

The Pennsylvania State University

The Graduate School

College of Engineering

**DETERMINING THE UNCERTAINTY OF A GPS-BASED
COLLISION VEHICLE DETECTION SYSTEM**

A Thesis in

Mechanical Engineering

by

Sanket R. Amin

© 2011 Sanket R. Amin

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Masters of Science

August 2011

The thesis of Sanket Amin was reviewed and approved* by the following:

Sean N. Brennan
Associate Professor of Mechanical Engineering
Thesis Advisor

Karl Reichard
Research Associate and Assistant Professor of Acoustics
Co-Advisor

H. Joseph Sommer III
Professor of Mechanical Engineering
Faculty Reader

Karen A. Thole
Professor of Mechanical Engineering
Department Head of Mechanical and Nuclear Engineering

*Signatures are on file in the Graduate School

ABSTRACT

Automotive manufacturers are researching forward collision warning systems (FCWS) to reduce the occurrence of rear-end collision accidents between vehicles. Traditionally these systems use forward scanning sensor technology such as RADAR or LIDAR to measure the distance between the equipped vehicle and other vehicles/obstacles ahead. The U.S. Army is using such technology on their ground vehicles but has noticed its performance is sometimes compromised due to environmental effects (caking of debris on sensors). This thesis presents the work of developing a FCWS that instead uses Global Position Satellite (GPS) technology and the available information associated as an alternative approach for collision avoidance in convoy situations. This approach however requires a vehicle to vehicle (V2V) network infrastructure to share local GPS data among vehicles.

Sponsorship from U.S. Army and Penn State's own Applied Research Laboratory (ARL) led to the fabrication of three low-cost, embeddable prototype units that were fielded on three Army Heavy Expanded Mobility Tactical Trucks (HEMTTs) vehicles navigating through desert test courses in convoy formation. These experiments proved the feasibility of such an alternative collision detection system.

The primary goal of this thesis is to evaluate how measurement errors/uncertainty affects performance of a GPS-based convoy collision avoidance system. A simple analytical framework is presented for merging system sensitivity analysis and measurement input error characterization results to determine the uncertainty in the output. The resulting metric is a dimensionless parameter corresponding to a range in the probability of collision. To test this approach, field data were analyzed and applied within the proposed framework.

A secondary focus of this thesis is to address a specific concern regarding the feasibility of GPS-based collision avoidance approach due to concerns about GPS accuracy. This thesis

includes identification of dominant GPS stochastic error sources using Allan Variance analysis. The research experimentally compares inter-vehicle distance accuracy, which is a core measurement of the system, between the GPS proposed approach and the traditional LIDAR-based approach in an attempt to address accuracy concerns. As vehicular communication systems such as vehicle to vehicle (V2V) emerge in the near future, a GPS-based FCWS will naturally provide a lower-cost alternative, or even supplemental, solution to the scanning technologies currently implemented. This work thus offers an immediate and substantial opportunity to save lives. While the target application of the work discussed here was for rear-end collisions such as might be encountered in military convoy operations, the solution could be adopted for the civilian commercial sector via straightforward application of existing technology.

Table of Contents

List of Figures.....	vii
List of Tables.....	ix
Acknowledgements.....	x
Chapter 1 Introduction	1
1.1 Motivation.....	3
1.2 Thesis Outline	7
Chapter 2 Collision Warning Systems Overview and Survey.....	8
2.1 Collision System Fundamentals	8
2.1 Existing Collision Detection Systems.....	11
2.2 Current Research Initiatives	13
Chapter 3 The Proposed Collision Warning System.....	15
3.1 Hardware Components	16
3.2 Software: Safe/Danger Decision Algorithm	18
Chapter 4 System Uncertainty Analysis Framework.....	21
4.1 Sensitivity Analysis	23
4.2 Error Analysis of Friction Coefficient	28
4.3 Uncertainty Modeling.....	32
4.4 Summary.....	39
Chapter 5 GPS Drift Characterization Using Allan Variance Analysis.....	40
5.1 Allan Variance A Review	42
5.2 Experiment Setup	47
5.3 Allan Variance Analysis and Results.....	50
5.3.1 San Jose FV-M8 GPS Allan Variance Analysis	51
5.3.2 ADXL-335 3-Axis Accelerometer Allan Variance Analysis.....	60
5.4 Summary.....	65
Chapter 6 Evaluating Inter-Vehicle Distance Measurement	67
6.1 Test Setup and Procedures	68
6.2 Error Analysis	70
6.3 Revised Uncertainty Analysis	77
6.4 Measurement Linearity	80
6.5 Measurement Hysteresis	84
6.6 Summary.....	87
Chapter 7 Conclusions	88
7.1 Summary.....	88

7.2 Future Work	91
Bibliography	95
APPENDIX A San Jose FV-M8 GPS Receiver	98
APPENDIX B MATLAB Code: Sensitivity Analysis.....	100
APPENDIX C MATLAB Code: Allan Variance Analysis for GPS & Accelerometer	103
APPENDIX D MATLAB Code: allan.m.....	108
APPENDIX E MATLAB Code: Calculate NOise Parameters.....	111
APPENDIX F IEEE 1588 Time Synchronization	112
APPENDIX G Continuous Parallel Process for Accessing Simulink Data In MATLAB.....	115
APPENDIX H LIDAR DATA Analysis.....	119

LIST OF FIGURES

Figure 1-1: Distribution of Crash Types (1994 data).....	4
Figure 1-2: Breakdown of 2003 U.S. Crash Data into Crash Types.....	4
Figure 1-3: Photograph of Rear-End Collision Accident Involving Military Equipment.....	6
Figure 2-1: Overview of VORAD VS-400 (From Eaton Corporation)	12
Figure 2-2: Mobileye AWS-400 Driver Interface	13
Figure 3-1: Concept of GPS-Based Collision Avoidance System.....	16
Figure 4-1: Analysis Framework for Estimating Confidence in Warning Parameter.....	22
Figure 4-2: Simulated RE-3 Test Profile Data	25
Figure 4-3: Relative Sensitivity of Parameters.....	27
Figure 4-4: Relative Sensitivity of Friction Coefficient and Driver Tuning Parameters	28
Figure 4-5: Skid Event on Rough Road Surface Raw Velocity Data	30
Figure 4-6: Comparison of Raw, Smoothed, and First Order Model Data for Vehicle 1	31
Figure 4-7: Sigma Uncertainty Analysis.....	36
Figure 4-8: Individual Parameter Error Contributions.....	36
Figure 4-9: Warning Parameter Uncertainty Results.....	37
Figure 4-10: Uncertainty Analysis Results Comparing Effect of Error Magnitude	38
Figure 5-1: Non-Overlapping versus Overlapping Samples Illustration.....	44
Figure 5-2: Typical Allan Deviation plot for a system	45
Figure 5-3: Comparison of Raw Latitude Measurements across Systems	48
Figure 5-4: Comparison of Raw Longitude Measurements across Systems	48
Figure 5-5: Drift Data Deviation in East-North Reference Frame.....	50
Figure 5-6: System 9366 Raw Latitude Measurement.....	51
Figure 5-7: System 9366 Latitude Measurement Allan Deviation Plot	52
Figure 5-8: System 9366 Individual Noise Parameter Influence.....	54
Figure 5-9: Dominant Noise Types at Longer Averaging Times	55
Figure 5-10: Random Walk Model.....	56
Figure 5-11: Comparison of All Allan Variance Results for Latitude Measurement	56
Figure 5-12: Comparison of All Allan Variance Results for Longitude Measurement	57
Figure 5-13: System 9584 East Deviation Allan Deviation Plot.....	59
Figure 5-14: System 9584 North Deviation Allan Deviation Plot.....	59
Figure 5-15: System 9366 Raw Accelerometer Data.....	60

Figure 5-16: System 9366 Acceleration X Allan Deviation and Full Error Model Results	61
Figure 5-17: Comparison of All Three Systems Acceleration X Allan Variance Results	63
Figure 5-18: Comparison of All Three Systems Acceleration Y Allan Variance Results	64
Figure 5-19: Comparison of All Three Systems Acceleration Z Allan Variance Results.....	64
Figure 6-1: Test Vehicles and Equipment.....	68
Figure 6-2: User Interface of LIDAR Data Acquisition Software.....	69
Figure 6-3: PTI Test Track Aerial Photograph.....	70
Figure 6-4: Estimates vs. Ground Truth, Lap Procedure	72
Figure 6-5: Absolute Error between Ground Truth and Estimates	72
Figure 6-6: Estimates Vs. Ground Truth with Outliers Removed	74
Figure 6-7: Absolute Error with Outliers Removed.....	74
Figure 6-8: Hybrid Error Distribution and Statistics.....	76
Figure 6-9: Real-Time Error Distribution and Statistics	76
Figure 6-10: Updated Uncertainty Analysis Results.....	78
Figure 6-11: Comparison of Case 1 and Case 2 Uncertainty Analysis Results.....	79
Figure 6-12: Evaluation of Case 3 Uncertainty Analysis Results.....	79
Figure 6-13: Estimates vs. Ground Truth, Forward/Reverse Procedure	81
Figure 6-14: Estimate Errors, Subset of Forward/Reverse Dataset	82
Figure 6-15: Estimates vs. Ground Truth.....	82
Figure 6-16: Estimate Errors vs. Ground Truth Inter-vehicle Distance	83
Figure 6-17: Error Behavior during Vehicle Approach	84
Figure 6-18: Close Inspection of Delay Effect.....	86

LIST OF TABLES

Table 1-1: Predominant Rear-End Crash Factors	5
Table 2-1: Minimum Driving Environment Detection Specifications.	9
Table 3-1: Sensor Box Parts List	18
Table 4-1: Warning Algorithm Parameters	22
Table 4-2: RE-3 Forward Collision Test Profile Used For Sensitivity Analysis Simulations	24
Table 4-3: Constant Parameters.....	25
Table 4-4: Estimated Friction Coefficient Results for Select Smooth and Rough Surface Skid Brake Events	32
Table 4-5: Statistical Results for Friction Coefficient Estimates.....	32
Table 4-6: “Realistic” Uncertainty Analysis	37
Table 4-7: Quantitative Detection Horizon Comparison	38
Table 5-1: Relation of Allan Variance and Noise Sources [35]	46
Table 5-2: Latitude Cross Correlation Results	49
Table 5-3: Longitude Cross Correlation Results	49
Table 5-4: System 9366 Latitude Noise Parameter Magnitudes	52
Table 5-5: Summary of Random Walk Noise Parameter for Latitude Measurement	55
Table 5-6: Summary of Random Walk Noise Parameters for Longitude Measurement.....	57
Table 5-7: Summary of Random Walk Noise Parameters for East Deviation Estimate	58
Table 5-8: Summary of Random Walk Noise Parameters for North Deviation Estimate.....	58
Table 5-9: System 9366 Accelerometer Full Error Model Noise Parameter Magnitudes.....	62
Table 5-10: Identified Acceleration X Noise Coefficients for All Three Systems	65
Table 5-11: Identified Acceleration Y Noise Coefficients for All Three Systems	65
Table 5-12: Identified Acceleration Z Noise Coefficients for All Three Systems	65
Table 6-1: Hysteresis Results for Hybrid System.....	85
Table 6-2: Hysteresis Results for Real-Time System	85

ACKNOWLEDGEMENTS

First I would like to thank Dr. Sean Brennan for providing me the opportunity to finish my masters' degree. I have learned a lot from you and thank you for your guidance and support over this past year. I admire your dedication to assist the students even at odd hours of the morning when we are asleep. I also admire your, almost super-human, ability to multi-task and making sure everybody is satisfied. I greatly respect you and look to you as an excellent mentor for the future as I begin working professionally, again.

I would also like to thank the US Army TACOM and the engineers at Penn State Applied Research Laboratory for providing the exciting project opportunity. The project topic and more importantly the funding provided me the invaluable opportunity to focus on completing my Masters degree as a full-time student. Thank you for the exciting work and being involved on such an important technology to increase soldier safety.

Next, I want to acknowledge the friendship and support of my fellow research students. You guys certainly made me feel young again and old at the same time. Special thanks go to Stephen Chaves for his partnership in working the collision project.

Finally, I cannot express enough gratitude and love to my parents Rajni and Ranjan Amin. You have been with me through the toughest times these past four years and made sure I never quit although it seemed to always be the easiest option. I hope to continue to make you proud of my future efforts.

Chapter 1

INTRODUCTION

In February 2010, the National Transportation Safety Board (NTSB) updated and released a “most wanted” list for transportation safety system improvements for aviation, rail, marine, and highway sectors. Within the highway sector, one of the most wanted items is to “prevent collision by using enhanced vehicle safety technology” [1]. Specifically the objective for this goal states: “reduce rear-end collisions through the use of adaptive cruise control and collision warning systems”. The NTSB believes that such technology will substantially reduce accidents [2]. The U.S. DOT estimates intelligent vehicle system technologies could prevent over 48% of rear-end, run-off-road, and lane change crashes [3].

Several aftermarket collision warning systems (CWS), or forward collision warning systems (FCWS), are already currently available for the commercial trucking industry. Auto manufacturers are even beginning to include such technology in newer luxury car models. The IIHS has identified 19 vehicles in 2011 offering forward collision warning [4]. These current systems typically rely on forward scanning sensor technologies such as Radar or LIDAR to scan the area in front of a vehicle. The U.S. Army is said to be using said technology on some of their ground vehicles but feedback from users indicates its performance is sometimes compromised due to environmental effects, specifically caking of debris (sand) on sensors. As a result of this specific concern the U.S. Army funded the Penn State Applied Research Laboratory to investigate alternative approaches to forward collision warning systems.

This Army-sponsored investigation seeks the development of a low-cost alternative solution to overcome performance limitations of the current technology by utilizing GPS technology already available on-most military ground vehicles. This alternative solution is different than typical CWS solutions because it requires information exchange between vehicles within a closed network, or vehicle-to-vehicle (V2V) infrastructure. The military is at the forefront of implementing such

communication capabilities with its research eventually paving the way for civilian and commercial vehicles. The Army-funded work led to the fabrication of three custom prototype hardware units that were actually demonstrated in three Army Heavy Expanded Mobility Tactical Truck (HEMTT) vehicles navigating through proving ground desert test courses in convoy formation to prove feasibility of such an alternative collision detection system.

The primary goal of this thesis is to evaluate how measurement errors/uncertainty affects performance of a GPS-based convoy collision avoidance system. A simple analytical framework is presented for merging system sensitivity analysis and measurement input error characterization results to determine the uncertainty in the output. The resulting metric is a dimensionless parameter corresponding to a range in the probability of collision. To test this approach, field data were analyzed and applied within the proposed framework.

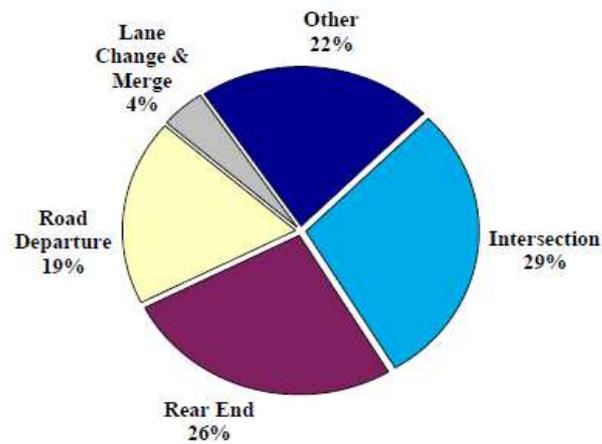
A secondary focus of this thesis is to address a specific concern regarding the feasibility of GPS-based collision avoidance approach due to concerns about GPS accuracy. This thesis includes identification of dominant GPS stochastic error sources using Allan Variance analysis. The research experimentally compares inter-vehicle distance accuracy, which is a core measurement of the system, between the GPS proposed approach and the traditional LIDAR-based approach in an attempt to address accuracy concerns. As vehicular communication systems such as vehicle to vehicle (V2V) emerge in the near future, a GPS-based FCWS will naturally provide a lower-cost alternative, or even supplemental, solution to the scanning technologies currently implemented. This work thus offers an immediate and substantial opportunity to save lives. While the target application of the work discussed here was for rear-end collisions as might be encountered in military convoy operations, the solution could be adopted for the civilian commercial sector via straightforward application of existing technology.

As a quick note, much of the development and preliminary in-field testing for this project was jointly completed with a previous masters student Stephen Chaves [5]. His focus for this project was

to identify the performance improvement gained by fusing GPS and accelerometer measurements using Kalman filtering methods. Some of the future work topics from his thesis are resolved in this thesis. Throughout this thesis, several references are made to specific sections of Stephen's thesis to provide more detailed information about certain topics. A review of Stephen's work is highly encouraged as advanced reading prior to study of this thesis.

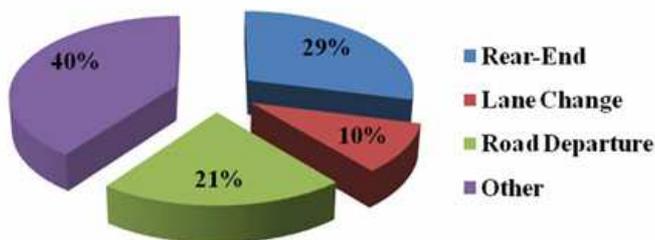
1.1 Motivation

Historical data shows a major percentage of auto accidents are specifically rear-end collisions. In 1993, the National Highway Transportation Safety Administration (NHTSA) performed a comprehensive survey of rear-end crashes based on 1990 police reported crash/accident data. The survey identified 1.5 million rear-end crashes, which constituted 23 percent of all crashes for that year [6]. A similar study was repeated using 1994 data and identified 1.66 million police-reported rear-end crashes, accounting for approximately 26 percent of all crash types as indicated by Figure 1-1 [7]. In addition, a NTSB report published in 2001 identified over 6 million crashes on U.S highways in the year 1999 [8]. Almost one-third of this total was specifically rear-end collisions. The same report detailed a study of a two year period from 1999 to 2001 where the NTSB investigated nine rear-end collisions which resulted in 20 fatalities and 181 injuries (three of the accidents involved buses and one accident involved 24 vehicles). The most recent analysis of crash data from 2003 found rear-end crashes accounting for 29% of all light-vehicle crashes and 22% of all heavy-truck crashes which was the highest category of crash types except for "Other" as indicated by Figure 1-2 [9].

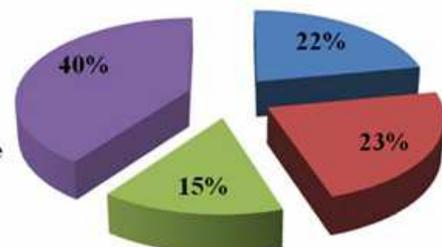


Source: IVI Problem Areas Description:
 Motor Vehicle Crashes - Data Analyses And IVI Program Emphasis [7]
Figure 1-1: Distribution of Crash Types (1994 data)

Total Light Vehicle Crashes



Total Heavy Truck Crashes



Source: Integrated Vehicle-Base Safety Systems First Annual Report [9]
Figure 1-2: Breakdown of 2003 U.S. Crash Data into Crash Types

Although many factors can contribute to a vehicle accident such as road surface quality, weather conditions, and sun glaring previous studies indicate driver distraction as a major leading factor. Table 1-1 shows the primary causal factors for rear-end crashes as classified by the NHTSA and the distribution of 1994 vehicle crash data into each category [7]. The four factors relating to either driver inattention or driver distraction accounted for approximately 92 percent of rear-end crashes. Of this subset, 27% was related to driver inattention combined with following too closely. With increasing use of cellular phones, navigation systems, and entertainment systems in vehicles there is an obvious concern that these distractions may increase crash statistics.

Table 1-1: Predominant Rear-End Crash Factors

Crash Causal Factor	Distribution (Percent)
Inattention	41
Inattention/following too close	27
External Distraction	14
Internal Distraction	10
Other	8
Total	100

Source: IVI Problem Areas Description:
Motor Vehicle Crashes - Data Analyses And IVI Program Emphasis [7]

In most rear-end accidents involving inattention, the driver does not have enough reaction time to either apply appropriate braking or take evasive action to avoid colliding with the preceding vehicle. Providing an early warning notification can increase the driver reaction time. Several previous studies have evaluated the potential benefits of implementing such technology. According to a 1992 study by Daimler-Benz, a CWS can potentially have a significant impact on accident safety statistics. The study found 60% of rear-end collisions could be prevented if the driver was given a warning 0.5 seconds ahead of the collision. In addition, 90% could be prevented if a warning was provided 1 second ahead [10]. Knipling attempted to model rear-end collision countermeasures to evaluate system effectiveness in preventing crashes, identify system functional requirements, and identify major factors that influence system performance [11]. One possible design system algorithm evaluated for 100 samples points using Monte Carlo simulation techniques yielded a system effectiveness of 77 percent. For those simulations that did result in a crash even with countermeasure warning given, the analysis identified a 42 percent reduction in crash severity, inferred from a reduction in vehicle velocity. Knipling goes on to mention that the importance of the results is not the

exact percentages that are obtained but more on the confidence that such technology has the potential to prevent crashes and reduce the severity in those that occur.

Rear-end collisions are also significantly occurring within the military sector, Figure 1-3. In 2004, Military.com posted a United Press International published article that identified 833 crashes, 50 deaths, and 223 injuries in 2003 during the Iraq war [12]. These statistics were claimed as the worst accident record in 10 years. Furthermore, rear-end collisions are a leading cause of convoy 'breakdowns' in theater [13]. Improving soldier safety is of top priority so as a result the U.S. Army TACOM provided funding for developing the GPS-based collision detection system to quickly provide forward collision warning capability. Regardless of whether for civilian or military application, the statistics previously discussed indicate a clear and recognizable need for CWS to reduce accidents and save lives.



Figure 1-3: Photograph of Rear-End Collision Accident Involving Military Equipment

1.2 Thesis Outline

The remaining content of this thesis is organized as follows. Chapter 2 reviews the collaborative research work by Chavez highlighting key pieces of information from his thesis related to the development of the prototype system. Chapter 3 presents a quick summary of the prototype collision detection system including a description of the hardware components and the collision detection algorithm. A framework for determining the system confidence/uncertainty is presented in Chapter 4 and includes a walkthrough example application using actual prototype system test data to estimate friction coefficient error. Since the inherent errors associated with GPS technology is obviously a concern, Chapter 5 describes the analytical process applied for evaluating the noise parameters of the GPS and accelerometer hardware using Allan Variance analysis. To address concerns related to whether GPS is accurate enough, Chapter 6 compares the inter-vehicle distance measurement error of the low-cost GPS sensor component used in the prototype hardware configuration against a highly accurate/high-cost GPS/INS, and then also against ground truth as measured by a LIDAR distance scanning sensor. The results of this section will show that although the low-cost solution may exhibit larger overall GPS errors, the inter-vehicle distance measurement error remains small due to GPS error correlation among units. Finally the conclusion and future work chapter summarizes the overall results of this thesis and provides recommendations for future efforts.

Chapter 2

COLLISION WARNING SYSTEMS OVERVIEW AND SURVEY

Since the early 1990s, collision detection has been a goal in the area of vehicle automation and driver assist, motivated highly by the statistics of crashes in the commercial sector. The objective of any collision warning system is to scan for vehicles, or objects, in the forward path of the equipped vehicle and alert the driver of any potential collision conditions. In this chapter a brief overview of collision systems is provided. The first section presents the fundamentals of collision detection as established by previous researchers. The following sections cover current research civilian and military research initiatives and finally a brief review of currently available commercial systems.

2.1 Collision System Fundamentals

There has been significant prior research in developing collision detection systems. Doi, et al. of Mazda R&D first identified four main technological components necessary for successful collision detection and avoidance: 1) driving environment detection, 2) path estimation algorithm, 3) safe/danger decision algorithm, and 4) longitudinal automatic brake control.

The first component, driving environment detection, is traditionally accomplished by forward scanning sensor systems. Previously developed systems have used a variety of sensor technologies including sonar, machine vision, video cameras, infrared, radar, and lasers [14]. Kamiya, et al. of Honda attempted to outline the minimum performance specifications necessary for successful collision detection incorporating forward-scanning radar systems and are shown in Table 2-1. The specifications for range accuracy and refresh time were considered more important.

Table 2-1: Minimum Driving Environment Detection Specifications.

Description	Specification
Longitudinal detecting range	≥ 100 m
Horizontal detecting area	≥ 350 mrad
Vertical detecting area	≥ 50 mrad
Horizontal resolution	≤ 5 mrad
Range accuracy	≤ 1 m
Refresh time	≤ 0.1 s

Source: Kamiya and et al. [15]

The second component is a path estimation algorithm which estimates the trajectory of the vehicle during curves to better determine whether the vehicle will collide with obstacles or vehicles that lay in that path. There has been significant previous research in the area of estimating vehicle trajectory producing simple and complex algorithm solutions; however, the path estimation problem is simplified for this work since vehicles are traveling in a convoy and the preceding vehicle's position can be relayed to following vehicles.

The third component, safe/danger decision algorithm, determines the collision probability between two vehicles by logically considering the kinematics of the vehicles. Overall the algorithm compares the current spacing distance to the critical spacing distance needed for the vehicles to arrive at a safe stop condition without colliding. Chaves provides a thorough literature review of these algorithms from previous researchers in Chapter 2 of his thesis. Modified versions of several formulas developed by Seiler et al. were implemented as the collision algorithms for this GPS-based collision system [16]. The original formulas are quickly reviewed here. In equation (2-1) v is the host vehicle's velocity, v_{rel} is the relative velocity between the host vehicle and preceding vehicle, α is the

maximum deceleration of the vehicles (assuming deceleration is same for both), τ is the delay time, and d_0 is the buffer distance between vehicles:

$$d_{warn} = \frac{1}{2} \left(\frac{v^2}{\alpha} - \frac{(v - v_{rel})^2}{\alpha} \right) + v\tau + d_0 \quad (2-1)$$

The maximum deceleration, time delay, and distance buffer are all considered tunable parameters that are adjusted depending on the current driving conditions and driver preference. Seiler et al. also presents a critical braking distance equation (Equation 2.2), which represents the inter-vehicle distance at which an active collision system would intervene and apply braking. The variables τ_{sys} and τ_{hum} represent system and human delays respectively.

$$d_{brake} = v_{rel}(\tau_{sys} + \tau_{hum}) + 0.5\alpha(\tau_{sys} + \tau_{hum})^2 \quad (2-2)$$

The previous equations are used to determine the collision system critical distances for warning the driver and applying appropriate braking. Of course, these distances must be compared to the actual inter-vehicle distance to determine the collision probability. To determine the probability of collision, a warning parameter equation is presented in Equation (2-3), also developed by Seiler et al. The warning parameter is a non-dimensional value that is simply calculated as the ratio of the current inter-vehicle distance, d , minus the braking distance, d_{brake} , to the warning distance, d_{warn} , minus the braking distance.

$$w = \frac{(d - d_{brake})}{(d_{warn} - d_{brake})} \quad (2-3)$$

If the value of the warning parameter is greater than or equal to one, the current driving situation is considered safe. Values between one and zero constitute unsafe conditions and different warnings are provided to the driver.

In context to the GPS-based collision system, the fourth component is considered as “warning or intervention” which encompasses both classifications of collision systems as either active or passive. An active collision system may take control of the vehicle (braking or steering) to avoid

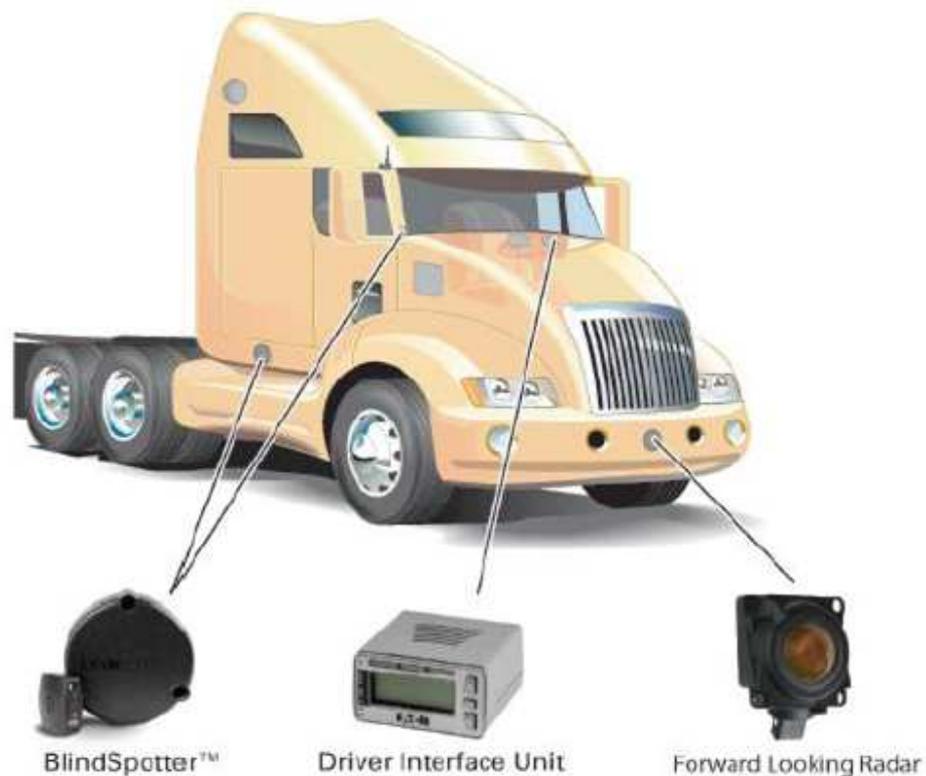
collision. In contrast, passive systems may simply provide a “warning” to the driver and requires them to actively respond. Most COTS systems are typically designed to be passive. The solution presented here is considered a passive system since it visually displays an alert to the driver. A well designed driver interface will properly alert the driver without producing a high number of false alarms; otherwise, the driver will develop low confidence in the alert. The specific representation of the alert has a role in how the alert is perceived by the driver. Lee et al. published a study in which two experiments were conducted to examine driver response to warning representation [17]. The study identified collision warning representation was better received by drivers with a graded representation proportional to the degree of threat was provided instead of a binary on/off when a certain threshold was exceeded. The graded warning did however produce more alerts but interestingly, the drivers trusted the graded warning more than the single-stage and did not indicate any annoyance from the extra alerts. A graded warning is an example of a likelihood alarm display (LAD) and previous studies have shown LADs can improve focus of alerts among multiple tasks [18].

2.1 Existing Collision Detection Systems

There are several commercial-off-the-shelf (COTS) collision detection systems that exist today, most of which are used in the trucking and public transportation industries. A variety of sensor technologies are used for vehicle detection but primarily rely on forward scanning sensors that must infer the position and speed of the vehicle ahead using sensor data.

The most well-known and successfully deployed commercial collision avoidance system is the Bendix VORAD (Vehicle On-board RADar) system specifically designed for the commercial trucking industry and heavy military vehicles. The latest model, VORAD VS-400, features a 77GHz radar with a detection range of 3 to 500 ft within speeds of 0.5 to 120 mph [19]. The VORAD system also detects vehicles/obstacles along the side of the equipped vehicle through a side object detection

system called BlindSpotter. The specifications and design of the VORAD driver interface were developed in collaboration with NHSTA and provides both auditory and visual alerts and exhibits a form factor which allows for integration into a truck dash [20]. In addition to relaying frontal and side collision alerts the visual display provides system status information (e.g., component failure, system availability, display settings). Figure 2-1 provides an overview of the VS-400 system.



Source: Eaton Corporation [21]

Figure 2-1: Overview of VORAD VS-400 (From Eaton Corporation)

Another commercial system is the Mobileye AWS (Advanced Warning System) which is strictly vision based and uses a single windshield-mounted camera for forward collision detection, lane departure warning, and headway distance monitoring. The application for this system includes commercial trucks as well as automotive sedans. Several automotive sedan manufacturers, BMW, Volvo, Buick, and Cadillac, have already deployed the Mobileye system in their models [22]. The

specifications for the latest system, Mobileye AWS-400 state a vehicle detection range of up to 70 meters with a sensor field of view of 40° x 30° (WxH). The driver interface for this system is capable of providing forward collision, lane departure, and heading monitoring information on a single interface as shown in Figure 2-2.



Source: Mobileye AWS-400 Website

Figure 2-2: Mobileye AWS-400 Driver Interface

To date, there is still no readily available commercial system that utilizes a cooperative collision detection architecture where information is exchanged between vehicles, e.g. a Vehicle to Vehicle interface (V2V), through wireless communication.

