

## INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

**The quality of this reproduction is dependent upon the quality of the copy submitted.** Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

# UMI

A Bell & Howell Information Company  
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA  
313/761-4700 800/521-0600



LAND-VEHICLE NAVIGATION SYSTEMS: AN  
EXAMINATION OF THE INFLUENCE OF INDIVIDUAL  
NAVIGATION AIDS ON SYSTEM PERFORMANCE

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

By  
Eric Charles Abbott  
March 1997

**UMI Number: 9723313**

---

**UMI Microform 9723313**  
**Copyright 1997, by UMI Company. All rights reserved.**

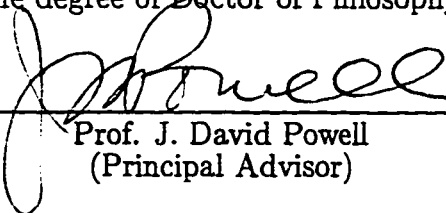
**This microform edition is protected against unauthorized  
copying under Title 17, United States Code.**

---

**UMI**  
**300 North Zeeb Road**  
**Ann Arbor, MI 48103**

© Copyright 1997 by Eric Charles Abbott  
All Rights Reserved

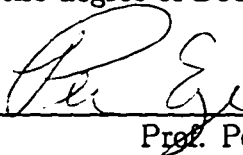
I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



---

Prof. J. David Powell  
(Principal Advisor)

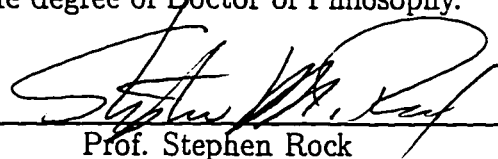
I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



---

Prof. Per Enge

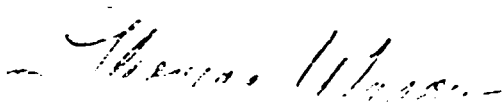
I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



---

Prof. Stephen Rock

Approved for the University Committee on Graduate Studies:



---

# Abstract

Traditionally, navigation systems have been very large, expensive and used only in aviation or military applications. However, recent advances in satellite-based positioning and the proliferation of small, low-cost motion sensors have made possible navigation systems that are small and inexpensive enough to be used in consumer products. Commercial consumer-grade navigation systems are, in fact, readily found today in Japan, Europe, and the United States, with one of the largest potential markets being in automobile navigation. Although the concept of in-vehicle navigation systems is not new, implementations of such systems are relatively recent. The research in this thesis advances the understanding of these systems through a quantitative examination of the impact that various navigation sensors have on the performance of a land-vehicle navigation system. A range of navigation sensor performance levels and their influence on vehicle positioning accuracy are examined. In addition, the impact of incorporating information from a digital map database in the navigation solution is also examined. The information produced by this research can help today's navigation system designers understand cost/performance tradeoffs in various candidate system designs. In addition, it can also help navigation system designers in the future, as the quality of navigation sensors improves through technological advancements. The work in this thesis can also be used to guide sensor designers—to reveal to them those sensor error parameters which contribute most to positioning error and to guide them into a design with appropriate performance tradeoffs.

Results show that, for a typical navigation system, positioning error is dominated by the accuracy of the position fixes provided by the Global Positioning System (GPS)

receiver when GPS position fixes are available and by the rate gyro's bias drift when GPS position fixes are not available. Furthermore, results show that the accuracy of the GPS fixes that are used has a significant impact on the relative contributions that various navigation sensor errors make. The implications of these results for navigation system design and sensor design are discussed. Finally, results show that using input from a digital map database to aid in navigation can degrade heading sensor calibration.

# Acknowledgements

I would first like to thank my advisor, Prof. David Powell, for his broad support. He skillfully gave me the right balance of freedom to pursue my own ideas and occasional nudges in the right direction. I am particularly grateful for his ability to see the big picture and for his broad experience. He knows the recipe for a good thesis, and I am grateful that he held my work to a high standard. I am also very grateful for his financial support, without which this thesis would not exist. I would also like to thank the members of my Reading Committee, Profs. Per Enge and Stephen Rock, for taking the time to read this thesis and give me their insightful comments. I am particularly grateful to Prof. Enge for his willingness to give me a research assistantship when funding was tight.

In addition, I am grateful to several corporations for their generous support of this research. I would like to express thanks to Gyration, Inc. and Daimler-Benz for their partial financial support of this research. Etak, Inc. and Navigation Technologies also generously supplied me with their electronic map databases of the Bay Area, and for this I am grateful.

The help of several fellow graduate students and colleagues—Y.C. Chao, Dr. Todd Walter, and Andrew Hansen—should not go without grateful thanks; and I am particularly grateful to Dr. Ran Gazit and Ping-Ya Ko for their insights and help in time of need. Naturally, there are many people outside of my professional circle that have supported me in my pursuit of this degree. My family and my wife's family have been extremely supportive, and, without them, I would never have arrived at Stanford. There is no doubt, however, that my wife has given me the most support. She is the one who has been there for me day-in and day-out, with her endless patience,

love, and genuine support. There is no other person who deserves greater praise. Finally, I give my greatest thanks to the Lord Jesus Christ. Without Him, I would be nothing, and it is His continuous guidance and love for me that gives my life joy and meaning. Psalm 27:1

# Contents

<b>Abstract</b>	<b>iv</b>
<b>Acknowledgements</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Historical Context . . . . .	1
1.2 Land-vehicle Navigation Concepts . . . . .	2
1.3 Research in This Thesis . . . . .	5
1.4 Previous Research . . . . .	6
1.5 Contributions . . . . .	8
1.6 Organization of the Following Chapters . . . . .	9
<b>2 Sensitivity Analysis</b>	<b>11</b>
2.1 Introduction . . . . .	11
2.2 The Equations of Sensitivity Analysis . . . . .	13
2.3 A Simple Example . . . . .	20
2.3.1 Problem Description . . . . .	20
2.3.2 Basic Analysis Results . . . . .	22
2.3.3 Extending the Analysis . . . . .	24
2.4 Extending Sensitivity Analysis to a Linearized Kalman Filter . . . . .	27
2.5 Summary . . . . .	34
<b>3 Sensor Error Models</b>	<b>35</b>
3.1 Introduction . . . . .	35

3.2	Rate Gyro Error Modeling . . . . .	36
3.2.1	Example Rate Gyros . . . . .	36
3.2.2	Rate Gyro Bias Drift . . . . .	37
3.2.3	Rate Gyro Scale Factor Error . . . . .	41
3.2.4	Equations for the Rate Gyro Error Model . . . . .	44
3.3	Magnetic Compass Error Modeling . . . . .	47
3.3.1	Compass Error Characteristics . . . . .	47
3.3.2	Equations for the Compass Error Model . . . . .	49
3.4	Odometer Error Modeling . . . . .	52
3.5	GPS Discussion and Error Modeling . . . . .	57
3.5.1	GPS with SA On . . . . .	58
3.5.2	GPS with SA Off . . . . .	58
3.5.3	Differential GPS . . . . .	59
3.5.4	Using GPS to Obtain a Heading Measurement . . . . .	62
3.6	Summary . . . . .	63
<b>4</b>	<b>Map-matching</b>	<b>65</b>
4.1	Introduction . . . . .	65
4.2	Factors in Successful Map-matching and Benefits of Map-matching to Navigation . . . . .	66
4.2.1	Initially Identifying the Correct Road . . . . .	68
4.2.2	Sustaining Successful Map-matching . . . . .	70
4.2.3	Using Map-matching Information to Aid Navigation . . . . .	71
4.3	Analyzing the Influence of Map-matching on Navigation System Performance . . . . .	74
4.4	A Detailed Description of a Map-matching Algorithm . . . . .	76
4.4.1	The Basis for Pattern-matching: Two Heuristic Observations . . . . .	76
4.4.2	Resolving Ambiguity with a Cost Function . . . . .	77
4.4.3	Updating Each Correction Over Time . . . . .	83
4.4.4	Using the Map-matched Position in the Kalman Filter . . . . .	84
4.4.5	Map-matching Results . . . . .	84

4.5	Summary . . . . .	88
<b>5</b>	<b>Analysis Details</b>	<b>89</b>
5.1	Introduction . . . . .	89
5.2	Simplifications . . . . .	89
5.2.1	Dealing with Trajectory Dependencies . . . . .	90
5.2.2	Simplifying Map-matching . . . . .	91
5.2.3	Choosing a Vehicle Trajectory . . . . .	92
5.2.4	Linearizing the Kalman Filter . . . . .	93
5.3	The Kalman Filter Equations . . . . .	94
5.3.1	The Kalman Filter Model Equations . . . . .	94
5.3.2	The Kalman Filter Measurement Equations . . . . .	99
5.4	The Equations of Sensitivity Analysis . . . . .	102
5.4.1	The Reference System Model Equations . . . . .	102
5.4.2	The Reference System Measurements Equations . . . . .	107
5.5	Summary . . . . .	108
<b>6</b>	<b>Results</b>	<b>109</b>
6.1	Introduction . . . . .	109
6.2	The Roles of Various Sensors While GPS Fixes Are Available . . . . .	110
6.3	The Influence of GPS Positioning Type . . . . .	114
6.3.1	The Influence of GPS on Sensor Calibration . . . . .	114
6.3.2	System Performance Without GPS Position Fixes . . . . .	118
6.4	The Influence of a Heading Measurement . . . . .	124
6.4.1	Using a Fluxgate Compass . . . . .	124
6.4.2	Using a GPS-based Heading Measurement . . . . .	127
6.5	The Influence of Vehicle Speed . . . . .	130
6.5.1	The Estimate of Cross-track Position . . . . .	130
6.5.2	The Estimate of Along-track Position . . . . .	131
6.6	The Influence of Odometer Resolution . . . . .	132
6.7	The Influence of Map-matching on Sensor Calibration . . . . .	134
6.7.1	The Effects of a Lane-change . . . . .	135

6.7.2	The Effects of Lateral Motion Within a Lane . . . . .	137
6.8	The Influence of Rate Gyro Scale Factor Errors and Turns . . . . .	140
6.9	Summary . . . . .	141
<b>7</b>	<b>Conclusions and Closing Remarks</b>	<b>144</b>
7.1	Conclusions . . . . .	144
7.1.1	Conclusions Drawn From Research Results . . . . .	144
7.1.2	Suggestions for Navigation System Design . . . . .	147
7.2	Closing Remarks . . . . .	148
<b>A</b>	<b>Numerical Values for Various Parameters</b>	<b>149</b>
	<b>Bibliography</b>	<b>153</b>

# List of Tables

2.1	Results of sensitivity analysis applied to an optimal filter . . . . .	22
2.2	Results of sensitivity analysis applied to a suboptimal filter . . . . .	24
6.1	Comparison of cross-track position error and GPS position error . . .	110
6.2	Relative contributions to mean-square error in cross-track position estimate . . . . .	111
6.3	Comparison of along-track position error and GPS positioning error .	112
6.4	Relative contributions to mean-square error in along-track position estimate . . . . .	113
6.5	Relative contributions to mean-square error in heading estimate . . .	115
6.6	Relative contributions to mean-square error in rate gyro bias estimate	116
6.7	Relative contributions to mean-square error in the odometer scale factor bias estimate . . . . .	117
6.8	Mean-square error in the odometer scale factor bias estimate for low bias drift . . . . .	118
6.9	Cross-track position error growth rate at 2 vehicle speeds . . . . .	131
6.10	RMS error in heading and rate gyro bias estimates at 2 vehicle speeds	131
6.11	Relative contributions to mean-square error in speed estimate for 3 odometer resolutions . . . . .	134
A.1	Spectral densities of process noise parameters in Kalman filter model equations . . . . .	150
A.2	Values for various constant parameters in Kalman filter model equations	150

A.3	RMS values for measurement noise parameters in Kalman filter measurement equations . . . . .	150
A.4	Spectral densities of process noise parameters in reference system equations . . . . .	151
A.5	Values for various constant parameters in reference system model equations . . . . .	151
A.6	RMS values for measurement noise parameters in reference system measurement equations . . . . .	152

# List of Figures

2.1	RMS error versus $\sqrt{Q}$ . . . . .	25
2.2	RMS error versus $\sqrt{R}$ . . . . .	25
2.3	Kalman gain as a function of process and measurement noise . . . . .	26
3.1	Forty-eight hours of data from two rate gyros . . . . .	38
3.2	Murata gyro transient . . . . .	39
3.3	Systron Donner gyro transient . . . . .	39
3.4	Gyrostar rate table results . . . . .	43
3.5	Gyrochip rate table results . . . . .	44
3.6	Compass data taken on a bridge and near power lines . . . . .	48
3.7	A schematic representation of an odometer . . . . .	53
3.8	Probability density function of $d_{k,k+1}$ . . . . .	55
3.9	Positioning error for Stanford's WAAS . . . . .	61
4.1	GPS position fixes overlaid on a map display . . . . .	67
4.2	Bias error due to map-matching on a two-lane road . . . . .	75
4.3	Lane changes on a multi-lane highway . . . . .	75
4.4	A nearly-constant correction applied to consecutive estimated positions	78
4.5	Many candidate corrections applied to a single estimated position . .	79
4.6	A demonstration of 3 possible corrections and their associated "costs"	80
4.7	Results showing the map-matching algorithm converge on the right road	86
4.8	The cumulative cost for each correction as a function of time . . . . .	87
4.9	Map-matched and Kalman filter location estimates versus time . . . . .	88

6.1	RMS error in cross-track position estimate after a GPS loss (SA on) .	119
6.2	RMS error in along-track position estimate after a GPS loss (SA on)	120
6.3	RMS error in cross-track position when using a 10-degree/hour gyro .	121
6.4	RMS error in cross-track position estimate after a GPS loss (SA off) .	122
6.5	RMS error in along-track position estimate after a GPS loss (SA off)	123
6.6	RMS error in cross-track position estimate after a GPS loss (DGPS) .	123
6.7	RMS error in along-track position estimate after a GPS loss (DGPS)	124
6.8	RMS error in cross-track position estimate when using a compass . .	125
6.9	Cross-track position error using a compass with and without magnetic disturbances . . . . .	126
6.10	RMS error in cross-track position estimate when using GPS attitude .	128
6.11	Comparison of RMS cross-track position error for 3 heading measure- ments . . . . .	129
6.12	Along-track position error with various odometer resolutions . . . . .	133
6.13	Heading error induced by map-matching during a lane-change . . . .	135
6.14	Gyro bias error induced by map-matching during a lane-change . . .	136
6.15	Heading error induced by map-matching as a result of motion within a lane . . . . .	138
6.16	Rate gyro bias error induced by map-matching as a result of motion within a lane . . . . .	139
6.17	Heading error induced by rate gyro scale factor error . . . . .	141

# Chapter 1

## Introduction

### 1.1 Historical Context

Traditionally, navigation systems have been very large, expensive and used only in aviation or military applications. However, recent advances in satellite-based positioning and the proliferation of small, low-cost motion sensors have made possible navigation systems that are small and inexpensive enough to be used in consumer products. Commercial consumer-grade navigation systems are, in fact, readily found today in Japan, Europe, and the United States, with one application being automobile navigation systems.

The concept of in-vehicle navigation systems is not new, but implementations of such systems have appeared only recently. Programs investigating the possibility of establishing an infrastructure to support widespread vehicle navigation began in the U.S. as early as the late 1960's. However, results from these studies deemed that the supporting infrastructure for such a system would be too expensive, and further study in the U.S. was dropped until the 1980's [53]. In the late 1980's, the U.S. government, recognizing that parts of the country's road system were taxed nearly to capacity, launched a campaign to promote the application of high-tech solutions to enhance roadway efficiency. Outlined in the National Program Plan for Intelligent Transportation Systems (NPP) [13], this campaign includes a strategy for improving the efficiency of the U.S. highway system over a 20-year period. The Plan's goals include

reducing highway congestion, fuel consumption, and the number of traffic accidents by providing drivers with real-time traffic information, route guidance, electronic toll collection, advanced vehicle collision avoidance systems, and automatic notification to authorities in the event of a traffic emergency. These ambitious renovations to the U.S. road system involve a number of diverse technologies, and knowledge of a vehicle's location lies at the heart of many services described in the NPP (e.g. route guidance and emergency response).

In Japan, research efforts in real-time automobile route guidance were begun in the 1970's with the goal of reducing traffic congestion. Throughout the 1970's and 1980's, the Japanese government, in cooperation with industry, was continuously involved in launching initiatives which helped to mature vehicle navigation technology [16]. Today, most Japanese car manufacturers offer factory-installed navigation systems in at least some of their models. Estimates indicate that, by the year 2000, per annum sales of vehicles with factory-installed navigation systems will reach 2.5 million [53].

## 1.2 Land-vehicle Navigation Concepts

This thesis deals with a specific technical aspect of vehicle navigation. However, because this is a relatively new field, the reader may not be familiar with the parlance of the vehicle navigation community. This section introduces the reader to several important concepts that are key to understanding the research in this thesis.

Simply put, the most basic function of a land-vehicle navigation system is to accurately identify the location of a vehicle. In many existing automobile navigation systems, this is typically achieved by an on-board computer that continuously collects data from sensors that are mounted inside the vehicle. The computer uses the sensor data to compute the vehicle's location and conveys this location to the driver by means of a graphical electronic display. Examples of positioning sensors in a typical navigation system include a Global Positioning System (GPS) receiver, a gyroscope, an electronic compass, and a tap into the automobile's odometer.

Although the purpose of the GPS is to provide its users with the ability to compute their location in 3-dimensional space, a land-vehicle navigation system cannot,

in general, continuously position a vehicle using a GPS receiver alone, and other navigation aids are necessary. In order to understand why this is so, one must first understand some basic facts about the GPS.

The GPS is a constellation of satellites in orbit around the Earth that is operated by the U.S. Department of Defense (DoD). Signals transmitted by the satellites can be received by appropriate equipment (a GPS receiver) on or near the Earth's surface, and the information in the signals can be utilized to compute the receiver's location in 3-dimensional space. The GPS can be used to perform 3-dimensional positioning worldwide under all weather conditions. However, in order to compute its location in 3-dimensional space, a GPS receiver must be able to lock onto signals from at least 4 different satellites. Moreover, the receiver must maintain its lock on each satellite's signal for a period of time that is long enough to receive the information encoded in the transmission. Achieving and maintaining a lock on 4 (or more) satellite signals can be impeded by solid objects that stand between the receiver and a satellite because the satellite signals are transmitted at a frequency (1.575 GHz) that cannot bend around or pass through solid objects. GPS receivers cannot be used indoors, for example, because the satellite signals cannot pass through a building's walls. Outdoors, tall buildings, dense foliage, or terrain that stand between a GPS receiver and a GPS satellite will block the satellite's signal. In urban or heavily-foliated environments, then, a GPS receiver may be unable to provide a position fix for indefinitely long periods of time. For this reason, an automobile navigation system cannot, in general, continuously position a vehicle using a GPS receiver alone.

Even if GPS position fixes *are* available, however, they contain errors and are accurate to only 100 meters (95% of the time). This error is unacceptably high because densely packed urban road networks generally contain roads that are less than 100 meters apart. (The inherent accuracy of the GPS is better than 100 meters. However, the signals from the GPS satellites have been *intentionally* degraded by the DoD for purposes of national security. This performance degradation is known as Selective Availability (SA), and only DoD-approved users have access to satellite signals without SA.)

Because GPS position fixes are inaccurate and may, at times, be unavailable altogether, many land-vehicle navigation systems utilize other navigation aids in conjunction with GPS position fixes to enhance overall system performance. These aids usually include some combination of sensors—e.g. low-cost gyroscopes, compasses, an odometer, inclinometers, and/or accelerometers. Any sensors other than GPS that are used to position the vehicle are collectively referred to as a *dead-reckoning unit*. Dead-reckoning sensors generally cannot be used alone to position a vehicle accurately for indefinitely long periods of time because dead-reckoning sensors, by definition, do not measure absolute position. Without an occasional measurement of absolute position, the error in a position estimate computed using dead-reckoning sensors alone grows without bound. Dead-reckoning sensors are utilized because they accurately measure changes in a vehicle's position over *short* time periods and can be used alone (for short time periods) if GPS position fixes become unavailable. GPS position fixes, in contrast, contain errors that are random and uncorrelated from one fix to the next, but the errors are bounded. The errors that appear in GPS position fixes and in the outputs of dead-reckoning sensors are therefore complementary in nature—dead-reckoning sensors smooth out the short-term GPS errors, and GPS fixes calibrate the dead-reckoning sensor drift over long time periods. Proper fusion of the GPS position fixes with the dead-reckoning sensor data can take advantage of these complementary errors, producing positioning performance that is better than could be obtained with either type of data alone.

In addition to GPS fixes and dead-reckoning sensors, many navigation systems utilize data from a digital map database to aid in navigation. A digital map database is essentially an electronic roadmap—a digitization of a local road network, with each road represented as a collection of points assumed to be connected in a dot-to-dot fashion. Information in a map database can be used to improve navigation accuracy if the vehicle is assumed to be traveling on a road stored in the database. The software algorithm that combines the sensor data with the map data to produce a position estimate is generally referred to as a *map-matching algorithm*. Map-matching algorithms are usually heuristic rules by which sensor data and information from the map database are processed to identify that road on which the vehicle is most likely to

be traveling. Map-matching algorithms described in the literature most often involve pattern-matching techniques that attempt to correlate the pattern created by several consecutive position fixes to a similar pattern of connected roads in the local road network. After a successful correlation is made, information about the matched road can be extracted from the database and used to calibrate errors in the navigation sensors.

### 1.3 Research in This Thesis

In light of the many possible combinations of navigation aids that can be used in these systems, one is led to question what criteria navigation system designers have used when selecting sensors for use in their vehicle navigation system. One could probably say with some certainty that the set of sensors selected by a design team is heavily influenced by the team's dual goals of maximizing the system's performance while minimizing its total cost. Unfortunately for system designers, however, system cost and performance are usually directly, rather than inversely, related—very accurate sensors may improve the performance of a system, but they tend to cost more than similar, less accurate sensors. Designers of land-vehicle navigation systems are therefore faced with trading off system cost and performance and must judiciously select that set of sensors deemed to be most cost-effective.

*The purpose of this thesis is to provide a quantitative and qualitative examination of the impact that individual navigation sensors have on the performance of various land-vehicle navigation systems.* The results of this research advance the understanding of the relationship between navigation *sensor* performance and overall *system* performance by means of analysis applied to various navigation systems. All of the navigation systems examined are similar in that they each utilize GPS position fixes and information from dead-reckoning sensors. The differences between systems lay primarily in *which* dead-reckoning sensors the systems utilize and the *accuracy* of the various sensor measurements. For example, many results are obtained for a system utilizing GPS position fixes, a rate gyro, and an odometer. This sensor set was chosen because it is frequently encountered in existing land-vehicle navigation systems. The

performance of this set of sensors is examined for various rate gyro performance levels and various GPS position fix accuracies. Other results are obtained for a system utilizing GPS position fixes, a rate gyro, an odometer, and a compass. The performance of this system is examined for various GPS position fix accuracies and for a range of compass errors. Still other results are obtained for a system utilizing GPS position fixes, a rate gyro, an odometer, and map-matching.

The quantitative results of this thesis immediately reveal the influence that individual navigation sensor error parameters have on navigation system performance. These quantitative results should therefore be valuable for identifying the most cost-effective navigation system designs. The qualitative results of this work should be valuable to the land-navigation community as a practical reference for future navigation system designs, and the analysis techniques used in this thesis should be a valuable model for navigation system analysts.

