, pp. 1700-1710.

[63] S. Pullen, J. Lee, G. Xie, P. Enge, "CUSUM-Based Real-Time Risk Metrics for Augmented GPS and GNSS," ION GPS/GNSS 2003, Sept. 9-12, 2003, Portland, OR.

[64] S. Pullen, M. Luo, S. Gleason, G. Xie, D. Akos, P. Enge, B. Pervan, "GBAS Validation Methodology and Test Results from the Stanford LAAS Integrity Monitor Testbed," *Proceedings of ION GPS 2000*, Salt Lake City, UT., Sept. 19-22, 2000, pp. 1191-1201.

[65] S. Pullen, M. Luo, B. Pervan, F.C. Chang, "Ephemeris Protection Level Equations and Monitor Algorithms for GBAS," *Proceedings of ION GPS 2001*. Salt Lake City, UT., Sept. 11-14, 2001, pp. 1738-1749.

[66] S. Pullen, M. Luo, T. Walter, P. Enge, B. Pervan, "The Use of CUSUMs to Validate Protection Level Overbounds for Ground-Based and Space-Based Augmentation Systems," *Proceedings of ISPA 2000*. Munich, Germany, July 18-20, 2000.

[67] S. Pullen, M. Luo, G. Xie, J. Lee, R.E. Phelts, D. Akos, G. Elkaim, P. Enge, "LAAS Ground Facility Design Improvements to Meet Proposed Requirements for Category II/III Operations," *ION GPS 2002*, Portland, OR., Sept. 24-27, 2002.

[68] S. Pullen T. Walter, "A Quick Look at Last Weekend's SVN/PRN 22 Clock Anomaly," Stanford University, Unpublished Presentation, Aug. 3, 2001.

[69] S. Pullen, T. Walter, P. Enge, "System Overview, Recent Developments, and Future Outlook for WAAS and LAAS," *Tokyo University of Mercantile Marine GPS Symposium*, Tokyo, Japan, Nov. 11-13, 2002.

[70] S. Pullen, G. Xie, P. Enge, "Soft Failure Diagnosis and Exclusion for GBAS Ground Facilities," *Royal Institute of Navigation: NAV 01 Conference*, Nov. 6-8, 2001, Westminster, London, England.

[71] S. Ross, *A First Course in Probability*, Prentice Hall, 1998. ISNB 0-13-746314-6. QA273.R83.

[72] C. Shively, "Derivation of Acceptable Error Limits for Satellite Signal Faults in LAAS," *Proceedings of ION GPS 1999.* Nashville, TN., Sept. 14-17, 1999, pp. 761-770.

[73] C. Shively, "LAAS CAT III Potential Availability Benefit from Dual Frequency (Preliminary Results)," Revision 1. McLean, VA., MITRE/CAASD, May 22, 2002.

[74] C. Shively, R. Braff, "An Overbound Concept for Pseudorange Error from the LAAS Ground Facility," *ION Annual Meeting*, San Diego, CA., June 26-28, 2000.

[75] T. Skidmore, F. van Graas, "DGPS Ground Station Integrity Monitoring," *ION National Technical Meeting*, San Diego, CA., Jan. 24-26, 1994.

[76] D. Stratton, "Development and Testing of GNSS-Based Landing System Avionics," *Proceedings of IEEE PLANS 2000.* San Diego, CA., March 13-16, 2000, pp. 90-97.

[77] U.S. Department of Defense, *Global Positioning System (GPS) Standard Positioning Service (SPS) Performance Standard*, October, 2001

[78] U.S. Federal Aviation Administration, *Local Area Augmentation System (LAAS) Status Report*. Washington, D.C., Dec. 1, 2003. Internet URL:

http://gps.faa.gov/Library/Data/LAAS/Congressional121603.doc

[79] U.S. Federal Aviation Administration, *Programs – Local Area Augmentation System*. Internet URL: http://gps.faa.gov/Programs/laas/howitworks-text.htm

[80] U.S. Federal Aviation Administration, *Specification: Category I Local Area Augmentation System Non-Federal Ground Facility*. Washington, D.C., FAA/AND710-2937, May 31, 2001.

[81] U.S. Federal Aviation Administration, *Specification: Performance Type One Local Area Augmentation System Ground Facility*. Washington, D.C., FAA-E-2937A, April 17, 2002. Internet URL: http://gps.faa.gov/Library/Data/LAAS/LGF2937A.PDF

[82] A.J. Van Dierendonck, D. Akos, S. Pullen, R.E. Phelts, P. Enge, "Practical Implementation Considerations in the Detection of GPS Satellite Signal Failures," *Proceedings of IAIN World Congress/ION 56th Annual Meeting*. San Diego, CA., June 26-28, 2000.

[83] A.J. Van Dierendonck, S. Russel, E. Kopitzke, M. Birnbaum, "The GPS Navigation Message," *Global Positioning System*, Vol. I, The Institute of Navigation, pp. 53-73, 1980.

[84] F. van Diggelen, "GPS and GPS+GLONASS RTK," *Proc. ION GPS-97*, pp. 139-144.

[85] B. Widrow, S. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, N.J.: Prentice-Hall, 2001.

[86] G. Xie, "An Analysis of Code-Carrier Divergence Test Performance with Different Time Constants," Stanford University, Unpublished Presentation, Nov. 23, 2003.

[87] G. Xie, "Integrity Monitor Testbed and its Software Development," Stanford University, Unpublished Manuscript, March 16, 2004.

[88] G. Xie, S. Pullen, M. Luo, P. Enge, "Detecting Ionospheric Gradients with the Cumulative Sum (CUSUM) Method," *AIAA 21st International Communications Satellite Systems Conference*, Apr. 15-19, 2003, Yokohama, Japan.

[89] G. Xie, S. Pullen, M. Luo, P.L. Normark, D. Akos, J. Lee, P. Enge, B. Pervan, "Integrity Design and Updated Test Results for the Stanford LAAS Integrity

Monitor Testbed," *Proceedings of the ION 57th Annual Meeting*, Albuquerque, NM., June 11-13, 2001, pp. 681-693.