2.2 Current Research Initiatives

The U.S. Department of Transportation has funded and facilitated most of the research and development work of collision detection systems since the early 1990s in the interest of incorporating the technology for vehicle automation and driver assistance systems. The latest Integrated Vehicle-

Based Safety Systems (IVBSS) Initiative is a partnership between the government, auto manufacturers and commercial vehicle industries to develop and field test systems that detect rear-end, lane-change, and lane-departure collision scenarios on light vehicles and heavy commercial trucks [23]. Currently the IVBSS team is performing in-field test of developed prototype hardware on 16 Honda Accords vehicles. Some of the overall goals of the testing are to evaluate the generation and quantity of false alarms, check system availability, understand algorithm warning logic, and assess alerts in perceived crash situations [24]. A detailed test plan is available online providing a description of test procedures to specifically evaluate the performance of forward collision warning functionality [25]. This document was a useful reference for outlining some of the test plans for the GPS-based collision detection system.

Chapter 3

THE PROPOSED COLLISION WARNING SYSTEM

As mentioned before, the interest for a GPS-based collision system was generated by the Army for addressing rear-end collisions in military convoys. Specifically the sponsoring customer, Army TACOM, was seeking a solution which could improve safety during convoy training exercises at their training centers. Overall, the solution needed to be low cost, have a small form factor, and be easy to install without requiring any vehicle modifications. The specific target application/scenario allows for a set of assumptions that simplifies the design and performance requirements of the prototype system.

The closed application assumptions are:

- The vehicles of interest are part of the convoy and therefore each vehicle will contain the necessary hardware. Vehicles or objects without the prototype hardware are considered external and will not be detectable by the system.
- Open-sky visibility is necessary for GPS receivers to acquire satellite lock-on and thus the collision avoidance system will only be used in open-sky locations or where GPS signals are readily available and not blocked.
- All vehicles within the convoy are driving in single column formation and vehicles do not enter or exit the formation, or change their ordering in the convoy, i.e. no passing
- The system is only concerned with rear-end collisions and does not try to detect collisions on the side of the vehicle.

Figure 3-1 provides a conceptual overview of the GPS-based collision detection system for military convoys. In order for proper collision detection between convoy vehicles, each vehicle in the convoy is equipped with a prototype hardware system. The vehicles then communicate with each other to transmit vehicle specific position and velocity information to determine collision probability.

This type of collision detection architecture is often referred to as a *cooperative* collision detection system [26].

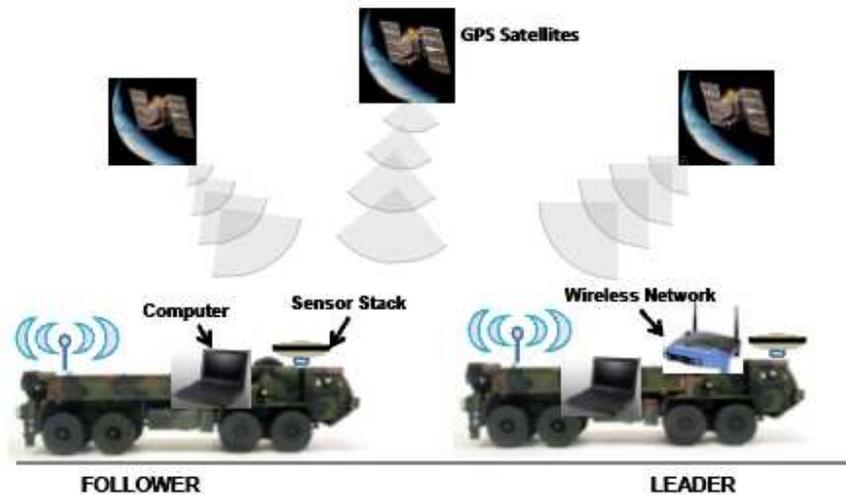


Figure 3-1: Concept of GPS-Based Collision Avoidance System

3.1 Hardware Components

Each hardware system is made up of two main components: a sensor module containing embedded hardware and a Dell Latitude 2100 Netbook computer. The primary component of the embedded sensor module is the FV-M8 GPS receiver unit manufactured by San Jose Navigation Technologies. The FV-M8 module outputs GPS position information at a rate of 5 Hz with a rated position accuracy of 3.3m CEP and is intended for embedded applications. Additional specifications are provided in Appendix A. The sensor module also contains the following embedded components: an Arduino Mega microcontroller, a Wiznet Ethernet module, and an ADXL-335 3-axis accelerometer. The Arduino microcontroller reads serial messages produced by the FV-M8 GPS module, which contain the GPS position, velocity, and time information, and voltage values from the 3-axis accelerometer and formats the data from both sensors into a new data message. This new data

message is transferred, using TCP/IP communication protocol, through a wired Ethernet connection, using the embedded Wiznet Ethernet module, to the Dell netbook computer. All the sensor stack components are placed in a watertight protection case affixed to a metal base with magnetic feet. The magnetic base allows the sensor module or “sensor box” to be easily attachable to any vehicle roof. It is also quickly mentioned that the watertight case was later retrofitted with an active cooling PC fan to blow air across the electronics after identifying overheating of the components during some field testing in the desert. As mentioned, the Dell netbook computer acquires data from the local sensor box but also exchanges it with remote computers onboard nearby convoy vehicles via wireless TCP/IP communication. The wireless network is representative of a growing concept called Vehicle-to-Vehicle (V2V) communication and is essential for cooperative collision detection. The computer is also responsible for processing the sensor data through a collision algorithm to determine the probability of the local vehicle colliding with the preceding vehicle. In this prototype version, the built-in computer monitor serves as the driver display and visually notifies the driver of collision conditions in real-time. Table 3-1 provides a list of all the components along with their cost and vendor source. Additional information regarding the hardware can be found the Chaves’ thesis [5].

Table 3-1: Sensor Box Parts List

Item	Model	Price	Vendor
GPS Receiver	FV-M8	\$100	www.sparkfun.com
GPS Shield Kit	-	\$20	www.adafruit.com
Microcontroller	Arduino Mega	\$65	www.sparkfun.com
Ethernet Module	WIZnet W5100	\$25	www.sparkfun.com
Ethernet Shield	-	\$15	www.adafruit.com
3-Axis Accelerometer	ADXL335	\$25	www.sparkfun.com
GPS Connector	-	\$2	www.sparkfun.com
PC Cooling Fan		\$10	RadioShack
Watertight Enclosure		\$25	Walmart
Computer	Dell Latitude 2100	\$350	Dell
Total		\$637	

3.2 Software: Safe/Danger Decision Algorithm

As mentioned in Chapter 2 there are four primary components of a collision detection system. This section focuses on the third component, the safe/danger decision algorithm, and its implementation in the proposed collision detection system. The safe/danger decision algorithm here is based on an algorithm developed by Seiler, et al. in [16]. Since the system proposed here is considered a passive collision *warning* system, equation (3-1) is defined in terms of critical warning distance instead of a critical braking distance, which is necessary for an active collision *avoidance* system.

$$d_{warn} = \frac{1}{2} \left(\frac{v^2}{\alpha} - \frac{(v - v_{rel})^2}{\alpha} \right) + v\tau + d_0 \quad (3-1)$$

The warning parameter calculation shown in equation (3-2) is the final output of the safe/danger decision algorithm. This calculation compares the calculated warning distance to the current inter-vehicle distance between the vehicles. If the value of the warning parameter is greater than or equal to one, the current driving situation is considered safe. As the inter-vehicle distance decreases the warning parameter value falls below one, representing unsafe conditions. A warning

parameter value exactly equal to zero represents a collision. The visual design of the collision system driver display is directly linked to the warning parameter.

$$w = \frac{d}{d_{warn}} \quad (3-2)$$

Seiler also proposed the addition of scaling factors to compensate for road friction uncertainty and individual driver behaviors. The friction, $f(\mu)$, and driving tuning, $g(driver)$, scaling factors scale the critical warning distance, d_{warn} , as shown in the following equation.

$$d_{warn,scaled} = d_{warn} * f(\mu) * g(driver) \quad (3-3)$$

Seiler suggests the driver scaling is accomplished via a dashboard knob. For example, for overly cautious drivers setting the driver tuning factor to 0.5 would warn the driver earlier of collision scenarios. Because this is a driver dependent variable, the driver tuning factor is set to one for the analysis of the warning parameter uncertainty in this thesis and its affects are ignored. However, the friction scaling factor is a function of the current, real-time tire road coefficient, which can be estimated.

Unlike currently available commercial systems that directly measure the inter-vehicle distance, the GPS-based collision system must calculate this measurement using latitude and longitude coordinates of each vehicle. The Haversine formula, described by the following list of equations, is used to calculate this distance because it considers the spherical surface of the earth and yields the great-circle distance between two sets of latitude and longitude coordinates [27].

$$\Delta lat = latitude_1 - latitude_2 \quad (3-4)$$

$$\Delta long = longitude_1 - longitude_2 \quad (3-5)$$

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(latitude_1) + \cos(latitude_2) + \sin^2\left(\frac{\Delta long}{2}\right) \quad (3-6)$$

$$c = 2 * \text{atan2}(\sqrt{a}, \sqrt{1-a}) \quad (3-7)$$

$$distance = R * c \quad (3-8)$$

Where R is the earth's radius (6371 km) and latitude/longitude positions are expressed in degrees.

Three prototype units were developed for the purpose of demonstrating their feasibility. The three units were mounted on three Army HEMTT ground vehicles and the trucks were driven through desert test courses at the U.S. Army Yuma proving grounds. The trucks were driven in a convoy, or single line, formation. In this pattern, the three collision detection systems then provided collision coverage for the middle vehicle with respect to the leader, and then also for the last vehicle with respect to the middle vehicle. A variety of test courses were utilized and various procedures were executed. Complete details on this testing are comprehensively covered in Chapter 5 of Chaves thesis [5]. The data collected during this feasibility testing was used by Chaves to evaluate whether application of Kalman filtering techniques to fuse the GPS velocity and local accelerometer data could improve the collision detection performance. For this thesis, the velocity data from the testing also served useful for analyzing the friction coefficient of the vehicles and its uncertainty. This is covered in the next chapter which overall focuses on analyzing the uncertainty in the collision detection warning parameter output.

Chapter 4

SYSTEM UNCERTAINTY ANALYSIS FRAMEWORK

The proposed prototype system for GPS-based collision detection has been presented in the previous chapter. In short, GPS and accelerometer sensors produce data as measurement inputs to a collision detection algorithm. The algorithm returns as output, collision probability in the form of a warning parameter. A key goal of this thesis is to determine the confidence in the warning parameter output by examining the sensitivity of parameters in the collision algorithm and identifying appropriate error statistics.

A conceptual framework for this analysis is shown in Figure 4-1. First, the warning parameter equation is examined for constituent signals/parameters. Combining equations 3-1 and 3-2 from the previous chapter, the overall warning parameter algorithm reduces to:

$$w = \frac{d}{\left[\frac{1}{2} \left(\frac{v^2}{\alpha} - \frac{(v - v_{rel})^2}{\alpha} \right) + v\tau + d_0 \right] * \mu * K_{driver}} \quad (4-1)$$

The parameters in equation (4-1) are identified and listed in Table 4-1. A sensitivity analysis is then applied to equation (4-1) to determine which parameters influence and dominate the variance of the warning parameter. In addition an error model is derived for each parameter as shown in the right path of the diagram in Figure 4-1. Finally the error models are combined with the percentage of variance influence to determine the overall variance of the warning parameter. The next subsections cover the remaining elements of this framework with particular focus on evaluating the propagation of friction coefficient error in the warning parameter uncertainty.

GOAL: Develop prediction of error/uncertainty in the warning parameter (w)

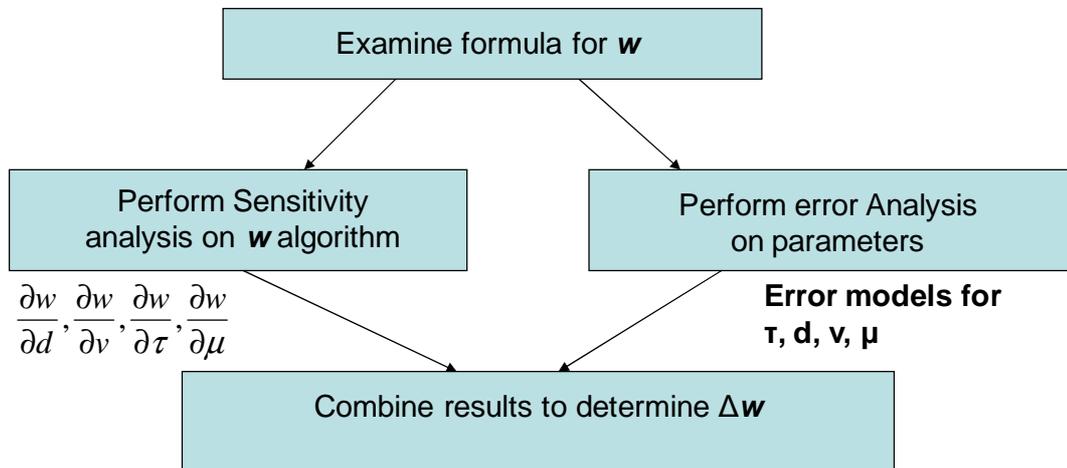


Figure 4-1: Analysis Framework for Estimating Confidence in Warning Parameter

Table 4-1: Warning Algorithm Parameters

Parameter	Description
Time Delay, τ , (second)	The delay in exchanging information between vehicle across the network
Vehicle Velocity, v , (m/s)	The speed of the subject vehicle (following vehicle)
Relative Vehicle Velocity, v_{rel} , (m/s)	The relative speed between the subject vehicle and the principal object vehicle (preceding vehicle)
Inter-Vehicle Distance, d , (m)	The distance between the vehicles
Buffer Distance, d_o , (m)	The distance between the subject vehicle and an imaginary “bumper” limit ahead
Friction Coefficient, μ ,	The tire-road surface interaction friction coefficient
Driver Adjustment, K_{driver} ,	A tunable parameter that would be adjusted by the driver to accommodate specific driving preferences
Maximum Deceleration, α , (m/s ²)	The maximum deceleration capability of the vehicle

4.1 Sensitivity Analysis

Sensitivity analysis is a classical technique that characterizes the performance of a system in response to variations in parameters that affect the system [28]. Sensitivity analysis can be used to: simplify/reduce complex models, investigate the robustness/stability of model outputs, and investigate the interaction between parameters. Specifically relevant to this work, sensitivity analysis is also useful in determining the confidence in the model and its outputs by evaluating individual parameter input uncertainties [29]. Hamby published a paper reviewing techniques for parameter sensitivity analysis in environmental models and provides a good overview of different sensitivity analysis methods [30]. Of the techniques described, the differential analysis approach, also known as the direct method, is recommended if the model is simple and described by explicit algebraic equations from which the partial derivatives are easily obtainable. In these cases the sensitivity of the output, Y with respect to the varied parameter, X is computed by the partial derivative of Y with respect to X times a normalization quotient X/Y, i.e.,

$$\varphi_i = \frac{\partial Y}{\partial X_i} \left(\frac{X_i}{Y} \right) \quad (4-2)$$

Where The calculated sensitivity coefficient, φ_i , represents the ratio of change in output to change in parameter input while all other parameters remain constant. This provides a measurement of how “sensitive” Y is to X, i.e. the relative sensitivity.

Using this simplistic approach, the sensitivity coefficient for each parameter was evaluated based on a test profile collision scenario. First, artificial time series data (X_i) for specific parameters was generated based on a collision test profile adopted from the Integrated Vehicle-Based Safety System (IVBSS) Light Vehicle Test Plan document, published by the U.S. Department of Transportation [31]. This document provides specific procedures to follow for validation testing of vehicle safety systems as they are introduced to the market in the future. Within this test plan document, there are six procedures specifically designed for testing forward collision warning

functionality. Of these six, procedure RE-3 was used for the sensitivity analysis effort because it represented a common, realistic collision scenario. In this scenario a subject vehicle (SV), travels at constant speed while approaching a preceding obstacle vehicle (POV) that is aggressively decreasing its speed. Relevant test profile specifications are listed in Table 4-2.

Table 4-2: RE-3 Forward Collision Test Profile Used For Sensitivity Analysis Simulations

Parameter, Unit	Value
SV Starting Velocity	20.1m/s (45 mph)
POV Starting Velocity	20.1m/s (45 mph)
POV Deceleration	3.5 m/s ² (7.8mph)
Initial Inter-Vehicle Distance, m	80 m (263 ft)

Based on the test profile specifications, the inter-vehicle distance, SV velocity, POV velocity and relative velocity between vehicles were all calculated using simple kinematic equations to produce artificial input data for the collision warning algorithm using MATLAB. Figure 4-2 shows the generated inter-vehicle distance data in the top plot, while the bottom plot shows the constant SV velocity, the decreasing POV velocity, and the increasing relative velocity between them. Artificial data for the remaining algorithm parameters was generated as constants, as listed in Table 4-3. These values were originally used during feasibility demonstration testing of the prototype hardware which is covered in detail in Chapter 5 of Chaves' thesis.

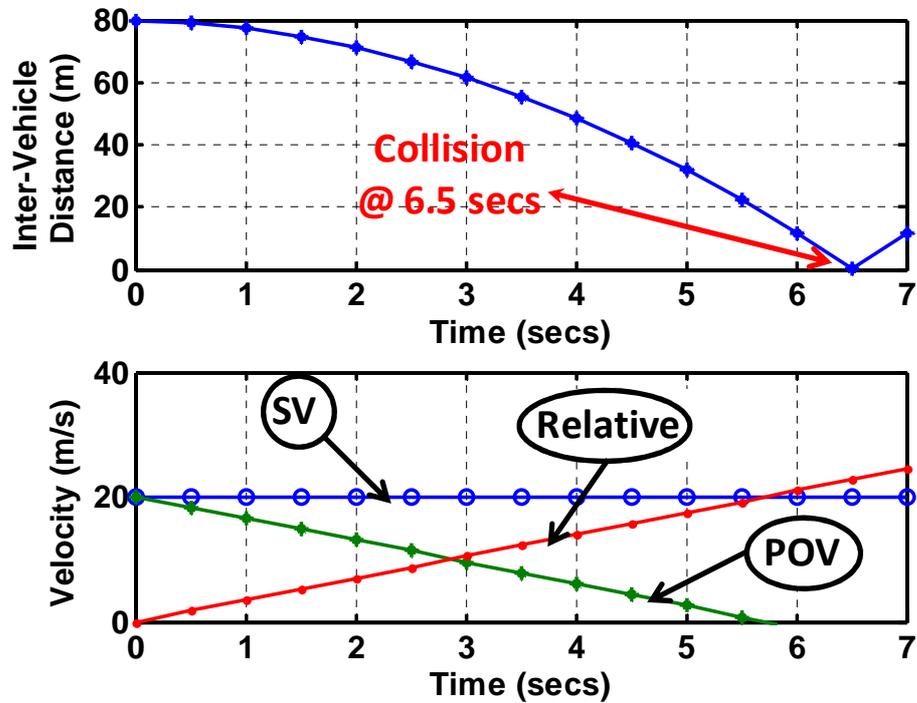


Figure 4-2: Simulated RE-3 Test Profile Data

Table 4-3: Constant Parameters

Parameter	Value
Time Delay, τ , (second)	1.4
Buffer Distance, d_o , (m)	5
Driver Adjustment, K_{driver} ,	1
Friction Coefficient, μ ,	0.8
Maximum Deceleration, α , (m/s^2)	8

The artificial data generated was then used in calculating the warning parameter value (w), via equation (4-1), the respective partial derivative ($\frac{\partial w}{\partial x_i}$), and subsequently the respective sensitivity coefficient (ϕ_i) for each corresponding data sample using equation (4-2). To calculate the partial

derivate value it was necessary to derive the partial derivative equations of the warning parameter, W, to each of the parameters. The partial derivative equations were hand-derived and validated using Mathematica software. The equations are listed as follows

$$\frac{\partial w}{\partial d} = \frac{\alpha}{\mu * K_{driver}(\alpha * d_0 + \alpha * \tau * v + v_{rel} - \frac{1}{2} * v_{rel}^2)} \quad (4-3)$$

$$\frac{\partial w}{\partial v} = \frac{-\alpha * d(a * \tau - v_{rel})}{\mu * K_{driver}(\alpha * (d_0 + \tau * v) + (v - \frac{1}{2} * v_{rel}) * v_{rel})^2} \quad (4-4)$$

$$\frac{\partial w}{\partial v_{rel}} = \frac{-\alpha * d(v - v_{rel})}{\mu * K_{driver}(\alpha * (d_0 + \tau * v) + (v - \frac{1}{2} * v_{rel}) * v_{rel})^2} \quad (4-5)$$

$$\frac{\partial w}{\partial \alpha} = \frac{d(v - \frac{1}{2} * v_{rel}) * v_{rel}}{\mu * K_{driver}(\alpha * (d_0 + \tau * v) + (v - \frac{1}{2} * v_{rel}) * v_{rel})^2} \quad (4-6)$$

$$\frac{\partial w}{\partial \tau} = \frac{-\alpha^2 * d * v}{\mu * K_{driver}(\alpha * (d_0 + \tau * v) + (v - \frac{1}{2} * v_{rel}) * v_{rel})^2} \quad (4-7)$$

$$\frac{\partial w}{\partial d_0} = \frac{-\alpha^2 * d}{\mu * K_{driver}(\alpha * (d_0 + \tau * v) + (v - \frac{1}{2} * v_{rel}) * v_{rel})^2} \quad (4-8)$$

$$\frac{\partial w}{\partial \mu} = \frac{-\alpha * d}{\mu^2 * K_{driver}(a * d_0 + a * \tau * v + v * v_{rel} - \frac{1}{2} * v_{rel}^2)} \quad (4-9)$$

$$\frac{\partial w}{\partial K_{driver}} = \frac{-\alpha * d}{\mu * K_{driver}^2(a * d_0 + a * \tau * v + v * v_{rel} - \frac{1}{2} * v_{rel}^2)} \quad (4-10)$$

The results of applying equation (4-2) to the individual algorithm parameters are displayed in Figure 4-3 and it provides a first-look at how the relative sensitivities of each parameter change over the test profile with respect to the warning parameter. Of all the parameters, the inter-vehicle distance parameter relative sensitivity remains constant. The relative sensitivity appears to increase for the

subject vehicle velocity as the vehicle approaches collision with the preceding vehicle. However, the relative vehicle velocity sensitivity is parabolic and increases to a certain bound and then decreases as the vehicles become closer. The buffer distance and time delay parameters both appear to decrease and reach horizontal asymptotes. The relative sensitivity of the maximum deceleration parameter increases, but also appears to reach a horizontal asymptote. The relative sensitivities of both the friction coefficient and driver tuning parameters first seem to appear to be zero. Figure 4-4 is a zoomed perspective of Figure 4-3. This perspective shows the friction coefficient and driver tuning relative sensitivity results are not actually zero but have some influence on the warning parameter albeit significantly smaller as compared to the other aforementioned parameters. The MATLAB m-script developed to simulate the RE3 test profile, analyze the relative sensitivities, and produce these figures is also provided in Appendix B.

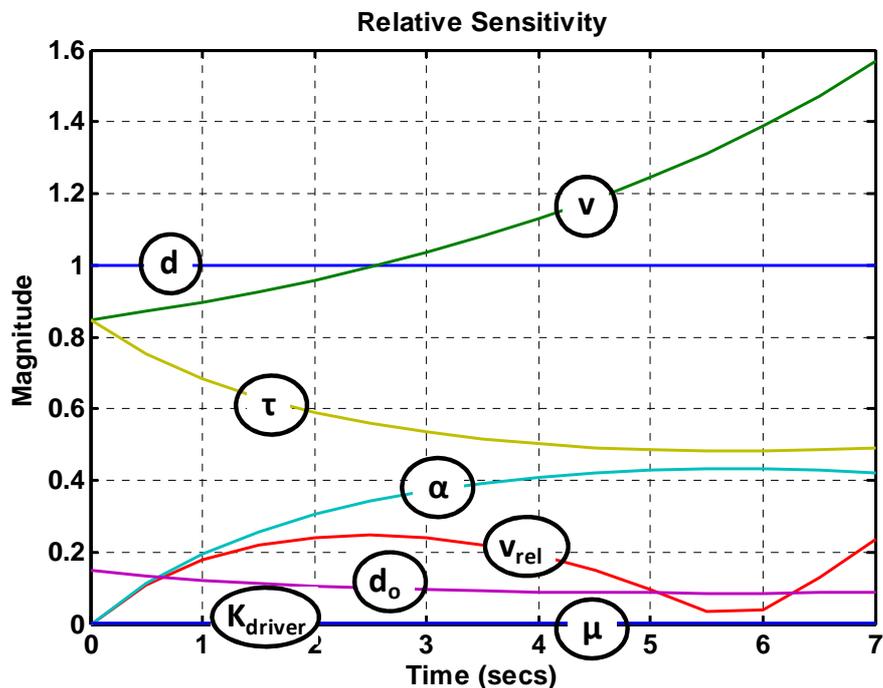


Figure 4-3: Relative Sensitivity of Parameters

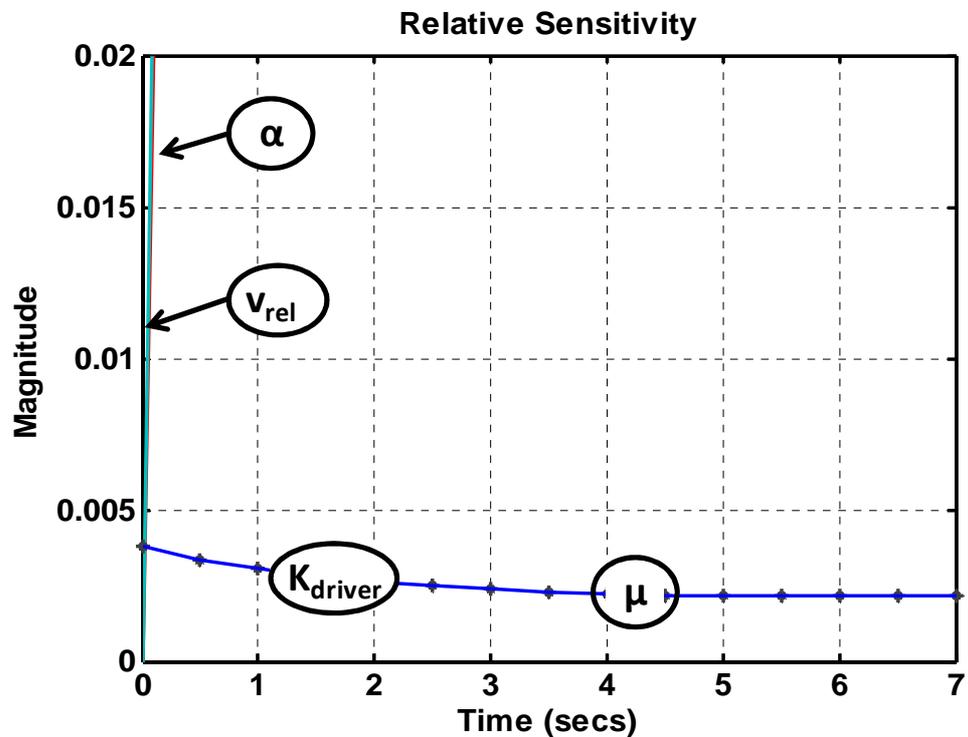


Figure 4-4: Relative Sensitivity of Friction Coefficient and Driver Tuning Parameters

4.2 Error Analysis of Friction Coefficient

As previously described with the framework shown in Figure 4-1 the overall uncertainty analysis also requires identification of the error bounds of each of the parameters. In this section error bounds are identified for the friction coefficient as an example. This parameter was specifically selected because the prototype hardware configuration does not provide for direct friction coefficient measurement. Instead a real-time assessment of friction must be inferred through other available measurements such as GPS velocity and/or accelerometer acceleration. Here a simple approach using GPS velocity to estimate the friction coefficient was adopted and so subsequently the errors associated with this approach are investigated.

GPS velocity data collected from field testing of the prototype hardware was used to estimate the friction coefficient and its variance. The data was collected as part of an effort to test and demonstrate the feasibility of the prototype collision detection system and hardware to the Army at

the Arizona Yuma Proving Grounds. The prototype hardware was installed on three HEMTT type vehicles. Several skid tests were performed for the specific purpose of collecting data that would later be useful for estimating the friction coefficient. The skid test procedure involved accelerating each vehicle to a steady state speed of approximately 30 mph, cruising at that steady state speed for a brief period of time, and then applying the brakes as aggressively as possible to initiate wheel lock and skidding. These tests were performed both on a smooth, paved road surface, specifically an airport taxiway tarmac surface, and also on an off-road, loose, dirt surface. The test procedure was performed on two different surfaces to better characterize the variation in friction coefficient related to changes in road surface smoothness. Of course, for smoother, paved, road surfaces, the friction coefficient is expected to be higher since the contact area is higher between the road and wheel.

During each skid event the vehicle velocity linearly decreased. This is visually confirmed in the raw data shown in Figure 4-5. The stair step pattern evident in the raw data exists because the update frequency of the GPS based velocity data (5Hz) was slower than the sampling rate of the data acquisition software (50Hz). The specific relationship between velocity and friction coefficient can be identified through derivation of basic kinematic principles. Starting with Newton's Second Law of Motion, the vehicle deceleration during skidding is equal to the opposite negatively affecting friction force as shown in equation (4-11). Where m is vehicle mass, a is vehicle acceleration, μ is friction coefficient, and g is gravity. By canceling the mass terms and then integrating this relationship, the relationship between vehicle velocity and friction coefficient is recognizable as shown in equation (4-12). Relating this equation to a basic first order linear equation ($y = m*x+b$), the friction coefficient and gravity represent the slope parameter, m , or the first order coefficient.

$$m * a = -\mu * g * m \quad (4-11)$$

$$v = -\mu * g * t + v_0 \quad (4-12)$$

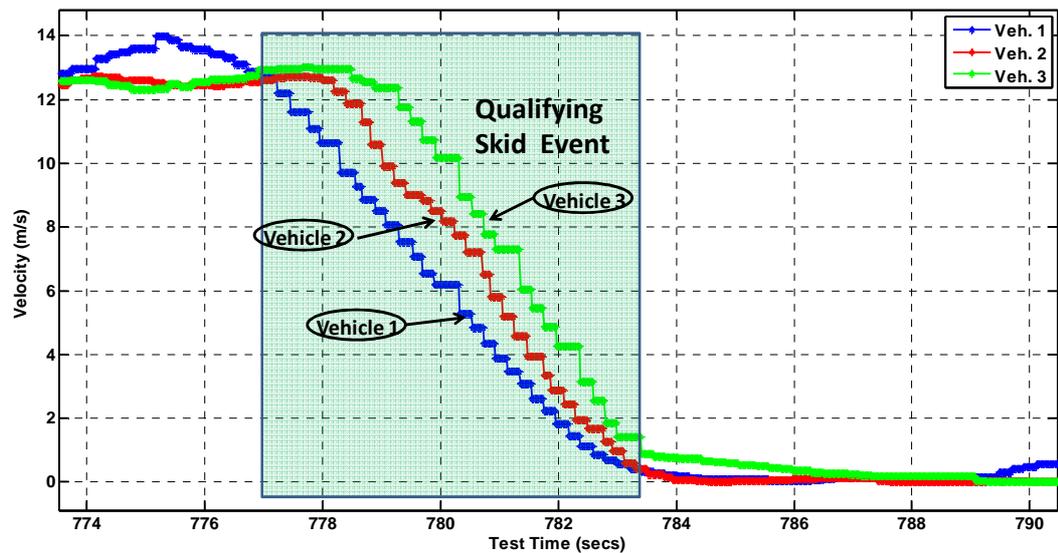


Figure 4-5: Skid Event on Rough Road Surface Raw Velocity Data

The specific analysis procedure implemented with the dataset was applied as follows. First the velocity data was graphed against test time and manually inspected for skid brake events as shown previously in Figure 4-5. After identifying the time interval during which the skid event occurs, the velocity data for that interval range was extracted as a data subset. Next, the subset data was smoothed using a moving average filter. The smoothing of the data eliminates the step change decreases evident in the raw data and improves the linear fit analysis subsequently applied. In the linear fit analysis, a regression analysis was applied to derive a first order model of the data. For this step the MATLAB Curve Fitting toolbox was used. The output of the toolbox is a first-order linear model with an estimate for the first order coefficient including 95% statistical confidence bounds. Since the first order friction coefficient value represents the combination of friction coefficient and gravity, this value must be normalized by 9.81m/s to eliminate gravity and arrive at an estimate for

the friction coefficient. For comparison, the raw data, the smoothed data after applying moving average filter, and the resulting first order linear model are shown in Figure 4-6 for vehicle 1 velocity data captured for the skid event shown previously in Figure 4-5.