## 1.4 Previous Research

Many papers and patents have been published which discuss various algorithms for combining the information obtained from various sensors and navigation aids for use in a land-vehicle navigation system [9, 17, 25, 24, 26, 28, 30, 29, 41, 34, 37, 39, 43, 48, 49, 50, 51, 56, 61, 62, 63, 64, 68, 69]. However, relatively little analytical or quantitative work seems to have been done to establish rationales for sensor selection. Nor has much work been done to quantify the relative contributions that various navigation sensors make to overall system performance.

The work in this thesis is most closely related to work in [10]. In [10], the authors examined the relative contributions that various navigation sensors made to the navigation errors in an aircraft's navigation system. The goal of the work in [10] is similar to the goal of this thesis. In addition, the analysis technique developed in [10] served as the inspiration for the analysis in this thesis. However, the problems being solved differ substantially, and the results in [10] do not carry over to the problem presented in this thesis. For example, in [10], the navigation system included a high-quality 3-axis inertial system, complemented with LORAN position fixes. In

this thesis, not only are entirely different navigation sensors used, but the quality of the sensors being examined (see Chapter 3) differs substantially from that of the sensors examined in [10]. The scope of this work is also more inclusive, examining several navigation systems and the use of other less-traditional navigation aids (e.g. input from a digital map). In addition, the research in this thesis required extensions to the theory presented in [10] that had to be developed by the author (see Section 2.4).

In other related work, the authors of [55] discuss the effects of inertial sensor quality on the performance of a navigation system; however, this work focuses on military-grade navigation systems, which are generally far too expensive to be practicable for commercial land-vehicle use. In [42], the author presents a simulation study in which the relative merits of two inertial navigation systems for use in a Mars rover are examined. Certain elements of the work in [42] are similar to elements of the research in this thesis, but there are important differences. For example, the author of [42] examined the sensitivity of the each navigation system's performance to perturbations in various sensor parameters. The author's results identified those sensor errors to which the total navigation error was most sensitive, thereby identifying the most important sensor errors. In this sense, the work in [42] is similar to the research in this thesis. However, the principle focus of the work in [42] was the evaluation of two navigation *systems*, not individual sensor contributions. The author's perturbation study did not quantify, for a *given* set of sensor parameters, the individual contributions that the sensor errors made to the total navigation error. Also, one of the systems was comprised of 3 accelerometers and 3 gyroscopes, but this combination of sensors is generally not found in existing automobile navigation systems. Finally, the author assumed that the vehicle moved at a maximum speed of 1.0 meter per second, a speed that is much lower than is typical of an automobile. In an earlier work, [32], the author enumerates various error sources in a particular vehicle navigation system. However, the navigation system examined used only LORAN-C to position the vehicle; dead-reckoning sensors were not utilized. Finally, in [35], the author presents a methodology for evaluating a land-vehicle navigation system by assigning it a "score" based on a host of criteria. The purpose of the scoring method

is to provide an objective basis by which to compare systems. However, the author's scoring system incorporates a wide variety of evaluation criteria, including functional features, cost, power consumption, reliability, etc. The author does not address the relative merits of individual navigation sensors.

## 1.5 Contributions

The contributions of this work include a quantification of the contributions that individual sensors and error parameters make to the performance of a land-vehicle navigation system. Part of this contribution includes an investigation into the role that low-cost motion sensors play in navigation system performance. However, it also includes an investigation into the impact that various types of GPS position fixes have on navigation system performance. Currently, the accuracy of GPS position fixes is intentionally degraded by SA. However, a policy statement recently issued by the White House indicates that SA will be turned off before the year 2006 [11], and the accuracy of GPS position fixes will improve significantly. In addition, a more accurate form of GPS positioning known as *differential GPS positioning* (DGPS) may soon become widespread. These changes in the accuracy of GPS position fixes could have a significant impact on the performance and evolution of land-vehicle navigation systems. This research investigates the impact that each type of GPS positioning has on navigation system performance.

The information produced by this research can help today's navigation system designers understand tradeoffs in various candidate system designs. However, it can also help navigation system designers in the future, when Selective Availability is turned off or DGPS becomes widely available. This contribution can also be used to guide sensor designers—to reveal to them those sensor error parameters which contribute most to positioning error and to guide them into a design with appropriate performance tradeoffs. Another part of this contribution is the application of analysis techniques to low-cost navigation systems. While a similar analysis technique was applied to a high-end inertial navigation system in [10], this is the first published analysis of a modern low-cost navigation system that includes low-cost sensors, GPS,

and map-matching. A second contribution of this work includes the development of error models for various low-cost dead-reckoning sensors and the design of a Kalman filter for data fusion. Finally, this work includes an original map-matching algorithm developed by the author. In summary, the contributions of this research are

- Quantitative analysis of the contributions that various dead-reckoning sensor errors make to the positioning accuracy of a low-cost navigation system
- Quantitative analysis of the impact of various types of GPS positioning on navigation system performance
- Development of original error models for various low-cost dead-reckoning sensors
- An original Kalman filter design for navigation sensor fusion
- Developed extensions to the theory of sensitivity analysis so that it may be applied to a linearized Kalman filter
- An original map-matching algorithm

## 1.6 Organization of the Following Chapters

In Chapter 2, the theoretical underpinnings for the work in this thesis are presented. In this chapter, an analysis tool known as *sensitivity analysis* is presented. The content of this chapter is independent of the specific problem being solved in this research—it has been written in a tutorial way, to give the reader an understanding of the analysis techniques used in this research without reference to specifics of the problem being solved. This chapter includes a discussion of sensitivity analysis as it appears in [10], but also presents extensions to the work in [10] that were developed by this author for this research. The next chapter, Chapter 3, lays out error models for various sensors that are commonly used in automobile navigation. Justification for the error models is discussed, and the mathematical models are also presented. Chapter 4 is devoted to a discussion of map-matching and the manner in which it is treated in this thesis. Specifically, various map-matching issues are discussed including factors

involved in successful map-matching and the benefits of map-matching to navigation system performance. Finally, a map-matching algorithm developed by the author is presented. This chapter could rightfully be included with Chapter 3, but map-matching issues deserve a chapter to themselves. Chapters 2, 3 and 4 together provide the complete basis for the analysis work in this thesis. Chapter 5 ties the information from the previous chapters together by presenting detailed equations of the analysis. This chapter also discusses some important assumptions that were made to simplify the analysis. Finally, Chapter 6 contains results, and Chapter 7 contains conclusions.

# Chapter 2

## Sensitivity Analysis

### 2.1 Introduction

As was mentioned in Chapter 1, existing land-vehicle navigation systems often include a GPS receiver and a dead-reckoning unit, and the errors appearing in GPS position fixes and the outputs of dead-reckoning sensors are complementary—GPS position fixes do not drift but contain random errors that tend to be uncorrelated from one fix to the next, and dead-reckoning sensors accurately measure changes in position over short time periods but drift over long time periods. Proper fusion of GPS position fixes with dead-reckoning sensor data can take advantage of these complementary errors, producing positioning performance that is better than could be obtained with either type of data alone. One algorithm for combining data with complementary characteristics is known as a *Kalman filter*. The advantages of the Kalman filter are such that it is well-suited for use in fusing GPS position measurements with data from inertial instruments, and, for this reason, is frequently utilized in the navigation community. Furthermore, the Kalman filter provides a mathematical framework that is conducive to analysis. For these reasons, a Kalman filter will be used in this research.

Many texts are devoted to a discussion of Kalman filtering ([19], for example), and the reader is referred to one of them for details. Suffice it to say here that a Kalman filter is a statistically optimal means for estimating quantities whose time histories

can be approximately modeled and for which a related measurement is available. In the state-space formulation of the Kalman filter, the quantities being estimated are generally encapsulated in a column matrix known as the filter's *state vector*. The time history of the state vector can presumably be modeled by linear stochastic differential equations referred to as the filter's *model equations*. The forcing function for these equations is assumed to be zero-mean white noise with a Gaussian distribution and a known spectral density matrix. In addition to the model equations, a measurement of a linear combination of the states is assumed to be available; the relationship between the measurement vector and the state vector is referred to as the *measurement equation*. The measurements may be corrupted with additive zero-mean white noise with a Gaussian distribution and another known spectral density matrix.

In order for a Kalman filter to produce a statistically optimal estimate of its state, the filter's model equations, measurement equations and spectral density matrices must exactly describe the actual dynamical and statistical properties of the system of interest. In other words, the time-history of the system's state must be described *exactly* by known linear stochastic differential equations driven by white Gaussian noise with known statistical properties. However, it is frequently the case that the dynamical equations that exactly describe the behavior of a system are not linear or are not known precisely. Moreover, it may be the case that the number of states required to accurately model the system would be so large that the computational requirements of the filter mechanization would exceed available computational capacity. Under these circumstances, the filter designer must resort to a reduced-state Kalman filter, capturing the essential behavior of the system with fewer states than are required to model it exactly. Whenever the equations in a Kalman filter mechanization do not model the behavior of the physical system at hand exactly for any reason—whether due to imprecise knowledge of the physical system or due to deliberate reduction of the state—the Kalman filter will produce a suboptimal state estimate. The filter is thus referred to as a “suboptimal” filter.

The error in the state estimate produced by a suboptimal filter will be greater than that produced by an optimal filter. The extent to which the performance of a suboptimal filter deviates from optimality may be quantified using a tool known as

*sensitivity analysis.* Sensitivity analysis can be extremely enlightening because it can reveal a great deal about various error mechanisms in a Kalman filter. Historically, sensitivity analysis has been used to quantify the sensitivity of a particular suboptimal Kalman filter's performance to perturbations in parameters which appear in the filter's equations. The research here, however, is not concerned with suboptimal filter design *per se*. Instead, the goal of this research is to quantify the relative contributions that various navigation sensors make to the total navigation error. Fortunately, the equations of sensitivity analysis can be formulated in such a way that the contributions that individual error sources make to the total error in a Kalman filter's estimate can be computed. Sensitivity analysis therefore has uses beyond its historical applications. In this chapter the equations of sensitivity analysis are derived, thus establishing the theoretical underpinnings for the analyses appearing in later chapters. First, the equations of sensitivity analysis are presented for a linear Kalman filter. Next, a simple example is given to illustrate the use of sensitivity analysis to the reader. At the end of the chapter, extensions that are required for this research are made to the basic equations of sensitivity analysis—specifically, the equations of sensitivity analysis are rederived for a linearized Kalman filter.

## 2.2 The Equations of Sensitivity Analysis

The subject of sensitivity analysis has been studied in [3], [8], [10], and [21]. The formulation in [10] is more general than those in [3] and [21] and is more relevant to this research than the formulation in [8]. The derivation that follows is therefore most closely related to that of [10]. The derivation presented in this section will be extended further in Section 2.4 to accommodate a special form of the Kalman filter that is utilized in the analyses in this thesis. (In the notation which follows, column matrices are denoted with lowercase bold type, scalars are denoted with lowercase plain type, and matrices are denoted with uppercase type.)

The derivation begins by assuming that the physical system at hand can be precisely modeled by a set of linear stochastic difference equations. All of the parameters governing these equations are assumed to be known, and the equations should model

the true behavior of the system as accurately as is reasonably possible. This set of equations will henceforth be referred to as the “reference” system because the sensitivity analysis will quantify the Kalman filter’s performance against (or, *in reference to*) this system. Finally, it should be noted that these equations need not be identical to the model equations of the Kalman filter in that they may include states that the Kalman filter cannot reliably estimate or constants whose true values are not known exactly.

This being said, we may begin the derivation. The reference system’s state evolves in time according to the following model:

$$\mathbf{x}_{r,k+1} = \Phi_{r,k} \mathbf{x}_{r,k} + \mathbf{w}_{r,k} \quad (2.1)$$

In this equation, the reference system state vector,  $\mathbf{x}_{r,k}$ , has dimensions  $n \times 1$ , the reference system *state transition matrix*,  $\Phi_{r,k}$ , has dimensions  $n \times n$ , and  $\mathbf{w}_{r,k}$  is the  $k^{th}$  element of an uncorrelated random sequence with covariance matrix  $Q_{r,k}$  and a Gaussian distribution. The quantity  $\mathbf{w}_{r,k}$  is generally referred to as *process noise*. The variable  $k$  is a discrete-time index representing time  $t = kT$ , where  $T$  is the discretization period of the system. The subscript “r” distinguishes these quantities as being associated with the reference system (as opposed to the Kalman filter).

The measurement vector,  $\mathbf{z}_{r,k}$ , has dimensions  $p \times 1$  and is assumed to be a linear combination of the system states:

$$\mathbf{z}_{r,k} = H_{r,k} \mathbf{x}_{r,k} + \mathbf{v}_{r,k} \quad (2.2)$$

where  $H_{r,k}$  is the reference system observation matrix and  $\mathbf{v}_{r,k}$  is the  $k^{th}$  element of a normally-distributed uncorrelated random sequence with covariance  $R_{r,k}$ .

The output of the system is defined by the  $s \times 1$  column matrix  $\mathbf{y}_{r,k}$ , which is a linear combination of the system states:

$$\mathbf{y}_{r,k} \equiv C_{r,k} \mathbf{x}_{r,k} \quad (2.3)$$

where  $C_{r,k}$  is a matrix of dimension  $s \times n$ .

The Kalman filter state vector,  $\mathbf{x}_{f,k}$ , which is an  $m \times 1$  column matrix, evolves in time according to the equation

$$\mathbf{x}_{f,k+1} = \Phi_{f,k}\mathbf{x}_{f,k} + \mathbf{w}_{f,k} + B_k\mathbf{u}_k \quad (2.4)$$

where  $\Phi_{f,k}$  has dimensions  $m \times m$ ,  $\mathbf{w}_{f,k}$  is process noise (an uncorrelated Gaussian sequence of covariance  $Q_{f,k}$ ),  $B_k$  has dimensions  $n \times j$ , and  $\mathbf{u}_k$  is a deterministic input to the filter of dimension  $j \times 1$ . Note that the subscript “f” in this (and subsequent) equations denotes a quantity that is associated with the filter (as opposed to the reference system).

The *actual* measurements read from the sensors and used by the filter are a linear combination of the *reference* states, the model for which is given in Equation 2.2. The filter’s *model* for the measurement vector, however, may be different from the actual measurement vector. Accordingly, the filter “believes” that the measurement taken is actually modeled by the equation

$$\mathbf{z}_k = H_{f,k}\mathbf{x}_{f,k} + \mathbf{v}_{f,k} \quad (2.5)$$

where  $\mathbf{v}_{f,k}$  is the  $k^{th}$  element of a normally-distributed  $p \times 1$  uncorrelated sequence with  $p \times p$  covariance matrix  $R_{f,k}$ .

At each timestep, a measurement is taken and the Kalman filter generates an estimate of the state vector. The state estimate is recursively computed using the well-known Kalman filter algorithm:

$$K_k = P_{f,k}^- H_{f,k}^T [H_{f,k} P_{f,k}^- H_{f,k}^T + R_{f,k}]^{-1} \quad (2.6)$$

$$P_{f,k}^+ = [I - K_k H_{f,k}] P_{f,k}^- \quad (2.7)$$

$$\hat{\mathbf{x}}_{f,k}^+ = \hat{\mathbf{x}}_{f,k}^- + K_k [z_{\tau,k} - H_{f,k} \hat{\mathbf{x}}_{f,k}^-] \quad (2.8)$$

$$\hat{\mathbf{x}}_{f,k+1}^- = \Phi_{f,k} \hat{\mathbf{x}}_{f,k}^+ + B_k \mathbf{u}_k \quad (2.9)$$

$$P_{f,k+1}^- = \Phi_{f,k} P_{f,k}^+ \Phi_{f,k}^T + Q_{f,k} \quad (2.10)$$

where  $K_k$  is the Kalman gain,  $P_k^+$  is the covariance of the estimate error after a measurement has been processed,  $\hat{\mathbf{x}}_{f,k}^+$  is the estimate of the state based on measurements up to and including timestep  $k$ ,  $\hat{\mathbf{x}}_{f,k}^-$  is the predicted state, and  $P_{k+1}^-$  is the covariance of the estimate error projected to timestep  $k+1$ . Note that the “+” superscript denotes a quantity *after* a measurement update, and the “-” superscript denotes a quantity *before* a measurement update. More details describing the Kalman filter can be found in [19].

The output of the Kalman filter is defined by the  $s \times 1$  column matrix  $\mathbf{y}_{f,k}$ , which is a linear combination of the estimate of the state:

$$\mathbf{y}_{f,k} \equiv C_{f,k} \hat{\mathbf{x}}_{f,k}^+ \quad (2.11)$$

where  $C_{f,k}$  is a matrix of dimension  $s \times m$ .

Finally, the error between the output of the reference system and the output of the Kalman filter is defined as

$$\mathbf{e}_k \equiv \mathbf{y}_{r,k} - \mathbf{y}_{f,k} \quad (2.12)$$

and define the covariance of  $\mathbf{e}_k$  as

$$P_{e,k} \equiv E[\mathbf{e}_k \mathbf{e}_k^T] \quad (2.13)$$

At this point, an important constraint on the exogenous input to the Kalman filter,  $\mathbf{u}_k$ , must be established. Namely, it is required that  $\mathbf{u}_k$  be the sum of a linear combination of the reference states and an additive noise term that is assumed to be normally-distributed and uncorrelated with any states or noise sources in the system. Mathematically,  $\mathbf{u}_k$  can be expressed as

$$\mathbf{u}_k = \Gamma_k \mathbf{x}_{r,k} + \psi_k \quad (2.14)$$

This assumption is not too restrictive, but may require  $\mathbf{x}_{r,k}$  to be augmented with states that define the time history of  $\mathbf{u}_k$ . Note that  $\Gamma_k$  has dimensions  $j \times n$  and  $\psi_k$

has dimensions  $j \times 1$ .

The goal of this derivation is to arrive at a recursive expression for the covariance matrix  $P_{e,k}$ . To this end, a single *combined state vector*,  $\mathbf{x}_{c,k}$ , is constructed by stacking  $\mathbf{x}_{r,k}$  and  $\hat{\mathbf{x}}_{f,k}$ :

$$\mathbf{x}_{c,k} \equiv \begin{bmatrix} \mathbf{x}_{r,k} \\ \hat{\mathbf{x}}_{f,k} \end{bmatrix} \quad (2.15)$$

so that

$$\mathbf{x}_{c,k}^- \equiv \begin{bmatrix} \mathbf{x}_{r,k} \\ \hat{\mathbf{x}}_{f,k}^- \end{bmatrix} \quad (2.16)$$

and

$$\mathbf{x}_{c,k}^+ \equiv \begin{bmatrix} \mathbf{x}_{r,k} \\ \hat{\mathbf{x}}_{f,k}^+ \end{bmatrix} \quad (2.17)$$

Utilizing Equations 2.1, 2.9, and 2.16, one can arrive at an expression that describes the propagation of  $\mathbf{x}_{c,k}^+$  in time from timestep  $k$  to timestep  $k+1$ :

$$\mathbf{x}_{c,k+1}^- = \begin{bmatrix} \Phi_{r,k} & 0 \\ B_k \Gamma_k & \Phi_{f,k} \end{bmatrix} \mathbf{x}_{c,k}^+ + \begin{bmatrix} \mathbf{w}_{r,k} \\ B_k \psi_k \end{bmatrix} \quad (2.18)$$

It is convenient to define  $\Phi_{c,k}$  and  $\mathbf{w}_{c,k}$  so that Equation 2.18 can be rewritten as follows:

$$\mathbf{x}_{c,k+1}^- = \Phi_{c,k} \mathbf{x}_{c,k}^+ + \mathbf{w}_{c,k} \quad (2.19)$$

where

$$\Phi_{c,k} \equiv \begin{bmatrix} \Phi_{r,k} & 0 \\ B_k \Gamma_k & \Phi_{f,k} \end{bmatrix} \quad (2.20)$$

and

$$\mathbf{w}_{c,k} \equiv \begin{bmatrix} \mathbf{w}_{r,k} \\ B_k \psi_k \end{bmatrix} \quad (2.21)$$

Utilizing Equations 2.8 and 2.17, one can produce an expression that describes the evolution of  $\mathbf{x}_{c,k}^-$  when a measurement is processed at timestep  $k$ :

$$\mathbf{x}_{c,k}^+ = B_{c,k} \mathbf{x}_{c,k}^- + K_{c,k} \mathbf{v}_{r,k} \quad (2.22)$$

where

$$B_{c,k} \equiv \begin{bmatrix} I & 0 \\ K_k H_{r,k} & I - K_k H_{f,k} \end{bmatrix} \quad (2.23)$$

and

$$K_{c,k} \equiv \begin{bmatrix} 0 \\ K_k \end{bmatrix} \quad (2.24)$$

The error term  $e_k$  is given by

$$e_k = C_{c,k} x_{c,k}^+ \quad (2.25)$$

where

$$C_{c,k} \equiv \begin{bmatrix} C_{r,k} & -C_{f,k} \end{bmatrix} \quad (2.26)$$

The covariance of  $x_{c,k}^-$  is derived from Equation 2.19 and is given by

$$P_{c,k+1}^- = \Phi_{c,k} P_{c,k}^+ \Phi_{c,k}^T + Q_{c,k} \quad (2.27)$$

where

$$Q_{c,k} \equiv \begin{bmatrix} Q_{r,k} & 0 \\ 0 & B_k \Psi_k B_k^T \end{bmatrix} \quad (2.28)$$

and  $\Psi_k$  is the  $j \times j$  covariance matrix of  $\psi_k$ .

The covariance of  $x_{c,k}^+$  is derived from Equation 2.22 and is given by

$$P_{c,k}^+ = B_{c,k} P_{c,k} B_{c,k}^T + K_{c,k} R_{r,k} K_{c,k}^T \quad (2.29)$$

Finally, the covariance of the error term  $e_k$  is given by

$$P_{e,k} = C_{c,k} P_{c,k}^+ C_{c,k}^T \quad (2.30)$$

The diagonal elements of  $P_{e,k}$  represent the mean-square value of the difference between  $y_{r,k}$  (the output of the reference system) and  $y_{f,k}$  (the output of the filter). As such, they collectively represent the “true” mean-square error in the filter’s estimates and are therefore the terms of interest.

Equations 2.27, 2.29, and 2.30 together with the initial condition for  $P_{c,k}^-$  form the so-called *equations of sensitivity analysis*. The power of sensitivity analysis lies in the fact that the diagonal elements of  $P_e$  are *linear* in  $R_r$ ,  $Q_r$ , and  $P_r(0)$ . Hence, the equations of sensitivity analysis are difference equations that are linear in their forcing functions ( $R_r$  and  $Q_r$ ) and their initial condition ( $P_r(0)$ ). This important result implies that *the total effect of all of the terms in  $R_r$ ,  $Q_r$ , and  $P_r(0)$  on  $P_e$  is equal to the sum of the effects caused by each individual term in  $R_r$ ,  $Q_r$ , and  $P_r(0)$* . Therefore, each term in  $R_r$ ,  $Q_r$ , and  $P_r(0)$  may be played through the sensitivity equations *one at a time*, with all other terms in these matrices set to zero; the resulting diagonal terms in  $P_e$  represent the contribution that the non-zero term *alone* makes to the mean-square error in the filter's state estimate. The sensitivity equations must be run many times—one time for each term in  $R_r$ ,  $Q_r$ , and  $P_r(0)$  whose nominal value is non-zero. After the individual contributions of each term in  $R_r$ ,  $Q_r$ , and  $P_r(0)$  have been computed, the contributions may be summed to obtain the total mean-square error in the Kalman filter's estimates.