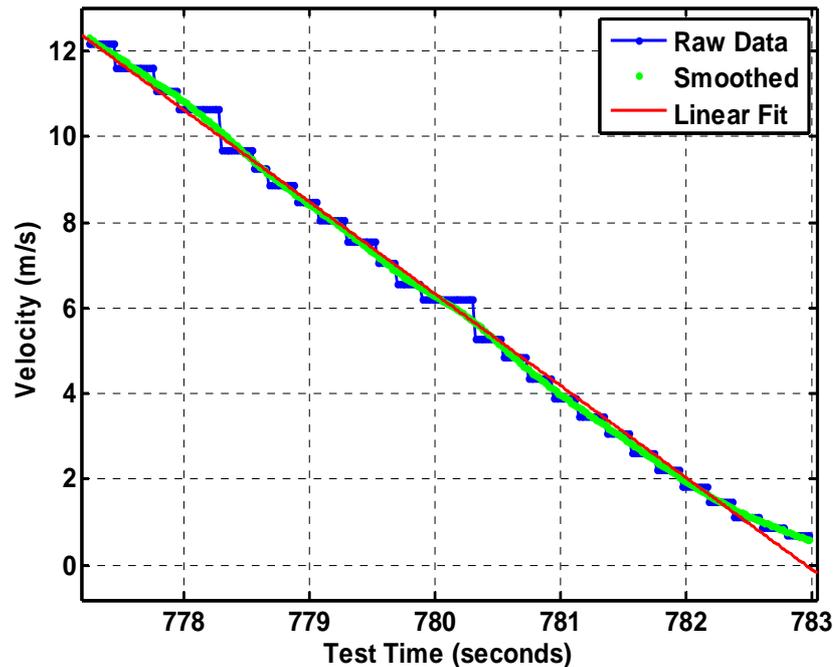


Figure 4-6: Comparison of Raw, Smoothed, and First Order Model Data for Vehicle 1

The analysis was applied to the velocity data collected from each of the three vehicles for four separate skid brake events providing a total of 12 characterized events. The resulting estimated friction coefficient values are listed in Table 4-4. Overall, the results appear consistent; the friction estimates are higher for the smooth surface than the rough surface, as expected. Interestingly, the coefficient values also appear to increase in relation to the order of the vehicles. This is perhaps due to higher road surface temperature as a result of the preceding vehicle braking. Using these results, the statistical mean, standard deviation and variance of the estimated friction coefficient for smooth road and rough road surfaces are listed in Table 4-5. For a conservative collision avoidance parameter estimate, the lowest value is recommended since it corresponds to the lowest capability in vehicle braking.

Table 4-4: Estimated Friction Coefficient Results for Select Smooth and Rough Surface Skid Brake Events

Unit	Road Surface	Friction Coefficient	95% Confidence Bounds
Event 1/Vehicle 1	Smooth	0.453	(0.448, 0.459)
Event 1/Vehicle 2	Smooth	0.484	(0.481, 0.486)
Event 1/Vehicle 3	Smooth	0.51	(0.506, 0.514)
Event 2/Vehicle 1	Smooth	0.448	(0.442, 0.453)
Event 2/Vehicle 2	Smooth	0.488	(0.487, 0.491)
Event 2/Vehicle 3	Smooth	0.528	(0.521, 0.534)
Event 3/Vehicle 1	Rough	0.162	(0.160, 0.165)
Event 3/Vehicle 2	Rough	0.2203	(0.2190, 0.2218)
Event 3/Vehicle 3	Rough	0.4321	(0.4214, 0.4429)
Event 4/Vehicle 1	Rough	0.219	(0.218, 0.2201)
Event 4/Vehicle 2	Rough	0.246	(0.244, 0.248)
Event 4/Vehicle 3	Rough	0.283	(0.282, 0.285)

Table 4-5: Statistical Results for Friction Coefficient Estimates

Road Surface	Mean	Standard Dev.	Variance	Lowest Value
Smooth Paved	0.4852	0.0312	0.009754	0.448
Rough	0.226	0.0442	0.00196	0.162

4.3 Uncertainty Modeling

In section 4.1, sensitivity analysis was applied to evaluate how the warning parameter is affected by specific parameter changes. In the previous section, an error analysis was performed to identify the error in the friction parameter categorized to smooth and rough road surfaces. In this section, the results of the previous analyses are combined together to evaluate the error in the warning parameter using experimental uncertainty analysis. Experimental uncertainty analysis is typically used to evaluate how errors propagate from measured quantities to derived quantities within a modeled relationship. The relationships can be complex so approximate solutions are often used which can provide preliminary but useful results. One such linearized approximation approach is obtained by application of the total differential. This approach is useful in estimating the absolute

error, Δf , of a function, f , based on the errors of the measured variables, Δx , Δy , ... as illustrated by the following conceptual equation.

$$\Delta f = f_x \Delta x + f_y \Delta y + \dots \quad (4-13)$$

Where f_x is the first derivative of f with respect to the parameter x . Equation (4-13) provides the sensitivity relationship between changes in x to changes in f . The error in a quantity, $\{\Delta x, \Delta y, \dots\}$, is the specific magnitude of change evaluated and is usually given as the standard deviation (σ). This is evaluated for all specific parameters that compose f . The parameters are assumed independent, and the sum of these components represents a unique combination of errors that is a conservative, worst-case scenario. That is, if each of the measured quantities are represented by a statistical distribution, then the total differential represents the unique case where the extreme values of the distributions are simultaneously in effect. The probability of these simultaneous occurrences of “worst parameters” actually occurring is nearly zero and so represents a worst-case scenario. In reality, the mean error of the derived quantity is expected to be smaller than the output of this estimation. This worst-case analysis is sometimes referred as the 1-norm or absolute change.

To analyze the warning parameter uncertainty, or Δw , the algorithm parameters listed in Table 4-1 represent the measured quantities, and the warning parameter is considered the derived quantity. The conceptual equation (4-13) of the total differential then becomes equation (4-14).

$$\begin{aligned} \Delta w = & \frac{\partial w}{\partial d} \Delta d + \frac{\partial w}{\partial v} \Delta v + \frac{\partial w}{\partial v_{rel}} \Delta v_{rel} + \frac{\partial w}{\partial \alpha} \Delta \alpha + \frac{\partial w}{\partial \tau} \Delta \tau + \frac{\partial w}{\partial d_0} \Delta d_0 + \frac{\partial w}{\partial \mu} \Delta \mu \\ & + \frac{\partial w}{\partial K_{driver}} \Delta K_{driver} \end{aligned} \quad (4-14)$$

The previously developed MATLAB code that was used to generate the test profile data and analyze the relative sensitivity was used again here to evaluate the uncertainty of the warning parameter. As an initial inspection, a +/- one standard deviation, $\Delta \sigma$, on all the parameters was evaluated which corresponds to a relative error of 68%. Equation (4-14) becomes the following equation.

$$\Delta w = \frac{\partial w}{\partial d} * 68\% * d + \frac{\partial w}{\partial v} * 68\% * v + \frac{\partial w}{\partial v_{rel}} * 68\% * v_{rel} + \frac{\partial w}{\partial \alpha} * 68\% * \alpha + \frac{\partial w}{\partial \tau} * 68\% * \tau + \frac{\partial w}{\partial d_0} * 68\% * d_0 + \frac{\partial w}{\partial \mu} * 68\% * \mu + \frac{\partial w}{\partial K_{driver}} * 68\% * K_{driver} \quad (4-15)$$

The top graph in Figure 4-7 shows Δw . The bottom graph compares the calculated warning parameter with the warning parameter uncertainty included, $w + \Delta w$, and without, w , (ideal system). The $w + \Delta w$ response appears almost twice the amount of w . The dotted horizontal redline corresponds to a warning parameter value of one, and represents the alert threshold between safe and unsafe driving conditions. Ideally an alert is presented to the driver when the warning parameter value becomes less than one and increases in severity as it approaches zero. In Figure 4-7, the resulting trend of $w + \Delta w$ crosses the alert threshold later in time compared to w trend (approximately 2 seconds after). This highlights the important fact that noisy measurement errors, which are inherent in a “real” system, increase warning parameter uncertainty and negatively affect the warning parameter response by delaying the collision alert to the driver.

Figure 4-8 compares the individual parameter errors to identify which terms dominate as the vehicles near collision. Based on this visual inspection, the dominant parameters appear to be the inter-vehicle distance, d , subject vehicle velocity, v , and time delay, τ . Vehicle deceleration, α , and relative vehicle velocity, v_{rel} , increase slightly but never become greater than other parameters. Finally the friction coefficient, μ , and driver tuning, K_{driver} parameters appear to contribute relatively very little error. One effect common to all parameters is that the errors reduce as the distance between vehicles decrease and a collision is about to occur.

The 1σ error on all the parameters is a very conservative scenario and is nearly impossible of existing in real conditions. An additional uncertainty analysis was processed using more appropriate absolute error values for the parameters, listed in Table 4-6, instead of the 1 sigma error values previously discussed. These values were selected within the following assumptions. The inter-vehicle

distance and system time delay delta magnitude values were adopted from Chaves' thesis [5]. Chaves previously conducted a similar uncertainty analysis of the warning parameter equation; however, his analysis focused on only two variables affecting the system: inter-vehicle distance (d), and system time delay (τ). (The results of his analysis are provided in section 8.2 of his thesis.) The error values selected for velocity were adopted from a paper published by David Bevly stating that a GPS receiver can provide three dimensional accuracy of velocity within an accuracy of 2-5 cm/s without differential corrections [32]. 5 cm/s was ultimately selected as it is the more cautious value. The error value for friction coefficient was selected as a result of the friction coefficient estimation analysis explained in the previous section, and more specifically the standard deviation for the rough road surface was selected since it produced the larger error. The absolute errors for the remaining parameters were set to zero because they are not considered measured quantities and either are tunable system parameters or calculated and therefore expected to have very minimal or at least controllable errors.

Figure 4-9 compares the previous Δw results between the two uncertainty analyses. Since the individual parameter errors are smaller compared to the 1σ deviation in the first analysis, Δw also decreased as expected. The warning parameter deltas were added to the ideal error free warning parameter and shown in Figure 4-10. The warning parameter with refined errors, $W+\Delta W(\Delta X)$, is closer to the ideal case than compared to the single standard deviation result, $W+\Delta W(\Delta \sigma)$. Table 4-7 summarizes the detection horizon achievable for each experiment and the percentage increase obtained over the worst case scenario which is the 1σ on all parameters simultaneously. Again, the detection horizon is considerably better for the refined case compared to the worst case. Logically the results also prove a system with a higher accuracy measurement sensor will provide a better detection horizon (earlier driver alert) than one containing poor measurements.

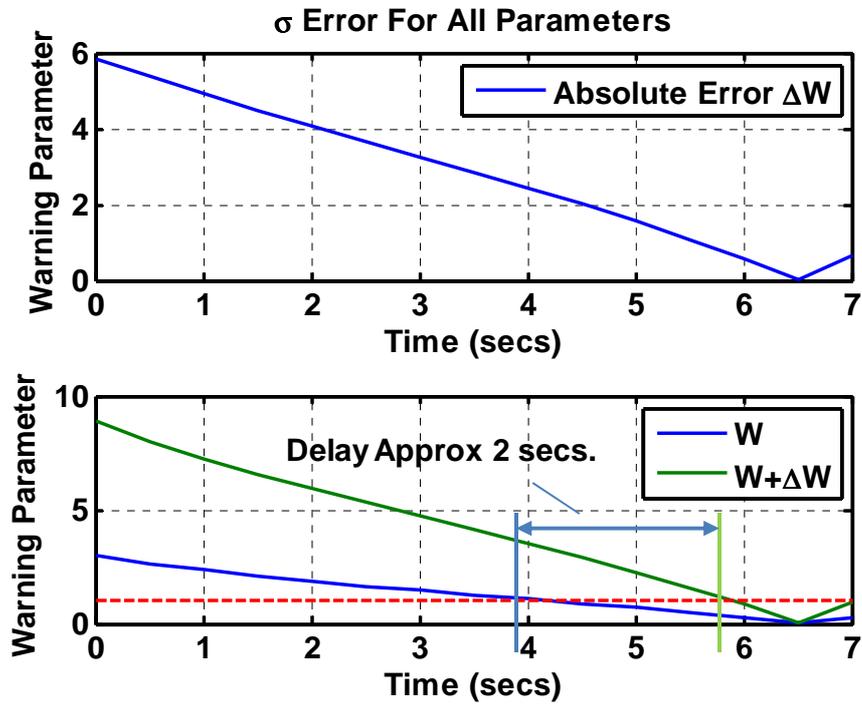


Figure 4-7: Sigma Uncertainty Analysis

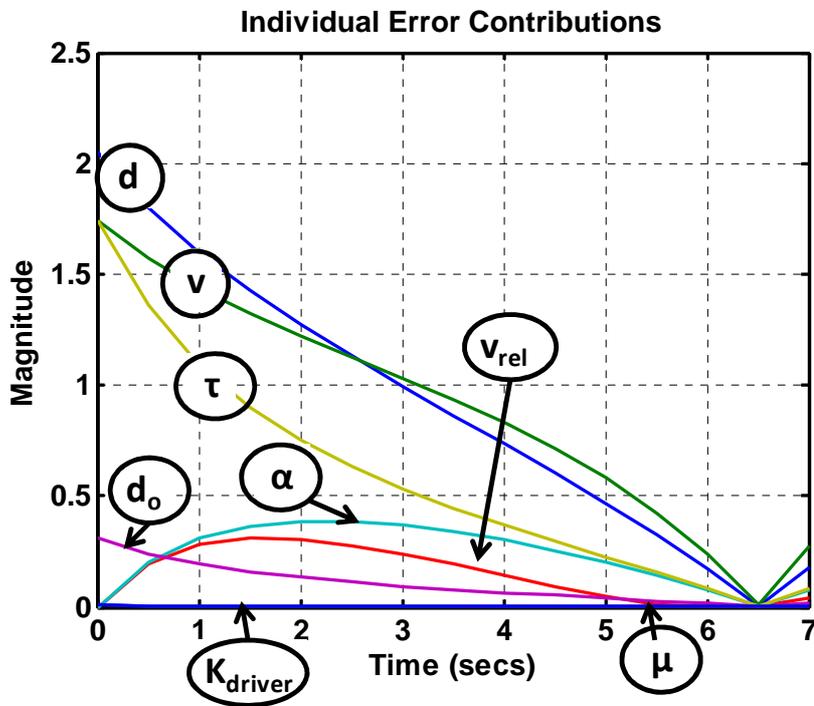


Figure 4-8: Individual Parameter Error Contributions

Table 4-6: “Realistic” Uncertainty Analysis

Parameter	Delta Value
Time Delay, τ , (second)	0.2
Vehicle Velocity, v , (m/s)	0.5
Relative Vehicle Velocity, v_{rel} , (m/s)	0
Inter-Vehicle Distance, d , (m)	0.7
Buffer Distance, d_0 , (m)	0
Friction Coefficient, μ ,	0.0442
Driver Adjustment, K_{driver} ,	0
Maximum Deceleration, a , (m/s ²)	0

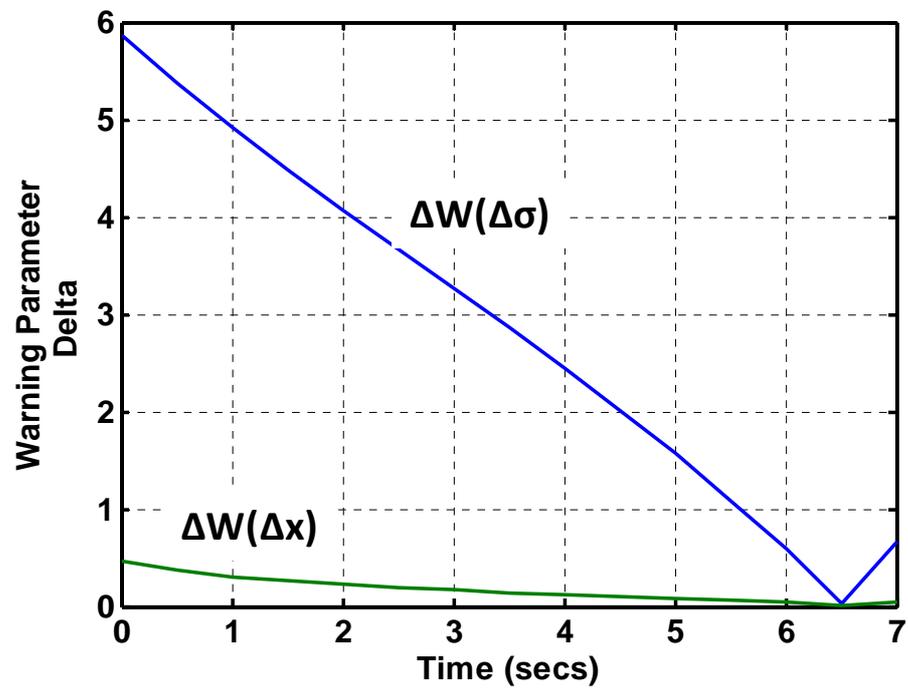


Figure 4-9: Warning Parameter Uncertainty Results

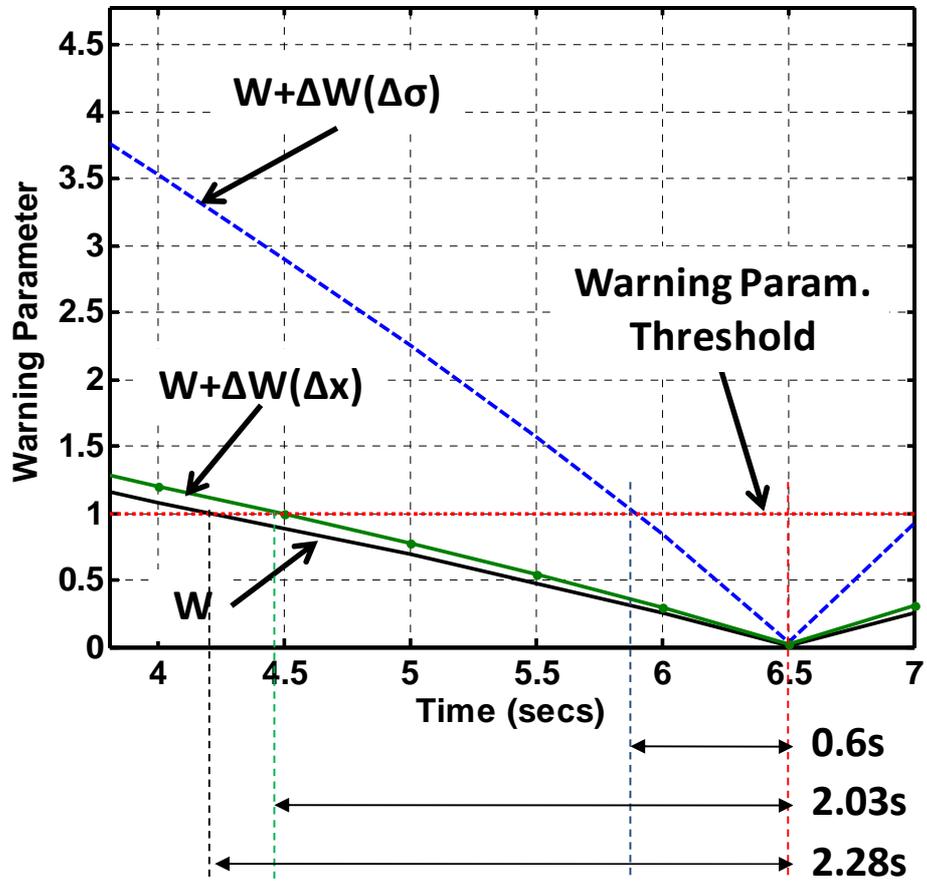


Figure 4-10: Uncertainty Analysis Results Comparing Effect of Error Magnitude

Table 4-7: Quantitative Detection Horizon Comparison

Experiment	Detection Horizon	Percent Increase
W	2.28 s	280%
$W + \Delta W(\Delta x)$	2.03 s	238%
$W + \Delta W(\Delta\sigma)$ (worst case)	0.6 s	N/A

4.4 Summary

In this chapter a analytical framework for evaluating uncertainty of the collision warning algorithm was first presented. The framework is simple and incorporates the use of sensitivity analysis to identify individual relative sensitivities of algorithm parameters to the output. The framework also relies on identification of parameter errors through appropriate methods. The relative sensitivity and parameter error information are combined in an uncertainty analysis to identify overall uncertainty behavior of the warning parameter. The results showed with measurement errors included, the warning parameter response is delayed compared to the ideal warning parameter response as illustrated in the top graph of Figure 4-7. Although the uncertainty analysis was performed using representative error values, these results are based on simulation. In the following chapters the errors of the inter-vehicle distance measurement is investigated in greater detail by first identifying stochastic error types in the GPS latitude and longitude measurements (Chapter 5) and then comparing the inter-vehicle distance measurement directly to ground truth (Chapter 6). A subsequent output of Chapter 6 is an estimate of inter-vehicle measurement error which is used to re-evaluate warning parameter uncertainty in Chapter 7.

Chapter 5

GPS DRIFT CHARACTERIZATION USING ALLAN VARIANCE ANALYSIS

A GPS-based collision detection system relies on GPS data as a primary measurement. As the previous chapter discussed, measurement error must be analyzed in order to assess system performance uncertainty. Chavez provides an overview of GPS technology and functionality in chapter 3 of his thesis, which also includes a description of common GPS measurement error sources [5]. A summary is provided here. GPS measurement errors typically fall into three categories: control segment errors, signal propagation errors, and receiver measurement errors. Errors in the control segment are associated with errors in the satellites such as offset in their position or onboard atomic clock bias. Signal propagation errors include all error sources that affect the transmitted GPS signal while in-flight from the satellites to the receiver. These primarily include atmospheric errors in the ionosphere and troposphere, which affect the speed of the signal and cause a skewed estimate of the range to the satellite.

The signal propagation, and control errors in particular, can often be corrected, as they are largely due to physical, deterministic processes. One solution to reduce control segment and signal propagation errors is Differential GPS (DGPS), which takes advantage of the fact that control segment and signal propagation errors are highly correlated and similar among nearby GPS receivers on the ground. Thus, a stationary GPS receiver located at a calibrated position can calculate the position error offset between the GPS inferred position and true position. This subsequently determines the amount of “correction” necessary for local conditions, which are then broadcasted to nearby GPS receivers to improve their accuracy. Only GPS receivers with DGPS compatibility are able to accept these corrections and correctly compensate their measurements. Some high-end DGPS compatible receivers can reduce position measurement accuracy from 10 meters to sub-meter levels.

The fact that control segment and signal propagation errors are correlated spatially and temporally provides an advantage for the GPS-based collision system of this work. That is, the error in relative distance between receivers is small compared to the individual position error of each receiver. In explanation, suppose there are two GPS receivers in a field, GPS A & GPS B, approximately 50 meters apart from each other in a field and the true position of each receiver is already known. If GPS A indicates a position 3m north of its true position, say due to ionosphere error, then GPS B is also likely to indicate a 3m north offset of its true position. This is because the ionosphere error will influence each GPS receiver the same amount. The ionosphere error could produce an increase in position error to 10m but again would affect both receivers the same amount. In any case the calculated relative distance between the receivers would remain the same even though individual position errors vary. This correlation of spatial error among receivers is well known but not clearly documented in regard to the noise models for mobile units that are correcting each other using simple differencing methods. This chapter examines position differences when nearby GPS receiver units are used. The assumption is that the primary error in the distance measurement between each receiver is largely independent of ionosphere severity and is only primarily influenced by receiver errors, which are the third classification of GPS errors.

GPS receiver manufacturers often provide a position rating for their products in terms of circular error probability (CEP). The Air Force Operational Test and Evaluation Center formally defines CEP as a circle, centered about the mean, whose boundary is expected to include 50% of the population within it when tabulated for very long time intervals [33]. The specifications for the San Jose embedded GPS modules used in this work specify a horizontal position accuracy of 3.3m CEP. This means that, for a given GPS position, the user can be 50% confident that they are within a 3.3m radius circle of that position. A smaller CEP value indicates a higher accuracy receiver. While CEP provides a good overall indication of GPS receiver accuracy, it does not provide any information about measurement stability or its degradation over time. For this, a more in-depth characterization is

required that can identify the specific error types including random bias. Identification of specific errors allows the development of appropriate noise models for improving the Kalman Filter implemented by Chaves for blending GPS and accelerometer data. Han et, al. performed a similar effort using Allan Variance (AVAR) analysis to determine the calibration model for a Kalman Filter fusing inertial sensors and GPS information, however they examined the stochastic errors of inertial sensors only and excluded GPS receivers [34]. This chapter presents the work accomplished for a drift analysis of the San Jose GPS receivers and ADXL 305 accelerometers using AVAR analysis.

5.1 Allan Variance A Review

To begin, an overview of Allen Variance (AVAR) analysis is first provided. Initially introduced in 1966 by Dr. David Allan for analyzing the noise characteristics of cesium beam frequency fluctuations for atomic clocks, AVAR analysis has become widely used and accepted as a preferred method for identifying stochastic noise processes. The National Institute of Standards and Technology (NIST) published a handbook for frequency stability analysis which includes a description of how to use and apply original AVAR, as well as modified versions for improved analysis [35]. The handbook also includes example datasets to test and verify modified AVAR algorithms or other stochastic noise modeling algorithms. The IEEE standard 647 specification also identifies AVAR as part of the standard test procedure of Single-Axis Laser Gyros [36]. Although originally developed to study the frequency stability of oscillators and clocks, the AVAR method is useful in general for characterizing the random processes of any data output and is selected here to characterize the stability of the GPS and accelerometer sensors by identifying their specific dominant noise types.

AVAR analysis focuses identification of primarily stochastic, or random, types of errors. Stochastic errors can be a mixture of several errors but the five most common sources are: quantization noise, white noise, random bias, random walk, and random ramp. Assuming the sources

are statistically independent, the AVAR is essentially the sum of the squares of the variances of the five common sources.

$$\sigma_y^2(\tau) = \sigma_Q^2 + \sigma_{Wh}^2 + \sigma_B^2 + \sigma_{Rw}^2 + \sigma_R^2 \quad (5-1)$$

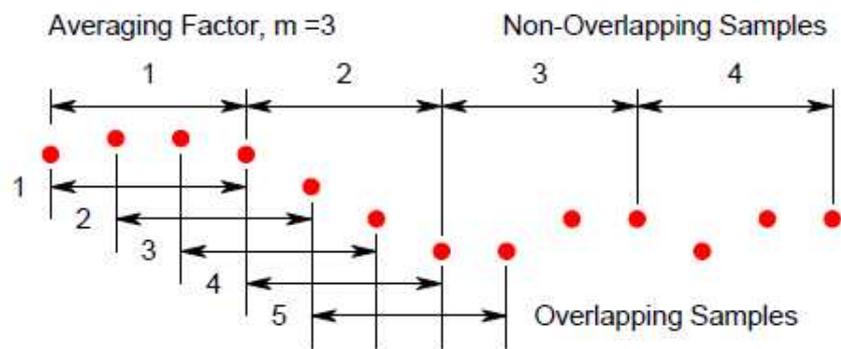
In Equation (5-1), $\sigma_y^2(\tau)$, is the AVAR, or total variance, result for a specific averaging time, τ , σ_Q^2 is the variance contribution by quantization, σ_{Wh}^2 is the variance contribution due to white noise, σ_B^2 is the variance contribution due to bias, σ_{Rw}^2 is the variance contribution due to random walk, and σ_R^2 is the variance contribution due to ramp noise.

An advantage of AVAR is that the computation is fairly simple and can be directly applied to time-series data collected from a stationary un-excited sensor. A stationary static measurement is preferred so only random processes influence changes in measurement. Given a sequence of measurements sampled at a constant rate, the data is first divided into clusters of a specific set size related to the averaging time, τ . For example, if data was collected at a rate of 50 Samples/second for a total duration of 100s and the selected τ was 0.1s then the size of each data cluster would be 5 samples. Next, the average is calculated for each cluster of grouped data. Then the difference is calculated between the calculated averages of successive clusters. The differences are then each squared, summed together, and finally multiplied by a scaling factor dependent on the number of clusters used. The formula for the original AVAR is defined in Equation (5-2) as:

$$\sigma_y^2(\tau) = \frac{1}{2(M-1)} \sum_{i=1}^{M-1} [y_{i+1} - y_i]^2 \quad (5-2)$$

Where $\sigma_y^2(\tau)$ is the calculated AVAR for a specific averaging time, τ ; y_i is the average value of the cluster i ; and M is the total number of clusters. The original AVAR specifies the division of the dataset into individual segments of length τ , so that the segment are non-overlapping; that is, each data sample is only used once in the calculation. For greater confidence, the NIST handbook for frequency stability analysis recommends overlapping the clusters. Figure 5-1 illustrates this

overlapping approach and compares it to the original AVAR analysis method. In the original approach each cluster contains a unique set of sample points and each subsequent cluster contains a different set of samples. This is illustrated in the top of Figure 5-1. The overlapping approach makes maximum use of the data by forming clusters which reuse samples across clusters. The reuse of samples is dependent on the cluster size and the amount of samples to shift by. In the bottom half of Figure 5-1 the cluster size is 3 samples and the shift is by 1 sample. Although the confidence of the estimate increases with this approach, the computational time also increases.



Source: NIST SP 1065 [35]

Figure 5-1: Non-Overlapping versus Overlapping Samples Illustration

The equation for the modified Overlapping Allan Variance is defined as

$$\sigma_y^2(\tau) = \frac{1}{2m^2(M - 2m + 1)} \sum_{j=1}^{M-2m+1} \sum_{i=j}^{j+m-1} [y_{i+1} - y_i]^2 \quad (5-3)$$

where \mathbf{m} is the averaging factor.

Once the calculation is performed for a specific averaging period, τ , it is repeated again for additional longer periods. Once AVAR values are obtained for several averaging periods, the square root of the AVAR values are calculated producing Allan Deviation (ADEV) values and plotted against the averaging periods. This produces a data plot similar to the sample diagram shown in Figure 5-2. This type of plot is commonly referred as Allan Deviation, or Sigma-Tau, plot. Generally

the log sigma and log tau is plotted to show the dependence of measurement stability as a function of averaging time.

The Allan Deviation plot is useful for identifying the five specific types of stochastic random processes described before by examining the slope of the curve. Provided the different noise sources are reasonably separated, different error terms will dominate different regions of τ .

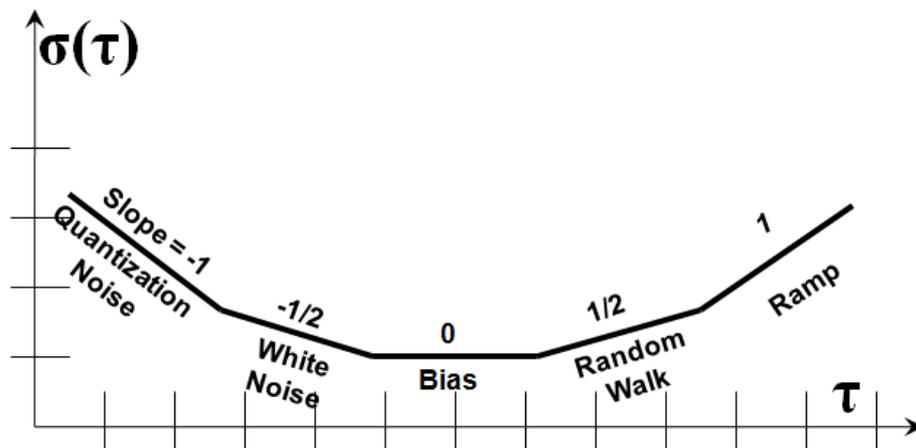


Figure 5-2: Typical Allan Deviation plot for a system

Table 5-1 provides a listing of the five common noise sources, the parameters of interest, their relation to the Allan Variance value and the region of τ at which they appear on the Allan Deviation plot. The magnitude of each noise parameter may be approximated using the plot. For example, to quantify the magnitude of quantization noise, one must first find where the slope of the Allan Deviation trend approximately equals -1. Then extends an imaginary line of the same slope until it intersects an extended line at $\tau = \sqrt{3}$. The y-value at this specific intersection then approximately represents the deviation due to quantization. This procedure is repeated again for the remaining noise types. This visual-fitting method provides a simple and quick way to identify the stochastic error sources but is limited in accuracy.

Table 5-1: Relation of Allan Variance and Noise Sources [37]

Noise types	Parameter of interest	Root Allan variance	Slope of root Allan variance plot
Quantization	Q	$\sigma_Q = \sqrt{3}Q/T$	-1
White noise	N	$\sigma_{wh} = N/\sqrt{T}$	-1/2
Random bias	B	$\sigma_R = B/0.6648$	0
Random walk	K	$\sigma_{Rw} = K\sqrt{T/3}$	+1/2
Ramp	R	$\sigma_R = RT/\sqrt{2}$	+1

The actual magnitude of each noise source can be also extracted via linear regression analysis. The goal of the linear regression analysis is to identify the coefficients for the following polynomial model through a least squares approach.

$$\sigma_y = A_1\tau^{-1} + A_2\tau^{-0.5} + A_3\tau^0 + A_4\tau^{0.5} + A_5\tau^1 \quad (5-4)$$

The resulting coefficients (A_1 through A_5) approximate the noise parameters of interest and are calculated using the following equations

$$Q = \frac{A_1}{\sqrt{3}} \quad (5-5)$$

$$N = A_2 \quad (5-6)$$

$$B = A_3 \times 0.6648 \quad (5-7)$$

$$K = A_4 \times \sqrt{3} \quad (5-8)$$

$$R = A_5 \times \sqrt{2} \quad (5-9)$$

By fitting a linear model to the root Allan Variance results, the coefficient results of the linear model are equal to the magnitude of the noise sources. This regression approach was used to quantify the noise source for the San Jose GPS receivers and ADXL 355 accelerometers. Results for the GPS receivers are presented first.