The catalog of the contributions that all terms in  $R_r$ ,  $Q_r$ , and  $P_r(0)$  make to each filter state at any instant in time is referred to as an *error budget* for that point in time. An error budget is a snapshot of the contributions that all error sources make to the total estimation error. Inspection of a complete error budget immediately reveals those error sources that contribute the most to the errors in the filter's estimates. Such results can contribute greatly to an understanding of the error mechanisms in a Kalman filter.

The error budget is also useful in other ways. Once a component error budget is established, one can easily compute the sensitivity of the filter's estimate error to changes in a particular error source. This can be done easily because it is not necessary to repeat the sensitivity analysis. This convenient result arises because the mean-square estimate error is a linear function of each error source (i.e. each term in  $R_r$ ,  $Q_r$ , and  $P_r(0)$ ). As a result of this linear relationship, an increase in the mean-square value of an error source will cause its contribution to the mean-square error in the estimate of each state to increase in direct proportion. For example, doubling the mean-square value of the random noise in a measurement will cause the contribution

of that noise term to the error in every state estimate to double, as well. This result assumes that the *real-world* (i.e. reference system) error sources change and that the Kalman filter design remains *fixed*, and is a convenient means for examining a filter's sensitivity to variations in real-world parameters.

The simple example that follows will demonstrate the use of sensitivity analysis and will demonstrate its usefulness in revealing the error mechanisms in a Kalman filter.

## 2.3 A Simple Example

### 2.3.1 Problem Description

Let us suppose that we seek to estimate the value of a quantity whose time history is described by a first-order Gauss-Markov process, and that a measurement of this quantity is available but is corrupted additive white noise. In this example, we shall utilize a Kalman filter to estimate this quantity, but will assume that the values for the parameters which govern this process are not known exactly or change over time. We shall employ sensitivity analysis to explore the manner in which such parameter changes affect the Kalman filter's estimate error and shall use our results to gain insight into the error mechanisms of the filter.

The reference system, which describes the *actual* evolution of the state  $x_r$ , is given by

$$\dot{x}_r = -\frac{1}{\tau_r}x_r + u_r \quad (2.31)$$

where  $u_r$  is normally-distributed white noise with spectral density matrix  $Q_r$ :

$$u_r = N(0, Q_r) \quad (2.32)$$

The Kalman filter's model equation is given by

$$\dot{x}_f = -\frac{1}{\tau_f}x_f + u_f \quad (2.33)$$

where

$$u_f = N(0, Q_f) \quad (2.34)$$

The actual measurement is given by

$$z = x_r + v_r \quad (2.35)$$

in which

$$v_r = N(0, R_r) \quad (2.36)$$

The value for the spectral density of the measurement noise that is used when computing the Kalman gain is denoted  $R_f$ . The value of  $R_f$  may not be equal to the value of the true spectral density,  $R_r$ , because we wish to simulate inaccurate knowledge of the spectral density of  $v_r$ . Therefore

$$v_f = N(0, R_f) \quad (2.37)$$

To maintain consistency with the discrete-time formulation in Section 2.2, the matrices for the sensitivity analysis are given in discrete time:

$$\Phi_c = \begin{bmatrix} \exp\left(-\frac{T}{\tau_r}\right) & 0 \\ 0 & \exp\left(-\frac{T}{\tau_f}\right) \end{bmatrix} \quad (2.38)$$

$$Q_c = \begin{bmatrix} \frac{Q_r \tau_r}{2} \left[1 - \exp\left(-\frac{2T}{\tau_r}\right)\right] & 0 \\ 0 & 0 \end{bmatrix} \quad (2.39)$$

$$x_c = \begin{bmatrix} x_r(0) \\ 0 \end{bmatrix} \quad (2.40)$$

$$P_{c,0}^- = \begin{bmatrix} \frac{Q_r \tau_r}{2} & 0 \\ 0 & 0 \end{bmatrix} \quad (2.41)$$

where  $x_r(0)$  is a random number with a mean of zero and a Gaussian distribution with variance  $\frac{Q_r \tau_r}{2}$ , and  $T$  is the discretization period of the system.

Contribution of $P_0$	0.0 (0 %)
Contribution of $Q_r$	0.212 (65.6 %)
Contribution of $R_r$	0.111 (34.4 %)
Total Mean-square Error	0.323
Filter's Predicted Mean-square Error	0.323

Table 2.1: Results of sensitivity analysis applied to an optimal filter

### 2.3.2 Basic Analysis Results

We begin the analysis with an optimal Kalman filter—that is, one for which  $\tau_f = \tau_r$ ,  $Q_f = Q_r$ , and  $R_f = R_r$ . Letting  $\tau_r = \tau_f = 1.0$ ,  $R_r = R_f = 1.0$ ,  $Q_r = Q_f = 1.0$ , and  $T = 1.0$ , we first run through the sensitivity analysis with  $R_r$  and all of the elements in  $Q_c$  and  $P_{c,0}$  set to their nominal values. In doing so, we find that, when the filter reaches steady state, the total mean-square error in the estimate of  $x$  is 0.323.

In general, the error in the filter's estimate of  $x$  results from 3 sources of uncertainty: uncertainty in the initial value of  $x$  (i.e.  $P_r(0)$ ), process noise ( $Q_r$ ), and measurement noise ( $R_r$ ). The equations of sensitivity can be utilized to quantify the contribution that each of these 3 sources of uncertainty makes to the total mean-square error in the estimate of  $x$ . This is accomplished by running through the sensitivity equations once for each error source (3 times in this example). To evaluate the effect of a single error source, the value of the error source of interest is kept at its nominal value, the other two error sources are set to zero, and the equations are run. In running through the sensitivity equations, the diagonal elements of  $P_e$  represent the contribution that the error source of interest makes to the mean-square error in the estimate of  $x$ . By repeatedly exercising the equations of sensitivity, the contribution that each error source makes to the total error in the estimate of the state can be quantified. Table 2.1 shows the contributions that each error source makes to the total steady-state estimate error and is the steady-state component error budget for this system.

The Kalman filter algorithm and the sensitivity analysis both computed the mean-square error in the filter's state estimate to be 0.323. Because the filter is optimal, we expect the filter's estimate of the mean-square error to be the same as that predicted

by the sensitivity analysis, and this is indeed the case. The data in Table 2.1 also show that the process noise contributes nearly twice as much as the measurement noise to the mean-square estimate error. This leads us to conclude that the estimate of  $x$  can be improved more effectively by decreasing the process noise than by decreasing the noise in the measurement—e.g. halving the measurement noise will have less impact on the filter’s estimate error than halving the process noise. If  $x$  had some physical meaning, these results could have important implications. If, for example,  $x$  represents a sensor bias, greater performance gains could be realized by decreasing the bias’ drift than by decreasing the noise in the sensor’s output. The quality of the sensor’s output could therefore be most effectively improved by focusing on decreasing the bias drift, or even trading off an increase in measurement noise for a decrease in bias drift. This result could have implications for the sensor’s design.

Because the sensitivity equations are linear in each source of uncertainty, the effect of a change in each error source can be calculated without re-running the equations. For example, we can easily quantify the change in the filter’s performance if the *actual* measurement noise ( $R_r$ ) is different from that modeled in the filter ( $R_f$ ). For example, let us suppose the  $R_r$  increases by a factor of 2.0. To compute the resulting mean-square error in the steady-state estimate of  $x$ , we simply multiply the individual contribution that  $R_r$  made to the total mean-square error (found in Table 2.1) by 2.0 and re-compute the total mean-square error. Doing so reveals that the contribution of the measurement noise increases to 0.222; the total mean-square error increases to 0.434; and the percent contribution that the measurement noise makes increases to 51.2%, surpassing the contribution of the process noise.

Other valuable insights can be gained from sensitivity analysis. For example, suppose that the actual time constant for the first-order Gauss-Markov process describing the reference state ( $\tau_r$ ) is 1.5 instead of 1.0. If the value of  $\tau_f$  remains at 1.0, then the filter will be suboptimal and the filter’s performance will likely degrade. In addition, the filter’s prediction of the mean-square error will likely be incorrect. Results from a sensitivity analysis applied to this filter/reference system configuration are tabulated in Table 2.2.

The results in Table 2.2 show that the actual performance of the filter will decline,

Contribution of $P_0$	0.0 (0 %)
Contribution of $Q_r$	0.303 (73.2 %)
Contribution of $R_r$	0.111 (26.8 %)
Total Mean-square Error	0.414
Filter's Predicted Mean-square Error	0.323

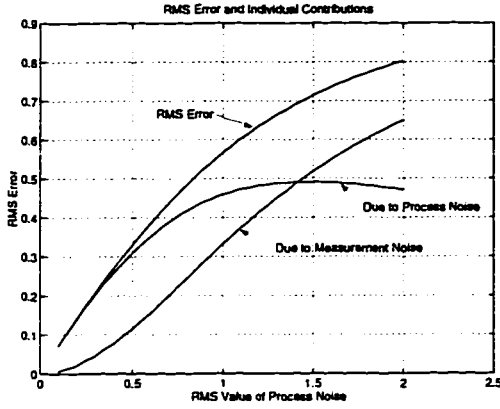
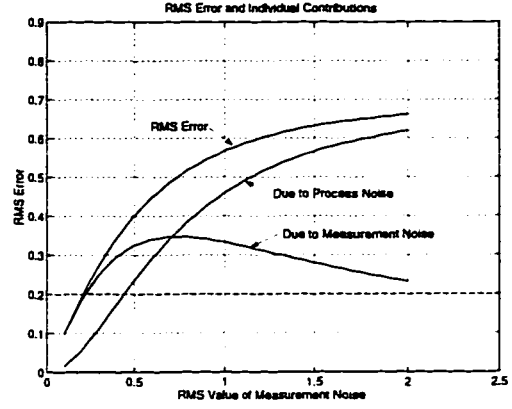
Table 2.2: Results of sensitivity analysis applied to a suboptimal filter

but the filter itself will not predict such a decline. As the results show, the contribution of the measurement noise did not change, but the contribution of the process noise increases by almost 50%. This result demonstrates the importance of having accurate models for sensor error sources and underscores (as did the results in Table 2.1) that, in this example, more significant performance gains could be realized by decreasing process noise than measurement noise.

### 2.3.3 Extending the Analysis

The results presented in the previous section demonstrate the salient features of sensitivity analysis. In this section, sensitivity analysis will be utilized to explore trends that appear in the filter's estimate of  $x$  when the process noise and measurement noise parameters vary over a wide range. The results of this examination provide insight into the fundamental behavior of a Kalman filter and, for this reason, will help to explain trends which appear in the results presented in Chapter 6. In the discussion that immediately follows, the filter is assumed to be optimal (i.e. the filter parameters are always equal to the reference system parameters). Hence, in this section,  $Q_f = Q_r$  and  $R_f = R_r$  in all simulations. Also, the RMS value of the process noise will be denoted  $\sqrt{Q}$ , and the RMS value of the measurement noise will be denoted  $\sqrt{R}$ .

Figure 2.1 shows the RMS error in the steady-state estimate of  $x$  versus  $\sqrt{Q}$ , with  $\sqrt{R}$  fixed at 1.0. Also shown in the figure are the individual contributions that process noise and measurement noise make to the steady-state error in the estimate of  $x$ . Figure 2.2 is similar to Figure 2.1, but shows the RMS error in the steady-state estimate of  $x$  versus  $\sqrt{R}$ , with  $\sqrt{Q}$  fixed at 1.0. This figure also shows the individual contributions that process and measurement noise each make to the steady-state RMS

Figure 2.1: RMS error versus  $\sqrt{Q}$ Figure 2.2: RMS error versus  $\sqrt{R}$ 

error in  $x$ .

As Figure 2.1 shows, the process noise is responsible for a *decreasing* fraction of the total RMS error as  $\sqrt{Q}$  grows *larger* relative to  $\sqrt{R}$ . When  $\sqrt{Q}$  is smallest, for example, it contributes nearly all of the error. Similarly, as Figure 2.2 shows, the measurement noise is responsible for a decreasing fraction of the total RMS error as its RMS value grows larger relative to  $\sqrt{Q}$ .

This may seem counterintuitive because one might guess that a noise input with a large RMS value would contribute a correspondingly large fraction of the RMS error. However, this is not the case. The trend arises because of the manner in which the Kalman filter weighs the measurement information against the model information. This weighting is numerically encapsulated in the Kalman gain. Loosely speaking, the Kalman gain can be thought of as the weighting applied to the information in a measurement—a “large” Kalman gain indicates that the measurement is weighed heavily, and a “small” Kalman gain indicates that the predicted state is weighed heavily. As the Kalman gain increases, then, the measurement information figures more prominently into the filter’s state estimate.

Figure 2.3 shows how the value of the steady-state Kalman gain varies with  $\sqrt{Q}$  and  $\sqrt{R}$ . The curve labeled “Varying Measurement Noise” shows how the steady-state Kalman gain changes as  $\sqrt{R}$  varies (with  $\sqrt{Q}$  fixed). Similarly, the curve labeled “Varying Process Noise” shows how the steady-state gain changes as  $\sqrt{Q}$  varies (with

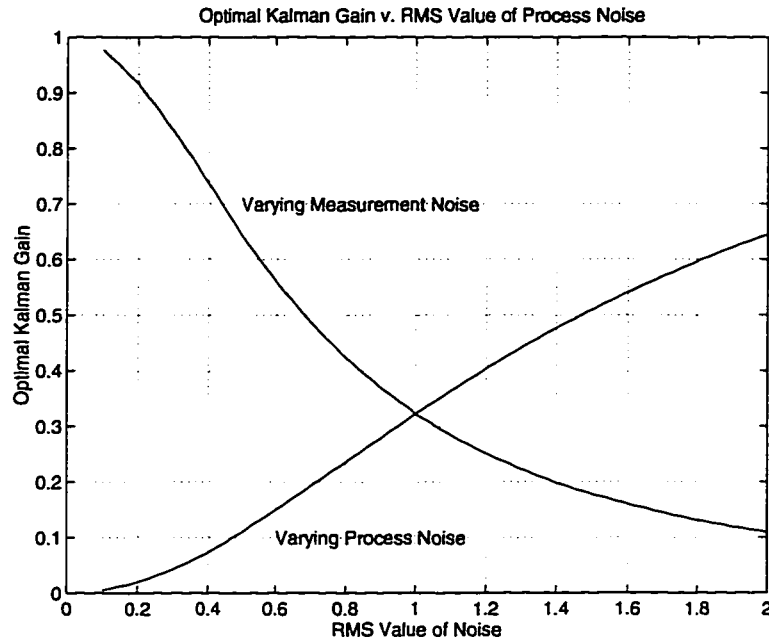


Figure 2.3: Kalman gain as a function of process and measurement noise

$\sqrt{R}$  fixed). As the figure shows, the Kalman gain is large when  $\sqrt{R}$  is small or  $\sqrt{Q}$  is large. Hence, when  $\sqrt{R}$  is small (or  $\sqrt{Q}$  is large), the measurement is weighed more heavily, and measurement errors contribute more to the total estimate error. Similarly, when  $\sqrt{Q}$  is small (or  $\sqrt{R}$  is large), the measurement is given less weight and the errors in the model contribute more to the total estimate error.

The relative weighting of the measurement against the model therefore explains why the measurement noise dominates the total estimate error when  $\sqrt{R}$  is small and why the process noise dominates the estimate error when  $\sqrt{Q}$  is small. Similar trends will be observed in the results presented in Chapter 6, and we shall refer back to this example when we encounter them. The reader should be aware, however, that the relative contributions of various error sources cannot always be explained simply by examining the Kalman gain. In general, the process noise, measurement noise, and system dynamics all come into play in determining how much each error source contributes to each state.

## 2.4 Extending Sensitivity Analysis to a Linearized Kalman Filter

For reasons to be given in Chapter 5, the Kalman filter analyzed in this research is a linearized Kalman filter. A linearized Kalman filter is one in which the model equations have been linearized about a known state trajectory, known as the *nominal state trajectory*. At the moment, this fact is important only insofar as it affects the formulation of the equations of sensitivity analysis. The specific equations of the linearized Kalman filter used in this research are not important here. In order to complete the theoretical picture for the work in this thesis, however, it is necessary to extend the theory presented earlier in this chapter. It is worth noting that the following derivation is original; past derivations of the equations of sensitivity analysis assume the presence of a Kalman filter having the form described in Section 2.2 ([3], [8], [10], [21]).

Before proceeding with the derivation, it is worth establishing a basic understanding of linearized Kalman filters. Linearized Kalman filters are generally employed when the evolution of the state vector is described by a nonlinear differential equation or the measurement vector is a nonlinear function of the state vector. In this research, certain model equations are nonlinear, and a linearized Kalman filter must therefore be employed. These equations have the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), t) + \mathbf{w}(t) + \boldsymbol{\mu}(t) \quad (2.42)$$

where  $\mathbf{f}$  is a vector of nonlinear functions,  $t$  is time,  $\mathbf{w}(t)$  is white noise, and  $\boldsymbol{\mu}(t)$  is a deterministic exogenous input.

In order to utilize a Kalman filter to estimate  $\mathbf{x}$ , this expression can be linearized about a nominal state trajectory (denoted  $\bar{\mathbf{x}}$ ) using a Taylor series expansion. This expansion can be carried out by first defining a *perturbation state vector* (denoted  $\delta\mathbf{x}$ ) such that

$$\mathbf{x} \equiv \bar{\mathbf{x}} + \delta\mathbf{x} \quad (2.43)$$

and then substituting the righthand side of Equation 2.43 into Equation 2.42 for

$\mathbf{x}$ . Generally, only the first-order terms in the expansion are kept, so that, after performing the expansion, we arrive at the approximation

$$\dot{\mathbf{x}} \approx \mathbf{f}(\bar{\mathbf{x}}, t) + F(t) \delta \mathbf{x} + \mathbf{w}(t) + \boldsymbol{\mu}(t) \quad (2.44)$$

where  $F(t)$  is a matrix given by

$$F(t) \equiv \left. \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \right|_{\mathbf{x}(t)=\bar{\mathbf{x}}(t)} \quad (2.45)$$

In light of Equation 2.43, the terms on the righthand side of Equation 2.44 can be put into two groups: terms describing the evolution of the nominal state trajectory and terms describing the evolution of the perturbation state. The nominal state trajectory is a known deterministic function of time, and is given by

$$\dot{\bar{\mathbf{x}}} \equiv \mathbf{f}(\bar{\mathbf{x}}, t) + \boldsymbol{\mu}(t) \quad (2.46)$$

Subtracting this equation from Equation 2.44, we arrive at the following model for the evolution of the perturbation state:

$$\dot{\delta \mathbf{x}} = F(t) \delta \mathbf{x} + \mathbf{w}(t) \quad (2.47)$$

which is a linear stochastic differential equation. As such, it is suitable for use as the model equation in a Kalman filter.

By utilizing a Taylor series expansion to linearize Equation 2.42, the problem of estimating the “full” state (which evolves according to a *nonlinear* function) has been reduced to a problem of estimating the perturbation state (which evolves according to a *linear* function). A linear Kalman filter that is employed to estimate the perturbation state vector is thus referred to as a *linearized* Kalman filter. An estimate of the “full” state ( $\hat{\mathbf{x}}(t)$ ) can be obtained by adding the filter’s state estimate ( $\delta \hat{\mathbf{x}}(t)$ ) to the (known) nominal state trajectory ( $\bar{\mathbf{x}}(t)$ ).

To accommodate a linearized Kalman filter, the equations of sensitivity analysis must be reformulated because, in general, the results of the sensitivity analysis

will depend on the nominal state trajectory. To proceed with the derivation of the equations of sensitivity analysis, we first define the reference system:

$$\delta \mathbf{x}_{r,k+1} = \Phi_{r,k} \delta \mathbf{x}_{r,k} + \mathbf{w}_{r,k} \quad (2.48)$$

which is the discrete-time form of Equation 2.47. The quantity  $k$  is a discrete time index,  $\Phi_{r,k}$  is the reference system's  $n \times n$  state transition matrix, and  $\mathbf{w}_{r,k}$  is process noise. This process noise is assumed to be an uncorrelated random sequence with covariance matrix  $Q_{r,k}$  and a Gaussian distribution.

The Kalman filter's model for the time history of the state vector,  $\delta \mathbf{x}_f$  is given by the equation

$$\delta \mathbf{x}_{f,k+1} = \Phi_{f,k} \delta \mathbf{x}_{f,k} + \mathbf{w}_{f,k} + B_k \mathbf{u}_k \quad (2.49)$$

This is very similar to Equation 2.4. However, the exogenous input  $\mathbf{u}_k$  is a function of the reference state, which, in the equation to follow, has been separated into its nominal and perturbation components:

$$\mathbf{u}_k = \Xi_k \bar{\mathbf{x}}_{r,k} + \Gamma_k \delta \mathbf{x}_{r,k} + \psi_k \quad (2.50)$$

where  $\psi_k$  is the  $k^{th}$  element of an uncorrelated Gaussian sequence that has covariance matrix  $\Psi_k$ . Notice that  $\mathbf{u}_k$  may depend on the nominal state trajectory differently than it depends on the perturbation state. We shall return to these time-update equations momentarily, but we now proceed by examining the measurement equations.