5.2 Experiment Setup

To evaluate the error characteristics of the prototype hardware, three prototype detection sensor packages were tested in the experiment. They are in fact the same three prototypes that were used in the Yuma feasibility testing that has been previously mentioned. All three units were inside an office building within approximately 3ft of a GPS repeater device. A GPS repeater receives GPS signals from the satellites at an antenna externally mounted outside the building, and then directly re-transmits the signals internally with some amplification. It is commonly used in tunnels to provide GPS functionality since the satellite constellation is not direct line of sight available. The units were placed inside nearby the GPS repeater station for more accurate comparison across hardware since the GPS receivers would be commonly receiving the same signal from the repeater station. Using the GPS repeater should also omit any control or signal propagation errors. The GPS data from the San Jose FV-M8 receiver and acceleration data from the ADXL335 3-axis accelerometer was collected, at a rate of 5 Hz, for approximately 24 hours to capture the full motion of the GPS satellite constellation from April 13, 2010 to April 14, 2010. The data collection process on all three units was started approximately at 4:00 pm. Two of the three systems successfully collected data continuously for the full duration of the experiment. In the following results discussion these two units are identified as System 9366 and System 9584. The data collection process for the third unit, identified as System 9365, first collected 11.4 hrs worth of data before shutting down due to a software error, but was re-started to collect an additional 7.22 hrs worth of data.

Figure 5-3 and Figure 5-4 show the raw latitude and longitude measurements, respectively, collected by all three systems. The figures identify the brief gap in the data for System 9365 related to the period at which the data collection process for this system was temporarily offline. Overall, the plots indicate fairly good agreement of the raw measurements across the three systems, although there is a brief period of non-correlation also identified in the figures. The cause of this measurement disagreement is unclear and appears to be random. For numerical quantification of the agreement, the

correlation between the datasets was calculated using Pearson's correlation formula. The results of are provided in

Table 5-2 and Table 5-3 respectively.

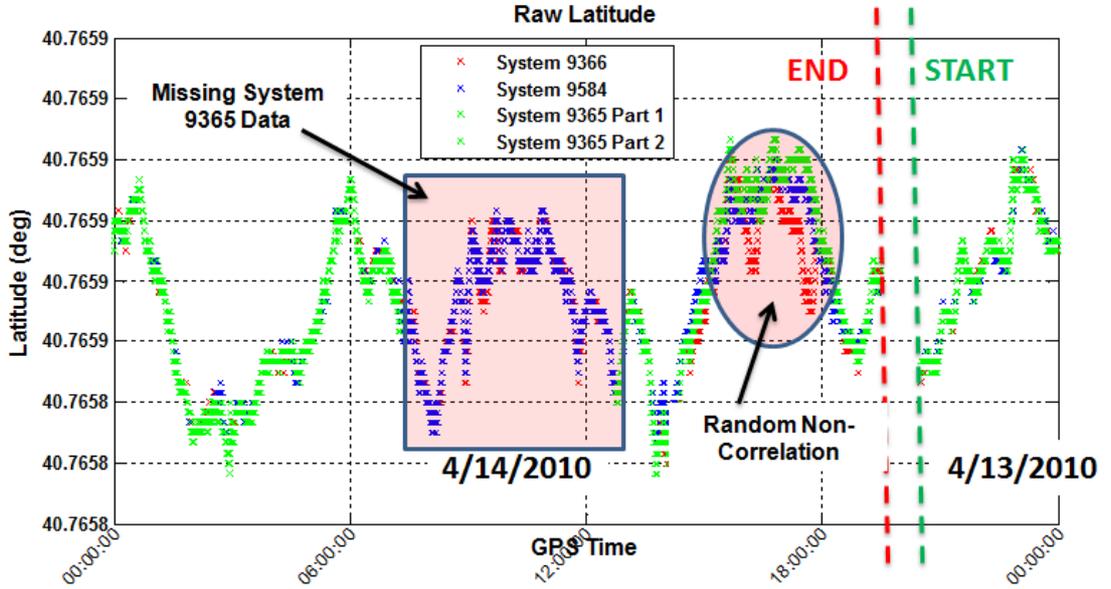


Figure 5-3: Comparison of Raw Latitude Measurements across Systems

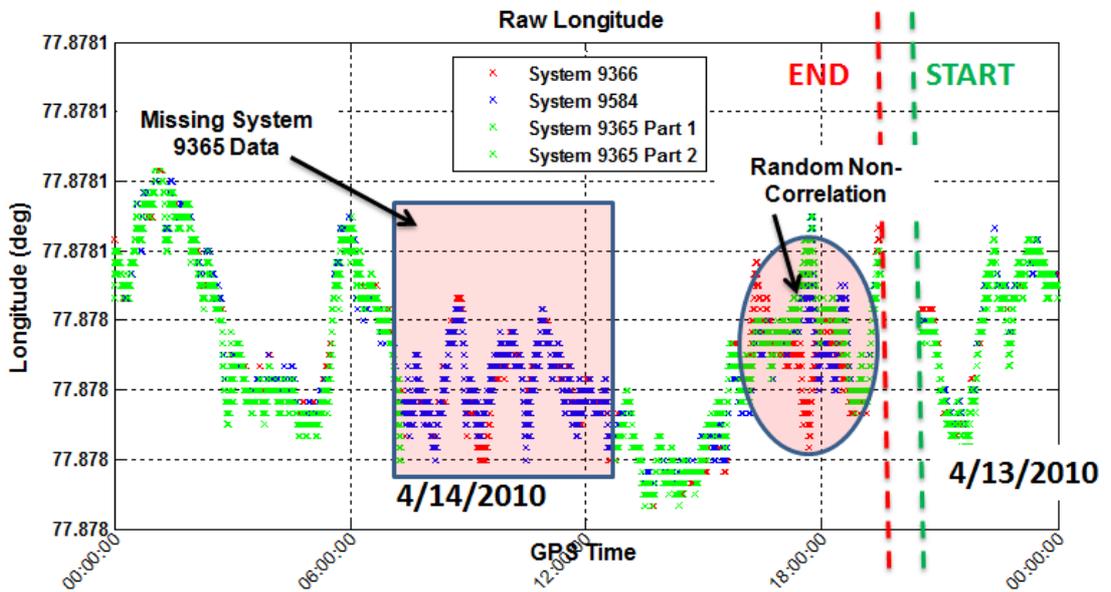


Figure 5-4: Comparison of Raw Longitude Measurements across Systems

Table 5-2: Latitude Cross Correlation Results

System	9366	9584	9365
9366	1	0.972	0.965
9684	0.972	1	0.989
9365	0.965	0.989	1

Table 5-3: Longitude Cross Correlation Results

System	9366	9584	9365
9366	1	0.963	0.947
9684	0.963	1	0.986
9365	0.947	0.986	1

The raw latitude and longitude measurements were translated to the East-North-Up (ENU) coordinate reference frame to better illustrate the change in position in units of meters. The position deviation from a mean position is shown in a scatter plot (Figure 5-5) using the ENU reference frame. The FV-M8 circular rated CEP specification of 3.3m is also shown in this figure. Although the CEP represents 50% percentile for coordinate fixes, all of the data from the three receivers lies within the CEP circle, indicating much higher than expected accuracy. On average, the deviation in east and north is 0.73m and 1.10m respectively. This high accuracy could be due to the use of a GPS repeater station mitigating other GPS errors effects such as multipath. To determine if this is the reason a similar experiment performed outside thereby eliminating the necessary use of the GPS repeater is recommended for future work.

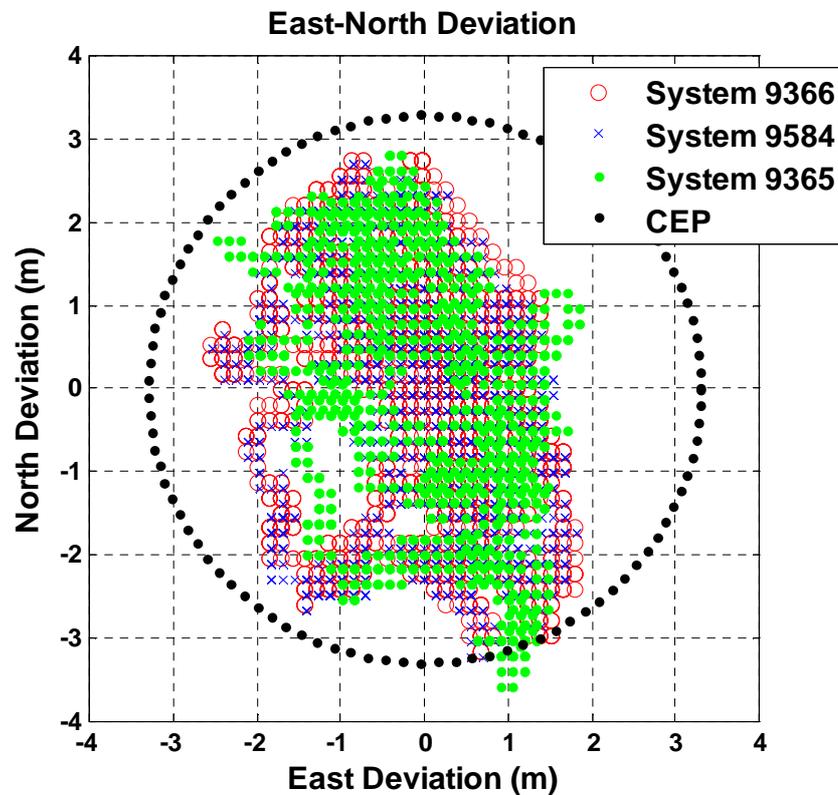


Figure 5-5: Drift Data Deviation in East-North Reference Frame

5.3 Allan Variance Analysis and Results

From visual inspection of the raw latitude and longitude trends in Figure 5-3 and Figure 5-4, it appears that random walk is the dominant stochastic noise source. AVAR analysis was applied to confirm this hypothesis. The data was processed using MATLAB and publicly available AVAR code (allan.m) from the Mathworks website [38]. This specific version (Version 3.0) includes functionality to calculate AVAR using the overlapping method. As mentioned before the overlapped version of Allan Variance produces higher confidence results. Before applying to the GPS datasets, the “allan.m” code was tested and verified on data provided in the NIST published Handbook of Frequency Stability Analysis. Additional code was developed and added to the MATLAB starter code to determine the noise parameter magnitudes using linear regression analysis. This code is provided in Appendix C.

5.3.1 San Jose FV-M8 GPS Allan Variance Analysis

The application of the Allan Variance analysis is first discussed in detail for System 9366 GPS data. The raw latitude measurement is again shown in Figure 5-6 to better illustrate the random drift in the data. The AVAR results for the latitude measurement are shown in Figure 5-7 on an Allan Deviation plot. The result of the linear regression analysis is also shown with the additional model produced curve shown in red. The correlation between the experimental data and model is 0.9995. From visual inspection, the positive slope of both the AVAR results and fitted model suggest that the dominant noise types are random walk and/or ramp. Using the resulting coefficients from the linear regression model and Equations (5-5) through (5-9) above, the calculated magnitude of the five noise variables are listed in Table 5-4.

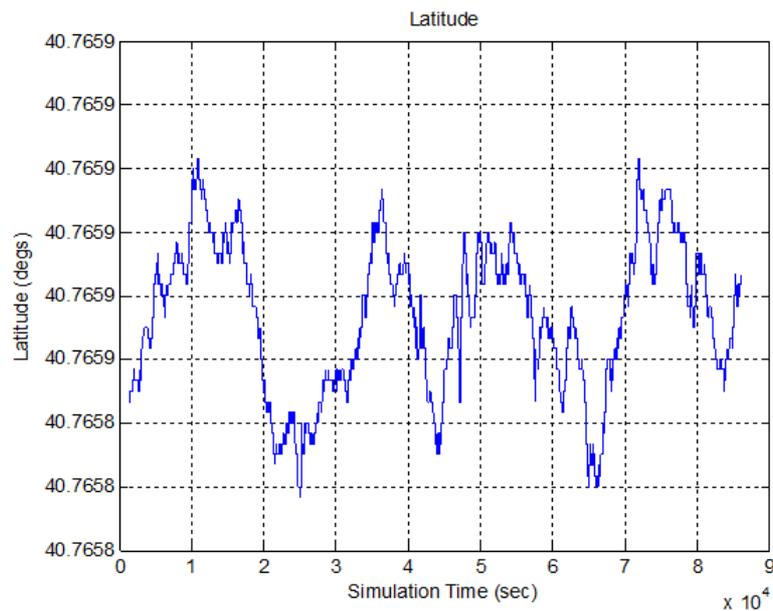


Figure 5-6: System 9366 Raw Latitude Measurement

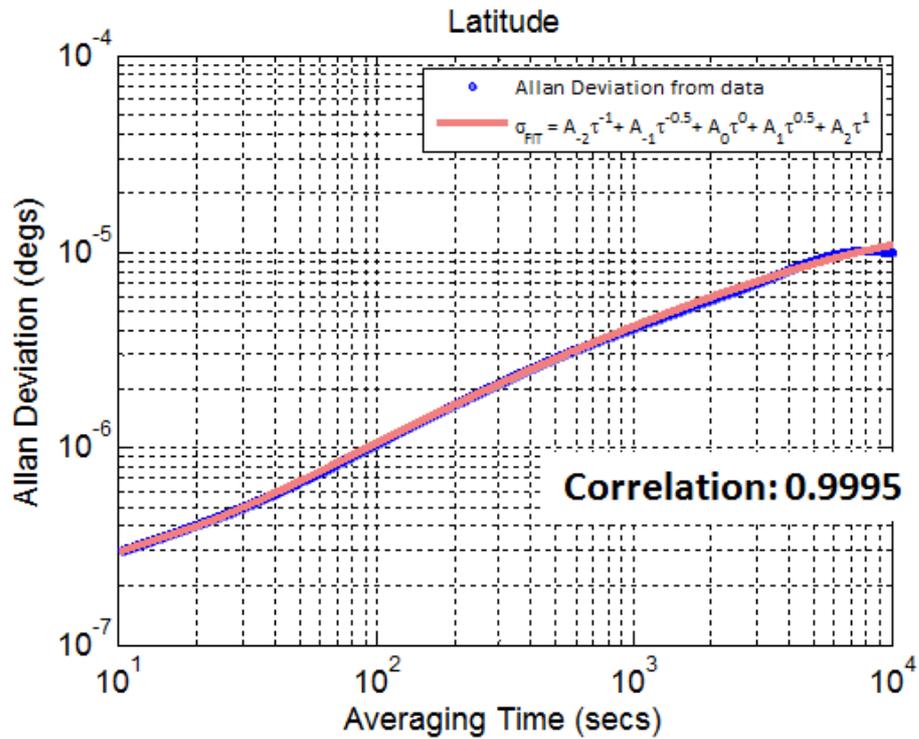


Figure 5-7: System 9366 Latitude Measurement Allan Deviation Plot

Table 5-4: System 9366 Latitude Noise Parameter Magnitudes

Noise Type	Magnitude
Quantization (<i>degs.sec</i>)	-2.749E-6
White Noise (<i>degs</i> \sqrt{sec})	3.916e-6
Bias (<i>degs</i>)	-6.838e-7
Random Walk (<i>degs</i> / \sqrt{sec})	3.129e-7
Ramp (<i>degs/secs</i>)	-8.89e-10

To better understand how each noise type individually contributes to the overall deviation the respective magnitudes from Table 5-4 were substituted into the appropriate noise equations 5-5 through 5-9 to calculate the noise coefficient terms A_1 through A_5 . These coefficients were then substituted into the root Allan Variance equation of (5-4), one at each time, and iterated through τ from 1 to 10000s to generate the result of noise type's behavior as a function of τ . Figure 5-8 compares the resulting trends of each noise type. Deviations at small averaging times are of primary

concern for the GPS-based collision system since the system is not able to update with high frequency. Closer inspection of the results at an interval of averaging times from 1 to 300 seconds shows the deviation positively increasing due to quantization (blue) and random walk (cyan) while negatively decreasing as a result of white noise (green), and ramp (purple). The bias noise type (red) steadily appears across all averaging times. The results are in general agreement with the theory of stochastic noise processes. Quantization and white noise tend to dominate at short averaging times and decay as averaging time increases. In contrast, as averaging time increases, the deviation due to random walk and ramp noise increases. Since these two noise types have specific impact on position fixes it would be particularly concerning if these noise types were large and began increasing at shorter averaging times where they could not be corrected. To demonstrate the influence at longer averaging times, the same results in Figure 5-8 are shown in Figure 5-9 except plotted on logarithmic scales. In this perspective the deviation contribution from the first three noise types is significantly diminished compared to random walk and ramp noise types. Although both random walk and ramp exist, the increasing trend of random walk increases deviation as averaging time grows and therefore corrupts the measurement. The decreasing trend of the ramp noise type suggests that the longer averages reduce the deviation thereby “correcting” the measurement. In reality this would mean that a random stochastic noise process is improving the measurement which is not realistically possible. For greater accuracy in the quantification of random walk noise, the linear regression analysis was applied again but instead ignoring the modeling of all terms except random walk. The resulting model fit curve is shown in Figure 5-10 along with the correlation agreement between the experimental data and model and magnitude of the random walk noise term. The influence of random walk noise is significant at longer τ values, however for the GPS collision system this error could be corrected by other measurements.

This analysis process was repeated for the raw latitude measurements of the two remaining systems. Figure 5-11 compares the AVAR result for all three systems, and the results are in

agreement. System 9584 and System 9365 exhibit positive slopes suggesting random walk as the dominant noise process. For both systems, full error models were first generated to confirm random walk as the primary stochastic noise type. Then, the linear regression analysis was applied again to model only random walk for greater accuracy.

Table 5-5 compares the final random walk noise parameter calculated using the simplified single noise term model. For all three receivers the degree of correlation between the model fit curve and experimental data is high. The random walk values obtained are also very similar suggesting that a single error model could be used to describe all three receivers and perhaps any San Jose FV-M8 receiver.

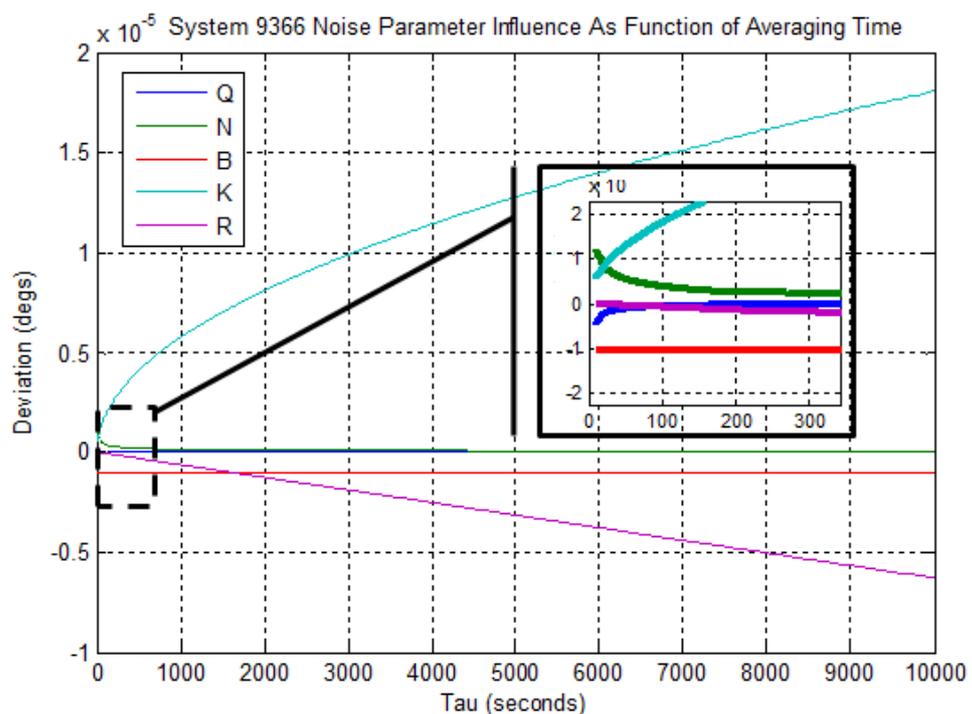


Figure 5-8: System 9366 Individual Noise Parameter Influence

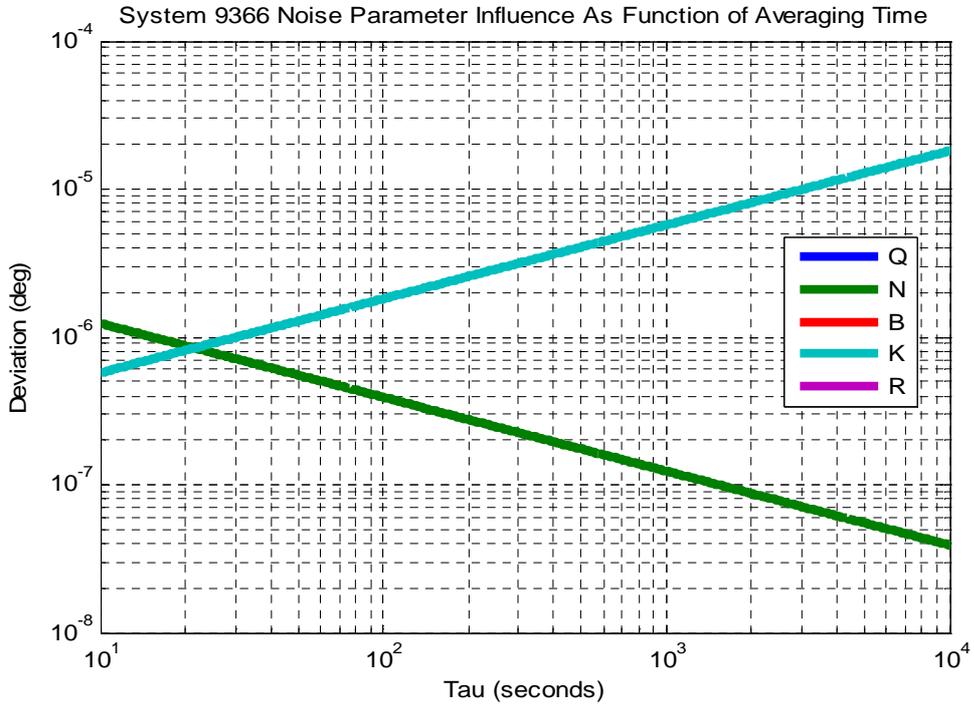


Figure 5-9: Dominant Noise Types at Longer Averaging Times

Table 5-5: Summary of Random Walk Noise Parameter for Latitude Measurement

System	Correlation	Random Walk (<i>degs/sec</i>)
9366	0.9926	1.893e-7
9584	0.994	1.834e-7
9365	0.997	1.760e-7

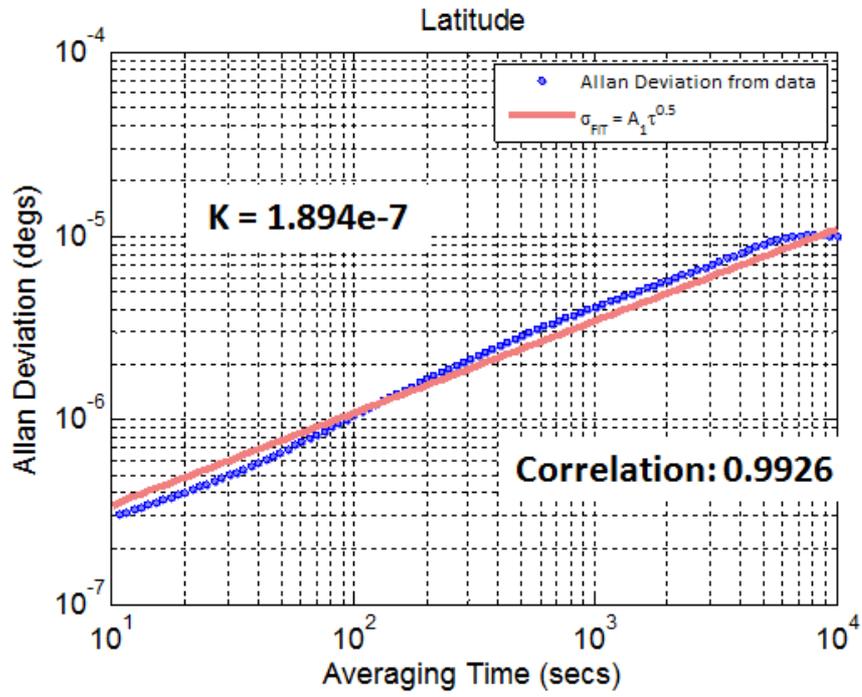


Figure 5-10: Random Walk Model

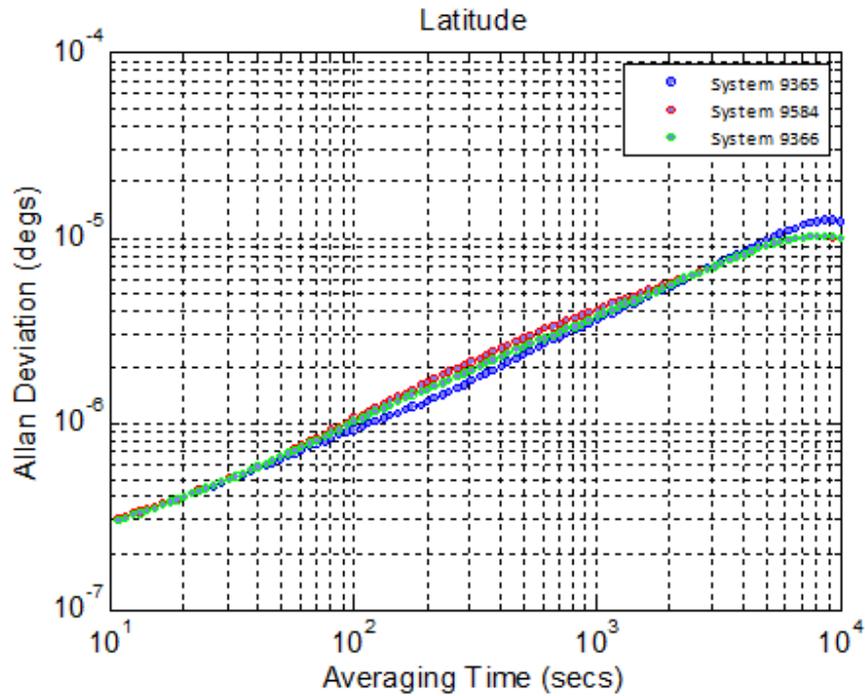


Figure 5-11: Comparison of All Allan Variance Results for Latitude Measurement

In similar fashion, the AVAR analysis was then applied to the longitude measurements from all three systems, and similar results were obtained. Figure 5-12 compares the longitude AVAR results on the Allan Deviation plot. Again, from visual inspection, the deviation is large at longer averaging times and continues to grow. Again for each longitude measurement full error models were first generated to characterize the influence of all five noise terms. After confirming random walk as the major noise source, the single noise term linear regression analysis was applied. The results of the regression analysis and calculation of the magnitudes of the noise parameters is provided in Table 5-6 below. Again the correlation values are high and the resulting noise values are very similar across the three receivers.

Table 5-6: Summary of Random Walk Noise Parameters for Longitude Measurement

System	Correlation	Random Walk (<i>degs/sec</i>)
9366	0.9973	1.82e-7
9584	0.9823	1.83e-7
9365	0.9933	2.00e-7

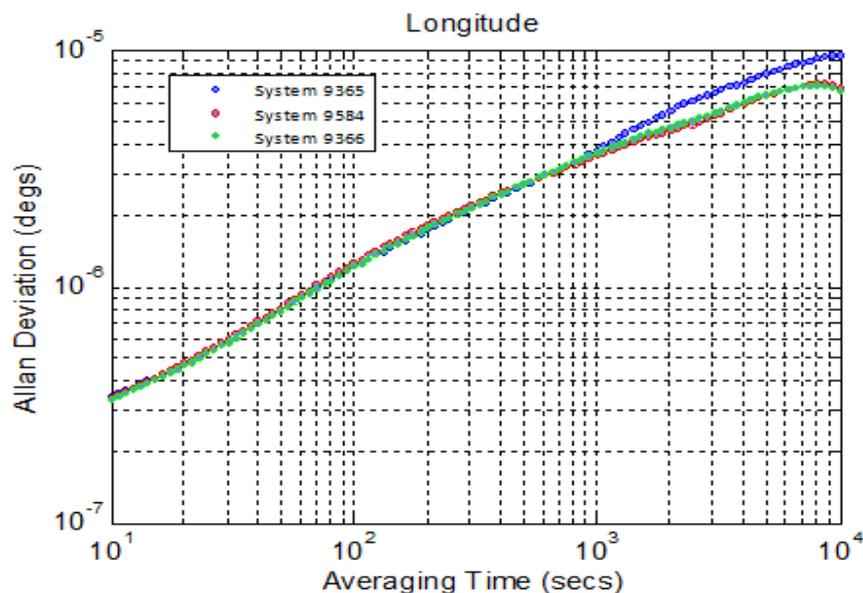


Figure 5-12: Comparison of All Allan Variance Results for Longitude Measurement

Although the magnitudes of the previous results appear very small, this is a result of the very small changes in the raw measurements which is in units of degrees. Specifically the changes in the raw measurements were observed on the order of $1e-6$. To evaluate the drift on a more realistic length scale, the Allan Variance analysis was also applied to the converted east and north deviations to obtain units of meters. The Allan Deviation plots for the East and North changes are shown in Figure 5-13 and Figure 5-14 respectively. Again the dominant noise type that appears from visual inspection of the plots still demonstrates random walk. On this scale the drift error is on the order of centimeters.

Table 5-7: Summary of Random Walk Noise Parameters for East Deviation Estimate

System	Correlation	Random Walk (<i>m/sec</i>)
9366	0.9791	0.0155
9584	0.9838	0.0156
9365	0.9939	0.0169

Table 5-8: Summary of Random Walk Noise Parameters for North Deviation Estimate

System	Correlation	Random Walk (<i>m/sec</i>)
9366	0.9937	0.021
9584	0.9949	0.0203
9365	0.9972	0.0196

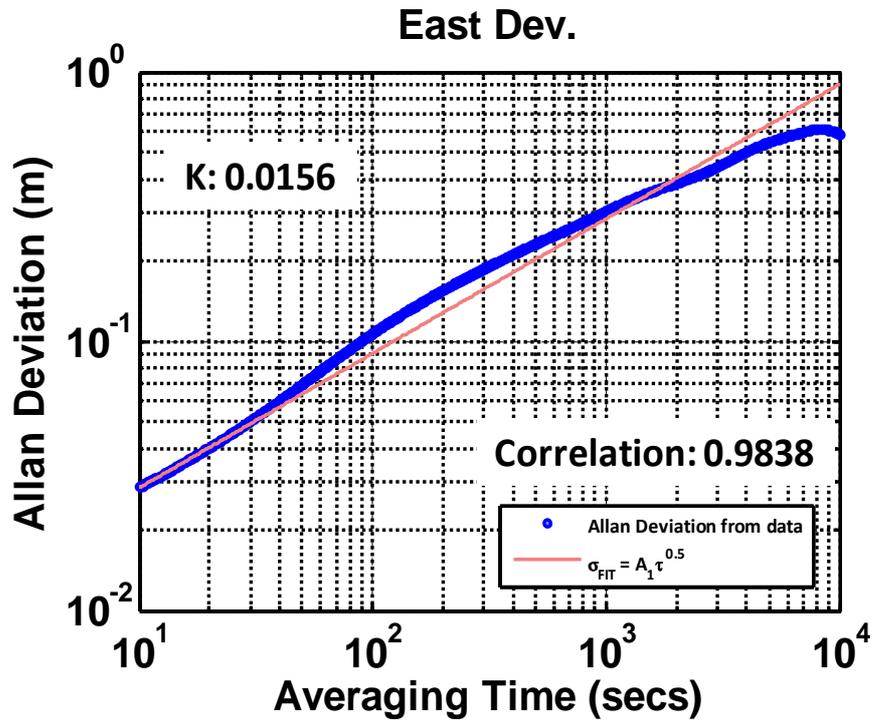


Figure 5-13: System 9584 East Deviation Allan Deviation Plot

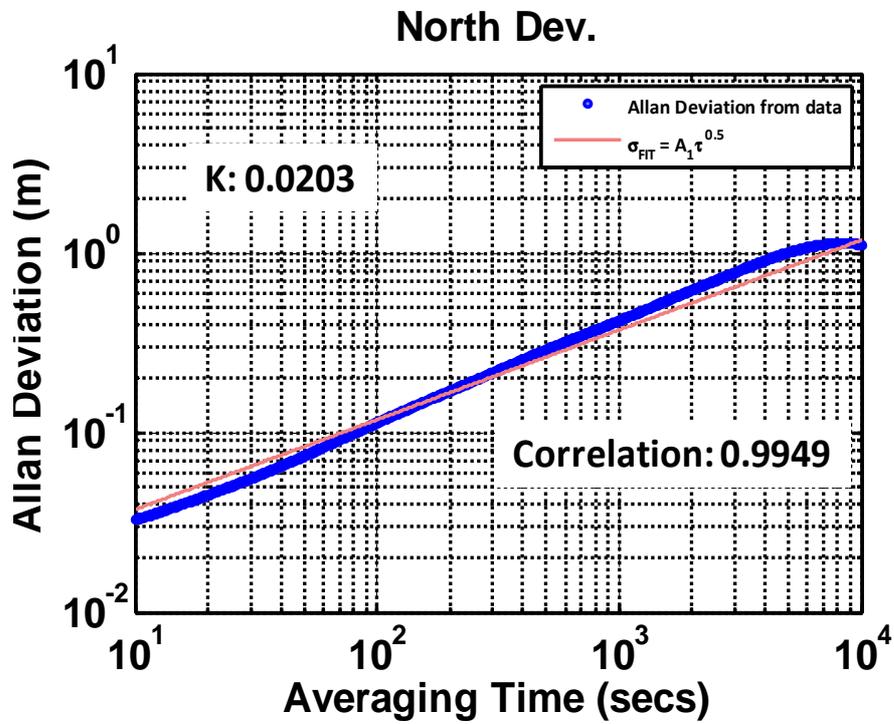


Figure 5-14: System 9584 North Deviation Allan Deviation Plot

5.3.2 ADXL-335 3-Axis Accelerometer Allan Variance Analysis

The AVAR analysis was also applied to the accelerometer data also collected during the 24 hour collection period. Figure 5-15 shows the raw measurements collected by System 9366. Based on visual inspection of the raw data it appears short term averaging time noise processes, such as quantization and white noise, are dominant. In contrast to the GPS data, there is no random walk easily detectable in the raw data.

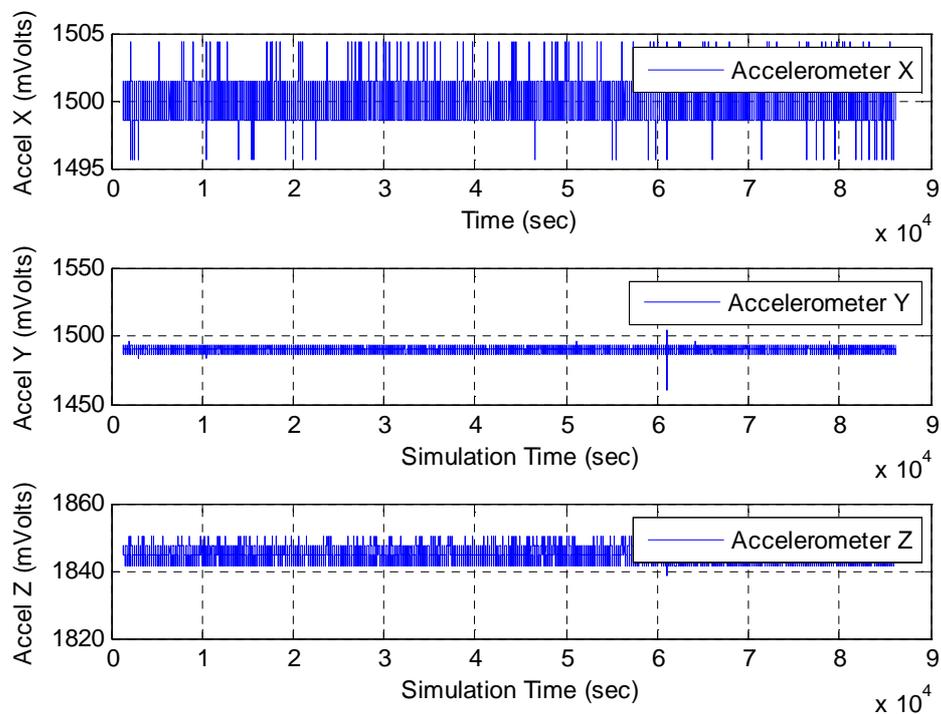


Figure 5-15: System 9366 Raw Accelerometer Data

The AVAR algorithm was applied next using the same time averaging interval (10 to 10000 seconds) as used for the GPS data. The resulting Allan Deviation plots and full error model for the acceleration x direction is shown in Figure 5-16. Similar results were obtained for the y and z directions. Unlike the GPS results, the Allan Deviation trend appears to exhibit all noise types. However, the calculations of the noise parameters from the coefficients of the linear regression model suggest quantization and bias as the dominant noise terms. Table 5-9 lists the noise parameters

magnitudes obtained for the x, y, and z directions using system 9366 collected data. Negative coefficients appear again for two of the noise types, white noise and random walk. Obtaining negative coefficients suggests the addition of that noise type reduces the overall variance of the sum of noise contributions. Or, this indicates that there are underlying contributors to the noise (such as dynamics of a filter) that are not being modeled. Or, it could indicate that the error in regression fitting is large relative to the coefficient of a particular noise model, e.g. that the noise is so small that the coefficient may seem negative just due to fitting error. The negative coefficients obtained are small and so it most likely that this last effect is occurring. Thus, these noise components are ignored hereafter along with the ramp noise term due to its small magnitude.

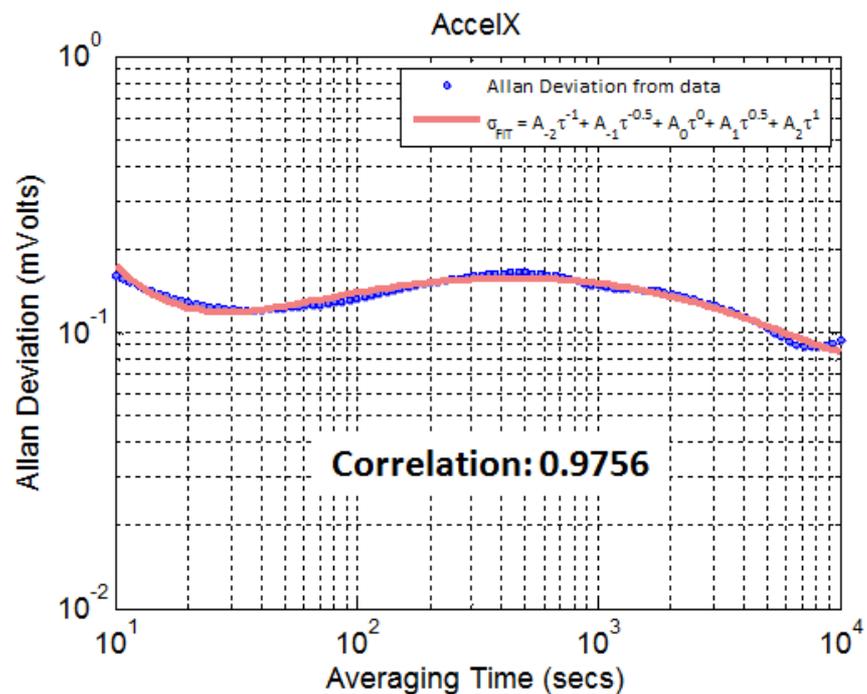


Figure 5-16: System 9366 Acceleration X Allan Deviation and Full Error Model Results

Table 5-9: System 9366 Accelerometer Full Error Model Noise Parameter Magnitudes

Noise Type	X	Y	Z
Model Fit Correlation	0.9756	0.9967	0.9460
Quantization (<i>mVolts.sec</i>)	2.042	0.351	2.038
White Noise (<i>mVolts\sqrt{sec}</i>)	-1.399	-0.044	-1.568
Bias (<i>mVolts</i>)	0.181	0.022	0.231
Random Walk (<i>mVolts/\sqrt{sec}</i>)	-0.0053	-0.001	-0.006
Ramp (<i>mVolts/secs</i>)	1e-5	4e-6	1e-5

To increase the accuracy of the quantization and bias noise noise terms, the AVAR analysis was re-applied but on the interval of 1 to 100 where these terms are dominant. The previously discussed “invalid” noise terms; quantization, white noise, and ramp, were also omitted from the subsequent linear regression analysis. Results of each sensor are compared in Figure 5-17, Figure 5-18, and Figure 5-19 grouped by sensing direction. The results from applying the linear regression analysis and extracting the quantization and bias noise terms are presented in Table 5-10, Table 5-11, Table 5-12 for x, y, and z axes respectively. The figures indicate a noticeable “offset shift” although the underlying shape of each curve appear similar especially in the range of less than 10 seconds. This is important to note as the GPS collision framework operates in the range of less than a second. It is therefore important to understand and determine the errors for very short averaging periods. Also because the underlying shape of each is similar and the difference in results only appears to be a shift, an appropriate noise model can still be developed and a scalar term can be incorporated to compensate for the shift.

The root cause of the shift was not investigated here but it is obviously related to some sort of variance among the sensors or interfacing hardware. It could possibly be due to errors or variances encountered during building the prototype GPS units. The placement and orientation of the sensors

was not precisely controlled and so there is likely some deviations in sensor orientation. Or it could be due to variances in the manufacturing tolerances of the sensors themselves. Particularly the breakout boards which interface the core sensor to the microcontroller contain a low pass filter, and although the production of the core sensor may be tightly controlled the breakout boards and subsequently the filter may not.

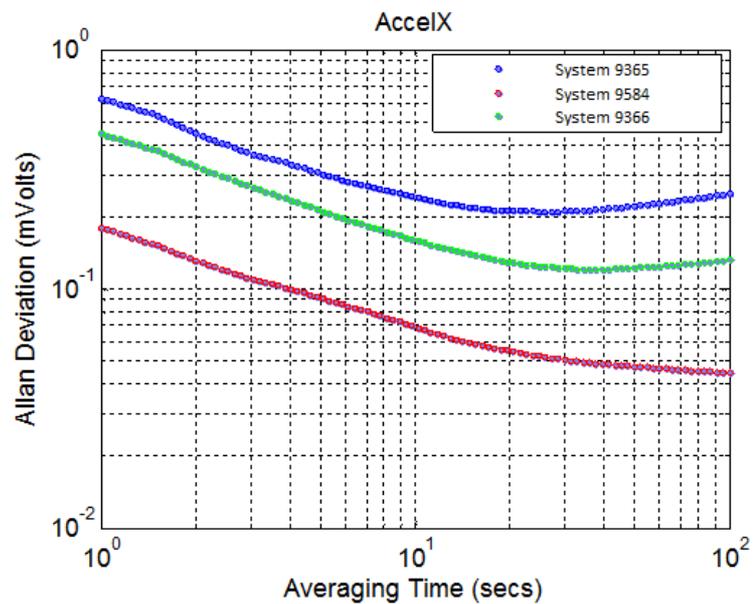


Figure 5-17: Comparison of All Three Systems Acceleration X Allan Variance Results

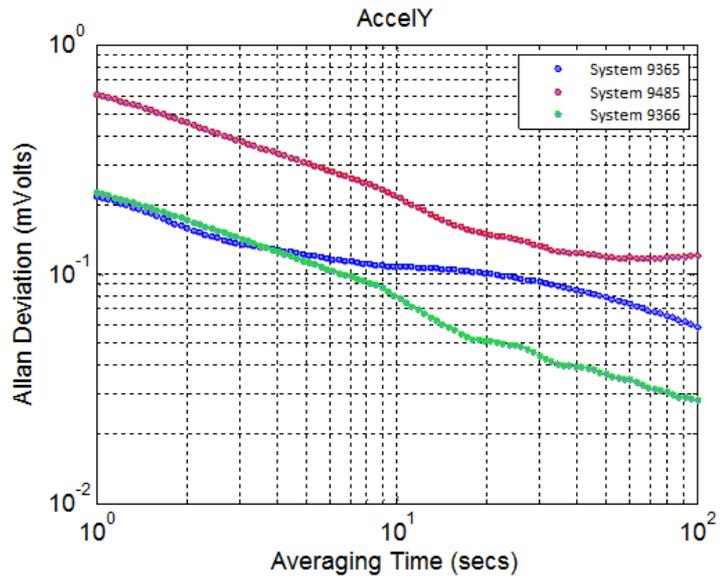


Figure 5-18: Comparison of All Three Systems Acceleration Y Allan Variance Results

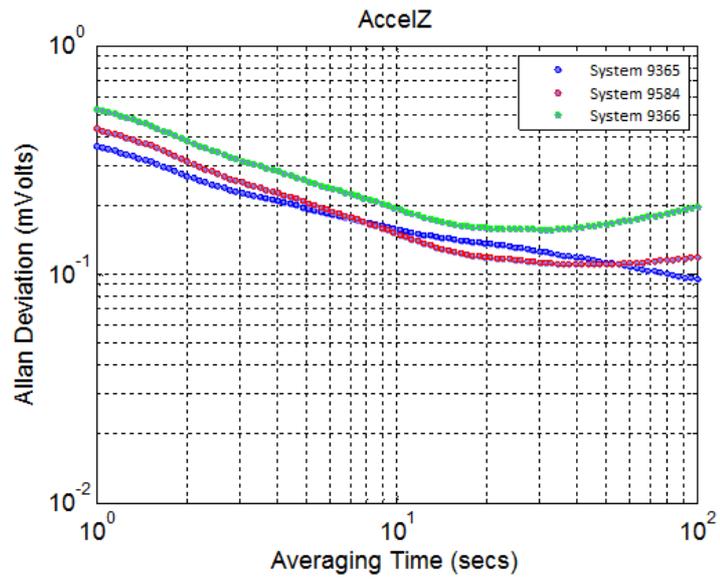


Figure 5-19: Comparison of All Three Systems Acceleration Z Allan Variance Results

Table 5-10: Identified Acceleration X Noise Coefficients for All Three Systems

System	Correlation	Quantization (mVolts.sec)	Bias (mVolts)
9366	0.989	0.231	0.0775
9584	0.982	0.099	0.0308
9365	0.991	0.268	0.136

Table 5-11: Identified Acceleration Y Noise Coefficients for All Three Systems

System	Correlation	Quantization (mVolts.sec)	Bias (mVolts)
9366	0.96	0.165	0.0219
9584	0.97	0.0309	0.0775
9365	0.969	0.0976	0.0523

Table 5-12: Identified Acceleration Z Noise Coefficients for All Three Systems

System	Correlation	Quantization (mVolts.sec)	Bias (mVolts)
9366	0.9883	0.2538	0.1024
9584	0.9886	0.233	0.0703
9365	0.9819	0.185	0.0743

5.4 Summary

This chapter presented a method for characterizing noise models of the GPS-based collision hardware sensors. Specifically, Allan Variance analysis was applied to GPS latitude/longitude and 3-Axis accelerometer data to quantify the stochastic noise processes in these sensors. Based on visual inspection of Allan Deviation plots and linear regression analysis derived models, estimates of noise magnitudes were produced. The GPS data clearly exhibited random walk as the primary noise type. Since the measurements are the results of receiver built-in processing algorithms, the absence of faster noise processes is understandable.

In contrast the accelerometer measurements displayed quantization and bias as the dominant noise types. The magnitudes of these noise types differed between the sensors and it could be possibly

due to hardware differences encountered either during the manufacturing process, the assembly of the prototype systems, or the built-in low-pass filter but none of these are certain causes. Although the magnitudes differ the underlying noise characteristics appear similar especially at short averaging time intervals. This is an important finding towards development of fusion algorithm, such as Kalman filters, which could benefit the collision algorithm. Since the underlying characteristics are the same and the results are only “shifted” one specific noise model can be created but could be adjusted by a scalar component to handle the shift offset.

Chapter 6

EVALUATING INTER-VEHICLE DISTANCE MEASUREMENT

One of the more important input variables in the collision algorithm is the physical distance between the vehicles from the leader's rear bumper to the follower's front bumper (or inter-vehicle distance). Current commercial collision systems measure this input directly using forward scanning sensor technologies, such as RADAR or LIDAR. The GPS-based collision system infers the inter-vehicle measurement by calculating the distance between two GPS coordinate current positions of the vehicles as measured by the GPS receivers. The inherent horizontal position uncertainty of a GPS receiver raises the concern of whether the inter-vehicle distance measurement can be accurate enough for use in determining collision probability. To determine this, a comparison of GPS to ground truth data of inter-vehicle distance is necessary. This chapter presents the experiment performed to collect GPS based and LIDAR based inter-vehicle measurement data and the subsequent analysis performed to compare the accuracy of the two methods.

Previous studies have evaluated the performance of collision warning systems however, most of this existing literature focuses on driver response and algorithm performance. Lu et al. evaluated the technical limitations of a developed frontal collision warning system for transit buses [39]. Their paper provides information regarding a test setup and procedure for evaluating the limitations of LIDAR- and RADAR-based collision detection systems. In their study, they used a string pot position transducer to measure inter-vehicle distance for ground truth. Instead, in this study, the LIDAR sensor itself collects inter-vehicle distance and this measurement is assumed ground truth. The test plan for this analysis was also influenced by test methods conducted by Birdsong who evaluated the performance of ultrasonic, laser range finder, and radar sensors through static and dynamic tests [40]. Kamiya et al. outlined specific requirements for a collision system suggesting that a range accuracy of 1m is necessary [15]. The following analysis discusses if the GPS-based collision system is capable of meeting that requirement.

6.1 Test Setup and Procedures

Two vehicles were used in the experiment: a 1994 GMC Sierra 2500 pickup truck and a 2005 Subaru Legacy sedan car. In all the test procedures, the sedan followed behind the truck so the GPS-based collision system calculated the inter-vehicle distance and the rear-end collision probability of the sedan crashing into the truck. Figure 6-1 captures all the hardware equipment used for this study.

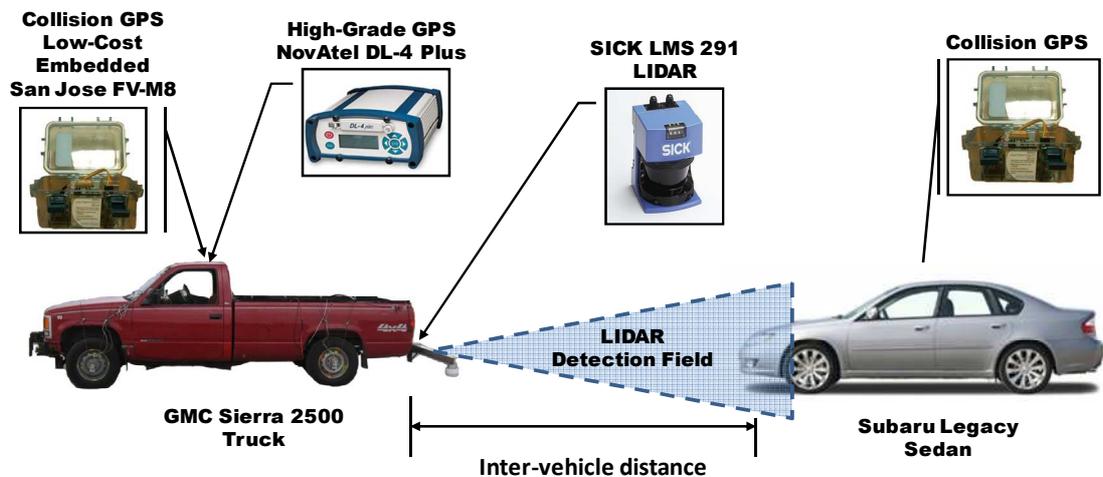


Figure 6-1: Test Vehicles and Equipment

A SICK LMS 291 LIDAR sensor system was mounted near the rear truck bumper with sensor scanning direction facing rear-wards to measure the distance (up-to 80 meters per sensor rated specifications) of the sedan's front bumper within a 0-180 hemispherical area. The SICK LIDAR sensor was configured to produce data at a rate of 37.5 Hz at 0.5 degree resolution. The acquisition and storage of the data from the sensor was handled by a MATLAB based GUI application, Figure 6-2, and the application was also designed to attach a GPS timestamp to the LIDAR data so measurement samples could be correlated to the GPS based measurements. The GPS timestamp is generated by a high-grade GPS receiver. The Novatel DL-4plus GPS receiver is able to provide precise GPS measurement with a rated position accuracy of 1.5m CEP (Circular Error Probability) at a collection frequency of up to 20 Hz.

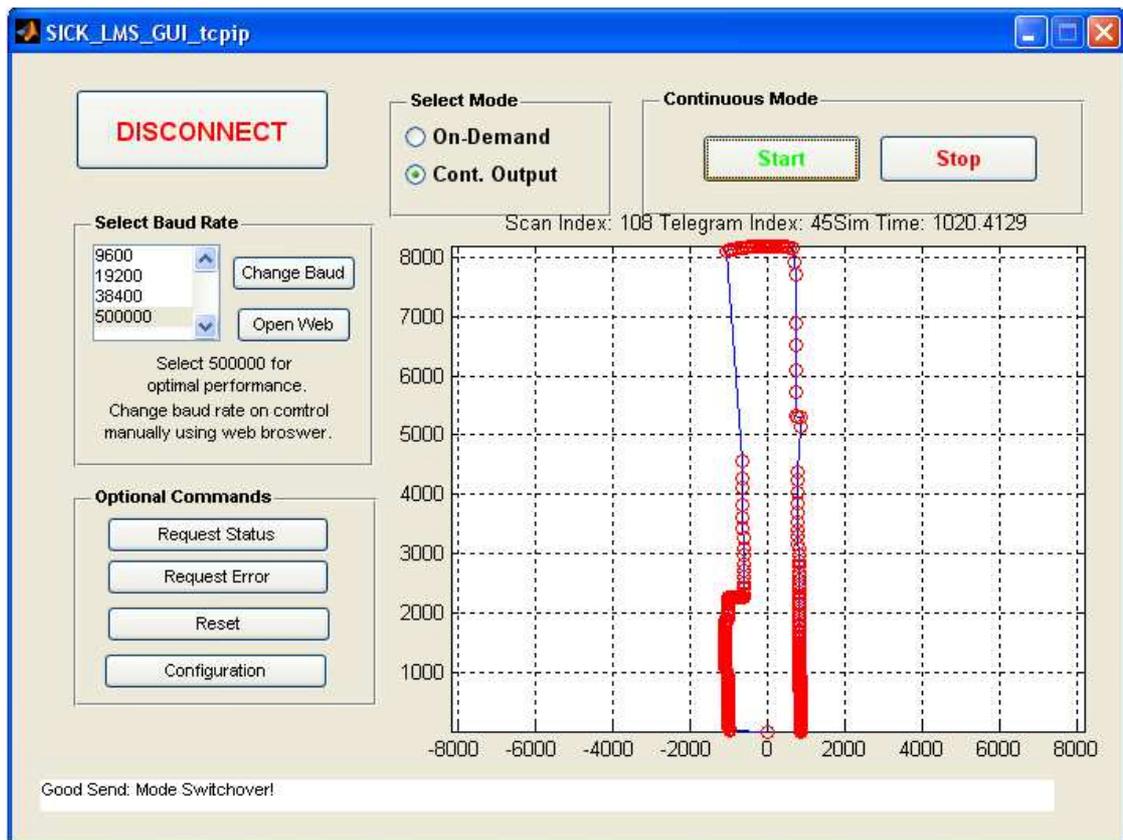


Figure 6-2: User Interface of LIDAR Data Acquisition Software

Two of the prototype hardware packages previously described in Chapter 3 were installed, one on each vehicle. One unit was mounted on the truck cabin roof, approximately 3.0 m distance from the LIDAR sensor. The other unit was mounted approximately in the center of the roof of the sedan. The distance of this unit from the sedan front bumper was approximately measured 2.36 meters.

The IEEE 1588 standard for precise time synchronization of multiple systems was programmed and implemented for this experiment to optimally synchronize the LIDAR, Novatel, and Collision GPS hardware systems. The GPS timestamp produced by the Novatel equipment was considered the master time and was shared wirelessly to the prototype GPS hardware units. Implementing this time synchronization protocol provides easier alignment of data in post processing

analysis and assures greater accuracy in making comparisons between the inter-vehicle distance ground truth and estimates. See Appendix F for more detailed information of this setup.

The test procedures were all performed at Penn State’s Pennsylvania Transportation Institute test track. The test track features a 5042 ft long oval shaped track ideal for safely conducting closed course experiments. An aerial photograph of the PTI test track facility with specific test feature highlight is shown in Figure 6-3. Performing the procedures at a test track provided a safe, controlled and reliable test environment free of other vehicles and obstacles. The collected data was post-processed using MATLAB software. For specific details regarding the data analysis procedure see Appendix H.

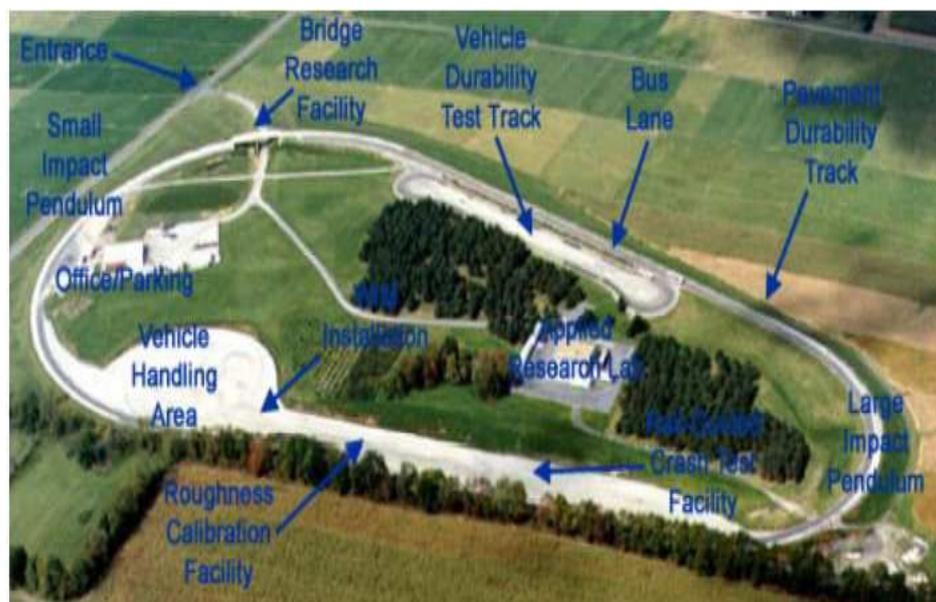


Figure 6-3: PTI Test Track Aerial Photograph

6.2 Error Analysis

The accuracy of the GPS-based inter-vehicle distance measurement compared to LIDAR was first evaluated. The GPS-based measurement was evaluated in two forms/estimates. The “hybrid” estimate was calculated using GPS information between the high-grade Novatel GPS unit mounted on the truck and the low-grade San Jose GPS receiver on the sedan. In contrast, the “real-time” estimate

only relies on the two low-grade GPS receivers from both vehicles and was the stored real-time calculation. This real-time estimate therefore includes wireless communication effects such as network latency. In addition to comparing GPS inter-vehicle spacing against LIDAR, the real-time estimate was compared to the hybrid spacing distance estimate to identify whether there are significant advantages to using a higher-cost, higher accuracy GPS system.

Figure 6-4 shows the measurements from two GPS-based inter-vehicle distance estimates versus LIDAR collected while the sedan follows the truck during a specific segment of the PTI test track. The absolute difference, or error, between the LIDAR and the two estimates is shown in Figure 6-5. From these two figures there are some noticeable outliers in the ground truth measurement. Although the sedan was driven directly behind the center (as best as possible) of the truck and as close as possible (approximately 20 meters) without risking safety, these outliers are primarily due to LIDAR target confusion with other roadside obstacles (vegetation, signs, buildings, etc) and also rough road surfaces causing small deviations in the LIDAR line of sight and subsequently causing missed detection of the sedan's front bumper. Also the test track includes a narrow bridge section that contains cement barriers bordering the track which also produced significant target confusion near the end of the lap. This degraded portion (errors induced by cement barriers) of data is not shown in Figure 6-4 or Figure 6-5 and was also omitted from the analysis.

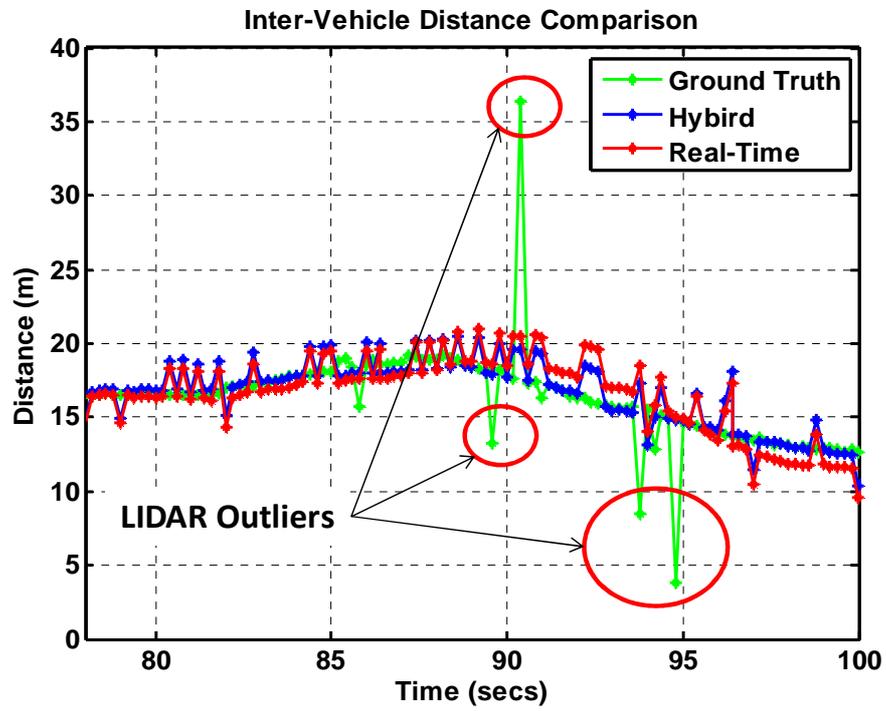


Figure 6-4: Estimates vs. Ground Truth, Lap Procedure

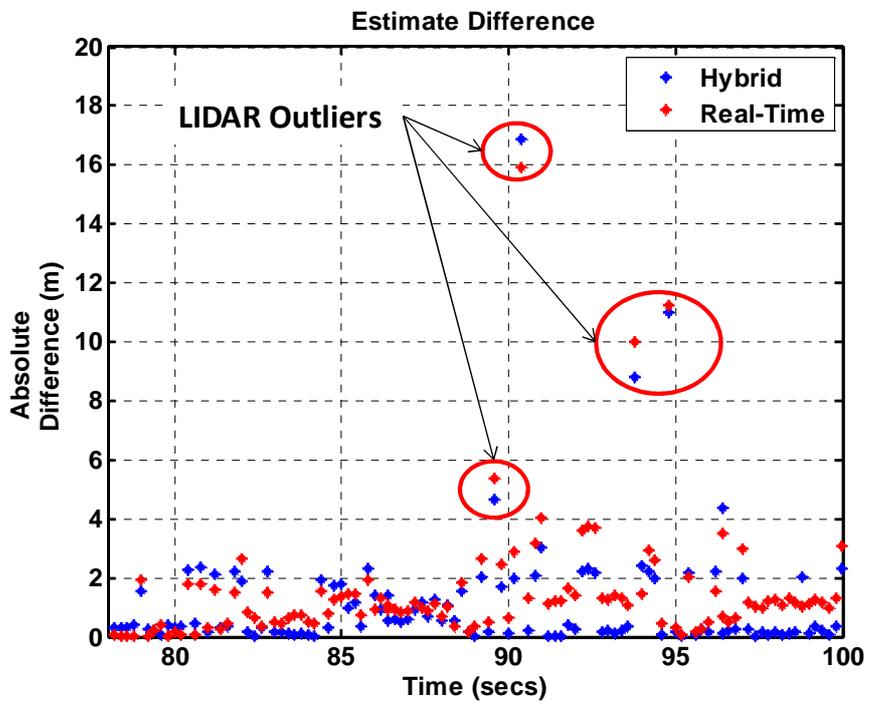


Figure 6-5: Absolute Error between Ground Truth and Estimates

Figure 6-6 and Figure 6-7 presents updates of the previous two figures with the LIDAR measurements outliers removed by first statistically identifying points outside of a 3 sigma deviation, then visually confirming the outliers by graphing the data, and finally manually removing those points from the data. All of this was accomplished using MATLAB.