Recalling Equation 2.2, the *actual* measurements read from the sensors and used by the filter are a linear combination of the *reference* states. The filter's *model* for the measurement vector, however, may be different from the actual measurement vector. In other words, the filter “believes” that the measurement of the “full” state is actually modeled by the equation

$$\mathbf{z}_{f,k} = H_{f,k} \mathbf{x}_{f,k} + \mathbf{v}_{f,k} \quad (2.51)$$

Because a linearized filter's state vector is a perturbation state ( $\delta \mathbf{x}_{f,k}$ ), however, the

measurement vector  $\mathbf{z}_{f,k}$  must be modified so that it is a linear function of the perturbation states only. The modified measurement vector is referred to as the *perturbation measurement*; the filter's *model* for the perturbation measurement is

$$\delta\mathbf{z}_{f,k} \equiv \mathbf{z}_{f,k} - H_{f,k}\bar{\mathbf{x}}_{f,k} \quad (2.52)$$

$$= H_{f,k}\delta\mathbf{x}_{f,k} + \mathbf{v}_{f,k} \quad (2.53)$$

The *actual* measurement vector may not be the same as the filter's model for the measurement, however, because the actual measurement of the “full” state is given by

$$\mathbf{z}_{r,k} = H_{r,k}\mathbf{x}_{r,k} + \mathbf{v}_{r,k} \quad (2.54)$$

The actual perturbation measurement vector, then, is

$$\delta\mathbf{z}_{r,k} \equiv \mathbf{z}_{r,k} - H_{f,k}\bar{\mathbf{x}}_{f,k} \quad (2.55)$$

Noting that

$$\mathbf{x}_{r,k} \equiv \bar{\mathbf{x}}_{r,k} + \delta\mathbf{x}_{r,k} \quad (2.56)$$

we can substitute from Equation 2.56 into Equation 2.54 for  $\mathbf{x}_{r,k}$ , then substitute the result into Equation 2.55 to arrive at an expression for the true perturbation measurement:

$$\delta\mathbf{z}_{r,k} = \mathbf{z}_{r,k} - H_{f,k}\bar{\mathbf{x}}_{f,k} \quad (2.57)$$

$$= H_{r,k}(\bar{\mathbf{x}}_{r,k} + \delta\mathbf{x}_{r,k}) + \mathbf{v}_{r,k} - H_{f,k}\bar{\mathbf{x}}_{f,k} \quad (2.58)$$

$$= H_{r,k}\delta\mathbf{x}_{r,k} + \mathbf{v}_{r,k} + H_{r,k}\bar{\mathbf{x}}_{r,k} - H_{f,k}\bar{\mathbf{x}}_{f,k} \quad (2.59)$$

The measurement at each timestep is utilized to update the state estimate according to

$$\delta\hat{\mathbf{x}}_{f,k}^+ = \delta\hat{\mathbf{x}}_{f,k}^- + K_k [\delta\mathbf{z}_{r,k} - H_{f,k}\delta\hat{\mathbf{x}}_{f,k}^-] \quad (2.60)$$

or, substituting from Equation 2.59 into Equation 2.60 for  $\delta \mathbf{z}_{r,k}$ ,

$$\begin{aligned} \delta \hat{\mathbf{x}}_{f,k}^+ &= \delta \hat{\mathbf{x}}_{f,k}^- + K_k \left[ H_{r,k} \delta \mathbf{x}_{r,k} + \mathbf{v}_{r,k} - H_{f,k} \delta \hat{\mathbf{x}}_{f,k}^- \right] \\ &\quad + K_k (H_{r,k} \bar{\mathbf{x}}_{r,k} - H_{f,k} \bar{\mathbf{x}}_{f,k}) \end{aligned} \quad (2.61)$$

We shall return to this expression shortly, but we must first derive a suitable expression for  $\delta \hat{\mathbf{x}}_{f,k}^-$ . Substituting from Equation 2.50 into Equation 2.49 for  $\mathbf{u}_k$ , an expression can be derived for the propagation of the state estimate from timestep  $k$  to timestep  $k + 1$ :

$$\delta \hat{\mathbf{x}}_{f,k+1}^- = \Phi_{f,k} \delta \hat{\mathbf{x}}_{f,k}^+ + B_k (\Xi_k \bar{\mathbf{x}}_{r,k} + \Gamma_k \delta \mathbf{x}_{r,k} + \psi_k) \quad (2.62)$$

We continue as we did in Section 2.2 by defining a combined state vector ( $\delta \mathbf{x}_{c,k}$ ) that is constructed by stacking  $\delta \mathbf{x}_{r,k}$  and  $\delta \hat{\mathbf{x}}_{f,k}$ :

$$\delta \mathbf{x}_{c,k} \equiv \begin{bmatrix} \delta \mathbf{x}_{r,k} \\ \delta \hat{\mathbf{x}}_{f,k} \end{bmatrix} \quad (2.63)$$

so that

$$\delta \mathbf{x}_{c,k}^+ \equiv \begin{bmatrix} \delta \mathbf{x}_{r,k} \\ \delta \hat{\mathbf{x}}_{f,k}^+ \end{bmatrix} \quad (2.64)$$

and

$$\delta \mathbf{x}_{c,k}^- \equiv \begin{bmatrix} \delta \mathbf{x}_{r,k} \\ \delta \hat{\mathbf{x}}_{f,k}^- \end{bmatrix} \quad (2.65)$$

Utilizing Equations 2.48, 2.62, 2.65, one can arrive at an expression that describes the propagation of  $\delta \mathbf{x}_{c,k}^+$  from timestep  $k$  to timestep  $k + 1$ :

$$\delta \mathbf{x}_{c,k+1}^- = \begin{bmatrix} \Phi_{r,k} & 0 \\ B_k \Gamma_k & \Phi_{f,k} \end{bmatrix} \delta \mathbf{x}_{c,k}^+ + \begin{bmatrix} \mathbf{w}_{r,k} \\ B_k \psi_k \end{bmatrix} + \begin{bmatrix} 0 \\ B_k \Xi_k \end{bmatrix} \bar{\mathbf{x}}_{r,k} \quad (2.66)$$

Equation 2.66 can be simplified to

$$\delta \mathbf{x}_{c,k+1}^- = \Phi_{c,k} \delta \mathbf{x}_{c,k}^+ + \mathbf{w}_{c,k} + \Xi_{c,k} \bar{\mathbf{x}}_{r,k} \quad (2.67)$$

where the meaning of  $\Phi_{c,k}$  and  $w_{c,k}$  can be inferred by comparing Equation 2.67 to Equation 2.66, and  $\Xi_{c,k}$  is defined as

$$\Xi_{c,k} \equiv \begin{bmatrix} 0 \\ B_k \Xi_k \end{bmatrix} \quad (2.68)$$

Utilizing Equations 2.61 and 2.64, one can arrive at an expression that describes the evolution of  $\delta x_{c,k}$  when a measurement is processed

$$\delta x_{c,k}^+ = B_{c,k} \delta x_{c,k}^- + K_{c,k} v_{r,k} - \Lambda_{f,k} \bar{x}_{f,k} + \Lambda_{r,k} \bar{x}_{r,k} \quad (2.69)$$

where  $B_{c,k}$  and  $K_{c,k}$  have the same meanings as in Equation 2.22 and

$$\Lambda_{f,k} \equiv \begin{bmatrix} 0 \\ K_k H_{f,k} \end{bmatrix} \quad (2.70)$$

and

$$\Lambda_{r,k} \equiv \begin{bmatrix} 0 \\ K_k H_{r,k} \end{bmatrix} \quad (2.71)$$

The goal of the analysis is to produce a recursive expression for  $P_{e,k}$  (defined in Equation 2.13) that is similar to Equation 2.30, but which applies to the linearized filter. To this end, we must first derive an expression that describes the propagation of the mean-square value of  $x_{c,k}^+$  from timestep  $k$  to timestep  $k+1$  (i.e. one that is analogous to Equation 2.27). This expression can be obtained by multiplying Equation 2.67 by its own transpose and taking the expectation of both sides:

$$\begin{aligned} P_{c,k+1}^- &= \Phi_{c,k} P_{c,k}^+ \Phi_{c,k}^T + Q_{c,k} + \Xi_{c,k} \bar{x}_{r,k} \bar{x}_{r,k}^T \Xi_{c,k}^T \\ &\quad + \Phi_{c,k} E[\delta x_{c,k}^+] \bar{x}_{r,k}^T \Xi_{c,k}^T + \Xi_{c,k} \bar{x}_{r,k} E[\delta x_{c,k}^+]^T \Phi_{c,k}^T \end{aligned} \quad (2.72)$$

where  $Q_{c,k}$  has the same meaning as in Equation 2.28 and  $E[\delta x_{c,k}^+]$  is the expected value of  $\delta x_{c,k}^+$ .

Next, we must derive an expression (analogous to Equation 2.29) that describes the update of the mean-square value of  $x_{c,k}^-$  after incorporation of  $\delta z_{r,k}$  into the state

estimate. This expression can be obtained by multiplying Equation 2.69 by its own transpose and taking the expectation of both sides:

$$\begin{aligned}
P_{c,k}^+ &= B_{c,k} P_{c,k}^- B_{c,k}^T + K_{c,k} R_{r,k} K_{c,k}^T \\
&\quad + \Lambda_{r,k} \bar{\mathbf{x}}_{r,k} \bar{\mathbf{x}}_{r,k}^T \Lambda_{r,k}^T + \Lambda_{f,k} \bar{\mathbf{x}}_{f,k} \bar{\mathbf{x}}_{f,k}^T \Lambda_{f,k}^T \\
&\quad - \Lambda_{r,k} \bar{\mathbf{x}}_{r,k} \bar{\mathbf{x}}_{f,k}^T \Lambda_{f,k}^T - \Lambda_{f,k} \bar{\mathbf{x}}_{f,k} \bar{\mathbf{x}}_{r,k}^T \Lambda_{r,k}^T \\
&\quad + \Lambda_{r,k} \bar{\mathbf{x}}_{r,k} E[\delta \mathbf{x}_{c,k}^-]^T B_{c,k}^T + B_{c,k} E[\delta \mathbf{x}_{c,k}^-] \bar{\mathbf{x}}_{r,k}^T \Lambda_{r,k}^T \\
&\quad - \Lambda_{f,k} \bar{\mathbf{x}}_{f,k} E[\delta \mathbf{x}_{c,k}^-]^T B_{c,k}^T - B_{c,k} E[\delta \mathbf{x}_{c,k}^-] \bar{\mathbf{x}}_{f,k}^T \Lambda_{f,k}^T
\end{aligned} \tag{2.73}$$

where  $E[\delta \mathbf{x}_{c,k}^-]$  is the expected value of  $\delta \mathbf{x}_{c,k}^-$ .

The recursive expressions for  $E[\delta \mathbf{x}_{c,k}^+]$  and  $E[\delta \mathbf{x}_{c,k}^-]$  are given by

$$E[\delta \mathbf{x}_{c,k+1}^-] = \Phi_{c,k} E[\delta \mathbf{x}_{c,k}^+] + \Xi_{c,k} \bar{\mathbf{x}}_{r,k} \tag{2.74}$$

$$E[\delta \mathbf{x}_{c,k}^+] = B_{c,k} E[\delta \mathbf{x}_{c,k}^-] + \Lambda_{r,k} \bar{\mathbf{x}}_{r,k} - \Lambda_{f,k} \bar{\mathbf{x}}_{f,k} \tag{2.75}$$

Equations 2.30, 2.72, 2.73, 2.74 and 2.75 together form the extended sensitivity analysis equations. Utilizing these equations for sensitivity analysis is complicated by the fact that Equations 2.72 and 2.73 are not linear in the same sense as Equations 2.27 and 2.29. Specifically, the expressions for  $P_{c,k}^+$  and  $P_{c,k+1}^-$  in Equations 2.72 and 2.73 are quadratic in  $\bar{\mathbf{x}}_{r,k}$ ,  $\bar{\mathbf{x}}_{f,k}$ ,  $E[\delta \mathbf{x}_{c,k}^+]$ , and  $E[\delta \mathbf{x}_{c,k}^-]$ . In contrast, the expressions for  $P_{c,k}^+$  and  $P_{c,k+1}^-$  in Equations 2.27 and 2.29 are linear in their forcing functions ( $Q_{r,k}$  and  $R_{r,k}$ ) and initial conditions ( $P_r(0)$ ). The principle of superposition therefore applies to the system of difference equations described by Equations 2.27 and 2.29 but not to that described by Equations 2.72 and 2.73. Therefore, *in general*, sensitivity analysis does *not* apply to the linearized Kalman filter in the same way as it applies to a truly linear Kalman filter.

Because Equations 2.72 and 2.73 are not linear in the appropriate sense, the contributions of individual error sources to the total mean-square estimate error cannot necessarily be separated from one another. *However*, Equations 2.72 and 2.73 still produce the true mean-square error in the estimates of the linearized Kalman filter's state. Furthermore, *if* Equations 2.72 and 2.73 simplify so that they are linear,

then the principle of superposition *does* apply and the contributions of individual error sources to the total mean-square estimate error *can* be quantified. Whether the equations simplify appropriately depends on whether the error models in the reference system are a function of the nominal trajectory (i.e. whether  $\Xi_k = 0$ ) and on how the nominal trajectory is defined. For example, if  $\Xi_k = 0$ , then Equation 2.72 simplifies to Equation 2.27. Also, if

$$H_{r,k}\bar{\mathbf{x}}_{r,k} \equiv H_{f,k}\bar{\mathbf{x}}_{f,k} \quad (2.76)$$

then Equation 2.73 reduces to Equation 2.29.

Fortunately, in this research, Equation 2.73 simplifies to Equation 2.29 and Equation 2.72 simplifies to Equation 2.27 under most circumstances. Details regarding the implementation of these equations will be presented in Chapter 5.

## 2.5 Summary

Sensitivity analysis is an elegant theoretical tool for quantifying the individual contributions that various error sources make to the total mean-square error in a Kalman filter's state estimate. Furthermore, its formulation is such that the effects of changes in the real-world error sources can be easily quantified with little computational effort. As the example in Section 2.3 illustrated, process noise, measurement noise, and initial uncertainty in the state each contribute differently to the total estimate error. The results implied that improvement in performance was a stronger function of process noise than measurement noise. These results could have important implications, depending on the physical meaning of the state. Finally, the example demonstrated that the filter's performance degrades if the filter's models are inaccurate. This is not unexpected, but, using sensitivity analysis, we were able to easily quantify the extent to which the erroneous model degraded the filter's performance. As was shown, a filter with an erroneous model can produce misleading results. All of these benefits of sensitivity analysis demonstrate that it is well-suited for the analysis in this thesis. In the chapters to follow, a Kalman filter and a reference system model will be presented, and the theory developed in this chapter will be applied to this system.

# Chapter 3

## Sensor Error Models

### 3.1 Introduction

The previous chapter supplied the theoretical foundation for the analysis in this thesis. As the previous chapter demonstrated, a complete sensitivity analysis requires two systems of stochastic equations: one that models the time history of the reference system's state and one that models the time history of the Kalman filter's state. For the problem addressed in this thesis, a detailed description of both systems of equations will be deferred until Chapter 5. Before the detailed equations can be presented, however, it is first necessary to discuss the performance characteristics of various sensors commonly found in land-vehicle navigation systems. In this chapter, each sensor's performance is discussed, and a mathematical model for the error in each sensor's output is given. Some models are derived in this chapter, while others are borrowed from the literature. The error models presented in this chapter will reappear in Chapter 5 as part of the reference system's and Kalman filter's stochastic equations.

It is important to point out that, for some sensors, the error models that appear in the Kalman filter's equations differ from those that appear in the reference system's equations. The reason for this has to do with the fact that some error models depend on parameters that cannot be reliably estimated by a Kalman filter. (The error model for the fluxgate compass' bias is an example.) Therefore, certain error models cannot

be mechanized in a Kalman filter and must be supplanted with simpler ones that can. However, such error models *can* generally be used in a sensitivity analysis. Differences between the error models in the Kalman filter and those in the sensitivity analysis will be noted.

## 3.2 Rate Gyro Error Modeling

### 3.2.1 Example Rate Gyros

Rate gyros are found in many existing land-vehicle navigation systems [38]. Two popular low-cost rate gyros are the Murata Gyrostar and the Systron Donner Gyrochip Horizon. Both of these gyros generate an analog signal that is amplified so that 1 volt of output corresponds to rotation rate of 45 degrees/sec. The maximum specified rotation rate for both gyros is 90 degrees/sec, a rate which far exceeds typical turn rates for an automobile. Both the Murata gyro and the Systron Donner gyro transduce rotation rate using a vibrating element. When the gyro is not rotating, the vibrating element continuously vibrates back and forth within a plane. When the element is subjected to rotation about a particular axis, coriolis forces cause the element to deflect out-of-plane; the amplitude of the out-of-plane motion is proportional to the rate of rotation. This out-of-plane motion is sensed and filtered by electronics inside the gyro, and the filtered signal serves as the gyro's output. (Excellent discussions of vibrating-element gyroscopes can be found in [58] and [59].) Both of these rate gyros have been tested by the author, and discussion of rate gyro error sources will center around these two particular gyros.

There are other rate gyros that may compete in the low-cost market in the future. Fiber-optic gyros (FOGs), which operate on a wholly different principle than the Murata and Systron Donner gyros, are generally known to exhibit significantly better performance than existing low-cost vibratory gyros [40]. Today, for example, Andrew Corporation produces a FOG known as the Autogyro. According to the specifications for this gyro, its performance is roughly 5 to 10 times better than that of the Murata gyro. Currently, FOGs of this caliber are more expensive than the Gyrostar and

Gyrochip; whether the cost of FOGs will, over the next few years, approach the current price range of low-cost vibratory sensors seems dubious [40].

### 3.2.2 Rate Gyro Bias Drift

Two errors which appear in the outputs of existing low-cost rate gyroscopes are additive white noise and bias drift. Tests have shown that, for the author's gyros, the RMS value of the white noise in the output of each gyro is approximately 0.6 mV (0.027 deg/s) and 1.0 mV (0.045 deg/s) for the Gyrostar and Gyrochip Horizon, respectively. The bias drift of each gyro was examined in several static tests. Figure 3.1 shows data collected concurrently from a Gyrostar and a Gyrochip Horizon. The data shown were collected over a 48-hour period, beginning immediately after power was applied to each gyro. The output of each gyro was filtered (with a continuous-time low-pass filter) and sampled at 50 Hz; this data was then filtered again (digitally) to reduce the noise in the samples. The sampled data were saved at 1 Hz. Even though filtered twice, the raw data is still somewhat noisy, and this noise obscures the drift in the gyros' outputs. For purposes of clarity, then, the raw data was averaged in blocks of time 1 minute long. (Averaging the data is really another form of low-pass filtering and is legitimate because it does not obscure long-term bias drift.) Finally, in order to compare the data from the two gyros more easily, the mean of each set of data (over the entire test) was removed. The modified gyro data is shown in Figure 3.1.

As the figure shows, the bias in both gyros drifts nearly identically with time. This suggests that the drift of both biases is caused by the same phenomenon—probably temperature variations. Other tests have been run in which the gyro's output was recorded as the gyro temperature was varied. These tests confirmed the fact that each gyro's bias is a strong function of temperature.

Static tests reveal that both gyros suffer from a significant transient bias drift immediately after power is applied to them. Figures 3.2 and 3.3 show the average output of each gyro for several minutes after power was applied. As the figures show,

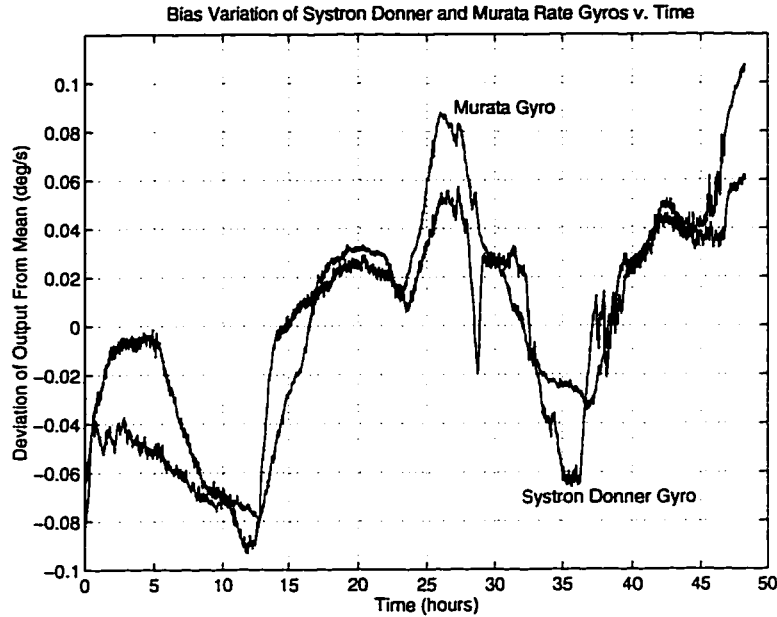


Figure 3.1: Forty-eight hours of data from two rate gyros

the mean output of both gyros approaches a steady-state value in a roughly exponential fashion. This phenomenon is probably the result of the gyro's self-heating—i.e. when each gyro is turned on, the electronics inside begin to dissipate heat and cause the temperature of the gyro to rise. As the temperature inside the gyro stabilizes, the mean output of the gyro changes at a slower rate. The outputs of both gyros take about the same amount of time to stabilize; however, note that the magnitude of the drift is larger for the Murata gyro than for the Systron Donner gyro.

A similar phenomenon was observed in [4]. In [4], the authors performed a thorough analysis of two low-cost rate gyros, one of which was the Murata Gyrostar. The authors developed error models for the rate gyros that were based on their data, and then evaluated the models against real gyro data. The error models in [4] account for the transient bias drift that results from self-heating. This author believes that it is more appropriate to ignore the effects of self-heating since this phenomenon is clearly transient; furthermore, the goals of the work in this thesis are most appropriately reached by examining the *steady-state* contributions that sensors make to a navigation system's performance. Therefore, in this work, the error model for the rate gyro

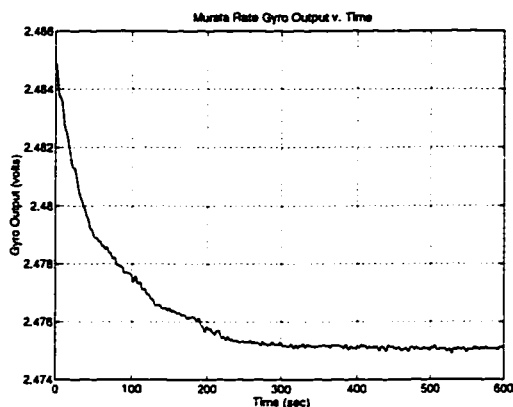


Figure 3.2: Murata gyro transient

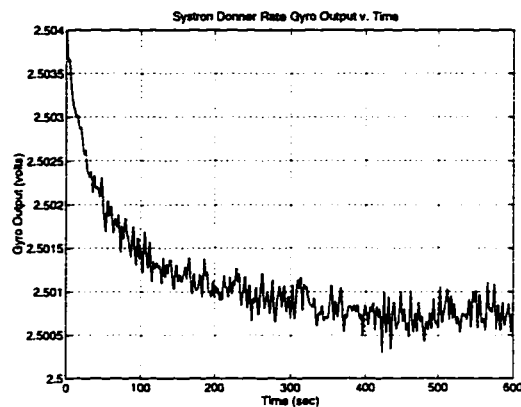


Figure 3.3: Systron Donner gyro transient

ignores this start-up transient bias drift.

It is important to understand the characteristics of the rate gyros' bias drift because the analysis in this thesis requires a model of the bias drift. If, in fact, the drift in the bias is due principally to changes in ambient temperature, then the relationship between bias drift and temperature is deterministic. Under these circumstances, the Kalman filter's model for the bias drift may depend explicitly on temperature. However, in order to mechanize a temperature-dependent model, a measurement of temperature would have to be available to the filter. To avoid using a temperature measurement, a simpler Kalman filter could be designed in which the bias' dependence on temperature is ignored altogether. In this case, the bias drift would be *modeled* as a random process, even though the bias drift is *not* truly random because it is actually a deterministic function of a measurable quantity (i.e. temperature). From the point of view of the Kalman filter, the bias drift could be modeled as a random process, since the Kalman filter would have neither a temperature measurement nor knowledge of the bias' relationship to temperature.

There are two ways, then, in which the bias drift can be modeled in the Kalman filter: 1.) model the bias drift as a random process with enough bandwidth to track worst-case drift resulting from changes in ambient temperature or 2.) include a temperature-measuring sensor (e.g. a thermistor) in the navigation system that could be used to calibrate the bias' temperature dependencies. In an implementation

adopting option 1, the Kalman filter's model for the bias drift would not explicitly depend on temperature—it would assume that the bias drift was a random process, and the temperature-dependent part of the bias drift would not be deliberately compensated. Because variations in the gyro's temperature (and therefore the rate gyro bias) would likely not have a mean of zero, a random walk could be employed as a reasonable model for the bias drift. (Other candidate models, such as a first- or second-order Gauss-Markov process are zero-mean processes and therefore would be less appropriate.) In contrast, in a filter design adopting option 2, a measurement of temperature must be available. The value of the bias as a function of temperature would presumably be known and could be stored in a software lookup table. Changes in the bias with temperature during normal operation could then be corrected by the navigation software: first, the temperature would be read from the temperature sensor; then, the value of the bias drift that corresponded to the temperature reading would be found in the lookup table; the bias error from the lookup table would then be subtracted from the gyro's output, and the *corrected* gyro reading would be fed into the Kalman filter as a measurement of heading rate.