There is also a theoretical error threshold shown in Figure 6-7 related to the CEP specification for the San Jose GPS receiver. As per specification, a single San Jose GPS receiver has a CEP rating of 3.3m. As mentioned before, the CEP accuracy rating is a 50% probability statistic valid for static measurements. This means that 50% of the GPS fixes will fall within a circular area with the radius equal to the specification. Logically a smaller CEP value indicates higher accuracy. Since two GPS receivers are used to estimate the inter-vehicle distance measurement, the, “worst case”, threshold here assumes the errors combine in a purely additive manner ($CEP_1 + CEP_2$) and so logically the threshold is drawn at 6.6m. It is then expected that 50% of the difference values in Figure 6-7 would be equally distributed above and below the thresholds. Instead, most of the samples fall significantly below the worst case threshold. Assuming the CEP specification for one receiver is accurate; it is concluded that the errors between the two GPS receivers are in fact not cumulative. This increase in *relative* positioning performance of the two GPS receivers is possibly due to a correlation of GPS errors among the receivers, similar to how DGPS systems work.

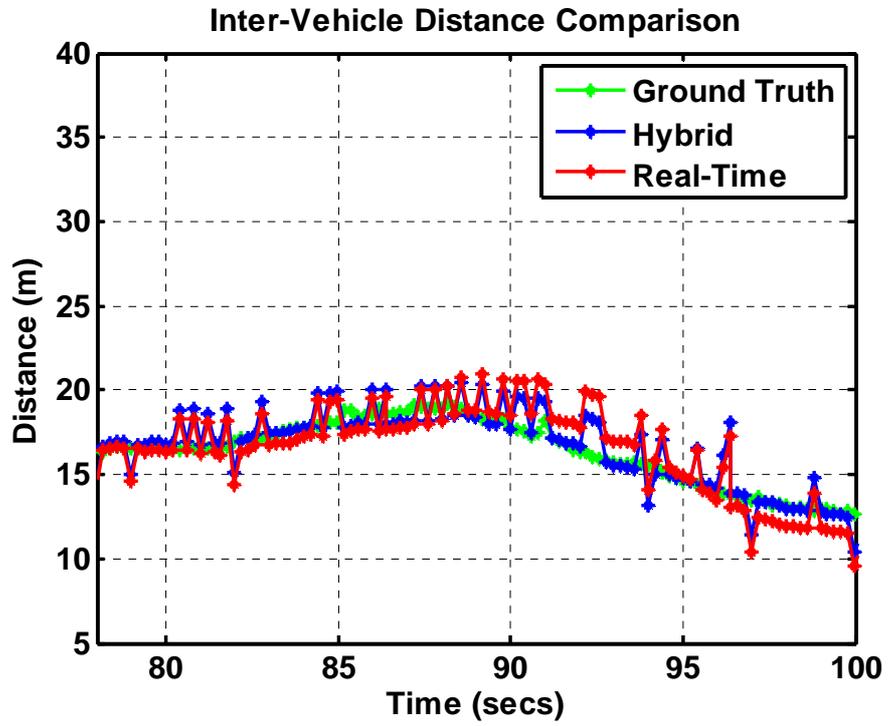


Figure 6-6: Estimates Vs. Ground Truth with Outliers Removed

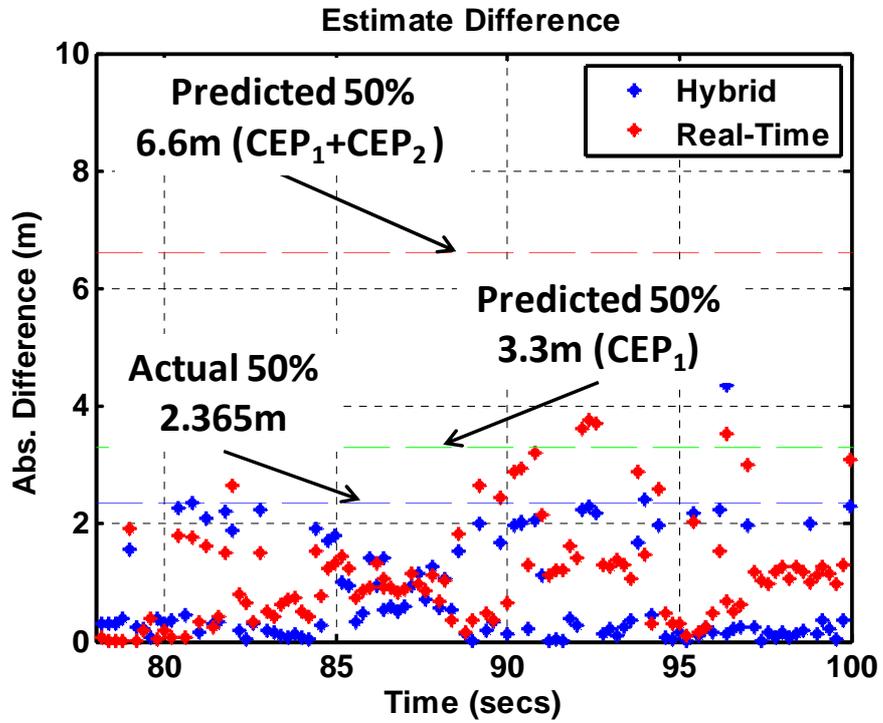


Figure 6-7: Absolute Error with Outliers Removed

Histograms of the hybrid and real-time error data are presented next in Figure 6-8 and Figure 6-9 respectively. The figures also show probability distribution results and inferred statistics. Since the absolute difference was previously evaluated, the half-normal probability distribution was selected for application. The half normal distribution is related to the standard normal distribution except it is appropriate for datasets where all data is positive, ($x \geq 0$), or only the magnitude of the data is recorded and sign is omitted, and the expected value is 0. The inferred statistical results suggest the error distribution for both hybrid and real-time methods is small. The statistical results indicate both the hybrid and real-time errors are lower than the CEP error specification for a single receiver and the previously discussed worst case scenario where CEP errors are cumulative. The variance statistic for both is also small showing the 95% confidence metric even falls within the single receiver CEP rating. It is mentioned that the error distribution and statistics is only slightly “worse” for the real-time dataset compared to the hybrid dataset. The mean error for the hybrid dataset is actually less than the 1m requirement as suggested by Kamiya. Recall that the hybrid dataset was produced from using a high-end GPS unit with a low-end GPS unit. The accuracy would then be expected to be better if two high-end GPS units were used. Although the mean error for the real-time dataset, which uses both low-cost GPS units, is over the 1m suggested requirement it is only over by 0.26m. The concern of whether GPS accuracy would be good enough for collision detection are answered by these results which suggests the accuracy is quite applicable.

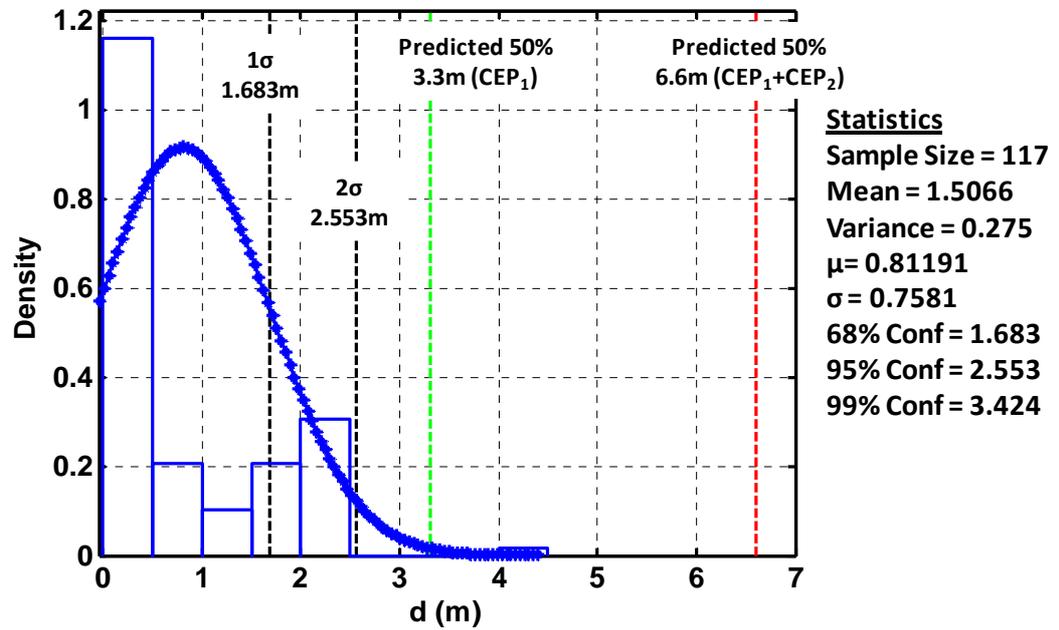


Figure 6-8: Hybrid Error Distribution and Statistics

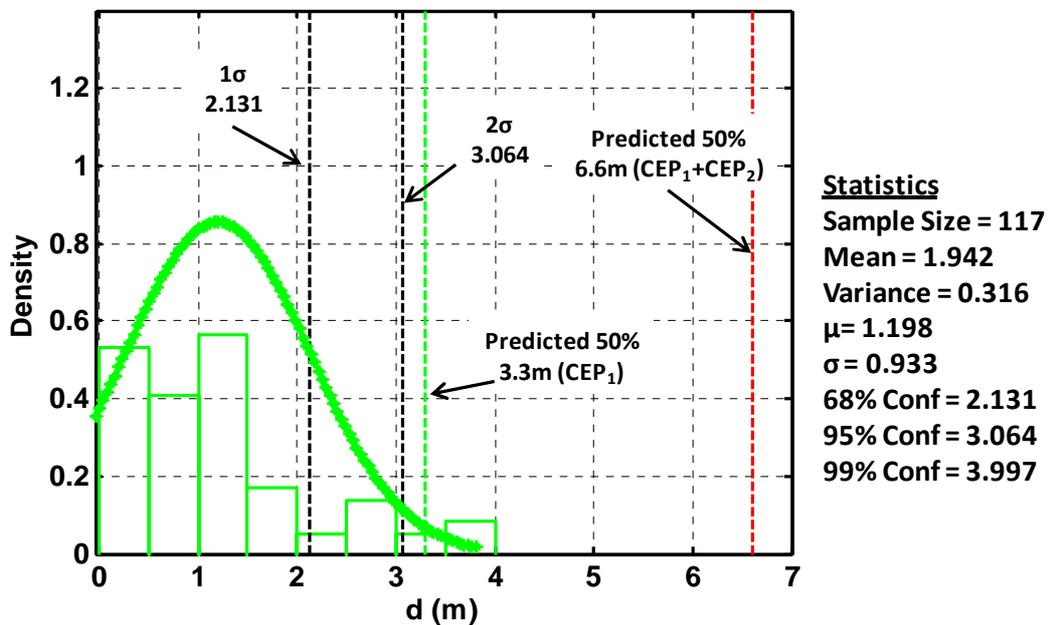


Figure 6-9: Real-Time Error Distribution and Statistics

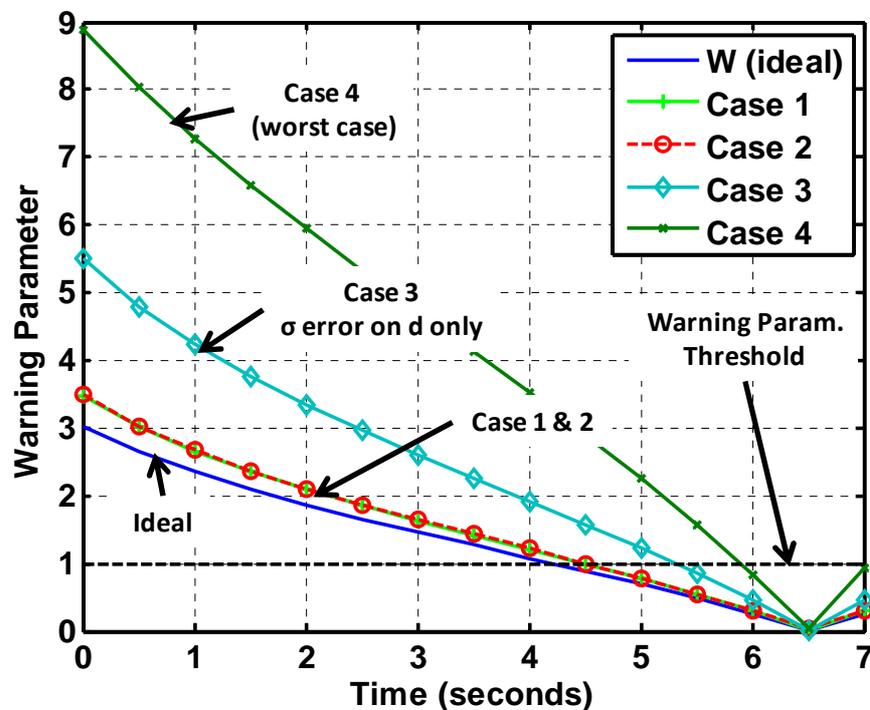
6.3 Revised Uncertainty Analysis

A significant outcome of the previous results was that this research gives numerical values for the estimated error associated with the GPS based inter-vehicle distance measurement. With this error more accurately quantified using actual experimental data; the uncertainty analysis first presented in section 4.3 is revisited to get a better estimate of warning parameter uncertainty.

Equation (4-14) and the delta values for each parameter listed in Table 4-6 were again used to calculate the warning parameter uncertainty, Δw . However, using the results of the previous inter-vehicle distance measurement analysis, only the delta value for d was changed to 1.198m. This represents a 71% error increase in d . For comparison, the result of this updated uncertainty analysis is shown in Figure 6-10 along with the two analyses previously presented in Chapter 4. Case 1 is the second analysis in Chapter 4 which uses the error values listed in Table 4-6. Case 4 is the first analysis of Chapter 4 where an error of 1σ was applied on all parameters. Recall this is the worst case scenario. Case 2 is the new uncertainty analysis result, using 1.198 meters for d . Comparing Case 1 and Case 2, there is barely a noticeable difference. The results almost appear the same; however, there is actually a difference between the two results which is shown in Figure 6-11. As highlighted in Figure 6-11, the increase in error from 0.7m (Case 1) to 1.198m (Case 2) for d resulted in a decrease in detection horizon of 0.02 seconds. This changes the original detection horizon of 2.03 seconds to 2.01s. This is a percentage decrease of only 1%. So even though the d error increased by 71%, the detection horizon only decreased by 1%. This suggests that the warning parameter is not highly sensitive to changes in inter-vehicle distance measurement errors.

To further support this claim, an additional uncertainty analysis (Case 3) was performed to evaluate the effect of a significant larger error on d compared to the other variables. For the Case 3 analysis, the error values listed in Table 4-6 were again used for all variables, except d . Each d value in the simulated data was changed by 68%, or one sigma deviation. Even with the large error on d , the Case 3 result is better than the Case 4 worst case result and Case 3 still provides an earlier detection

horizon. Case 3 crosses the warning parameter threshold at approximately 5.3 seconds and since collision occurs at 6.5 seconds in the simulated scenario the detection horizon is approximately 1.2 seconds. Compared to Case 1 and Case 2, this is a detection horizon decrease of 1.28 seconds. This is a percentage change decrease of 48% although the error increase is at least 68%. These results show that, as spacing errors increase by a factor of X , that the warning parameter error will not increase by the same factor.



W (ideal) – No error on any variable

Case 1 – Errors using Table 4-6 values

Case 2 – Table 4-6 values except d increased from 0.7m to 1.198m

Case 3 – 68% error on d but Table 4-6 values for remaining variables

Case 4 – 68% error on all variables

Figure 6-10: Updated Uncertainty Analysis Results

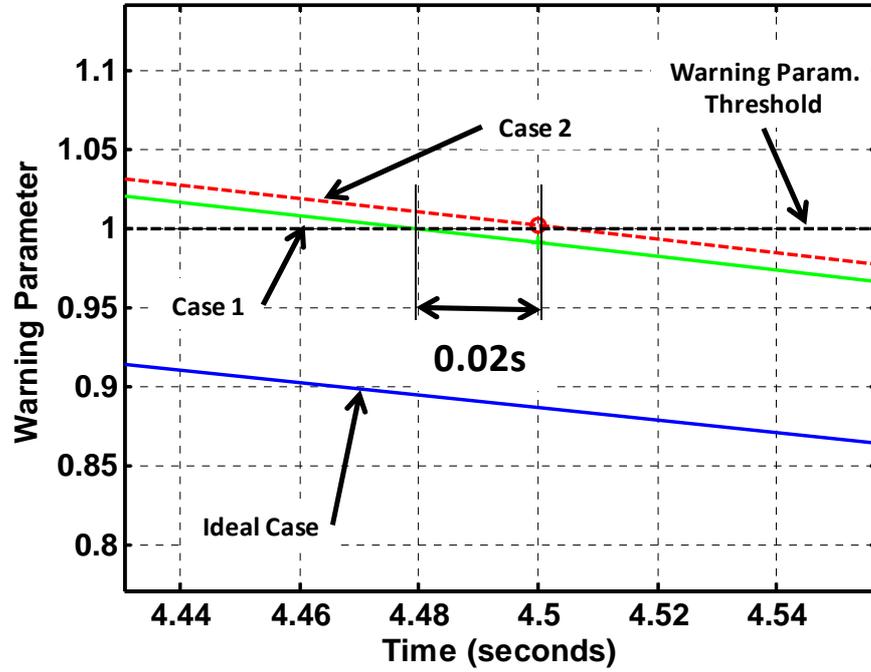


Figure 6-11: Comparison of Case 1 and Case 2 Uncertainty Analysis Results

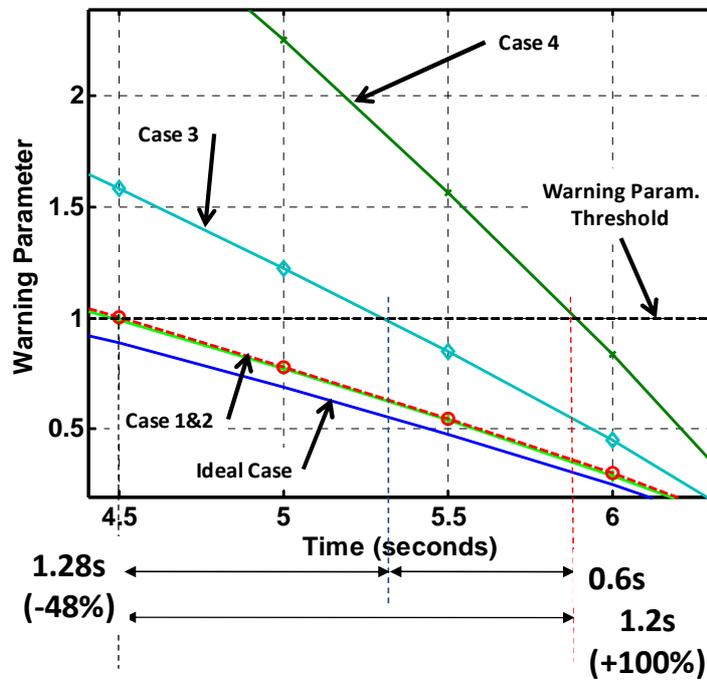


Figure 6-12: Evaluation of Case 3 Uncertainty Analysis Results

6.4 Measurement Linearity

The previous analysis results identified how Δd is distributed. It is also important to understand the dynamics of this error and whether it changes as the actual distance between vehicles increases and decreases, in other words the linearity of the error. Particularly as the d decreases it is important to verify that the error does not significantly increase or behave in nonlinear manner which would be difficult to model and correct. This section investigates that concern.

Instead of using the same dataset as before, which is problematic because it is difficult to ensure the fidelity of long-distance LIDAR readings (due to uncertainty where the LIDAR is striking the vehicle), a different test procedure was planned and executed to produce data specifically for this analysis. In this procedure, the truck was parked at one end of the large space vehicle handling area, identified in Figure 6-3, and the sedan was located directly behind within the LIDAR detection window. The sedan was then driven in reverse away from the truck thereby increasing the inter-vehicle distance until before exiting the LIDAR detection range (approximately 80m). After a brief stationary pause the sedan was then driven forward towards the truck to decrease the inter-vehicle distance. The goal was to simulate an impending collision. The same procedure was repeated for several iterations. This procedure provided data that was significantly “cleaner” than the data collected while driving around the track.

Figure 6-13 compares the collected data for the two estimates, hybrid and real-time, against the LIDAR ground truth. Again, from this preliminary visual inspection, the estimates show generally good correlation to the LIDAR ground truth. The difference between the estimates and the ground truth was again calculated and the result for a specific subset period of the dataset is shown in Figure 6-14. It is noticeable that the error is larger when the sedan is transient, either towards or away, compared to when it is stationary. This is expected as most GPS receivers improve position fix based on previous data through some type of low-pass or averaging filter. Thus, both a time lag and velocity dependence is introduced through filtering. It is also noticeable that the error is generally positive

during the approach maneuver and negative during reverse. When the GPS estimates are plotted against the LIDAR data as shown in Figure 6-15, this effect appears as hysteresis. It is believed that this hysteresis is due to a combination of the sedan travel direction and an inherent “delay” of the estimates in comparison to the LIDAR. This hysteresis effect is further examined later.

Figure 6-15 also provides a preliminary visual inspection of the linear behavior of the error for both GPS-based estimates as the sedan is in motion. As the ground truth d increases or decreases, it is expected that the hybrid and real-time d estimates change linearly as well. Figure 6-15 confirms that they do follow an expected linear behavior.

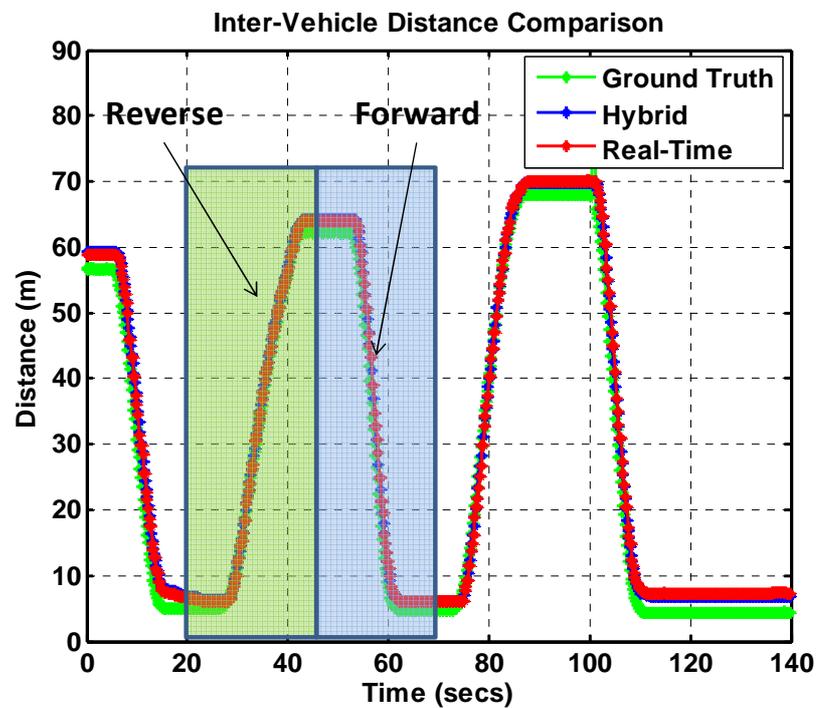


Figure 6-13: Estimates vs. Ground Truth, Forward/Reverse Procedure

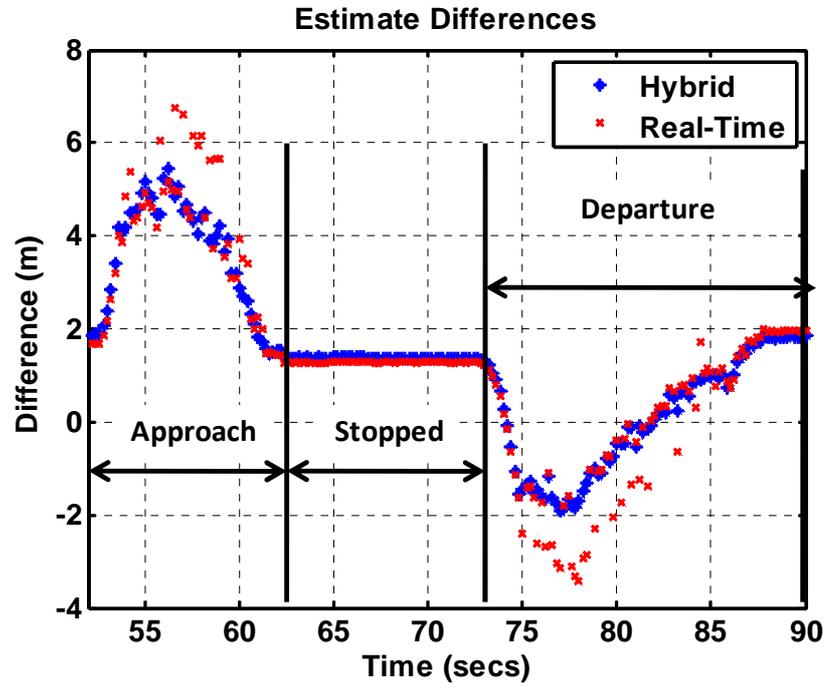


Figure 6-14: Estimate Errors, Subset of Forward/Reverse Dataset

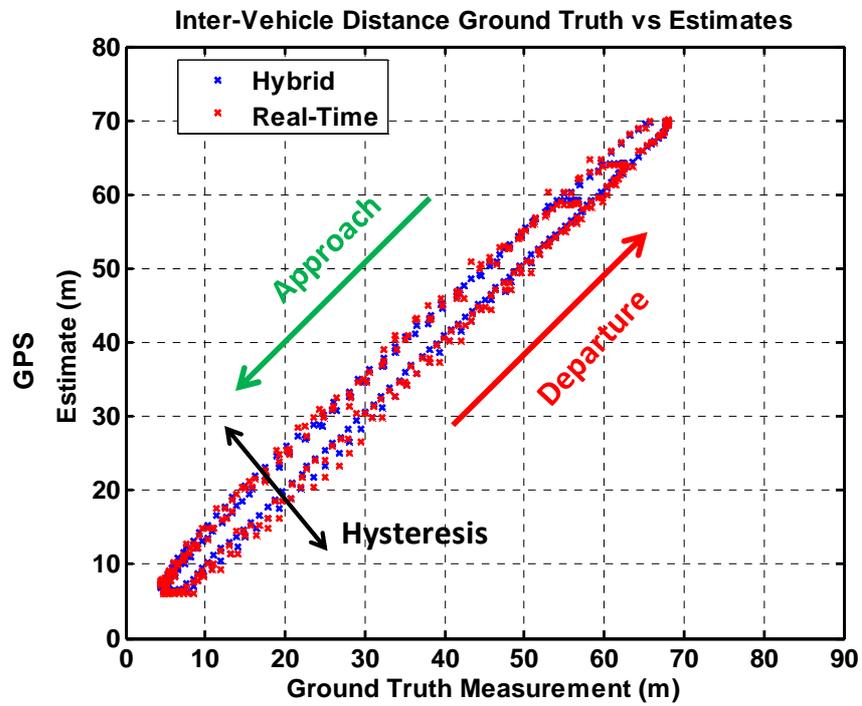


Figure 6-15: Estimates vs. Ground Truth

Since the estimates demonstrate a linear behavior, it is expected that their errors would also be linear as a function of distance. Again it is more important to confirm that the error does not consequently grow as the sedan approaches the truck. Figure 6-16 is similar to Figure 6-15 except that it is a plot of the errors of the estimates (the results trended in Figure 6-14) versus LIDAR. Again, the hysteresis effect is observable but a general linearity is also observed during the approach and reverse maneuvers. Figure 6-17 focuses only on the subset of error data during the approaches. To quantify the linearity, the MATLAB curve fitting toolbox was used to apply a linear fit to this data. The results are also shown in Figure 6-17. Although there is a slightly positive slope in the error, it is believed that this is related to the hysteresis effect. More importantly, the results clearly show that the error for either estimate does not significantly grow or behave non-linearly as the gap decreases between the vehicles.

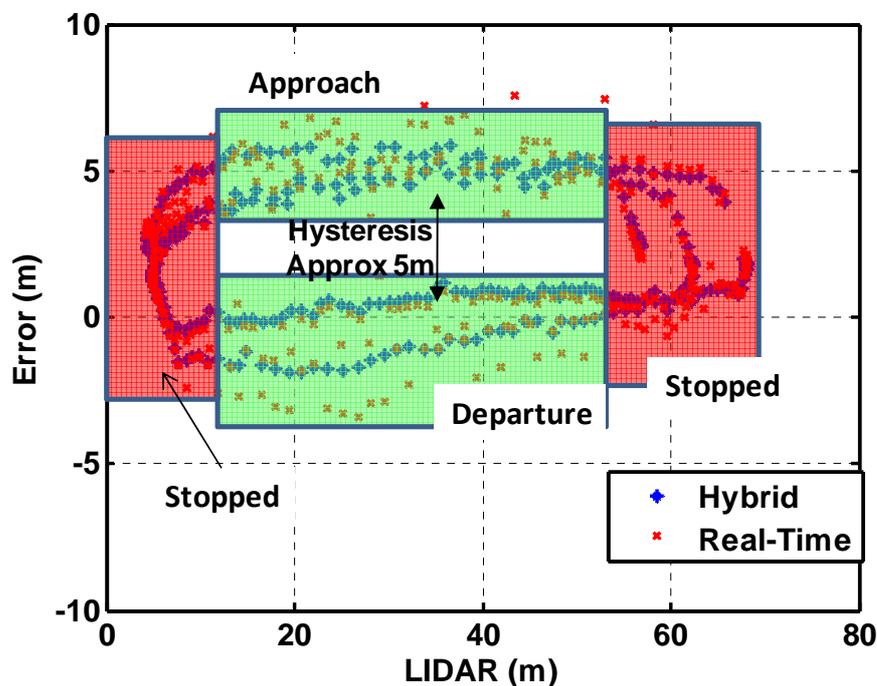


Figure 6-16: Estimate Errors vs. Ground Truth Inter-vehicle Distance

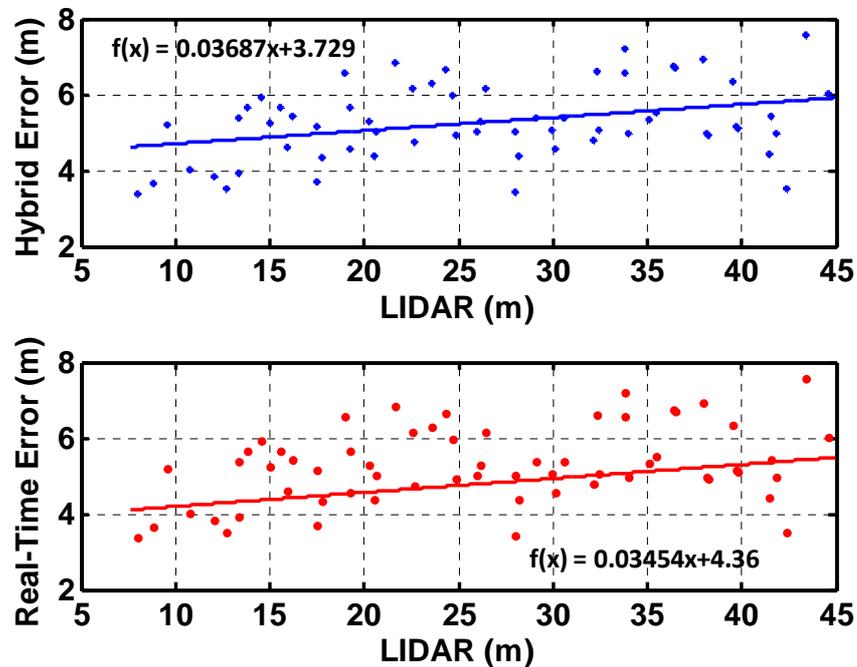


Figure 6-17: Error Behavior during Vehicle Approach

6.5 Measurement Hysteresis

Figure 6-16 also visually confirms the hysteresis effect by showing the shift in the error during approach and departure. From the figure, the hysteresis spread is approximately 5.0m. Table 6-1 and Table 6-2 quantify the mean and variance for each of the “events” during the testing. The hysteresis is also quantified by calculating the difference between the mean of the forward and reverse events. Although the exact cause of the hysteresis was not determined, the spread is believed to be due to several factors which are briefly discussed here. First, the travel direction of the vehicle affects the sign of the error. The error results shown in Figure 6-16 are calculated by subtracting time correlated samples of LIDAR based d from GPS based d . Since the LIDAR sampling rate is faster, the LIDAR based d changes ahead of the GPS estimates. Therefore it is expected that the sign of the error is positive when the sedan is closing in and signed negative during reverse.