It may appear that including a temperature measurement and lookup table to calibrate the rate gyro bias (i.e. option 2) would result in performance superior to an implementation that ignored the bias' dependency on temperature. However, this is not necessarily the case. First, temperature variations probably occur over time periods that are much longer than the sample period of the filter. (The data in Figure 3.1 demonstrate that bias variations occur over many minutes, and the sample period of the Kalman filter designed for this research is 0.5 seconds.) As the ratio of the time-constant of the bias variations to the filter's sample period approaches infinity, a random walk model will more closely approximate the bias variations. Hence, the actual bias variations could probably be accurately modeled (in the Kalman filter) as a random walk, and the filter should be able to easily track bias variations without a measurement of temperature, as long as complementary sensor measurements are available. Second, adding a temperature sensor adds cost, complexity, and measurement errors to the navigation system. Consumer vehicle navigation systems are very cost-sensitive products, and any benefits in performance

gained by adding a temperature sensor would have to be weighed carefully against the attendant cost. Furthermore, the functional relationship between bias drift and temperature may vary from gyro to gyro. Therefore, it may be necessary to customize the data held in a temperature-versus-bias lookup table for each gyro, adding further to production costs. Finally, errors in the temperature sensor's output, such as drift and noise, would have to be appropriately modeled or compensated.

The Kalman filter in this work does not assume the presence of a temperature measurement. It therefore includes a random walk process to model the rate gyro bias. The empirical data in Figure 3.1 was used as a starting point to derive parameters that govern this model. Note that the particular data set in Figure 3.1 is probably *not* best modeled as a random walk. However, in a real system, the bias may change in a variety of ways—it could be constant, change in a stepwise fashion, or oscillate (as is roughly demonstrated in Figure 3.1)—depending on how the gyro's temperature changes. Therefore, the Kalman filter model must be flexible enough to track various types of bias variations. The most appropriate model for this is a random walk. The parameters for the bias models were chosen conservatively, so that the Kalman filter could track worst-case variations in the bias.

The philosophy behind the model for the gyro bias for the reference system is somewhat different than it is for the Kalman filter. In the reference system equations, we seek to utilize a model that most closely emulates the *true* bias variations. Therefore, the model for the bias in the reference system is *not* a random walk. Instead, the data in Figure 3.1 was used to directly derive a model for the bias drift that emulates that particular data set. The model chosen is a second-order Gauss-Markov process. Also, the angular error produced by the bias drift in this model is approximately 30 degrees/hour RMS; this is consistent with the gyros' specifications and with data cited in the literature.

### 3.2.3 Rate Gyro Scale Factor Error

The errors modeled for both rate gyros include bias drift and additive white noise. However, it is important to note that the Kalman filter error models presented in

this thesis (and in [4]) ignore rate gyro scale factor errors. Results have shown that ignoring scale factor variations is reasonable because it is extremely difficult for a Kalman filter to calibrate a rate gyro's scale factor errors under certain circumstances. Specifically, a Kalman filter cannot estimate the rate gyro's scale factor error unless the error appears in the rate gyro's output. When the rate gyro is not rotating, the output of the gyro contains no significant information about its scale factor, and the rate gyro's scale factor error cannot, therefore, be determined from the rate gyro's output. Hence, if a rate gyro is part of a vehicle navigation system, it is only when the vehicle is actually *rotating* that the gyro's scale factor errors can be observed in the rate gyro's output. However, land-vehicles typically move in straight lines for long periods of time, and turns occur abruptly and last only a short time. Because of this, a navigation Kalman filter will generally not be able to estimate its rate gyro's scale factor error accurately. It has been this author's experience that the Kalman filter developed for this research does a poor job of estimating the rate gyro's scale factor, even if the vehicle's movement includes turns. In addition, including a gyro scale factor error as a state in the Kalman filter complicates analysis because it introduces a nonlinearity into the filter's equations.

Hence, the design for the Kalman filter used in this research assumes that the rate gyro's scale factor is constant and equal to the nominal scale factor for Murata's rate gyro. No attempt is made to estimate the gyro's scale factor error. However, because scale factor error may exist in a real gyro, it is important to investigate its influence on navigation system performance. Therefore, the reference system's model for the rate gyro's output includes scale factor error. The manner in which the scale factor error figures into the rate gyro's error equations will be given in Section 3.2.4.

Empirical data has been gathered from both a Gyrostar and Gyrochip to determine each rate gyro's scale factor accuracy. Figures 3.4 and 3.5 show results that were obtained when the Murata and Systron Donner rate gyros were tested using a rate table. (A rate table is an apparatus with a platform that can be made to rotate at a constant speed with very high accuracy.) For each test, the gyroscopes were mounted on the table and rotated at each of 13 different speeds, from -60 degrees/sec to 60 degrees/sec. Each gyro's output was recorded at each rotation rate for several

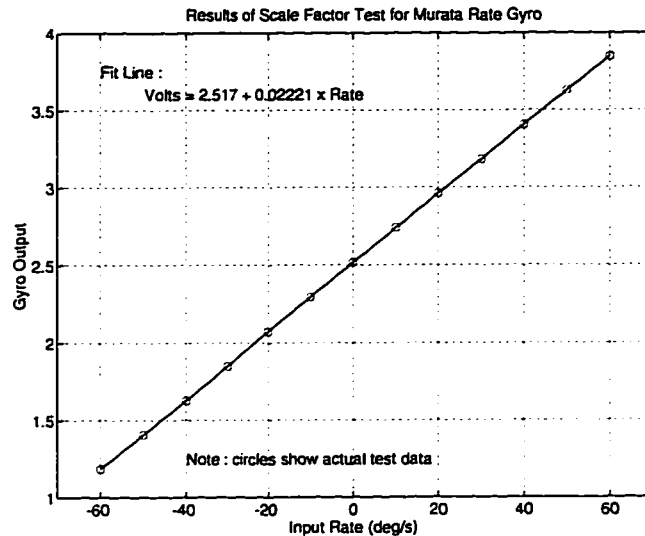


Figure 3.4: Gyrostar rate table results

minutes. Figures 3.4 and 3.5 show the average output of the Murata and Systron Donner gyros, respectively, at each speed. The circles show the actual test data, and the line through the points was fit to the data in a least-squares sense. The equation for the fit line is shown on the plots.

The scale factor reported by the gyro specifications is 45 degrees/sec/volt, or 0.0222 volts/degree/sec. As the equations on the plots show, both gyros' scale factors are very near their specified values. The scale factor measured for the Systron Donner gyro is only 1.4% larger than the nominal scale factor given in the gyro's specifications; the scale factor error measured for the Murata gyro is too small to be considered significant in comparison to the error inherent in the test.

How much a scale factor error contributes to heading error depends on how much the vehicle turns. Theoretically, if a rate gyro's output is integrated directly to measure a change in heading, then the computed heading change will be in error by the same percent as the rate gyro scale factor. For example, a scale factor error of 1.0% will result in a heading error of approximately 0.90 degrees after a 90-degree turn. If the vehicle never turns, then scale factor error contributes virtually nothing to errors in the heading estimate. Hence, one should expect the largest heading error to appear

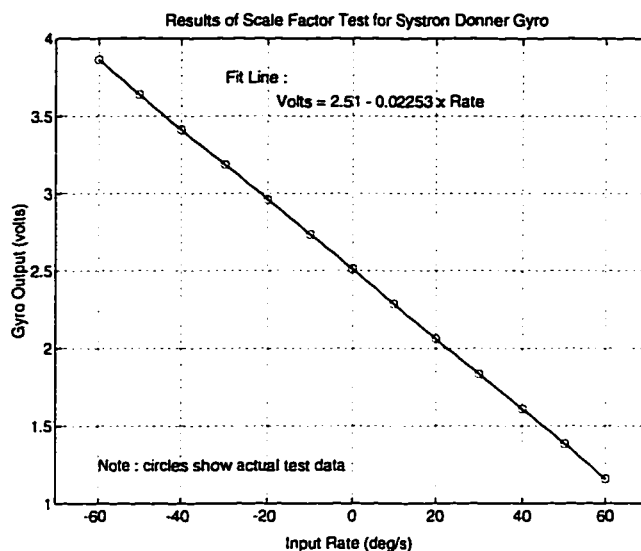


Figure 3.5: Gyrochip rate table results

when the vehicle makes turns which sweep through large angles. Because it is extremely difficult for the filter to estimate the rate gyro's scale factor and because scale factor error generally does not contribute to navigation error except when the vehicle is turning, the deleterious effects of scale factor error can be mitigated by increasing the filter's measurement noise parameter for the rate gyro in proportion to the turn rate. Doing so will cause the Kalman filter to weigh the rate gyro measurement less, thereby reducing the negative impact of the scale factor error on performance.

### 3.2.4 Equations for the Rate Gyro Error Model

The output of a rate gyro ( $V_{out}$ ) is usually an analog voltage that varies (nominally) linearly with the rotation rate ( $\omega_T$ ) of the gyro. For a non-ideal rate gyro, the output voltage is biased and corrupted with noise:

$$V_{out} = V_{out,T} + V_b + v \quad (3.1)$$

where  $V_b$  is a bias and  $v$  is white noise in the output. The quantity  $V_{out,T}$  is the "ideal" gyro output (unbiased and uncorrupted by noise). For an ideal gyro, the

output voltage is related to the rotation rate of the gyro about its sensitive axis by

$$\omega_T = K_T(V_{out,T} - V_{nominal}) \quad (3.2)$$

where  $K_T$  is the true scale factor and  $V_{nominal}$  is the nominal output of the rate gyro when it is not rotating about its sensitive axis. For both the Murata Gyrostar and the Systron Donner Gyrochip Horizon, the nominal output is 2.50 volts.

The heading rate measured from a non-ideal rate gyro contains three sources of error—scale factor error, bias error and white noise error. (Other error sources exist [2, 12, 60], but justification for ignoring them will be given presently.) The heading rate measured from a non-ideal gyro ( $\omega_{meas}$ ) is therefore

$$\omega_{meas} = K_{nominal}(V_{out} - V_{nominal}) = (K_T + \delta K)(V_{out} - V_{nominal}) \quad (3.3)$$

$\delta K$  is the scale factor error. (Recall that, even though the Kalman filter does not attempt to estimate the rate gyro's scale factor error, the reference system's model *does* include scale factor error. For this reason, scale factor error is included in the model for the rate gyro's output.) The relationship between the measured angular velocity,  $\omega_{meas}$ , and the true angular velocity,  $\omega_T$ , is therefore given by

$$\omega_{meas} = (K_T + \delta K)(V_{out,T} + V_b + v - V_{nominal}) \quad (3.4)$$

$$\omega_{meas} = \left(1 + \frac{\delta K}{K_T}\right)(\omega_T + \omega_{b,T} + v_T) \quad (3.5)$$

where

$$\omega_{b,T} \equiv K_T V_b \quad (3.6)$$

$$v_T \equiv K_T v \quad (3.7)$$

The measured heading rate therefore contains several terms: the true heading rate ( $\omega_T$ ), the true heading rate bias ( $\omega_{b,T}$ ), the true white noise in the output ( $v_T$ ), and three error terms that are proportional to the fractional error in the nominal scale factor. These three error terms appear if the assumed scale factor ( $K_{nominal}$ ) is not

equal to the true scale factor; when the reading from the gyro is converted from a voltage to a heading rate, these errors will be introduced. Equation 3.5 is used in the equations of sensitivity analysis. It should be noted that, in this research, the scale factor error in the reference system model is assumed to be constant.

The Kalman filter design ignores the scale factor error and therefore assumes that the gyro measurement is given by

$$\omega_{meas} = \omega_T + \omega_{b,T} + v_T \quad (3.8)$$

The Kalman filter's model for the bias error is a random walk process given by

$$\dot{\omega}_b = u_{\omega_b} \quad (3.9)$$

where  $u_{\omega_b}$  is zero-mean Gaussian white noise. The justification for choosing this model was given in Section 3.2.2. In the equations of sensitivity analysis, the model for the *true* bias error, denoted  $\omega_{b,T}$  is give by the sum of a second-order Gauss-Markov process and a random constant:

$$\omega_{b,T} = \varpi_b + \varphi_b \quad (3.10)$$

where

$$\dot{\varphi}_b = 0 \quad (3.11)$$

$$\dot{\varpi}_b = \alpha_b \quad (3.12)$$

$$\dot{\alpha}_b = -\beta_{\alpha_b}^2 \varpi_b - 2\beta_{\alpha_b} \alpha_b + u_{\alpha_b} \quad (3.13)$$

where  $\beta_{\alpha_b}$  is related to the time-constant of the process and  $u_{\alpha_b}$  is zero-mean Gaussian white noise.

It should be noted that, for various reasons, several sources of error that may appear in the output of a rate gyro have been ignored. Each of these error sources will now be briefly defined and an explanation will be given as to why they were ignored. The error sources that were ignored are g-sensitivity, cross-axis sensitivity,

and nonlinearity. The g-sensitivity of a rate gyro causes errors to be introduced into the gyro's output as a result of linear acceleration. The output of an ideal rate gyro would be entirely insensitive to acceleration. However, this quantity has been ignored because a simple calculation can show that the typical acceleration encountered in a real automobile is quite small: 0.27-g for a vehicle accelerating from 0 MPH to 60 MPH in 10 seconds. Furthermore, experience suggests that, most of the time, automobiles accelerate at even lower rates and often travel at nearly constant speeds. This error is therefore likely to be a very small contributor to the overall navigation error. Cross-axis sensitivity causes errors to be introduced into the gyro's output as a result of rotations about an axis perpendicular to the axis of sensitivity. This error has been ignored because an automobile typically rotates about a vertical axis only. Finally, nonlinearity errors are introduced into the gyro's output because the relationship between angular speed and the gyro's output is not truly linear. This error source has been ignored because rate table test results have shown that nonlinearity errors are quite small in the range of turning rates typically encountered in an automobile.

## 3.3 Magnetic Compass Error Modeling

### 3.3.1 Compass Error Characteristics

A magnetic compass is an electronic device that measures its heading relative to magnetic North by measuring the direction of the Earth's local magnetic field. Compasses are generally implemented with magnetometers, a Hall effect sensor, or a set of orthogonal coils referred to as a "fluxgate." It seems generally true that, of the existing compass implementations, the fluxgate compass is most commonly used in existing land-vehicle navigation systems [38].

Results will show that accurate heading measurements can be an extremely valuable positioning aid. However, accurate heading measurements can be difficult to obtain with a magnetic compass because disturbances in the magnetic field near the compass can induce large errors in the compass' output. Sources of magnetic disturbance encountered in vehicle navigation include power lines, motors (e.g. a fan

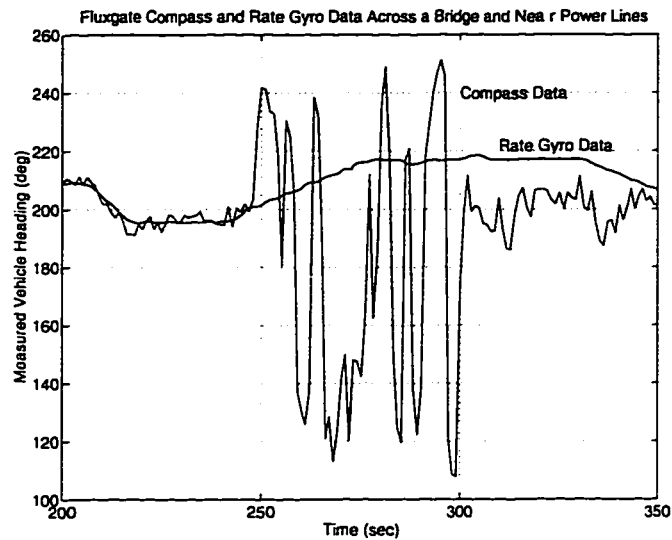


Figure 3.6: Compass data taken on a bridge and near power lines

inside the vehicle), and residual magnetism in local metal structures such as bridges, buildings and even the vehicle's chassis.

Data collected from a fluxgate compass have verified that a compass can exhibit very large measurement errors. Figure 3.6 shows data collected from a fluxgate compass taken in a vehicle that was being driven across a nearly-straight bridge in the vicinity of power lines. Included in the plot is the integral of data that were simultaneously collected from the Systron Donner rate gyro. The integrated gyro data provide a measure of the vehicle's heading history that is independent of the compass reading. As the figure illustrates, the gyro data indicate that the vehicle's heading is changing very little, while the compass data shows swings larger than 100 degrees. Two other independent gyroscopes were sampled concurrently and their data verify this result. Clearly, the compass reading is in error, probably as a result of magnetic disturbances induced by the power lines and the metal in the structure of the bridge.

Because compasses are susceptible to large errors, error compensation schemes for the fluxgate compass have received significant attention. Some compensation methods depend on having other sensors available, such as an angular velocity sensor [46]

or GPS [36]. Other approaches involve calibrating the compass errors by generating a lookup table of the errors as a function of heading [5, 41, 48] or involve some type of basic prefiltering technique [65, 52]. Still other approaches involve gimbaling the compass to prevent the compass from tilting relative to a local horizontal plane, thereby avoiding tilt-induced errors in the compass' output [67]. Finally, some fluxgate compass manufacturers specify a method by which the user can calibrate the errors in the compass. Such a calibration process is designed to eliminate systematic measurement errors that are a function of heading. Usually, calibration involves rotating the compass through at least 360 degrees, while digital electronics in the compass generate a lookup table of heading errors. This type of calibration can compensate for systematic errors that affect the gyro *at the time of calibration*. However, this type of calibration is ineffective against random errors that arise during operation and changes in the systematic errors that occur after calibration.

An analytical study of fluxgate compass errors has shown that the errors that appear in a compass' output can be mathematically modeled as a function of magnetic heading [44]. In [44], the authors derived the following expression for errors that appear in the compass reading:

$$\Theta_b = A \sin(\Theta) + B \cos(\Theta) + C \sin(2\Theta) + D \cos(2\Theta) + E \quad (3.14)$$

where  $\Theta_b$  is the compass' bias error,  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $E$  are constants and  $\Theta$  is the true magnetic heading of the compass. It has been this author's experience that this model is difficult to utilize in a Kalman filter because of the relatively large number of unknown parameters that must be estimated and because Equation 3.14 is nonlinear in  $\Theta$ . The estimates of the parameters  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $E$  were unstable when this model was mechanized in a Kalman filter. In [70], this model was also rejected, but for other reasons. Also, in [65], this model was cited but apparently not used.

### 3.3.2 Equations for the Compass Error Model

Experience and the literature indicate that the errors in the fluxgate compass' output can be modeled as the sum of three components: a bias that is a systematic function

of heading, a random (but time-correlated) error resulting from external magnetic disturbances, and white noise. Therefore, the following model was used in the sensitivity analysis to model the total error in the fluxgate compass' output:

$$\Theta_{meas} = \Theta + \Theta_b + v_\Theta \quad (3.15)$$

where  $\Theta$  is the true heading and  $v_\Theta$  is additive white noise in the measurement. The model for the bias is the sum of three terms:

$$\Theta_b = \theta_o + \phi + \vartheta \quad (3.16)$$

where  $\theta_o$  is a random constant and

$$\dot{\vartheta} = \eta \quad (3.17)$$

$$\dot{\eta} = -\beta_\eta^2 \vartheta - 2\beta_\eta \eta + u_\eta \quad (3.18)$$

$$\phi = A \sin(\Theta) + B \cos(\Theta) + C \sin(2\Theta) + D \cos(2\Theta) \quad (3.19)$$

In Equations 3.17 through 3.19,  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $\theta_o$  are constants, the values for which were chosen from data in [44];  $\vartheta$  is a time-dependent random error described by a second-order Gauss-Markov process whose characteristics are determined by  $\beta_\eta$  and the RMS value of  $u_\eta$ . Note that  $\phi$  is a heading-dependent bias,  $\theta_o$  is a constant bias, and  $\vartheta$  represents magnetic disturbances. Equations 3.16 through 3.18 and a linearized form of Equation 3.19 represent the reference system's model for the fluxgate compass' bias error.

However, this model is *not* used in the Kalman filter for two reasons. First, as was mentioned, the model described by Equation 3.19 is nonlinear and has several unknown constants; attempts to mechanize this model in a Kalman filter resulted in unstable estimates. Second, the errors caused by external magnetic disturbances (denoted  $\vartheta$  in the model above) occur at unpredictable times and with unpredictable magnitude and therefore defy reliable predictive modeling.

Choosing a Kalman filter model for the compass' bias is therefore difficult because the filter must perform adequately when faced with measurement errors that are

unpredictable. It would be unwise to ignore the measurement errors altogether, since poorly modeled measurement errors can have a detrimental effect on the performance of a Kalman filter. One solution is to use a large value for the variance of the noise in the filter's heading measurement and/or use a large value for the variance of the process noise in the filter's model for the compass' bias. This is inadvisable, however. Using a large value for the measurement noise variance attempts to compensate for time-correlated bias errors by indicating to the filter that uncorrelated errors are large. Also, a constant measurement variance indicates that the RMS additive white noise in the measurement is constant. Loosely speaking, then, the weighting that the filter applied to every compass measurement would be the same, whether or not the measurement contained large errors. This is inadvisable because, while some compass measurements might be corrupted with large bias errors, others would not. Using a large value for the process noise in the bias model would probably extend the time it takes for the filter to reach a steady-state bias estimate and increase the filter's RMS error in the steady-state estimate of the compass' bias and heading. So, while these solutions might improve the performance of the filter under worst-case conditions, there is a better solution.

A more appropriate solution to this problem is to include a fault-detection algorithm in the navigation software that can determine when the compass errors are too large. This algorithm could cause the Kalman filter to ignore the compass data when the compass errors exceed a tolerable threshold. Sensor fault-detection algorithms already exist, and it is not necessary to describe one here. One advantage of using a fault-detection algorithm is that it eliminates the need for a "worst-case" error model in the Kalman filter and avoids the attendant degradation in filter performance. Another advantage of this solution is that the compass data are utilized only if the measurement errors are below an acceptable threshold. Because the error in a compass reading can be so large that it entirely obscures the useful data in the measurement, ignoring the compass reading altogether can be appropriate.

In the presence of a fault-detection algorithm, the data from the compass could be ignored if a large disturbance were detected. If the data were *not* being ignored, then the data should contain "small" disturbances, a heading-dependent error, and

white noise. The Kalman filter’s model for the bias ( $\Theta_b$ ) must therefore be chosen so that it is capable of tracking the changes in the bias as the vehicle’s heading changes. Although the model given in Equation 3.14 may appear to be a legitimate model, its complexity prevents its successful use. Therefore, the model of the compass’ bias drift that was chosen for use in the Kalman filter is simply a random walk:

$$\dot{\Theta}_b = u_{\Theta_b} \quad (3.20)$$

In addition to a bias, the measurement is assumed to include additive noise, the RMS value of which was chosen based on data collected from a fluxgate compass during on-road in-vehicle testing. (The parameters governing this model can be found in Appendix A).

### 3.4 Odometer Error Modeling

An odometer measures the curvilinear distance traveled by a vehicle. This section includes an analysis of the errors that appear in an odometer’s output. Equations describing the use of odometer data in particular navigation systems appear in [65], [66], [7], and [52]; a particularly detailed analysis is given in [66]. In [70] and [43], the authors discuss various error sources in odometry and actual data is presented in [43], but no formal analyses are presented. The following analysis is slightly different from other analyses in the literature.