Table 6-1: Hysteresis Results for Hybrid System

	Mean (m)	Variance
Stationary	2.05	0.4
Forward	4.8	0.73
Reverse	0.04	0.85
Hysteresis	4.76	

Table 6-2: Hysteresis Results for Real-Time System

	Mean (m)	Variance
Stationary	2.16	0.63
Forward	5.27	1
Reverse	-0.44	1.12
Hyteresis	5.71	

The magnitude of the spread is dependent on the sedan traveling velocity and the sampling characteristics of the sensors. As the relative velocity between the vehicles increases the error in the inter-vehicle distance will increase proportionally because the sensor sampling frequency is constant. Subsequently a measurement “delay” will be introduced because the sensor will not be able to report as quickly as the rate of changing position. The delay effect is easier to confirm in Figure 6-18 which focuses on the lower left corner of the data from Figure 6-15 but only for the real-time estimate. As the sedan approaches the truck and slowly decelerates to a stop; the data points stop changing along the x-axis because the LIDAR measurement has settled. But the values continue to change along the y because of the slower response of the GPS based measurement. Upon departure, the LIDAR responds first as evident by data points changing in x but remain locked in y, which again demonstrates the delay effect. Although not shown, this delay affect also occurs at the opposite end of the test.

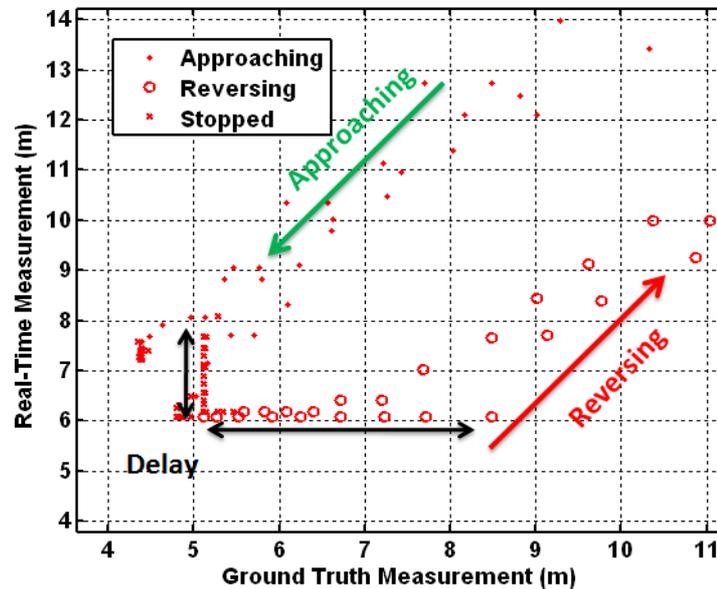


Figure 6-18: Close Inspection of Delay Effect

The approximate error of the measurement delay can be quantified knowing vehicle traveling velocity and sensor sampling rate. For example, the San Jose GPS sensors are designed with an internal update rate of 5 Hz and the average maximum velocity achieved during the testing was 20 mph (8.9408 m/s) while driving in the forward direction. Dividing this maximum speed by the GPS update rate returns an approximate error of 1.8m. Similarly this error exists for the LIDAR measurement as well. However, since the LIDAR system samples at a higher rate (37.5Hz) the maximum velocity delay error would be instead 0.24m using the same calculation. Since error is compared against the LIDAR measurement in Figure 6-16, the two values could be combined resulting in an estimate of 2.0m for measurement error strictly due to delay. Since this 2.0m error would exist in both directions, the hysteresis would appear as 4.0m in Figure 6-16 due to the delay error alone. Since the real-time system relies on wireless communication to exchange vehicle telemetry with surrounding vehicles there is also likely delay due to wireless network latency and subsequently additionally increased error. This may explain the larger hysteresis result in the real-time system when compared to the hybrid system in Table 6-1 and Table 6-2. From field test results the GPS timestamp difference

between receivers would nominally jump from 0.2 seconds to 0.4 seconds. This means on average there was an additional 0.2 second delay introduced due to network latency. The additional 0.2 seconds of delay due to network latency would add an additional 3.6m of hysteresis spread to the original 4.0m estimate. The predicted hysteresis error could be in the worst case 7.6m. In summary based on this rough analysis, the hysteresis evident in Figure 6-16 is within reasonable expectation. It is important to note that this hysteresis spread represents the total combined error in both directions of travel. The large magnitude of approximately 5m is not the expected error the GPS collision system would experience.

6.6 Summary

In this chapter, the error of inter-vehicle distance measurement, d , was specifically analyzed. Measured d values were compared between the GPS hybrid system (a high grade GPS system used in conjunction with a low grade system), the real-time system (prototype low grade GPS collision system), and a LIDAR sensor. Error was evaluated by comparing the measured d values of the hybrid and real-time system against the LIDAR. Analysis results reported the error to be within 1.3 meters. The effect of this error on the collision warning parameter was subsequently analyzed. Even with large Δd the effect on the warning parameter is minimal compared to other parameter variations. The results of this section should address concerns of whether GPS could be accurate enough compared to the current forward scanning techniques. The linearity of Δd error was also investigated. Results showed that d and subsequently its error behave linearly throughout increasing and decreasing spacing. Through a specific test procedure executed to gather data for the linearity analysis, a hysteresis in the error was identified. Although the exact cause for the hysteresis was not determined several factors inherent to the data acquisition scheme and test procedure execution could be contributing. Overall the results of this chapter show that d as measured by GPS is stable and useable for collision detection.

Chapter 7 CONCLUSIONS

7.1 Summary

Previous surveys of accidents by the NHSTA identify rear-end collisions to be a major percentage of overall vehicle accidents. Most of these accidents are a result of driver inattention and are due to a driver following the preceding vehicle too closely. Forward collision warning detection systems are currently being developed to address these safety concerns. In most systems, the distance between the vehicles is actively measured by a forward scanning sensor system such as Radar or LIDAR. When this inter-vehicle distance measurement falls below a certain threshold, an alert is provided to the driver. However, these forward scanning type sensor technologies have performance limitations. They require direct line of sight and infer preceding vehicle velocity from changes in inter-vehicle distance. Also in certain environments/conditions the sensors can become covered with particulate (sand, dust, water) that adversely affect the measurement.

This thesis investigated alternative methods for a forward collision warning (FCW) system. In particular, an alternative solution was investigated which utilizes GPS technology. For military applications, GPS was an attractive option since it is readily available on most vehicles and is low-cost to incorporate. A GPS-based collision detection system was designed, fabricated and successfully demonstrated on three HEMTT vehicles at the YUMA proving grounds. The details of the prototype systems that were developed and tested were presented in Chapter 3.

The work of this thesis focused on developing a methodology to evaluate measurement uncertainty in the system and how it affects the overall output which is the alert to the driver. A framework for evaluating this uncertainty was first presented in Chapter 4. The framework combines the results of an error analysis applied to each of the input measurements and sensitivity analysis of the overall system algorithm. The results of the sensitivity analysis identified how each of the input parameters contributes to the warning parameter as a subject vehicle approaches a leading vehicle.

Dominant parameters included the inter-vehicle distance, velocity, relative velocity, and system timing parameters.

The analysis showed that parameters affecting the driver's warning each have a varying level of influence on the warning parameter. Specifically, the driver tuning parameter and friction coefficient parameter exhibited minimal influence. For the error analysis, only the friction coefficient parameter was characterized using the velocity measurement from the GPS sensor. Using the worst-case friction coefficient values from this analysis along with representative error values for the other parameters, the final uncertainty analysis was applied. The key takeaway of this analysis was that, when parameters of the warning algorithm are in error, the alert to the driver can have increasing error. In a field application, a full understanding of the inherent measurement errors through this analysis could be used to correct the warning parameter calculation so that it approximates an ideal error-free calculation, thus providing a better alert to the driver.

The remaining chapters of the thesis focused on GPS position accuracy affects on inter-vehicle distance measurement. First Chapter 5 took a closer examination of sensor noise by evaluating the stochastic noises of the GPS sensor. To identify these sensor errors, a technique known as Allan Variance analysis was applied. Allan Variance has been widely used and adopted by several organizations for identifying stochastic noise processes of sensors. The dataset for this analysis included a full period of 24 hours where the sensors were undisturbed so that only noise would affect the measurement. For the GPS sensors, random walk appeared to be the most significant noise process. This was clearly evident in the raw time domain data. There was good agreement of this specific noise type across all three GPS sensors in both the latitude and longitude directions. A similar Allan Variance analysis was performed on the accompanying accelerometer sensor.

In contrast to the GPS error, the correlation of noise in the 3-axis accelerometers was not as evident. From the time domain analysis, it appeared that quantization and possibly bias were the most significant noise contributors. The result of the Allan variance analysis did not conclusively indicate a

specific noise model for accelerometers from one vehicle to another, and there was no strong agreement across the three sensors or even in the three directions. This may be due to the accelerometer orientation inconsistencies during the time of assembly/fabrication, manufacturer variation, differences in the electrical environment of each sensor, etc. In addition, the results of an autocorrelation analysis did confirm that the noise in the accelerometers was not random and that there is some correlation between the successive samples of data. This is possibly due to the low-pass filter that is part of the breakout interface circuit board used to interface the core accelerometer sensor to the embedded microcontroller.

Finally, chapter 6 analyzed the inter-vehicle distance measurement accuracy by comparing this measurement between the low-cost GPS solution used in the prototypes, a mixture of a low-cost unit and a high-end GPS unit, and then a rearward scanning LIDAR sensor. The LIDAR sensor was assumed as ground truth. The results of this comparison showed the inherent lag or delay in the inter-vehicle distance measurement when using GPS. On a positive note, the accuracy of the GPS-based measurements closely matched the LIDAR measurement. The low-cost GPS sensors have a CEP position accuracy of 3.3m. And since two GPS units are used to calculate the inter-vehicle measurement, the worst case error was expected to be 6.6m if the errors are combined. Instead, it was shown that the error was generally less than this worst case estimate due to the correlation benefits of multiple sensors operating simultaneously, similar to differential GPS. In other words, inter-vehicle distance measurement errors were reduced due to local error correlation.

In summary, a GPS-based alternative solution for forward collision warning/detection was introduced. The system was produced in the form of a small number of prototypes, and their use as was demonstrated through actual field testing in convoy-like situations. However, incorporating additional techniques could greatly improve the performance of the system. In a separate but parallel study, Stephen Chaves evaluated the use of Kalman filtering to fuse the velocity as measured by the

GPS with velocity derived from the accelerometer. In this thesis, a framework for analyzing system error and uncertainty was investigated.

7.2 Future Work

The analytical framework used to determine expected measurement uncertainty is not specific to the GPS-based collision detection system and could be also applicable to traditional and currently developing solutions. Regardless of the solution or technology used it is important to characterize the measurement errors and evaluate their effect so the uncertainty can be properly handled, especially if a sensor deteriorates or becomes completely unusable during operation. There are additional topics that could be addressed as future work to gain additional benefits.

Improved characterization of input measurement errors

The results of the error analysis applied to the friction coefficient were only used in the uncertainty analysis while estimates from previous literature were used for the remaining parameters. For better uncertainty analysis, actual error values for all the input parameters could be derived from appropriate test data. This would involve defining the proper test procedures that includes the use of accurate ground truth measurements. Ideally if a test procedure could be designed where the subject vehicle virtually collides with the vehicle, or object in front, the truest collision data could be gathered including variations in normal driver response.

Evaluate other collision scenarios

The collision scenario investigated in most of the work in this thesis involved two vehicles traveling at a constant cruising speed at some distance and then the leading vehicle aggressively stops. This scenario was selected as it closely represents typical convoy driving missions. Future work should involve investigating other scenarios. Examples are: both vehicles are traveling at different speeds, leading vehicle braking while following vehicle is accelerating. Additional collision test cases for consideration are also provided in [25].

Evaluate other methods for friction coefficient estimation

The friction coefficient estimation analysis used here simply involved interpretation of the slope profile of the velocity data during skid braking. Additional methods to estimate friction coefficient exist and should be investigated and their resulting errors should be compared.

Verify stochastic noise models

The result of the Allan variance analysis provides a stochastic noise model. Although it was not considered in this work, this model of noise variation could be used to corrupt a clean measurement signal and then compared against an actual signal to verify the validity of the noise model. A method for this analysis and results has been reported by Jerath in a technical report produced for the Federal Highway Administration [41][39].

Evaluate less conservative uncertainty analysis algorithms

The uncertainty analysis equation used here represents the most conservative uncertainty estimator, using the delta or sigma terms for each measurement. This method is sometimes referred as the 1-norm approach. For better results, a less conservative algorithm may be used for evaluation such as 2-norm approach which involves the use of the variance of the measurement. Other non-linear methods also exist that should also be evaluated.

Use better ground truth measurement for inter-vehicle measurement comparison

In Chapter 6, the LIDAR sensor was used as ground truth but suffered from some target confusion. For future similar testing, it is recommended an industrial string-pot be used to get better ground truth measurement accuracy. The LIDAR or other forward scanning sensor technologies could be compared to the ground truth as well as the GPS for a true comparison. Further, using such a simple analog potentiometer, one could confirm that the delays seen in comparing the GPS data to LIDAR are solely due to GPS and not due to processing errors in LIDAR.

Thorough survey of inter-vehicle networks

In the GPS-based FCWS approach, vehicle-to-vehicle (V2V) communication is absolutely necessary. Otherwise, this alternative solution is not possible. Here V2V was established through wireless communication and a simple commercial of the shelf (COTS) network architecture (802.11N). A standard protocol (TCP/IP) was used in this development phase. During testing, this configuration presented limitations in range and stability and subsequently the test procedures were tailored appropriately. For future work, other wireless network architectures should be evaluated on the basis of these criteria and to have additional security. Network security is critical for military applications. Inter-vehicle communication, for example using DSRC radios, can provide significant benefits other than just FCWS and is currently a popular area of research largely supported by the U.S DOT's Research and Innovative Technology Administration (RITA) as part of the Intelligent Vehicle-Based Safety Systems (IVBSS) initiative [42] [40]. A number of studies have already been conducted and serve as a good starting point for this task [43][41], [44][42], [45][43].

Outside Environment Drift Characterization

In the future it would be beneficial to perform a similar analysis of higher-end GPS receivers to examine any changes in noise terms. Also, the GPS measurements in this experiment were collected inside a building acquired from a GPS repeater station. Future work should also include the characterization of GPS drift in an outside environment.

Multi-sensor Fusion

In the proposed prototype system, GPS information is fused with a basic 3-axis accelerometer. Data from the accelerometer could be used to improve the vehicle telemetry estimates, for example to determine when the vehicle is experiencing rapid acceleration changes to provide better input on relative vehicle velocity, and friction. Additional sensors were considered during the prototype design process including: vehicle odometer and speed, camera, and map information. These sensors were ignored due to core focus on GPS and design limitations (system must be vehicle non-intrusive, plug and play compatible, etc). For future work, these sensors should be still considered in improving the robustness of the system. Additionally as conceptual intelligent highway systems become reality, real-time road condition information will also be beneficial.

Real-Time Implementation

The analytical approach presented here was applied off-line to experimental and simulated datasets. Real-time methods for identifying sensor error could be developed. The uncertainty analysis could also be implemented in an embedded environment thus providing real-time assessment of measurement uncertainty. If uncertainty changes in a negative manner, error correction models could be called upon to provide temporary accuracy until the issue is resolved. The real-time implementation could also be used to identify when sensors are degraded or failing.

BIBLIOGRAPHY

- [1] National Transportation Safety Board. (2010, Feb) NTSB. [Online]. http://www.nts.gov/recs/brochures/MostWanted_2010.pdf
- [2] National Transportation Safety Board. (2010, February) NTSB. [Online]. http://www.nts.gov/recs/mostwanted/prevent_collision_tech.htm
- [3] Noblis, "Intelligent Transportation Systems Benefits, Costs, Deployment, and Lessons Learned: 2008 Update," Annual Report 2008.
- [4] Insurance Institute for Highway Safety, "Big Strides," *Status Report*, vol. 46, no. 3, March 2011.
- [5] Stephen Chaves, "Using Kalman Filtering To Improve a Low-cost GPS-based Collision Warning System for Vehicle Convoys," Department of Mechanical Engineering, Masters Thesis 2010.
- [6] Ronald R NHTSA Knipling, Jing-Shiarn IMC Inc Wang, and Hsiao-Ming IMC Inc Yin, "Rear-End Crashes: Problem Size Assessment and Statistical Description," Office of Crash Avoidance Research, Department of Transportation, Technical Report 1993.
- [7] Ray Resendes, "Ivi Problem Areas Description: Motor Vehicle Crashes - Data Analyses And Ivi Program Emphasis," Technical Report 1999.
- [8] National Transportation Safety Board, "Vehicle and Infrastructure-based Technology For the Prevention of Rear-end Collisions," Special Investigation Report 2001.
- [9] University of Michigan Transportation Research Institute, "Integrated Vehicle-Base Safety Systems First Annual Report," Progress Report 2005.
- [10] D. R. Ankrum, "Smart Vehicles, Smart Roads," *Traffic Safety*, no. 93, pp. 6-9, 1992.
- [11] Ronald R Knipling, "IVHS Technologies Applied to Collision Avoidance: Perspectives on Six Target Crash Types and Countermeasures," Technical Paper presented at Saftey & Human Factors session of 1993 IVHS America Annual Meeting 1993.
- [12] United Press International. (2004) Military.com. [Online]. http://www.military.com/NewsContent/0,13319,FL_deaths_071204,00.html
- [13] Carrie Deming. (2009, April) United States Army Acquisition Support Center. [Online]. http://www.usaasc.info/alt_online/article.cfm?iID=0904&aid=07
- [14] A. Nmngani and M. Akyurt, "A Review of Vehicle Collision Avoidance Systems," in *Saudi Engineering Conference*, Dhahran, 2002, pp. 413-428.
- [15] H. Kamiya and et al., "Intelligent Technologies of Honda ASV," , 1996.
- [16] P. Seiler, B. Song, and J. K. Hedrick, "Development of a Collision Avoidance System," vol. 98PC-417, 1998.

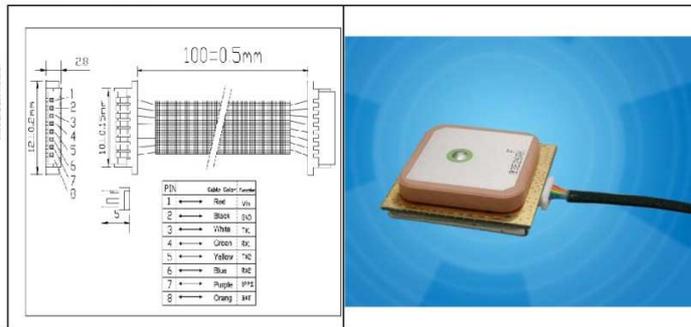
- [17] John D. Lee, Joshua D. Hoffman, and Elizabeth Hayes, "Collision warning Design to Mitigate Driver Distraction," in *Conference on Human Factors in Computing Systems*, vol. 6, Vienna, Austria, 2004, pp. 65-72.
- [18] R. D. Sorkin, B. H. Kantowitz, and S. C. Kantowitz, "Likelihood Alarm Displays," vol. 30, no. 4, pp. 445-459, 1988.
- [19] Eaton Corporation, VORAD Collision Warning System for Military Vehicles, 2007.
- [20] James Brown, Marvin McCallum, John Campbell, and Christian Richard, "Integrated Vehicle-Based Safety System Heavy Truck Driver-Vehicle Interface (DVI) Specifications (Final Version)," Washington, D.C., Final Specifications 2007.
- [21] Eaton Corporation, VORAD VS-400 System Packages, 2008, URL:http://www.roadranger.com/ecm/groups/public/@pub/@eaton/@roadranger/documents/content/ct_063188.pdf.
- [22] Mobileye. (2008) Mobileye. [Online]. <http://www.mobileye.com/about>
- [23] U.S. Department of Transportation. Research and Innovative Technology Administration. [Online]. <http://www.its.dot.gov/ivbss/index.htm>
- [24] University of Michigan Transportation Research Institute, "Integrated Vehicle-Based Safety Systems: Third Annual Report," 2008.
- [25] Visteon Corporation, "Integrated Vehicle-Based Safety Systems Light Vehicle Verification Test Plan," 2008.
- [26] S. E. Shaldiver and S. K. Tan, "Analysis of Vehicle Positioning Accuracy Requirements for Communication-Based Cooperative Collision Warning," vol. 10, no. 3, pp. 131-140, 2006.
- [27] R. W. Sinnott, "Virtues of the Haversine," *Sky and Telescope*, vol. 68, no. 2, p. 158, 1984.
- [28] Paul Frank, *Introduction to System Sensitivity Theory*. New York: Academic Press, 1978.
- [29] Lucia Breierova and Mark Choudhari, "An Introduction to Sensitivity Analysis," Massachusetts Institute of Technology, D-4526-2, 1996.
- [30] D. M. Hamby, "A Review of Techniques for Parameter Sensitivity Analysis of Environmental Models," *Environmental Monitoring and Assessment*, no. 32, pp. 135-154, 1994.
- [31] Visteon Corporation, "Integrated Vehicle-Based Safety Systems Light Vehicle Verification Test Plan," U.S. Department of Transportation, 2008.
- [32] David Bevly, "Global Positioning System (GPS): A Low-Cost Velocity Sensor for Correcting Inertial Sensor Errors on Ground Vehicles," *Journal of Dynamic Systems, Measurement, and Control*, vol. 126, pp. 255-264, June 2004.
- [33] Circular Error Probable (CEP), Air Force Operational Test and Evaluation Center Technical Paper 6, Ver 2, 1987.

- [34] Songlai Han, Jinling Wang, and Nathan Knight, "Using Allan Variance To Determine The Calibration Model of Inertial Sensors For GPS/INS Integration," 2009.
- [35] W. J. Riley, "Handbook of Frequency Stability Analysis," SP 1065, 2008.
- [36] "IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Laser Gyros," IEEE, IEEE Std. 647, 2006.
- [37] Hyunseok Kim, Jang Gyu Lee, and Chan Gook Park, "Performance Improvement of GPS/INS Integrated System Using Allan Variance Analysis," 2004.
- [38] Fabian Czerwinski. (2010, Feb) www.mathworks.com. [Online]. <http://www.mathworks.com/matlabcentral/fileexchange/26659>
- [39] X.Y. Lu, S.E. Shladover, and W.B. Zhang, "Quantitative testing of a frontal collision warning system for transit buses," *Institution of Engineering and Technology Intelligent Transportation Systems*, pp. 215-223, 2007.
- [40] Charles Birdsong, Peter Schuster, John Carlin, Daniel Kawano, and William Thompson, "Test methods and results for sensors in a pre-crash detection system," *SAE Commercial Vehicle Conference*, 2005.
- [41] Kshitij Jerath, "Next Generation Vehicle Positioning Techniques for GPS-Degraded Environments to Support Vehicle Safety and Automation Systems.," FHWA, Interim Technical Report BAA DTFH61-09-R-00004, 2010.
- [42] U.S. Department of Transportation. (2011, February) Research and Innovative Technology Administration IVBSS. [Online]. <http://www.its.dot.gov/ivbss/index.htm>
- [43] S. Biswas, R. Tatchikou, and F. Dion, "Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety," *IEEE Communications Magazine* 44, pp. 74-82, 2006.
- [44] J. Blum, L. Eskandarian, and L. Hoffman, "Challenges of inter-vehicle adhoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, pp. 347-351, 2004.
- [45] J. Luo and J. Hubaux, "A survey of inter-vehicle communication," School of computer and communication sciences, EPEL, Technical Report IC/2004/24, 2004.
- [46] J.D Lee, D.V. McGehee, T.L. Brown, and M.L. Reyes, "Driver distraction, warning algorithm parameters and driver response to imminent rear-end collisions in a high fidelity driving simulator," US Department of Transportation, NHSTA, 2002.
- [47] W. Janssen and L. Nilsson, *Driving Future Vehicles*, A. M. Parkes and S. Franzen, Eds. Washington, D.C.: Taylor & Francis, 1993.
- [48] Larry J. Levy. (1997) The KF: Navigation's Integration Workhorse. [Online]. <http://www.cs.unc.edu/~welch/kalman/Levy1997/index.html>
- [49] T. H. Witte and A. M. Wilson, "Accuracy of non-differential GPS for the determination of speed over ground," *Journal of Biomechanics*, pp. 1891-1898, 2004.

APPENDIX A SAN JOSE FV-M8 GPS RECEIVER



GPS Engine Board Model: FV-M8



©2008 San Jose Technology, Inc. All specifications subject to change without notice.

Specifications:

PHYSICAL CONSTRUCTION		PERFORMANCE		
Dimension	L30mm*W30mm*H8.6mm	Built-in Antenna	Highly-reliable ceramic patch	
Weight	15 grams	Sensitivity	-158dBm	
		SBAS	1 channel (Support WAAS, EGNOS, MSAS)	
		DGPS	RTCM Protocol	
Receiving frequency	1575.42MHZ; C/A code	Receiver architecture	32 parallel channels	
Connector	8pin connector with 1.0mm pitch	Start-up time	Hot start	1 sec. typical
			Warm start	35 sec. typical
			Cold start	41sec. typical
Mounting	Soldering	Position accuracy	Without aid	3.3 m CEP
			DGPS (RTCM)	2.6 m

www.sanav.com

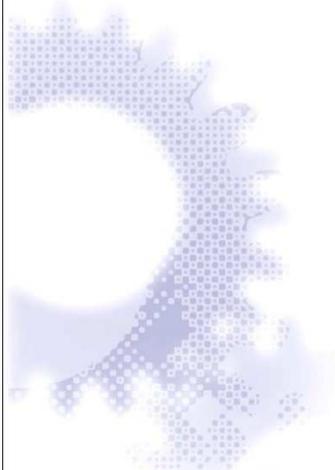
San Jose Technology, Inc. | 11F., No.2, Sec. 4, Zhongyang Rd., Tucheng City, Taipei County 236, Taiwan (R.O.C.) | Tel: 886-2-22694456 | Fax: 886-2-22694451 | sanav@sanav.com

Appendix Figure 1: GPS Specification Page 1



Construction	Full EMI Shielding		Velocity accuracy	0.1 Knot RMS steady state		
ENVIRONMENTAL CONDITIONS			Update Rate	1 ~ 5Hz		
Temperature	Operating: -30 ~ +80 °C		Power Supply	3.3~5V +- 5%		
	Storage: -40 ~ +85 °C					
COMMUNICATION			Current Consumption	Acquisition	63mA	
				Tracking	59mA (first 5 minutes)	
					42mA (after 5 minutes)	
					33mA (after 20minutes)	
Protocol	NMEA V3.01					
Signal level	UART @ 2.8V * 2					
INTERFACE CAPABILITY			Baud Rate	4800 bps (default) &		
Standard	Default	RMC, GGA, GSV*5,		4800/9600/38400/57600/11520		
Output		VTG, GSA*5		0 bps are adjustable		
Sentences	Optional	GLL, ZDA				

©2008 San Jose Technology, Inc. All specifications subject to change without notice.



Appendix Figure 2: GPS Specification Page 2

APPENDIX B
MATLAB CODE: SENSITIVITY ANALYSIS

```

% The following test procedures are taken from the IVBSS Light Vehicle
Test
% Plan published by the NHTSA

%% FCW RE-3
% Rear End Conflict with an Aggressively Slowing POV
% Vehicle)
% Parameters
% Subject Vehicle Velocity = 20.1 m/s (45 mph)
% Principal Vehicle Velocity = 20.1 m/s (45mph)
% Principal Vehicle Deceleration = 3.5 m/s^2 (7.8mph)
% Relative Velocity = 11.2 m/s
% Initial Intervehicle Distance = 80 m (263ft)
% Warning Distance = 5 m (11.5ft)
clear
% close all

xsv(1) = 0; %Subject Vehicle Starting Position (m)
xpv(1) = 80; %Principal Vehicle Starting Position (m)
vsv(1) = 20.1; %Subject Vehicle Velocity (m/s)
vpv(1) = 20.1; %Principal Vehicle Velocity (m/s)
apv(1) = -3.5; %Principal Vehicle Deceleration

vrelative(1) = abs(vsv(1)-vpv(1)); %Relative Velocity (m/s)
d(1) = 80; %Starting intervehicle distance
%First calculate the Total Time of Experiment

dt = .5;
t = 0;
i = 2;
%Create Distance Measurement For Given Set of Test Conditions
for i = 2:15
    % while t < T
    xsv(i) = vsv(1)*dt+xsv(i-1);
    vpv(i) = vpv(i-1)+apv*dt; %Principal Vehicle Change in Velocity
    vsv(i) = vsv(1);
    xpv(i) = vpv(i)*dt+xpv(i-1);
    d(i) = abs(xsv(i)-xpv(i));
    vrelative(i) = abs(vpv(i)-vsv(1));
    t = t+dt;
    time(i) = t;
    i = i+1;
end

%Create Figure of Test Profile
figure
subplot(211)
plot(time,d)

```

```

ylabel('Inter-Vehicle Distance (m)');
xlabel('Time (secs)');
grid
subplot(212)
plot(time,vsv,time,vpv,time,vrelative);
legend('2nd Vehicle','Lead Vehicle','Relative Velocity')
ylabel('Velocity (m/s)');
xlabel('Time (secs)');
grid

%Define parameters for algorithm
numpts = length(d);
interdistance = d; %intervehicle distance (d)
velocity = vsv(1)*ones(1,numpts); %velocity of subject vehicle (v)
% vrelative = vrel*ones(1,numpts); %calculated above
accel = 8*ones(1,numpts); %deceleration capability of subject vehicle (a)
timedelay = 1.4*ones(1,numpts); %system time delay (tau)
buffdistance = 5*ones(1,numpts); %buffer distance (do)
friction = 0.8*ones(1,numpts); %tire friction coefficient (mu)
drivertune = ones(1,numpts); %driver tuning coefficient (Kdriver)

for i = 1:numpts
    d = interdistance(i);
    v = velocity(i);
    vrel = vrelative(i);
    a = accel(i);
    tau = timedelay(i);
    do = buffdistance(i);
    mu = friction(i);
    dtune = drivertune(i);

    %Calculate warning parameter
    WP(i) = (d)/((0.5*(v^2/a - (v-vrel)^2/a)+v*tau+do)*mu*dtune);

    %Derived Partial Equations
    %Distance
    dd(i) = (a)/(dtune*mu*(a*do+a*tau*v+v*vrel-0.5*vrel^2));
    Sdd2(i) = abs(dd(i)*d/WP(i));

    %Velocity
    dv(i) = (a*d*(-1*a*tau-vrel))/(dtune*mu*(a*(do+tau*v)+(v-
0.5*vrel)*vrel)^2);
    Sdv2(i) = abs(dv(i)*v/WP(i));

    %Relative Velocity
    dvrel(i) = -(a*d*(v-vrel))/(dtune*mu*(a*(do+tau*v)+(v-
0.5*vrel)*vrel)^2);
    Sdvrel2(i) = abs(dvrel(i)*vrel/WP(i));

```

```

%Deceleration
da(i) = (d*vrel*(v-.5*vrel))/(dtune*mu*(a*(do+tau*v)+(v-
0.5*vrel)*vrel)^2);
Sda2(i) = abs(da(i)*a/WP(i));

%Delay Time
dtau(i) = -(a^2*(d)*v)/(dtune*mu*(a*(do+tau*v)+(v-0.5*vrel)*vrel)^2);
Sdtau2(i) = abs(dtau(i)*tau/WP(i));

%Buffer Distance
ddo(i) = -(a^2*d)/(dtune*mu*(a*(do+tau*v)+(v-0.5*vrel)*vrel)^2);
Sddo2(i) = abs(ddo(i)*do/WP(i));

%Friction Coefficient
dmu(i) = -(a*d)/(dtune*mu^2*(a*(do+tau*v)+(v-0.5*vrel)*vrel)^2);
Sdmu2(i) = abs(dmu(i)*mu/WP(i));

%Driver Tuning Coefficient
ddtune(i) = -(a*d)/(dtune^2*mu*(a*(do+tau*v)+(v-0.5*vrel)*vrel)^2);
Sddtune2(i) = abs(ddtune(i)*dtune/WP(i));

end

figure
TIME=time;
plot(TIME,Sdd2,TIME,Sdv2,TIME,Sdvrel2,TIME,Sda2,TIME,Sddo2,TIME,Sdtau2,TIM
E,Sdmu2,TIME,Sddtune2)
title('Relative Sensitivity');
ylabel('Magnitude');
xlabel('Time (secs)');
grid
legend('d','v','v_{rel}','\alpha','d_o','\tau','\mu','K_{driver}');