For the analysis that follows, we first consider Figure 3.7, which is a functional representation of any one of a number of odometer implementations. This figure shows a cross-section of a rotating shaft or gear in the vehicle. Rigidly mounted on the rotating shaft are several evenly-spaced “trigger points” which pass a “pick-up sensor” that is mounted to the body of the vehicle. The odometer operates in such a way that the pick-up sensor generates a single digital pulse when any one of the trigger points passes it.

For an odometer comprised of an optical shaft encoder, as in [34], the trigger points represent the slots in the encoder wheel and the pick-up sensor represents an

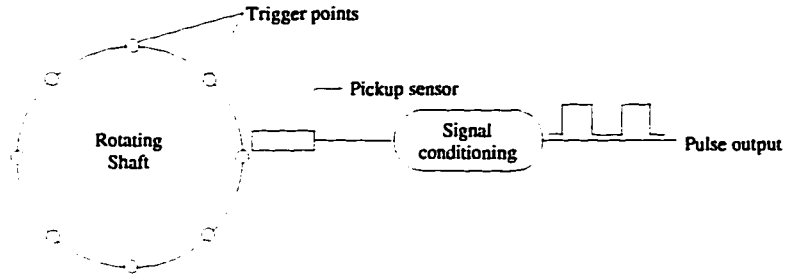


Figure 3.7: A schematic representation of an odometer

opto-electric device that generates a digital pulse when a slot passes through its field of vision. For an odometer comprised of a series of magnets and a pick-up coil, as in [43] and [48] (and in this research), the trigger points represent the magnets, and the pick-up sensor represents the coil and any necessary signal-conditioning circuitry.

However the odometer is physically implemented, it will be assumed in the following analysis that the odometer is any sensor that generates a constant integer number of digital pulses for each revolution of a rotating shaft on the vehicle. It is further assumed that the rotation rate of the shaft is (approximately) linearly proportional to the forward speed of the vehicle, and that this rotation rate is independent of whether the vehicle is turning. (An odometer on a vehicle's drive shaft would satisfy these assumptions.) Finally, it will be assumed that the odometer readings are taken at points in time separated by a constant sampling period,  $T$ , and that, between sampling points, the cumulative number of odometer pulses,  $N$ , is stored. In the following analysis,  $k$  is an integer that refers to the sample taken at time  $t = kT$ .

By way of definition, we first define the *true odometer scale factor*,  $S_{true}$ , to be the curvilinear distance traveled by the vehicle between two consecutive pulse outputs of the odometer. The value of  $S_{true}$  depends on the radii of the vehicle's tires and is therefore not necessarily constant because the radii of the vehicle's tires may vary with the vehicle's speed, the tires' air pressure, or the progressive wear of the vehicle's tires [43]. It will be assumed that significant variations in  $S_{true}$  take place over a time period that is much longer than one sampling period. Therefore, we will consider  $S_{true}$  to be constant over each sampling period and treat it as an unknown quantity. We next define the *nominal odometer scale factor*,  $S_o$ , to be a known *constant* that

is approximately equal to the true odometer scale factor,  $S_{true}$ . Finally, we define the odometer scale factor bias,  $\Delta S$ , to be the difference between  $S_{true}$  and  $S_o$ :

$$\Delta S \equiv S_{true} - S_o \quad (3.21)$$

Note that  $\Delta S$  is not necessarily constant, nor is it known exactly. Note also that  $S_{true}$ ,  $S_o$ , and  $\Delta S$  all have dimensions of distance traveled per pulse.

We now seek to derive an expression for the error in the odometer measurement. We begin the analysis by assuming that, when the  $k^{th}$  sample is taken, the pick-up sensor is located randomly, with uniform distribution, between any two trigger points on the shaft. Next, the quantity  $d_k$  is defined as the forward distance that the vehicle must travel in order to cause the next trigger point to pass the pick-up sensor. The quantity  $d_k$  is random and has a uniform distribution from 0 to  $S_{true}$ , denoted

$$d_k = U(0, S_{true}) \quad (3.22)$$

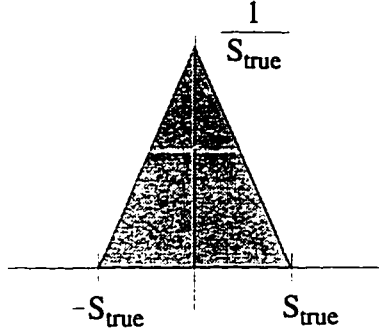
If the vehicle subsequently moves forward by some arbitrary distance  $D_{true}$  over the next  $T$  seconds, then the odometer will generate  $N$  pulses. At the start of the  $k+1^{th}$  sampling time, the pick-up sensor may be located anywhere between two trigger points. Let us define  $d_{k+1}$  as the forward distance that the vehicle must travel in order to cause the next trigger point to pass the pick-up sensor. The forward distance traveled from timestep  $k$  to timestep  $k+1$  is  $D_{true}$  and is related to  $N$  and  $S_{true}$  by

$$D_{true} = S_{true}(N - 1) + d_k + (S_{true} - d_{k+1}) \quad (3.23)$$

or

$$D_{true} = S_{true}N + d_k - d_{k+1} \quad (3.24)$$

The righthand side of Equation 3.24 contains the difference of two random variables,  $d_k$  and  $d_{k+1}$ , both of which are uniformly distributed from 0 to  $S_{true}$ . This difference is also a random variable, which shall be denoted  $d_{k,k+1}$ , whose distribution is the convolution of two probability density functions:  $U(0, S_{true})$  with  $U(0, -S_{true})$ . This distribution is shown in Figure 3.8. Therefore,

Figure 3.8: Probability density function of  $d_{k,k+1}$ 

$$D_{true} = S_{true}N + d_{k,k+1} \quad (3.25)$$

The quantity  $d_{k,k+1}$  is quantization error that arises because the odometer discretizes the distance traveled by the vehicle into segments that are each  $S_{true}$  in length.

We now turn our attention to the measurement that is made by the odometer. The measurement that is made by the odometer is the distance  $D_{meas}$ , the product of  $S_o$  and  $N$ :

$$D_{meas} \equiv S_o N \quad (3.26)$$

In general,  $D_{meas}$  will not be equal to  $D_{true}$ , not only because  $S_o$  is not generally equal to  $S_{true}$ , but also because  $D_{true}$  contains the random quantity  $d_{k,k+1}$ . We seek, as the result of this analysis, a mathematical expression for the difference between  $D_{meas}$  and  $D_{true}$ . To that end, we next define the error in the measurement,  $D_{error}$ , such that

$$D_{meas} = D_{true} + D_{error} \quad (3.27)$$

then, substituting from Equations 3.21 and 3.26 into Equation 3.27, we arrive at

$$D_{meas} = (S_{true} - \Delta S)N \quad (3.28)$$

If we next solve Equation 3.25 for  $S_{true}$  and substitute the resulting expression into

Equation 3.28, we arrive at

$$D_{meas} = D_{true} - d_{k,k+1} + N \Delta S \quad (3.29)$$

Finally, setting the righthand sides of Equations 3.27 and 3.29 equal to each other, we arrive at an expression for the error in the distance measured by the odometer:

$$D_{error} = N \Delta S - d_{k,k+1} \quad (3.30)$$

Equation 3.30 shows that the error in the measured distance has two components: a non-random component that is proportional to the distance traveled and a random component that is distributed as shown in Figure 3.8. Equation 3.30 is the basic error equation for the odometer. However, this equation must be developed further before we will arrive at a suitable error equation. Before developing Equation 3.30 further, however, a model for the time history of  $\Delta S$  must be derived.

The variations in  $\Delta S$  over time depend on vehicle speed, vehicle loading, tire pressure, temperature, and tire wear. For a given vehicle loading, the most significant of these factors over short time periods are vehicle speed and tire pressure [43]. The effects of vehicle loading are not investigated in [43]. Furthermore, in this work, the influence of vehicle loading on the odometer scale factor is assumed to be constant in all simulations because the load within a vehicle generally does not change while the vehicle is in use. Therefore, there is no need to include vehicle-loading effects in the model for the odometer scale factor's time history. Tire wear can change the odometer scale factor significantly [43]. However, this wear takes place over the lifetime of the tire, and can therefore be considered constant over short time spans.

It is surprisingly difficult to empirically measure small changes in an odometer scale factor because such testing requires extremely accurate position measurements (more accurate than stand-alone GPS can provide) and a long straight track on which the vehicle can travel. Because of the difficulties associated with testing the time-varying component of an odometer's scale factor, such tests were not performed. As suggested in Equation 3.21, the odometer scale factor,  $S$ , has been modeled as the sum of a (known) constant  $S_o$ , and a time-varying bias, denoted  $\Delta S$ . The bias has been

modeled as the sum of a first-order Gauss-Markov process and a speed-dependent term:

$$\Delta S = S_b + K_{S_b} V \quad (3.31)$$

where

$$\dot{S}_b = -\frac{1}{\tau_{S_b}} S_b + u_{S_b} \quad (3.32)$$

The quantity  $K_{S_b}$  is a constant,  $V$  is the vehicle's speed, and  $u_{S_b}$  is zero-mean white noise with a Gaussian distribution.

The value of  $K_{S_b}$  used in this research was obtained from empirical results in [43]. The parameters governing the random part of this model (Equation 3.32) are difficult to choose. No useful empirical data describing the random variations in a scale factor odometer have been found in the literature, and it is difficult to obtain accurate empirical data. Therefore, for the filter model, worst-case values for  $\tau_{S_b}$  and the RMS value of  $u_{S_b}$  were selected based on estimated worst-case tire temperature and pressure variations. (The reader is referred to Appendix A for the particular values of these quantities.) For the reference system's model, a range of parameter values was used in order to explore the range of bias variations that would likely occur in a real system.

Substituting from Equation 3.31 into Equation 3.30 for  $\Delta S$ , we arrive at the final expression for the odometer error equation:

$$D_{error} = S_b N + K_{S_b} V N - d_{k,k+1} \quad (3.33)$$

### 3.5 GPS Discussion and Error Modeling

As of the writing of this thesis, the only type of GPS positioning available to civilian users worldwide is unaided positioning corrupted by SA. It is possible, however, that SA will be turned off, and free differential corrections may become available over the entire continental United States in the near future. In either case, the accuracy of GPS position fixes available to civilian users would improve significantly. This may have significant impact on land-vehicle navigation design because the improved positioning

accuracy may cause the relative impact of dead-reckoning sensors on overall system performance to change. For example, improved positioning accuracy may improve dead-reckoning sensor calibration and may, therefore, permit navigation system designers to relax requirements on dead-reckoning sensor performance. This possibility is a compelling reason to investigate the impact that each type of GPS positioning has on the contributions that dead-reckoning sensors make to overall system performance.

### 3.5.1 GPS with SA On

The equations used to model SA-induced positioning errors form a second-order Gauss-Markov process. These equations were first presented in [18] and have been shown to accurately model position error induced by Selective Availability. In [18], the author derived this model from 4 sets of GPS measurements, each of which was approximately 5 hours long. Each data set was taken at a different time of day. The subscript  $x$  in the following equations denotes quantities associated with the x-axis in a locally horizontal xy coordinate frame that is fixed to the Earth; the equations modeling the position error along the y-axis are identical (except for the subscripts) and are therefore not included.

$$\dot{\lambda}_x = \xi_x \quad (3.34)$$

$$\dot{\xi}_x = -\beta_x^2 \lambda_x - 2\beta_x \xi_x + u_x \quad (3.35)$$

$$u_x = N(0, \sigma_x^2) \quad (3.36)$$

This model for SA-induced positioning error is included in both the Kalman filter's model equations and in the reference system's model equations. (The values for the parameters that govern these equations can be found in Appendix A.)

### 3.5.2 GPS with SA Off

The error model for this type of GPS positioning has proven the most difficult to justify because no real data could be found from which a model could be empirically derived. The only source of data was found in [54, Chapter 11]. This reference does

not include any differential equations which model the time history of this type of GPS position error. However, the data in this text show that the RMS bias error in each pseudorange is approximately 5.1 meters and the RMS value of the uncorrelated noise in each pseudorange is anywhere from 0.4 to 1.4 meters, depending on how much averaging the receiver does. With no measurement data available and no other source of modeling information, the model used for this research is given by

$$\dot{\lambda}_x = \xi_x \quad (3.37)$$

$$\dot{\xi}_x = -\beta_x^2 \lambda_x - 2\beta_x \xi_x + u_x \quad (3.38)$$

$$u_x = N(0, \sigma_x^2) \quad (3.39)$$

Note that the form of this model is identical to the one for GPS positioning error with SA *on*. The difference is that the parameters that govern the equations ( $\beta_x$  and  $\sigma_x$ ) have different numerical values. This model was also used in [18] to simulate GPS with SA off. (The values for the parameters that govern these equations can be found in Appendix A.)

### 3.5.3 Differential GPS

Unlike unaided GPS, the use of DGPS requires a source of differential corrections. As of the writing of this thesis, there is no single widely-available source of free differential corrections. However, a few commercial sources of corrections exist, and a few sources of free corrections exist in restricted geographical areas. Commercial differential corrections are services to which users can subscribe. For a fee, subscribers are given access to differential corrections that are broadcast on a radio frequency in their locale.

Two other sources of differential corrections may be available to navigation systems for land-based vehicles, although one is not yet widely available. The U.S. Coast Guard (USCG) has established differential GPS correction stations covering the U.S. coasts and inland waterways. This system provides differential GPS corrections to

the marine community free of charge [22]. The radiobeacons broadcasting the corrections transmit nondirectionally at a frequency of 285-325 kHz with enough power to reach a user 10 to 175 miles away [14, 23]. Therefore, although not designed primarily for land-based GPS users, the corrections may be receivable by land-based GPS users in the vicinity of the U.S. coasts and inland waterways. A similar differential correction service may be placed inland to cover the entire continental U.S. [20] (For excellent discussions of the USCG DGPS system and its performance characteristics, the reader is referred to [1], [14], and [15].)

A second source of differential corrections that may become available to land-vehicle navigation systems is the Wide Area Augmentation System (WAAS). The WAAS is a GPS-based navigation system currently being developed by the Federal Aviation Administration for the aviation community. According to current plans, differential GPS corrections would be broadcast free of charge over the entire U.S. by a set of geosynchronous communications satellites [53, Chapter 4].

Although DGPS position fixes are generally much more accurate than unaided GPS position fixes, accessing DGPS corrections is generally not without some cost. Commercial differential correction services, for example, add to system cost through the subscription costs and the cost of the equipment required to access the broadcast corrections. The USCG DGPS system would require equipment (in addition to a GPS receiver) to receive the broadcast corrections. In contrast, current indications are that WAAS corrections will be broadcast in such a way that an ordinary GPS receiver with an internal software modification will be able to receive them [53, Chapter 4].

An error model for DGPS position fixes was derived by the author using data obtained from Stanford University's experimental WAAS. An example of such data is shown in Figure 3.9. The data from which the error model was derived consisted of position fixes taken at 1 Hz for 7 hours. The differential correction computed using Stanford's WAAS was applied to each position measurement taken over that 7-hour period. The error in each differentially corrected position fix was calculated by subtracting the antenna's location (which was known) from its measured location. The calculated position error was then utilized (by the author) to derive the following model for the bias error in the DGPS position fixes.

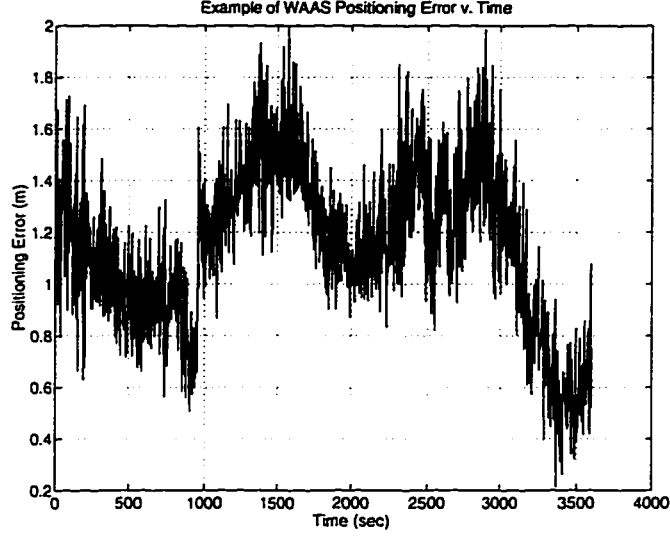


Figure 3.9: Positioning error for Stanford's WAAS

The total bias error ( $\Delta p$ ) is modeled as the sum of two biases ( $\lambda$  and  $\xi$ ), each of which is modeled as a first-order Gauss-Markov process. The time-constants ( $\tau_\lambda$  and  $\tau_\xi$ ) and RMS values of the two biases are not equal. The following model was obtained for each component (i.e. x and y) of the bias error:

$$\Delta p_x = \lambda_x + \xi_x \quad (3.40)$$

where

$$\dot{\lambda}_x = -\frac{1}{\tau_\lambda} \lambda_x + u_\lambda \quad (3.41)$$

$$u_\lambda = N(0, \sigma_\lambda^2) \quad (3.42)$$

and

$$\dot{\xi}_x = -\frac{1}{\tau_\xi} \xi_x + u_\xi \quad (3.43)$$

$$u_\xi = N(0, \sigma_\xi^2) \quad (3.44)$$

This model is used both in the Kalman filter model equations and the reference system model. (The values for the parameters that govern these equations can be found in Appendix A.)

### 3.5.4 Using GPS to Obtain a Heading Measurement

As was demonstrated by Figure 3.6, a compass is susceptible to magnetic disturbances and, as a result, its output may contain errors that occur at unpredictable times and have an unpredictable magnitude. Even if navigation software were able to successfully minimize the impact of compass errors by identifying (and ignoring) erroneous compass data, ignoring the data effectively renders the sensor useless; how frequently the sensor data would be unusable depends on the nature of the compass' magnetic environment. One way to avoid the problems associated with a compass is to replace it with a sensor that can measure absolute heading *without* relying on the Earth's magnetic field.

Conveniently, an appropriately modified GPS receiver can be used to measure absolute heading. (In the literature, much attention has been given to measuring attitude using GPS, usually for aircraft and spacecraft applications.) In addition to an appropriately modified GPS receiver, two GPS antennas, mounted in a fixed location relative to each other, would be required. These additional hardware requirements would increase the total cost of a navigation system. However, the added cost may be worthwhile because attitude measurements from GPS are quite accurate. They are immune to drift and insensitive to the magnetic disturbances which can plague compasses. Furthermore, the bias in the measurement is small if an appropriately calibrated state-of-the-art receiver is used, although the bias can be a function of temperature [53]. According to [53], the RMS error in a heading measurement (in radians) is approximately  $\frac{0.005}{L}$ , where  $L$  is the distance between the antennas (in meters). Assuming the antennas are 20 centimeters apart (which is a reasonable distance), then the RMS measurement error would be approximately 1.5 degrees. (The error in the heading may be larger than this, however, for a two-antenna attitude system if the vehicle is not horizontal.)

Another significant point in favor of using a GPS-based heading measurement is that, for an appropriately designed receiver, only *one* satellite would usually need to be in view to make the measurement, rather than the four that are required for a position measurement. This is very important, because it means that a GPS receiver could produce a heading measurement even if it could not produce a position measurement. Data collected by this author while driving in downtown San Francisco over a 30 minute period showed that the GPS receiver had 4 satellites in view approximately 10% of the time, but *always* had at least 1 satellite in view. Therefore, it is possible that, even in environments in which a GPS receiver could not produce a position fix, the GPS receiver could produce a measurement of absolute heading.

At this time, there are commercially available GPS receivers with multiple antenna inputs that are capable of measuring attitude. Unfortunately, this author knows of no *low-cost* commercially available GPS receivers that have been modified to measure only one axis of attitude. Therefore, while, in principle, it may be feasible and beneficial to use GPS-based heading measurements in an automobile navigation system, currently available GPS receivers that are capable of making attitude measurements are too expensive. Even so, it is worthwhile to explore the navigation performance improvement that is achieved when a GPS-based heading measurement is available.

In this research, the Kalman filter and reference system model equations for the GPS-based heading measurement errors are similar—in both systems, the measurement includes a bias and white noise; the bias is modeled as constant in the reference system and as a slowly-varying quantity in the Kalman filter.

### 3.6 Summary

In this chapter, error models for various sensors which commonly appear in automobile navigation systems have been presented. Some of these error models were derived from real data sensor data, while others were taken from the literature. These error models represent the first step toward developing a navigation Kalman filter and equations of sensitivity analysis. The Kalman filter will be developed in Chapter 5. Application of the sensitivity analysis to this Kalman filter will parallel the example in Chapter

2, and details will be given in Chapter 5. Before deriving and analyzing the Kalman filter, however, it is first necessary to discuss the roles that a digital map database plays in automobile navigation.

# Chapter 4

## Map-matching

### 4.1 Introduction

A digital map database is essentially an electronic roadmap—a digitization of a local road network, with each street stored as a group of points that are assumed to be connected in a dot-to-dot fashion. Map databases in existing land-vehicle navigation systems are usually stored on a CD-ROM or hard disk and usually contain a record of every road in a particular geographical area. They may also contain much more than just geographical information about each road, including speed limits, directionality (i.e. whether a road is one-way), address information, connectivity (i.e. whether roads which *appear* to intersect when viewed from above actually do), and the type of each road (e.g. freeway, residential sidestreet, etc.). However, map databases generally do *not* contain information on driveable surfaces other than roads (e.g. parking lots). Map databases are part of many navigation systems because the vehicle's location is usually conveyed to the driver on an electronic display that shows the local road network. But map databases can be used for more than just display purposes—information in a map database can also be used to aid in navigation if a vehicle is assumed to be traveling on a road stored in the database.

The software algorithm that employs information from the map database to aid in navigation is generally referred to as a *map-matching algorithm*. Broadly speaking, map-matching is the process by which data from the navigation sensors and map

database are combined to identify that road on which the vehicle is most likely to be traveling. There are many different implementations of map-matching algorithms that appear in the technical and patent literature [27, 47, 31, 26, 29, 30, 9, 33, 24]. Most map-matching algorithms described in these references consist of a set of heuristic rules by which sensor and map data are processed; other methods (most notably [37] and [57]) use more rigorous probabilistic methods.

The main purposes of map-matching are 1.) to provide information from the map database to aid in navigation and 2.) to convey a meaningful vehicle position to the driver (i.e. a position that lies on a road). Note that a position estimate derived from the navigation sensors alone may *not* coincide with the coordinates of a road in the map database due to error in the position estimate, error in the map database, or because the vehicle is not traveling on a road (i.e. it may be in a parking lot). If the map-matching algorithm “believes” that the vehicle is on a road, information from the database can be employed to correct the apparent navigation error and to generate a position that appears meaningful to the driver.

This chapter discusses several issues associated with map-matching and its effects on vehicle navigation. Factors in successful map-matching and the relationship between map-matching and sensor calibration are discussed. The manner in which this research examines the impact of map-matching on navigation system performance is described. Finally, at the end of the chapter, a map-matching algorithm developed by the author is described in detail.

## 4.2 Factors in Successful Map-matching and Benefits of Map-matching to Navigation

In order to be a useful aid to navigation or to convey a meaningful location to the vehicle’s driver, a map-matching algorithm must correctly identify the road on which the vehicle is traveling. The algorithm’s ability to do this successfully depends in a complex way on details of the algorithm’s design, the quality of the navigation sensor data, and the correctness of the information in the map database. Developing

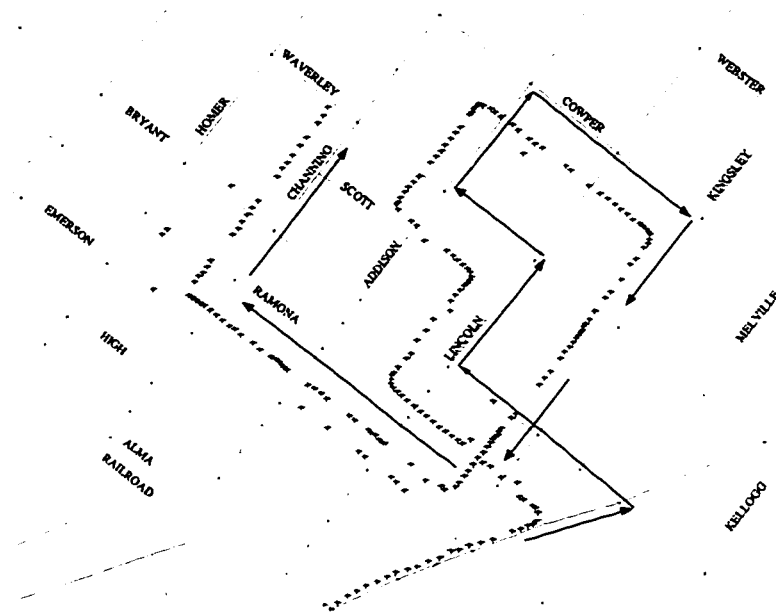


Figure 4.1: GPS position fixes overlaid on a map display

a map-matching algorithm that always works well is difficult because which road the vehicle is actually on is ambiguous—there may be many candidate roads in the vicinity of the vehicle’s estimated location. This ambiguity can be especially difficult to resolve because roads are often laid out in a grid-like fashion, and a map-matching algorithm cannot take advantage of unique geometric features in the road network to pinpoint the vehicle’s location. This concept is, perhaps, more clearly understood by examination of Figure 4.1.

Figure 4.1 shows a set of consecutive GPS position fixes (shown with “+” symbols) superimposed on a display of the surrounding road network. The position fixes were collected from a low-cost GPS receiver mounted inside an automobile. (The GPS receiver’s antenna was mounted to the vehicle’s roof.) The map display was drawn utilizing information from a digital map database produced by Etak, Inc. The figure also shows arrows that represent the path that the vehicle *actually* took. Comparison of the path represented by the position fixes with the arrows shows that the position fixes are offset from the road due to the influence of Selective Availability. The figure also illustrates that the GPS position fixes occasionally exhibit gaps and large lateral

jumps. Such gaps and jumps in the position fixes can be “smoothed” when the GPS position fixes are properly fused with data from dead-reckoning sensors. However, results have shown that the bias in the position fixes cannot be eliminated even when dead-reckoning sensors are utilized because dead-reckoning sensors do not measure absolute position. The figure illustrates that the path actually taken by the vehicle may be ambiguous when the sensor data alone is examined.

Successful map-matching, then, depends on the successful execution of two tasks: first, to correctly resolve this ambiguity after initial startup, given sensor data that is erroneous and map database information that *may be* erroneous and, second, to continuously maintain a “lock” on the correct road as the vehicle moves about. Whether a map-matching algorithm will be successful depends very much on the strengths and weaknesses of the particular algorithm. It is therefore difficult to make general statements about how successful map-matching algorithms are and how much map-matching improves navigation accuracy.

### 4.2.1 Initially Identifying the Correct Road

There are elements that are common to many map-matching algorithms, and from these we can, perhaps, identify some of the ingredients in successful map-matching. Most of the heuristic map-matching algorithms that appear in the literature attempt to solve the ambiguity problem by employing a set of rules that process sensor and map data to implement some form of pattern-matching [27, 47, 31, 26, 29, 30, 9, 33]. Pattern-matching algorithms generally attempt to correlate the pattern created by many consecutive position fixes to a similar pattern of roads in the surrounding road network. When the pattern created by the position fixes is deemed to be sufficiently similar to a pattern of connected roads in the road network, a map-matching algorithm will generally select a single road from the database as that road on which the vehicle is most likely to be. Once this has been established, the navigation software may begin to include information from the map database to aid in sensor calibration.

Successful pattern-matching obviously depends on how well the pattern created by consecutive position fixes (from the navigation sensors) matches a unique pattern

of roads in the vicinity and on how unique that pattern is. The shape of the pattern created by the position fixes depends on the quality of dead-reckoning sensor data and the quality of GPS fixes obtained. At one extreme, for example, accurate DGPS fixes may always be available. If this is the case, the need for map-matching may be obviated because the DGPS fixes should lie on a road stored in the map database. If they do not, then the error is probably in the map database. At the other extreme, GPS fixes may not be available at all, and the navigation system has only dead-reckoning sensor data with which to navigate the vehicle. Under these circumstances, the map-matching algorithm's performance will depend on the drift characteristics of the dead-reckoning sensors and the accuracy with which the sensor errors were initially calibrated. The overall performance of the navigation system will depend on the map-matching algorithm's ability to continuously calibrate the drift of the dead-reckoning sensors. In a third scenario (one that is more likely for existing navigation systems), GPS (with SA on) fixes are available. Because of SA, the error in the position fixes will drift slowly within 100 meters of the actual vehicle location and may increase as the environment becomes increasingly hostile to accurate GPS positioning. In addition, results will show (in Chapter 6) that typical dead-reckoning sensors are not accurate enough to allow SA-induced position error to be estimated accurately. Therefore, because city streets may be closer together than 100 meters, the street on which the vehicle is actually traveling may be difficult to determine. Under these circumstances, the success of a map-matching algorithm becomes difficult to predict because it depends on particulars of the algorithm and the situation under which the algorithm is being tested—one algorithm may succeed where another would fail, depending on the particular strengths and weaknesses of each implementation.

Successful pattern-matching also depends on how well the information in the map-database represents the real world. Map-databases may contain erroneous information. For example, roads that actually exist may not be in the database or they may be incorrectly placed [45]. Even if, at some point in time, a given database is wholly correct, errors would be introduced as new roads are constructed and old ones are closed or changed. Some work has been done to examine the effect of database errors

on the performance of a particular navigation system [6]. In [6], the authors simulated errors in a map database by altering the contents of the database so that some roads were distorted from their actual shape or were taken out altogether. The authors quantified the effects of these database errors by simulating vehicle motion over the modified areas of the database and examining the performance of the vehicle's navigation system. The authors found that, for their navigation system, database distortions had a measurable effect on the navigation system's performance; when roads were missing from the database, the map-matching algorithm utilized by the navigation system placed the vehicle on the wrong road by choosing a road that *was* in the database. However, the work in [6] is limited, and the authors point out that their work is not extensive enough to support general conclusions.

#### 4.2.2 Sustaining Successful Map-matching

Once a map-matching algorithm “believes” that it has correctly identified the road on which the vehicle is traveling, it must subsequently maintain a “lock” on the correct road as the vehicle proceeds. Its ability to do this successfully will depend, of course, on the quality of the sensor data and the strengths and weaknesses of the algorithm. However, there are some situations which are generally difficult for most map-matching algorithms. For example, one difficult situation for many map-matching algorithms arises when a vehicle is traveling on a highway and approaches an exit ramp heading off the highway. In this situation, if the exit ramp splinters off the highway at a shallow angle, it may be difficult for a map-matching algorithm to resolve whether the vehicle remained on the highway or headed onto the exit ramp. Whether a particular map-matching algorithm would be successful in this situation would probably depend heavily on the quality of the raw heading sensor data.

Another difficult situation arises when a vehicle travels on a straight road for a long distance, then turns; if there are other nearby roads which branch off the straight road, the map-matching algorithm may have difficulty resolving the road onto which the vehicle actually turned. The reason for this has to do with the fact that along-track error tends to accumulate when a vehicle travels on a long straight road, even

if map-matching is in use. Traveling on a long straight road for an extended period allows along-track positioning error to accumulate due to bias error in the odometer's scale factor. For example, if the odometer scale factor estimate is accurate to 1% (typical), along-track error will grow to 50 meters if the vehicle travels in a straight line for 5 kilometers. If GPS position fixes are available, the along-track error will be bounded by the error in the GPS position fix; however, if the GPS position fixes are corrupted by SA, this could be as large as 100 meters.

A third difficult situation arises when the vehicle travels off the road and onto a driveable surface which is *not* in the map database. Examples of common driveable surfaces that are generally not included in map databases include parking lots and decks. In these situations, the map-matching algorithm must deliberately abandon the assumption that the vehicle is on a road. Obviously, the navigation software must proceed using only sensor data and must stop using information in the map database to aid in navigation. One can imagine that developing a robust algorithm to perform well under these circumstances would be difficult.

Finally, in geographical areas in which roads are densely arranged in a regular grid, a navigation system may have difficulty identifying the road on which the vehicle is traveling. This problem is particularly difficult because the roads are not only close together, but the regular layout of the roads impedes effective pattern matching. Choosing one road from among two (or more) that are closely-spaced and parallel is cited by the authors of [6] as one of the most difficult problems in vehicle positioning.

### 4.2.3 Using Map-matching Information to Aid Navigation

Once a map-matching algorithm chooses a road on which it believes the vehicle is likely to be traveling, the navigation software can make use of information in the map database to aid in positioning the vehicle. Precisely *what* information is used and *how heavily* the map information influences the final position estimate depends on the particulars of the map-matching algorithm. For example, in [26], one of the earliest patents on map-matching, the algorithm *sometimes* uses the heading of the matched road to correct errors in readings from a compass; whether the correction is made

depends on the degree of confidence in the map-matched position. In contrast, the system described in [39], [24], and [25] incorporates map database information into the navigation solution only when the vehicle turns a corner. In yet another system (described in [9]), the map-matched position is continuously fed back to calibrate the dead-reckoning sensors. Means for calibrating navigation sensors using map-matched positioning are also mentioned in [31] and [29], but details are not given. In each of the navigation systems for which sufficient information is available, information from the map database is used differently. For this reason, it is difficult to make general statements about the extent to which map-matching improves navigation accuracy.

However, it seems generally true that the *location* and/or *direction* of the road can be extracted from the database and used as a “measurement” of cross-track position and/or heading. It is important to emphasize that the map-matched position provides the navigation system with an accurate estimate of the vehicle’s position *perpendicular* to the road, but not parallel to it. (This may be more clearly understood if one realizes that the system’s position error can be resolved into two orthogonal components: one parallel to the road and one perpendicular to it.) Knowledge of the road’s location can be used to correct only the perpendicular component of the position error. One implication of this fact is that map-matching can be used to continuously calibrate a navigation system’s heading sensors. Errors in a compass reading and a rate gyro’s bias, for example, can be calibrated if the map-matched position information is incorporated into the navigation solution appropriately. The odometer scale factor, on the other hand, cannot be continuously calibrated. (It should be noted that correct map-matched positioning can provide very accurate 2-dimensional position measurements when the vehicle turns, because a turn is a unique feature in the map database that locates the vehicle precisely. This is recognized and taken advantage of in [39], for example.)

Although map-matching can be utilized as a navigation aid, the use of map-matching to calibrate dead-reckoning sensors is not always beneficial. Errors in the map-matched position can skew the sensor calibration, causing the navigation system to produce erroneous estimates of certain sensor errors. For example, the map-matching algorithm may mistakenly “believe” that the vehicle is traveling on a road

which it is not really on. If the “wrong” road and the “right” road diverge, then the sensor calibration could be ruined. This is an obvious example. However, there are much more subtle ways in which map-matching can skew sensor calibration, even if the map-matching algorithm always identifies the correct road.

Specifically, map-matching can degrade heading sensor calibration as a result of the vehicle’s side-to-side motion on a road. For example, as the vehicle travels down a straight road, correctly map-matched positions will lie along a straight line because the road’s representation in the map database would consist of coordinates lying on a straight line. However, human drivers do not drive in perfectly straight lines and cause their vehicles to wander slowly side-to-side several decimeters. If the map-matched position is used as a measurement of the vehicle’s cross-track position, these “measurements” of cross-track position would indicate that the vehicle is traveling in a straight line; however, the system’s heading sensors (e.g. a rate gyro and/or compass), which measure the vehicle’s *true* motion, would indicate that the vehicle is *not* traveling in a straight line. The system’s sensor-fusion algorithm (a Kalman filter, for example) would therefore be presented with conflicting information from the sensors and map-matching algorithm. Presumably, this algorithm would utilize both pieces of information, weighing each one into its calculation of the vehicle’s position. Whatever weighting scheme is used (as long as neither piece of information is entirely ignored), the sensor-fusion scheme would resolve the conflict by concluding that each measurement was partly in error. As a result, it would attribute some of the apparent vehicle motion to bias drift in the rate gyro or compass. The sensor-fusion algorithm would therefore produce erroneous estimates of the errors in the gyro or compass readings. In sum, then, as a result of a driver’s tendency to wander laterally within a lane (or to change lanes on a multi-lane roadway), even a perfect map-matching algorithm can induce errors in sensor calibration. In this research, the effects of map-matching on sensor calibration will be examined from this point of view.

### 4.3 Analyzing the Influence of Map-matching on Navigation System Performance

The results to be expected from the type of analysis just described may not be immediately obvious. As has been mentioned, a subtle error in sensor calibration appears as a result of the vehicle's lateral motion on a road. This section attempts to demonstrate more clearly why this error arises.

Two-lane roads (i.e. two individual lanes, the traffic on which moves in opposite directions) are usually represented in map databases with one set of coordinates. Each coordinate is a latitude/longitude pair (or equivalent) that should lie on the physical road. If there is no error in the digitization of the database, these coordinates should coincide with the centerline of the road. The cross-track position "measurement" that would be obtained from the map database, then, is a position on a line coinciding with the centerline of the road. If the vehicle travels down the center of either lane, then the error in the map-matched position would be a bias approximately half the width of one lane. Figure 4.2 shows this error. As the driver causes the car to wander from side-to-side within the lane, this bias would change with time.

Divided multi-lane highways present other difficulties. Because the vehicle has the ability to change lanes on a multi-lane highway, the vehicle's actual location can "float" relative to the centerline of the freeway as the driver changes lanes. Therefore, the bias in the map-matched positions would change with time. Figure 4.3 graphically illustrates the concept being described.

The research in this thesis focuses on map-matching by examining the effects that lane changes and lateral motion within a lane have on the calibration of dead-reckoning sensors. As was mentioned, in this research, the map-matched positions are supplied to the navigation Kalman filter as cross-track position "measurements." Because these cross-track position "measurements" lie on a line and the actual vehicle motion is not straight, the errors in the map-matched positions will induce errors in the calibration of the dead reckoning sensors, particularly the heading sensors. How severely the calibration of the dead-reckoning sensors is affected will probably depend on the nature of the vehicle's motion—e.g. more rapid lane-changes will probably

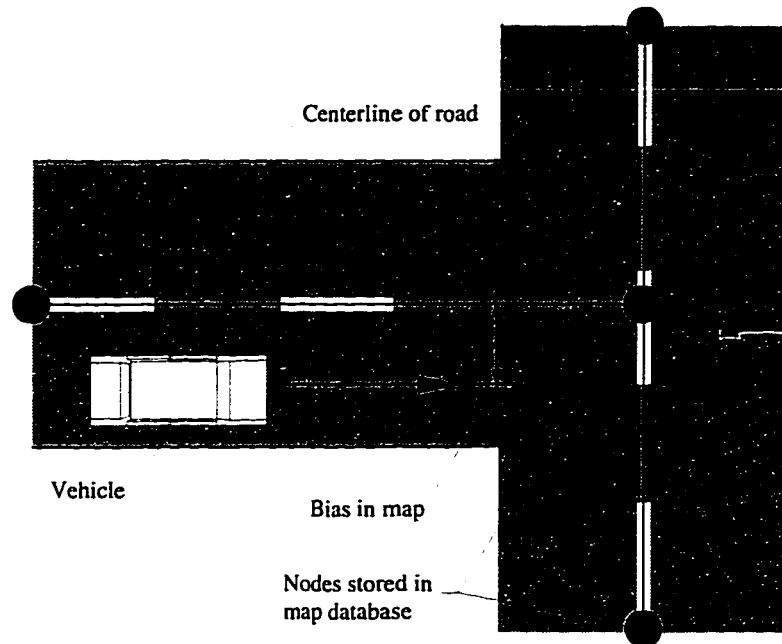


Figure 4.2: Bias error due to map-matching on a two-lane road

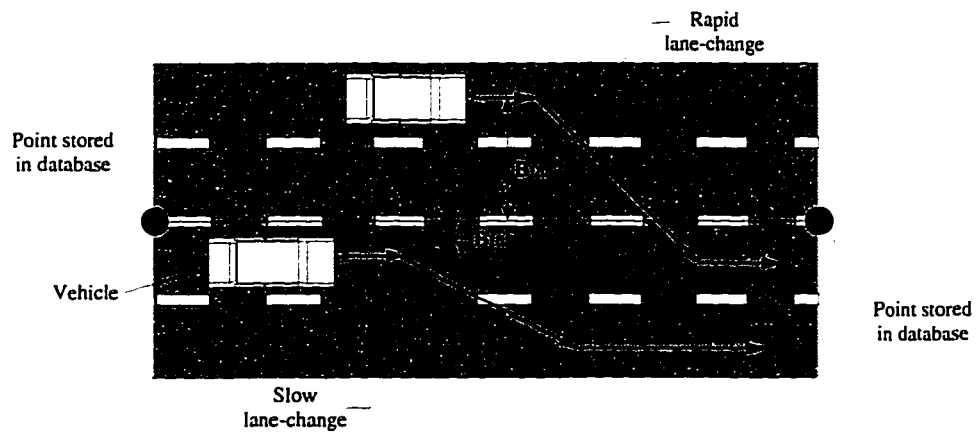


Figure 4.3: Lane changes on a multi-lane highway

induce larger errors than slow lane-changes. These calibration errors will be quantified in this research and will be attributed to map-matching. Details regarding the manner in which map-matching is utilized by the Kalman filter designed for this research are given in Chapter 5, and results can be found in Section 6.7.

## 4.4 A Detailed Description of a Map-matching Algorithm

This section presents an example map-matching algorithm developed by the author. A detailed description of the algorithm follows. Briefly stated, the map-matching algorithm starts by assuming that the vehicle's actual location is somewhere in the vicinity of the first GPS position fix that is obtained. (If a GPS position fix is not available, the algorithm must be initialized manually with an approximate location and heading.) Based on this first GPS fix, the algorithm assumes that every location on every road in the vicinity is a possible location for the vehicle. As time progresses, the algorithm attempts to identify the "most likely" location for the vehicle by implementing a form of pattern-matching; after a "most likely" location is identified with some degree of confidence, the algorithm supplies the coordinates of the "most likely" location to the Kalman filter as a measurement of position. The filter treats this position measurement as being very accurate in the cross-track direction, but very inaccurate in the along-track direction. Thenceforth, the map-matching algorithm continuously updates the "most likely" location and supplies it to the filter as a measurement of position. Details of the pattern-matching approach, which lie at the heart of the map-matching algorithm, follow.

### 4.4.1 The Basis for Pattern-matching: Two Heuristic Observations

The pattern-matching approach that is used to identify and update the "most likely" location for the vehicle is based on two heuristic observations. First, the vehicle's position (as estimated by the Kalman filter from sensor data *alone*) generally does not

fall directly on any particular street, but is, instead, usually *offset* from the street by a slowly-changing distance. Hence, the pattern traced on the map by many consecutive position fixes closely resembles the actual path taken on the road network. Second, the actual path of the vehicle becomes evident *over time* because that pattern created by many consecutive estimated positions correlates highly to a unique path through the road network. Figure 4.1, shown previously, illustrates these concepts. This figure shows GPS position fixes overlaid on a display from a digital map database. Comparison of the path represented by the position fixes with the vehicle's actual path (shown with arrows) demonstrates that the path actually taken by the vehicle may be ambiguous when the sensor data alone is examined. The core idea of the algorithm developed in the following section is the means by which this ambiguity is resolved.

#### 4.4.2 Resolving Ambiguity with a Cost Function

In this research, the solution to the ambiguity problem is premised on the observation that the discrete position fixes obtained from sensor data alone could be put onto the correct road on the map if a *correction*, or *shift*, were added to each estimated position. Such a correction would, in general, have components in both the x and y coordinate directions (i.e. East- and North-pointing axes in a local East-North-Up frame) that would change slowly, as the error in the individual position estimates drifted. This concept is illustrated in Figure 4.4. Figure 4.4 shows (with arrows) that a nearly constant correction applied to each position estimate causes the position estimate to shift to the actual location of the vehicle on the road. Examination of the real data in Figure 4.1 also suggests that this is true if the sudden jumps in the fixes are ignored. (Note that, even if the GPS position fixes contain sudden jumps, their influence on the navigation system's estimate of the vehicle's location can be greatly reduced when the GPS position fixes are combined with dead-reckoning sensor information. As a result, the combined GPS/dead-reckoning position solution is generally "smooth" and offset from the road by a slowly-changing amount.)

Thus, it appears that the vehicle's estimated location *could* be shifted to the

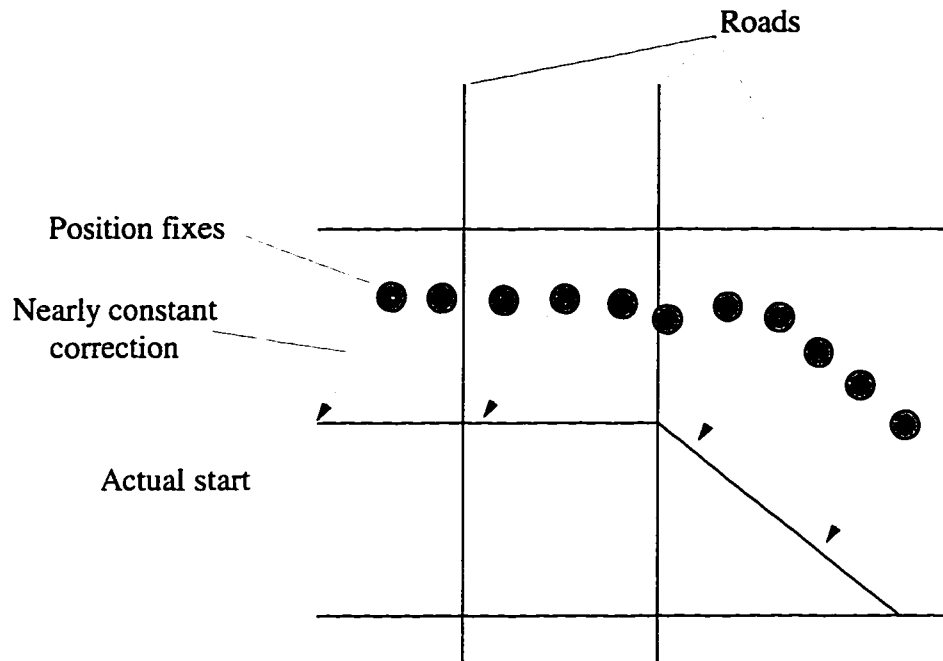


Figure 4.4: A nearly-constant correction applied to consecutive estimated positions

actual vehicle's location if a 2-dimensional correction were added to the vehicle's estimated xy-position. However, *which way* to shift the estimated position is initially ambiguous because which road the vehicle is on is unknown—if there are several roads in the vicinity of the first position estimate, then there are many candidate position corrections that could be applied to it to shift it to a nearby road. However, as long as the vehicle is on a physical road for which the database has a record, then there exists a *unique* xy-correction that can be applied to the estimated vehicle location to cause it to coincide with the vehicle's actual location. Figure 4.5 illustrates this idea. As the figure suggests, the estimated location for the vehicle is in error because the navigation sensors have errors in their outputs; which correction is the right one is therefore ambiguous because vehicle's actual location is unknown. It is the job of the map-matching algorithm to resolve this ambiguity and determine which correction is the right one.