```

APPENDIX C
MATLAB CODE: ALLAN VARIANCE ANALYSIS FOR GPS & ACCELEROMETER

```

function [AWLS] = allanVariance(data, dataRate, noiseModel,tau, do_debug)
%% Allan variance analysis
%-----
-
% Created by   : Kshitij Jerath; Email: kjerath@psu.edu
% Dated      : 05 April 2010
% Edits by Sanket Amin, April 15 2010
%-----
-
% INPUTS
% data       : Data on which Allan variance analysis is to be performed
% [n x 1]
% dataRate   : Sampling frequency (e.g. 25 Hz)
% noiseModel : Array indicating noise sources to include in model
%             E.g. [1 0 0 0 0] corresponds to quantization error
%                 [0 1 0 0 0] corresponds to angle random walk
%                 [0 0 1 0 0] corresponds to bias instability
%                 [0 0 0 1 0] corresponds to rate random walk
%                 [0 0 0 0 1] corresponds to rate ramp
%                 [0 0 0 1 1] corresponds to rate random walk and
%                 rate ramp in noise model, and so on;
% tau       : Tau vector, if empty will default to calculation
%             internally
% debug     : if value of one will produced plots
%
% NOTE : Empty inputs such as allanVariance([],[],[1 0 0 0 0]) sets the
% values of data and dataRate to default white noise and 100Hz
% respectively
%-----
-
% OUTPUT
% AWLS      : Weighted least squares estimate for coefficients
%-----
-

%% Code follows

disp('Processing...');
if(isempty(do_debug))
    do_debug = 1; %Plotting allan variance and weight least squares fit
end

if(isempty(data))
    warning('No data entered - setting default data to white noise');
    data = 10.*randn(100000,1);
end
if(isempty(dataRate))
    warning('Data rate not entered - default data rate = 100 Hz');
    dataRate = 100;
end

```

```

if (isempty(tau))
    len = length(data);
    ordermax = numel(num2str(fix(len/dataRate)))-2; % Maximum possible
correlation time based on data length
    order = fix(log10(1/dataRate));
    tau1 = [1, 2, 3, 4, 5, 6, 7, 8, 9];
    tau2 = [];

    while(order < ordermax)
        tau2 = [tau2, (10^order).*tau1];
        order = order + 0.5;
    end
    tau = sort(tau2);
    RootAllanVar = zeros(1,length(tau));

    %% Calculation Loop - varying tau
    for(count = 1:1:length(tau))
        t = round(tau(count)*dataRate);
        numDivisions = floor(len/t);
        Avg = zeros(1,numDivisions);
        Diff = zeros(1, numDivisions-1);
        for(index = 1:1:numDivisions)
            Avg(index) = (sum(data(t*(index-1)+1:t*index)))/t;
        end
        for(index = 1:1:numDivisions-1)
            Diff(index) = Avg(index+1) - Avg(index); %in degrees per sec
        end
        RootAllanVar(count) = sqrt(0.5*mean(Diff.*Diff )); % Calculating
root allan variance
        if(mod(count,15) == 0)
            disp('Processing...');
        elseif(mod(count,10) == 0)
            disp('Processing..');
        elseif(mod(count,5) == 0)
            disp('Processing. ');
        end
    end
end
end

%% Calculation Using allan.m (3.0)
% This section added by Sanket Amin
%Prep Data
Data.freq = data; %Raw Frequency Measurement
Data.rate = dataRate; %

% Allan Variance
Tau = tau;
[avar]=allan(Data, Tau);

% Plotting data and Allan variance
if(do_debug == 1)

```

```

figure
subplot(311)

loglog(Tau,avar.sig,'o','Markersize',3,'Markerfacecolor',[0.5,0.5,0.95]);
title('Standard Deviation');
ylabel('Sig');
h = xlabel('Tau Correlation Time (in seconds)');
set(h,'FontSize',13);
h = ylabel('Root Variance (deg/sec)');
set(h,'FontSize',13);
grid on;
subplot(312)

loglog(Tau,avar.sig2,'o','Markersize',3,'Markerfacecolor',[0.5,0.5,0.95])
;
title('Normal Allan Deviation');
ylabel('Sig');
h = xlabel('Tau Correlation Time (in seconds)');
set(h,'FontSize',13);
h = ylabel('Root Allan Variance (deg/sec)');
set(h,'FontSize',13);
grid on;
subplot(313)

loglog(Tau,avar.osig,'o','Markersize',3,'Markerfacecolor',[0.5,0.5,0.95])
;
title('Overlapped Allan Deviation');
ylabel('Sig');
h = xlabel('Tau Correlation Time (in seconds)');
set(h,'FontSize',13);
h = ylabel('Root Allan Variance (deg/sec)');
set(h,'FontSize',13);
grid on;

end

%% Data Fitting to obtain coefficients
RootAllanVar = avar.osig; %Using Normal Allan Deviation
weight = 1./RootAllanVar; % Needed for performing weighted least squares
% weight = (ones(length(RootAllanVar),1))';
TAU = [];
TAU2 = [];

if(noiseModel(1) == 1)
    TAU = [TAU;tau.^(-1)];
    TAU2 = [TAU2;weight.*tau.^(-1)];
end
if(noiseModel(2) == 1)
    TAU = [TAU;tau.^(-0.5)];
    TAU2 = [TAU2;weight.*tau.^(-0.5)];
end
if(noiseModel(3) == 1)
    TAU = [TAU;tau.^(0)];
    TAU2 = [TAU2;weight.*tau.^(0)];
end

```

```

end
if(noiseModel(4) == 1)
    TAU = [TAU;tau.^(0.5)];
    TAU2 = [TAU2;weight.*tau.^(0.5)];
end
if(noiseModel(5) == 1)
    TAU = [TAU;tau.^(1)];
    TAU2 = [TAU2;weight.*tau.^(1)];
end

%invTAU = (inv(TAU2*TAU2'))*TAU2;
%AVARwt = weight.*RootAllanVar;
%AWLS = invTAU*AVARwt;

AVARwt = weight.*RootAllanVar;
AWLS = TAU2'\AVARwt';

% [estimates, model] = fitcurvedemo(tau,RootAllanVar);
% [sse, FittedCurve] = model(estimates);

%% Plotting fitted curve
if(do_debug == 1)
    figure

loglog(tau,RootAllanVar,'o','Markersize',3,'Markerfacecolor',[0.5,0.5,0.95
]);hold on;
    h = xlabel('Averaging Time (secs)');
    set(h,'FontSize',13);
    h = ylabel('Allan Deviation (degs)');
    set(h,'FontSize',13);
    h = gca;
    set(h,'FontSize',13)
    % legend('Processed data','Fitted curve');
    grid on;

    AVARfit = AWLS'*TAU;
    plot(tau,AVARfit,'r','Linewidth',3,'Color',[0.95,0.5,0.5]);
    corr([RootAllanVar',(AWLS'*TAU)'])
    legend1 = '\sigma_{FIT} = ';
    flag = 0;

    if(noiseModel(1)==1)
        leg1 = ' A_{-2}\tau^{-1} ';
        legend1 = strcat(legend1,leg1);
        flag = 1;
    end
    if(noiseModel(2)==1)
        leg2 = ' A_{-1}\tau^{-0.5} ';
        if(flag == 0)
            legend1 = strcat(legend1,leg2);
            flag = 1;
        else

```

```

        legend1 = strcat(legend1, '+', leg2);
    end
end
if(noiseModel(3)==1)
    leg3 = ' A_{0}\tau^{0} ';
    if(flag == 0)
        legend1 = strcat(legend1, leg3);
        flag = 1;
    else
        legend1 = strcat(legend1, '+', leg3);
    end
end
if(noiseModel(4)==1)
    leg4 = ' A_{1}\tau^{0.5} ';
    if(flag == 0)
        legend1 = strcat(legend1, leg4);
        flag = 1;
    else
        legend1 = strcat(legend1, '+', leg4);
    end
end
if(noiseModel(5)==1)
    leg5 = ' A_{2}\tau^{1} ';
    if(flag == 0)
        legend1 = strcat(legend1, leg5);
        flag = 1;
    else
        legend1 = strcat(legend1, '+', leg5);
    end
end
legend2 = legend('Allan Deviation from data', legend1);
% legend1 = legend('Root Allan Variance from data', '\sigma_{FIT} =
A_{-2}\tau^{-1} + A_{-1}\tau^{-0.5} + A_{0}\tau^{0}', '\sigma_{FIT} = A_{-
2}\tau^{-1} + A_{-1}\tau^{-0.5} + A_{0}\tau^{0}+
A_{1}\tau^{0.5}', '\sigma_{FIT} = A_{-2}\tau^{-1} + A_{-1}\tau^{-0.5} +
A_{0}\tau^{0}+ A_{1}\tau^{0.5}+ A_{2}\tau^{1}', 'FontSize', 8);
set(legend2, 'FontSize', 9, 'FontName', 'Calibri');
end

```

APPENDIX D
MATLAB CODE: ALLAN.M

```

function [avar]=allan(data, tau)

% Compute various Allan deviations for a constant-rate time series
% [AVAR]=allan(DATA, TAU)
%
% INPUTS:
% DATA should be a struct and has the following fields:
% DATA.freq    the time series measurements in arb. units
% DATA.rate    constant rate of time series in (Hz)
%              (Differently from previous versions of allan.m,
%              it is not possible to compute variances for time-
%              stamp data anymore.)
% TAU is an array of the tau values for computing Allan deviations
%
% OUTPUTS:
% AVAR is a struct and has the following fields (for values of tau):
% AVAR.sig      = standard deviation
% AVAR.sig2     = Allan deviation
% AVAR.sig2err  = standard error of Allan deviation
% AVAR.osig    = Allan deviation with overlapping estimate
% AVAR.osigerr = standard error of overlapping Allan deviation
% AVAR.msg     = modified Allan deviation
% AVAR.msgerr  = standard error of modified Allan deviation
% AVAR.tsig    = timed Allan deviation
% AVAR.tsigerr = standard error of timed Allan deviation
% AVAR.taul    = measurement interval in (s)
% AVAR.tauerr  = errors in tau that might occur because of initial
%              rounding
%
% NOTES:
% Calculations of modified and timed Allan deviations for very long time
% series become very slow. It is advisable to uncomment .msig* and .tsig*
% only after calculations of .sig*, .sig2* and .osig* have been proven
% sufficiently fast.
%
% No pre-processing of the data is performed.
% For constant-rate time series, the deviations are only calculated for
tau
% values greater than the minimum time between samples and less than half
% the total time.
%
% versionstr = 'allan v3.0';
% FCz OCT2009
% v3.0 faster and very plain code, no plotting; various Allan deviations
%      can be calculated; script and sample data are available on
%      www.nbi.dk/~czerwin/files/allan.zip
%      (Normal, overlapping and modified Allan deviations are calculated
in one function,
%      in strong contrast to MAHs approach of splitting up among various
functions. This might be beneficial for individual cases though.)

```

```

%
% MAH 2009
% v2.0 and others
%
% FCz OCT2008
% v1.71 'lookfor' gives now useful comments; script and sample data are
%     available on www.nbi.dk/~czerwin/files/allan.zip
% v1.7 Improve program performance by mainly predefining matrices outside
%     of loops (avoiding memory allocation within loops); no changes to
%     manual
%
% early program core by Alaa MAKDISSI 2003
% (documentation might be found http://www.alamath.com/)
% revision and modification by Fabian CZERWINSKI 2009
%
% For more information, see:
% [1] Fabian Czerwinski, Andrew C. Richardson, and Lene B. Oddershede,
% "Quantifying Noise in Optical Tweezers by Allan Variance,"
% Opt. Express 17, 13255-13269 (2009)
% http://dx.doi.org/10.1364/OE.17.013255

n=length(data.freq);
jj=length(tau);
m=floor(tau*data.rate);

avar.sig      = zeros(1, jj);
avar.sigerr   = zeros(1, jj);
avar.sig2     = zeros(1, jj);
avar.sig2err  = zeros(1, jj);
avar.osig     = zeros(1, jj);
avar.osigerr  = zeros(1, jj);
% avar.msig   = zeros(1, jj);
% avar.msigerr = zeros(1, jj);
% avar.tsig   = zeros(1, jj);
% avar.msigerr = zeros(1, jj);

tic;

for j=1:jj
    % fprintf('.');

    D=zeros(1,n-m(j)+1);
    D(1)=sum(data.freq(1:m(j)))/m(j);
    for i=2:n-m(j)+1
        D(i)=D(i-1)+(data.freq(i+m(j)-1)-data.freq(i-1))/m(j);
    end

    %standard deviation
    avar.sig(j)=std(D(1:m(j):n-m(j)+1));
    avar.sigerr(j)=avar.sig(j)/sqrt(n/m(j));

    %normal Allan deviation

```

```

avar.sig2(j)=sqrt(0.5*mean((diff(D(1:m(j):n-m(j)+1)).^2)));
avar.sig2err(j)=avar.sig2(j)/sqrt(n/m(j));

%overlapping Allan deviation
z1=D(m(j)+1:n+1-m(j));
z2=D(1:n+1-2*m(j));
u=sum((z1-z2).^2);
avar.osig(j)=sqrt(u/(n+1-2*m(j))/2);
avar.osigerr(j)=avar.osig(j)/sqrt(n-m(j));

%      %modified Allan deviation
%      u=zeros(1,n+2-3*m(j));
%      z1=D(1:m(j));
%      z2=D(1+m(j):2*m(j));
%      for L=1:n+1-3*m(j)
%          u(L)=(sum(z2-z1))^2;
%          z1=z1-y(L)+y(L+m(j));
%          z2=z2-y(L+m(j))+y(L+2*m(j));
%      end
%      avar.msigerr(j)=avar.msig(j)/sqrt(n-m(j));
%      uu=mean(u);
%      avar.msig(j)=sqrt(uu/2)/m(j);
%
%      %timed Allan deviation
%      avar.tsig(j)=tau(j)*avar.msig(j)/sqrt(3);
%      avar.tsigerr(j)=avar.tsig(j)/sqrt(n-m(j));

% toc

end;

avar.taul=m/data.rate;
avar.tauerr=tau-avar.taul;

toc;
end

```

APPENDIX E
MATLAB CODE: CALCULATE NOISE PARAMETERS

```
function [NoiseVals] = CalculateNoiseParameters(ACoeffs)

Q = ACoeffs(1)/sqrt(3);
N = ACoeffs(2);
B = ACoeffs(3)*0.6648;
K = ACoeffs(4)*sqrt(3);
R = ACoeffs(5)*sqrt(2);

NoiseVals = [Q;N;B;K;R];
```

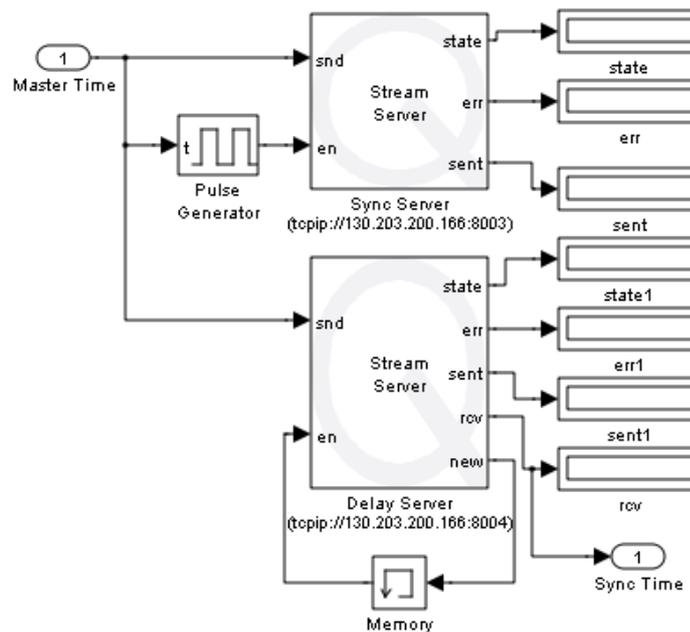
APPENDIX F

IEEE 1588 TIME SYNCHRONIZATION

The collection of data from multiple systems often presents a challenge during post analysis when each system functions off of separate timing hardware. To prevent this, the systems should be enabled with time synchronizing functionality to synchronize each of the independent clocks to a master time clock so that timestamps associated with the data are completely aligned. A specific standard, officially entitled “*Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*”, for this has been agreed upon and is maintained by the IEEE society. Information for this standard, including its implementation, can be found at the following website link: <http://ieee1588.nist.gov/>. In short summary, a high precision timing hardware is selected as a master clock on a system. This “master” system periodically communicates with other “slave” systems, usually over Ethernet, sharing the master clock value. Each slave system compares the master clock value to its own local slave clock value and determines the offset value between the clocks. Additionally each slave system sends a message to the master system and awaits a response to measure the delay in the communication infrastructure. Having measured the offset and delay a slave system is capable of routinely fully synchronizing its clock to the master clock.

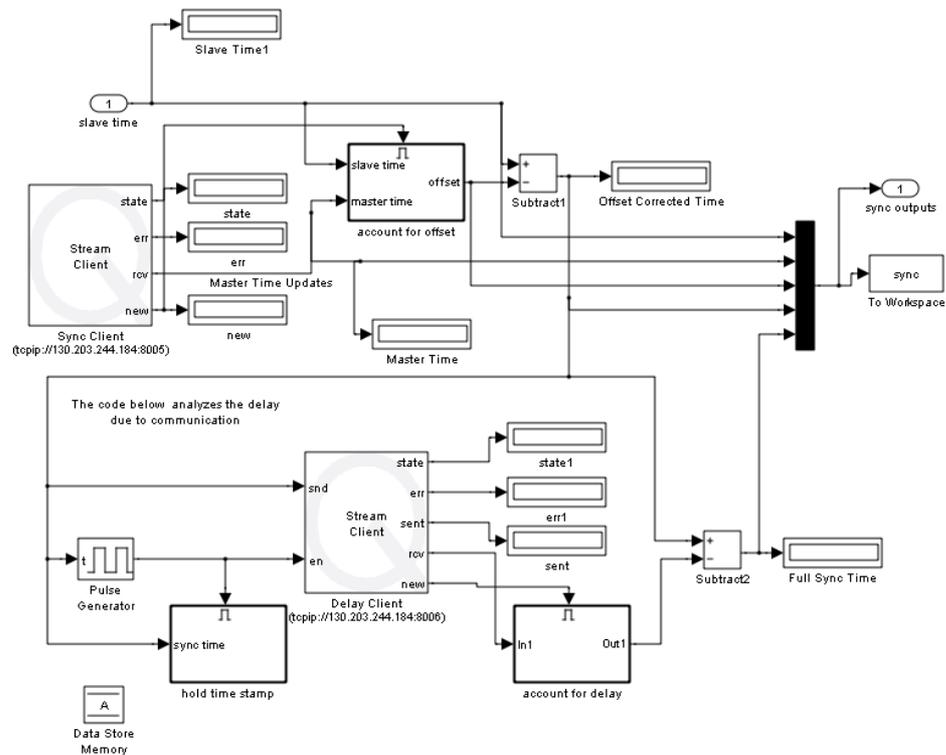
In the LIDAR experiment procedure, the IEEE 1588 standard was implemented across three laptop computers each collecting GPS data from three respective receivers. Although GPS is universally aligned and usually serves as a master time, while conducting previous experiments a drift in GPS time difference was observed between the low-cost San Jose GPS receivers, used in the prototype hardware packages, versus the high-cost Novatel GPS unit. The IEEE 1588 standard was followed and implemented in Mathwork’s Simulink software to synchronize the data collected on the laptop with the Novatel GPS unit attached to the laptops collecting GPS and collision data with the San Jose GPS receivers attached.

Appendix Figure 3 shows the Simulink diagram constructed for the master system. The input *master time* value is provided as the GPS timestamp produced by the high-end Novatel GPS unit. This GPS timestamp value represents the number of seconds that have elapsed since the beginning of the week and does not include leap second corrections.



Appendix Figure 3: Master System Simulink Diagram

Appendix Figure 4 shows the Simulink diagram built for the slave systems. Here the input *slave time* is provided as the GPS timestamp produced by the lower-cost San Jose GPS receivers. This GPS timestamp represents the number of seconds that have elapsed since the beginning of the day at 0:00:00. To compensate in the difference of the GPS timestamp formats the number of seconds since the beginning of the week is added to the San Jose GPS timestamp before providing as input to the slave block diagram. In this diagram the slave time, master time, measured offset, offset corrected time, and fully corrected time are all collected and saved in a data file for later analysis.



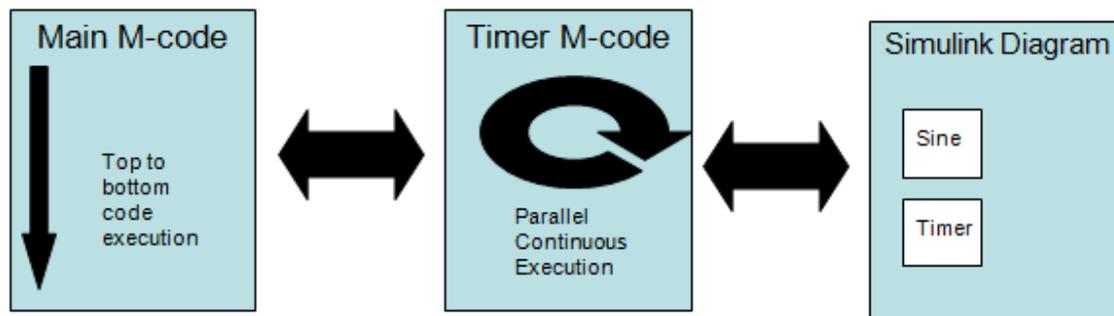
Appendix Figure 4: Slave System Simulink Diagram

The Simulink diagrams shown above were designed with plug and play use in mind and are available for use in other systems. They can be easily adopted into existing Simulink block diagram code as subsystems. It should be mentioned that QuaRC is used to communicate the corrections via wireless Ethernet. QuaRC is a 3rd party add-on for Simulink developed by Quanser to allow real-time processing of Simulink diagrams.

APPENDIX G CONTINUOUS PARALLEL PROCESS FOR ACCESING SIMULINK DATA IN MATLAB

Utilizing a timer function in a MATLAB script allows the user to break away from the traditional top-down execution of scripts and allows the user to incorporate parallel processes into their programming. One particularly useful feature of utilizing this is the ability to extract values from Simulink simulations at set intervals, allowing the user to have regularly updated variables from their simulation. The following provides a step-by-step example of how to use MATLAB to sample the value of a dynamic Simulink variable.. In this example we will create a Simulink diagram containing a timer and a sine function. Using a separate timer in MATLAB, the value of the Simulink blocks will be sampled and plotted every .1 seconds. Appendix Figure 5 is given as a schematic of overall code execution.

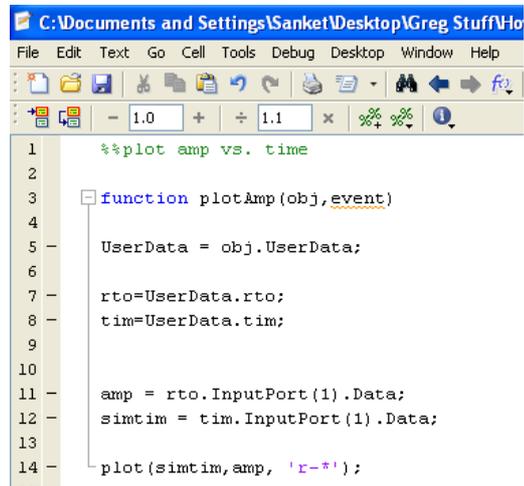
NOTE: One caveat of this technique is its “jitteriness” while executing.



Appendix Figure 5: Software Paradigm

1. First, a simple Simulink diagram is drawn using a timer and a sine function with two display blocks. The values of the two display blocks are what will be imported and plotted in MATLAB. The sine function block configuration is amplitude of 5 and a frequency of .01 radians/sec. The diagram uses a .1 second time step.

3. The function `plotAmp` is called when the timer executes. It adds the current values of the amplitude and time display blocks to a plot.



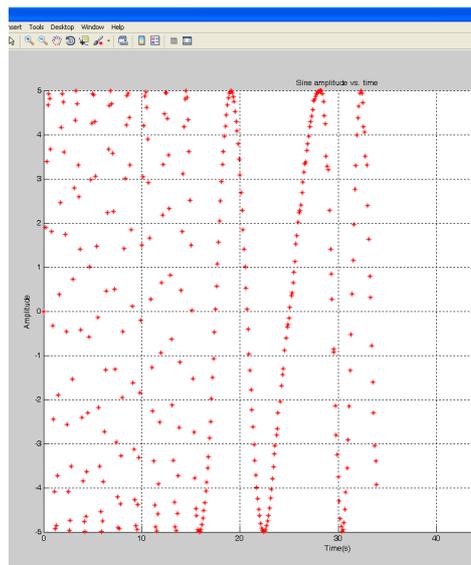
```

1 %%plot amp vs. time
2
3 function plotAmp(obj,event)
4
5     UserData = obj.UserData;
6
7     rto=UserData.rto;
8     tim=UserData.tim;
9
10
11     amp = rto.InputPort(1).Data;
12     simtim = tim.InputPort(1).Data;
13
14     plot(simtim,amp, 'r-+');

```

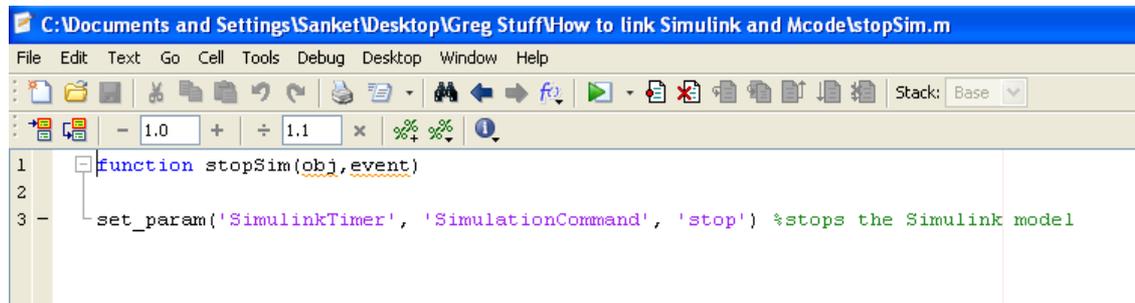
Appendix Figure 8: MATLAB M-Code for Timer Script

4. This is the plot that is created as `plotAmp` executes. As the timer executes, a new point is added to the plot. Note the beginning section of the graph where the points are scattered. These points were plotted while the program ran with lower precedence and exemplify the jittery nature of this technique.



Appendix Figure 9: Captured Simulink Results

5. When the user stops the timer by typing `stop(t2)` into the command line, the function `stopSim` is executed and the Simulink program is stopped.



The screenshot shows a MATLAB M-file editor window titled "C:\Documents and Settings\Sanket\Desktop\Greg Stuff\How to link Simulink and Mcode\stopSim.m". The editor displays the following code:

```

1 function stopSim(obj,event)
2
3 set_param('SimulinkTimer', 'SimulationCommand', 'stop') %stops the Simulink model

```

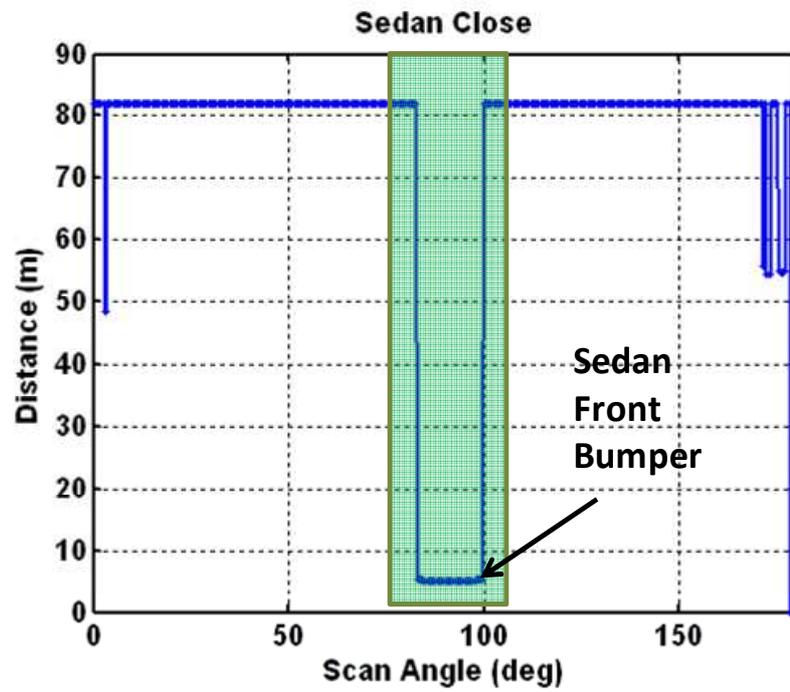
Appendix Figure 10: MATLAB M-Code for Stop Script

MATLAB References - topics found in the MATLAB help navigator

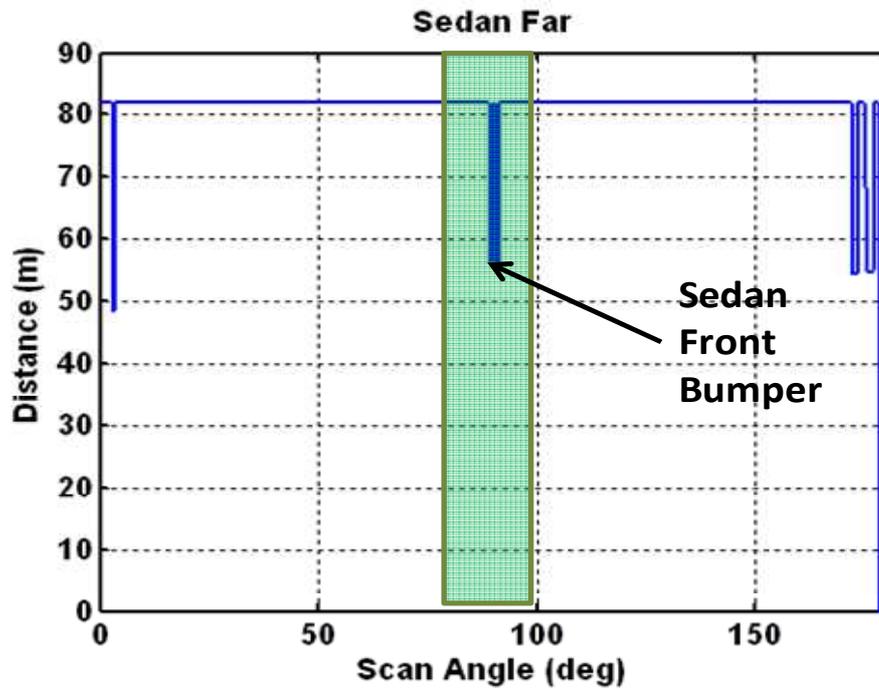
1. Timer Functions
 - a. Creating timer Objects
 - b. Timer Object Execution Modes
 - c. Working with Timer Object Properties
 - d. Starting a Timer
 - e. Deleting Timer Objects from Memory
2. Controlling Simulink Execution
 - a. `set_param`
 - b. Using the `set_param` Command
3. Real Time Objects
 - a. `Simulink.RunTimeBlock`

APPENDIX H LIDAR DATA ANALYSIS

The collected LIDAR and GPS data was post processed using MATLAB. First specific subsets of the data were extracted in relation to the specific test procedure executed. The subsets were classified as “Slow LVS”, “Fast LVS”, “1st Lap LVM”, and “2nd Lap LVM”. In each of these subsets of data, the LIDAR sensor data was first processed to infer inter-vehicle distance. Each scan of the LIDAR sensor produces 361 values representing the measured distance to an obstacle within a vision range of 0 to 180 degrees in ½ degree increments. For the data collected during the LVS procedure, a few scans were first manually visually reviewed to identify the region in the scan field occupied by the sedan. In the explanation of the results that follow, when the sedan is stationary and near the rear-end of the truck, approximate distance 20 meters or less, this event is referred as “In”. Accordingly when the sedan is distant from the truck, approximate distance of 60 meters or more, this event is referred as “Out”. **Error! Reference source not found.** and **Error! Reference source not found.** plot the scan data when the sedan was “In” and then “Out” respectively. The flattened shape of the obstacle, clearer visibility in **Error! Reference source not found.**, represents the front bumper of the sedan. As the sedan backs away from the truck, the obstacle width decreases due to the decrease in LIDAR scan angle resolution. Upon visual inspection, the sedan was consistently identified in an interval between 84 and 94 degrees. In an automated analysis approach, the minimum of the distance values in this sub-interval was observed as the ground truth inter-vehicle distance measurement.



Appendix Figure 11: Sampled LIDAR Scan Data While Sedan Is Near Rear-End of Truck



Appendix Figure 12: Sample LIDAR Scan Data While Sedan is Distant to Rear-end of Truck

During testing, in real-time, the GPS-based inter-vehicle distance estimate was calculated and stored in the data file. Since this real-time estimate includes the effect of wireless communication between the two computers the inter-vehicle estimate was again calculated off-line using the raw GPS information from the prototype hardware units as an additional comparison. Additionally, the inter-vehicle distance estimate was calculated using the raw GPS information but using the data collected by the high-quality Novatel GPS/IMU hardware instead of the prototype unit installed on the truck. To compare against the ground truth, an offset of 5.3 meters is removed from the estimate versions to compensate for the physical offset of the GPS collision hardware packages from the LIDAR sensor on the truck and from the front bumper on the sedan.