The map-matching algorithm developed here resolves this ambiguity by making use of a series of identical error (or “cost”) functions—one error function is created

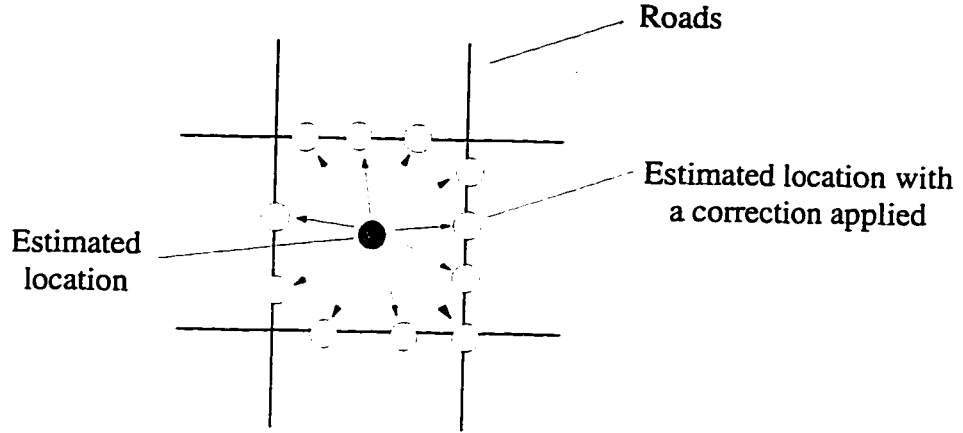


Figure 4.5: Many candidate corrections applied to a single estimated position

for every candidate correction. The error function is designed so that its *cumulative* sum for a given correction increases rapidly over time if the corrected location is inconsistent with the information from the map database. The most likely location of the vehicle is therefore chosen to be the corrected location whose error function has the smallest cumulative sum.

To illustrate the concept of a cost function in this context, we consider the following example. Suppose that, at some instant in time, the Kalman filter's estimate of the vehicle's position is at the location shown by the shaded circle in Figure 4.6. Furthermore suppose that there are three candidate corrections (labeled 1, 2, and 3) that shift the vehicle's estimated location to nearby roads (shown by the white circles in Figure 4.6). Notice that the estimated vehicle heading is nearly parallel to the vertical roads and that traffic on one of the vertical roads is restricted to movement in one direction. For each of the three candidate positions shown in Figure 4.6, let us introduce and evaluate the cost function

$$C_k^j \equiv \psi_x^2 (\epsilon_{y,k}^j)^2 + \psi_{xy}^2 \epsilon_{x,k}^j \epsilon_{y,k}^j + \psi_y^2 (\epsilon_{y,k}^j)^2 + \psi_\theta^2 (\epsilon_{\theta,k}^j)^2 + \psi_d^2 (\epsilon_{d,k}^j)^2 + \dots \quad (4.1)$$

where  $C_k^j$  is a "cost" associated with the  $j^{\text{th}}$  correction evaluated at timestep  $k$ , and the quantity  $\epsilon_{d,k}$  is a directionality error defined as 1 if a one-way restriction is violated and 0 if a one-way restriction is not violated; each  $\psi$  is a weight chosen by the designer.

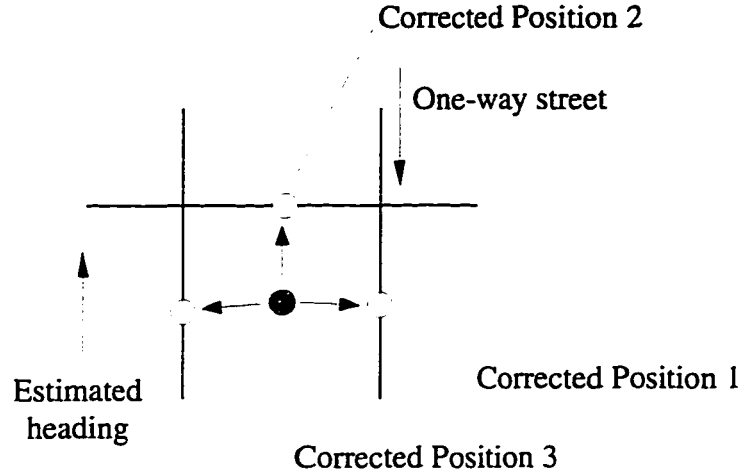


Figure 4.6: A demonstration of 3 possible corrections and their associated “costs”

The ellipsis marks in Equation 4.1 indicate that more terms than those shown can be included in this cost function definition, depending on how much information is available in the map database. The other quantities are defined as

$$\epsilon_{x,k}^j \equiv x_{m,k} - x_{c,k}^j \quad (4.2)$$

$$\epsilon_{y,k}^j \equiv y_{m,k} - y_{c,k}^j \quad (4.3)$$

$$\epsilon_{\theta,k}^j \equiv \theta_{m,k} - \hat{\theta}_k \quad (4.4)$$

where  $(x_{c,k}^j, y_{c,k}^j)$  is the  $j^{th}$  corrected xy-location of the vehicle,  $(x_{m,k}, y_{m,k})$  is an arbitrary point on a road in the map database,  $\theta_{m,k}$  is the heading of the road at  $(x_{m,k}, y_{m,k})$ , and  $\hat{\theta}_k$  is the Kalman filter’s estimate of heading. (Note that  $\theta_{m,k}$  and  $\epsilon_{d,k}$  are determined once a point on the map  $(x_{m,k}, y_{m,k})$  is chosen.) The  $j^{th}$  corrected location  $(x_{c,k}^j, y_{c,k}^j)$  is computed according to

$$x_{c,k}^j = \hat{x}_k + \Delta x_{c,k}^j \quad (4.5)$$

$$y_{c,k}^j = \hat{y}_k + \Delta y_{c,k}^j \quad (4.6)$$

where  $(\hat{x}_k, \hat{y}_k)$  is the estimated xy-location of the vehicle,  $\Delta x_{c,k}^j$  is the  $j^{th}$  x-coordinate correction and  $\Delta y_{c,k}^j$  is the  $j^{th}$  y-coordinate correction.

The manner in which each correction  $(\Delta x_{c,k}^j, \Delta y_{c,k}^j)$  is obtained is at the heart of this map-matching algorithm and is determined using the cost function defined in Equation 4.1. At this point, we seek to qualitatively evaluate  $C_k^j$  ( $j = 1, 2, 3$ ) for each of the 3 corrections depicted in Figure 4.6. (Note that the time index  $k$  is not relevant to this example, but is kept in the notation for consistency.) Our goal is to identify which of the 3 candidate corrections is most likely to be the one that shifts the estimated vehicle location to the true vehicle location. To achieve this goal, we seek to identify one particular value of  $C_k^j$  for each correction, denoted  $C_k^{j*}$ . The quantity  $C_k^{j*}$  is the minimum value of  $C_k^j$  that is obtained after evaluating  $C_k^j$  at all points on every road in the map database that lie in the vicinity of  $(x_{c,k}^j, y_{c,k}^j)$ . In other words, for each correction,  $C_k^j$  is repeatedly evaluated at every point  $(x_{m,k}, y_{m,k})$  in the vicinity of  $(x_{c,k}^j, y_{c,k}^j)$ , with the goal of finding the minimum value of  $C_k^j$ . The general form of  $C_k^j$  is such that, as the point  $(x_{m,k}, y_{m,k})$  at which  $C_k^j$  is evaluated moves farther from  $(x_{c,k}^j, y_{c,k}^j)$ , the value of  $C_k^j$  increases quadratically; since the algorithm's goal is to minimize  $C_k^j$ , it is not necessary to evaluate  $C_k^j$  at every point in the entire database. In sum, then, for each correction, the goal is to find that point on the map  $(x_{m,k}^*, y_{m,k}^*)$  that minimizes the value of  $C_k^j$ .

If the cost function (Equation 4.1) were evaluated for each of the corrected positions shown in Figure 4.6, then Corrected Position 1 would incur additional cost because the road on which it lies is restricted to one-way travel in the wrong direction. Corrected Position 2 would also incur additional cost because the heading of the road on which it lies differs significantly from the estimated heading of the vehicle. Finally, Corrected Position 3 would incur no additional cost for violating a one-way restriction, nor would it incur much additional cost for errors in heading. Therefore, if the weights (denoted with a  $\psi$  in Equation 4.1) are chosen appropriately, the correction with the minimum cost would be Corrected Position 3. As time passes, the vehicle would presumably move about on the road network, and consecutive position estimates would form a pattern that is similar to the actual path of the vehicle. At regular intervals, the cost function for each correction ( $C_k^j$ ) must be re-evaluated, and its minimum value ( $C_k^{j*}$ ) must be found. The cumulative sum of the minimum cost must be tracked individually for each correction. The most likely location for the

vehicle is determined to be that corrected position for which the *cumulative* sum of  $C_k^{j*}$  is minimum over all  $j$  and  $k$ . Mathematically, the most likely location for the vehicle after the  $k^{th}$  update is deemed to be the  $j^{th}$  corrected position if

$$S_k^j = S_k^* \quad (4.7)$$

where

$$S_k^* \equiv \min_{j=1, \dots, N} \left\{ \sum_{i=1}^k C_i^{j*} \right\} \quad (4.8)$$

and  $N$  is the total number of candidate corrections.

If the vehicle traces a path that correlates highly to a unique path in the road network, those corrected positions that lie on the wrong road should suffer significant increases in their associated cumulative cost because they should be inconsistent with the pattern of roads in the road network. In contrast, corrected positions that lie near or coincide with the vehicle's true location should maintain a low cumulative cost because they are most consistent with the information in the database.

To summarize, the pattern-matching method presented here recognizes that the difference between the vehicle's actual position and its estimated position generally changes slowly. Therefore, a nearly-constant correction can be added to a vehicle's estimated position to put it in the correct location on the map. However, there are many possible position corrections that could be added to the vehicle's estimated position to put it on a nearby road. Therefore, which correction is the right one is ambiguous and must be resolved. In the algorithm just described, this ambiguity is resolved by evaluating a cost function for each correction over time; that correction with the minimum cumulative cost is deemed to be the most likely correction. The cumulative cost for each correction should therefore be inversely related to the likelihood that the correction shifts the vehicle's estimated location to its true location—i.e. corrected locations with a low cumulative cost are more likely to coincide with the vehicle's actual location than those locations with a high cumulative cost.

### 4.4.3 Updating Each Correction Over Time

Because the Kalman filter's estimate of position will generally drift relative to the actual location of the vehicle, corrections should not remain constant, and must, instead, vary over time. To allow the corrections to vary with time, the algorithm at hand utilizes an adaptive scheme that updates each correction.

At each update (update  $k$  for example), each correction is processed sequentially. The first step in the update of the  $j^{\text{th}}$  correction is to add the xy-correction  $(\Delta x_{c,k}^j, \Delta y_{c,k}^j)$  to the filter's estimate of the vehicle's position  $(\hat{x}_k, \hat{y}_k)$ , as was described in the previous section:

$$x_{c,k}^j = \hat{x}_k + \Delta x_{c,k}^j \quad (4.9)$$

$$y_{c,k}^j = \hat{y}_k + \Delta y_{c,k}^j \quad (4.10)$$

The corrected position  $(x_{c,k}^j, y_{c,k}^j)$  may not coincide exactly with a road stored in the map database because the error in the estimated position changes over time. Therefore, it may be necessary to modify the correction. For each corrected position, then, a search is initiated for that point  $(x_{m,k}^*, y_{m,k}^*)$  which lies on a road in the map database and minimizes  $C_k^j$ . Once  $(x_{m,k}^*, y_{m,k}^*)$  is found, the correction is modified so that the corrected position  $(x_{c,k}^j, y_{c,k}^j)$  “steps” toward  $(x_{m,k}^*, y_{m,k}^*)$  by some amount. The correction is updated according to

$$\Delta x_{c,k+1}^j = \Delta x_{c,k}^j + \lambda(x_{m,k}^* - x_{c,k}^j) \quad (4.11)$$

$$\Delta y_{c,k+1}^j = \Delta y_{c,k}^j + \lambda(y_{m,k}^* - y_{c,k}^j) \quad (4.12)$$

where

$$\lambda \in [0, 1] \quad (4.13)$$

The value of  $\lambda$  controls how far the corrected position “steps” toward  $(x_{m,k}^*, y_{m,k}^*)$ . For example, if  $\lambda$  is zero, then the correction does not change. If  $\lambda$  is unity, the point  $(\hat{x}_k + \Delta x_{c,k+1}^j, \hat{y}_k + \Delta y_{c,k+1}^j)$  coincides *exactly* with the point  $(x_{m,k}^*, y_{m,k}^*)$ . Experience has shown that using  $\lambda = 1$  is risky because the point  $(x_{m,k}^*, y_{m,k}^*)$  is not *always* a

sensible choice for the location of the vehicle. Making the corrected position “step” a fraction of the way toward  $(x_{m,k}^*, y_{m,k}^*)$  (e.g. with  $\lambda \approx 0.7$ ) helps hedge against this risk.

#### 4.4.4 Using the Map-matched Position in the Kalman Filter

Over time, the cumulative cost  $S^j$  associated with most corrections becomes very large. The cumulative cost of a correction is therefore a convenient criterion for determining whether the correction can be eliminated as unlikely. If the cumulative cost for a given correction exceeds a predetermined threshold, for example, that correction can be eliminated. Over time, as the vehicle’s path traces out a pattern that is unique on the road network, more corrections can be eliminated as their cumulative cost exceeds the threshold. The number of remaining candidate corrections therefore decreases over time. Once the number of candidate corrections drops to some small number, the chances that one of the remaining corrections is the right one should be fairly high, and the corrected position can be utilized by the Kalman filter as a “measurement” of position.

Logistically, several matrices in the Kalman filter mechanization must be enlarged (in real time) to accommodate the new measurement. In addition, because map-matching generally provides position information that is accurate only in that direction which lies perpendicular to the road, the filter must be “told” that the measurement is accurate only in one dimension. This involves assigning appropriate values to the Kalman filter parameters that describe the error in the map-matching “measurement.” When the map-matched position information is fed back into the filter as a position measurement, the filter can utilize the additional information to aid in the calibration of its dead-reckoning sensors just as naturally as it would if it were receiving the information from another sensor.

#### 4.4.5 Map-matching Results

This section contains results obtained using the map-matching method described in the previous section. Most of the results are provided in graphical form as plots that

show symbols representing the vehicle's position overlaid on a road network. All of the results shown were generated from the same set of real sensor data collected in a test vehicle.

Figure 4.7 shows results after the first 78 seconds of data processing. This figure demonstrates the map-matching algorithm's performance as it autonomously identifies the road on which the vehicle is traveling, without having any *a priori* information about the vehicle's location. The figure shows a road network overlaid on which are two sets of symbols: crosshairs and squares. The crosshairs (“+” symbols) in the figure represent the Kalman filter's best estimate of the vehicle's location; initially, the crosshairs do not lie on a road due to errors in the GPS position fixes. As the figure shows, this set of crosshairs appears to stop abruptly. The reason for this is that, at the point where the crosshairs stop, the map-matching algorithm began to supply its position “measurements” to the Kalman filter. The Kalman filter's estimated positions do not really stop, as it appears; instead, the filter's estimated location simply “jumps” to map-matched location.

The squares represent the location at which the map-matching algorithm believes the vehicle to be. Recall that the map-matching algorithm deems the vehicle's most likely location to be at that corrected location with the minimum cumulative cost. Therefore, the squares in Figure 4.7 represent those corrected positions for which  $S_k^j = S_k^*$ . As Figure 4.7 shows, the map-matching algorithm chooses the wrong position correction for the first five timesteps—the first five corrected positions lie on the southwest side of Abrams Court, but the vehicle was actually traveling on the northeast side of Abrams. The algorithm failed because both sides of Abrams Court are nearly parallel. The sixth corrected position, however, is in (approximately) the right location, as are subsequent corrected positions.

Figure 4.8 shows numerical values of  $S_k$  for all the corrections being tracked by the algorithm. At the beginning of the run, there are many candidate corrections. The map-matching algorithm's purpose is to resolve which correction, when applied to the filter's estimated position, produces a corrected position that matches the vehicle's actual location. The magnitude of  $S_k^j$  is inversely proportional to the likelihood that corrected position  $j$  is the actual location of the vehicle. As the figure shows, the

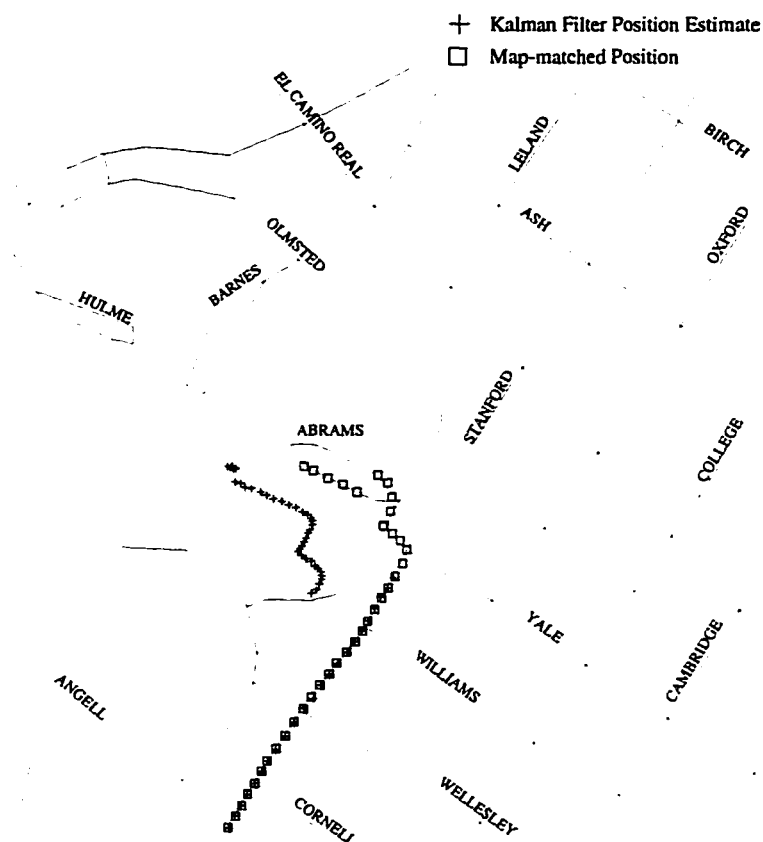


Figure 4.7: Results showing the map-matching algorithm converge on the right road

cumulative cost for most of the corrections grows rapidly. This suggests that most of the corrections that were candidates at the start of the run are incorrect. As shown in the figure, only two candidate corrections survive more than 30 correction updates. Examination of these two corrections revealed that both of them cause the corrected position to lie on the correct road, and the corrected positions are, in fact, actually very close to one another.

Inspection of Figure 4.8 shows that the cumulative cost for a group of several corrections ends abruptly. The endpoints of these curves are circled in the figure. The reason that these curves end abruptly is because the map-matching software detected that these corrections were redundant and eliminated them. In other words, these corrections were very nearly equal to other corrections; when the map-matching

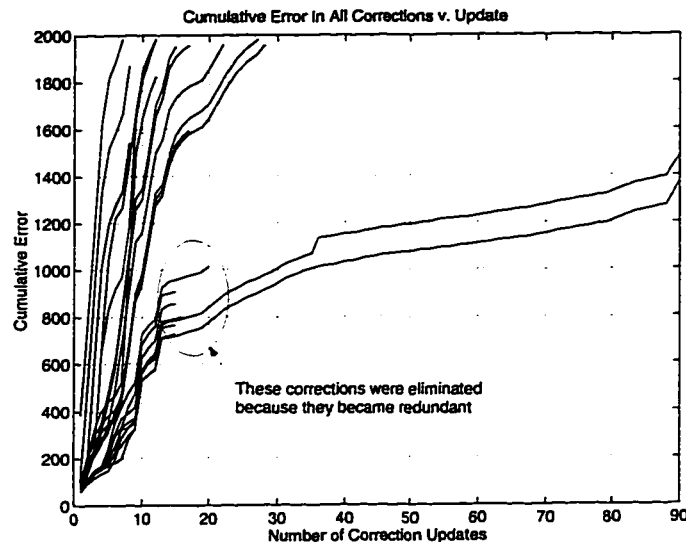


Figure 4.8: The cumulative cost for each correction as a function of time

software detects that two or more corrections are very nearly equal, it will eliminate all but one correction to reduce its computational burden. The reason some corrections can become nearly equal is because they are periodically updated, and the adaptive updating scheme (see Equations 4.11 and 4.12) does not restrict the corrections from becoming equal. As a result, corrections that are initially in the vicinity of one another frequently converge over time.

To demonstrate that the map-matching algorithm works correctly, it was applied to the data set whose GPS position fixes are shown in Figure 4.1. Figure 4.9 shows the Kalman filter's estimate of the vehicle's location (shown with "+" symbols) and the map-matching algorithm's estimate of the vehicle's location (shown with squares). The symbols are difficult to distinguish because they overlay each other. The geographical region shown in Figure 4.9 is the same as that shown in Figure 4.1. Comparing Figure 4.9 with Figure 4.1, it is clear that the Kalman filter/map-matching system smoothes the jumps and eliminates the bias in the GPS position fixes.

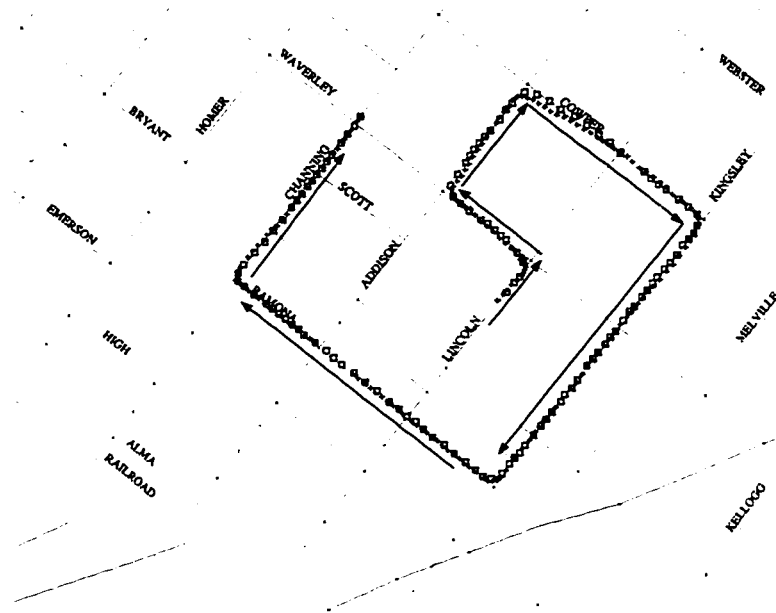


Figure 4.9: Map-matched and Kalman filter location estimates versus time

## 4.5 Summary

This chapter began with a discussion of various issues associated with map-matching. A discussion of various factors influencing successful map-matching was included. It was emphasized that the success of a particular map-matching algorithm depends on the particular strengths and weaknesses of the algorithm, and, for this reason, it is difficult to make general statements regarding the circumstances under which a map-matching algorithm will be successful. In addition, it is difficult to make general statements regarding the benefit of map-matching as a navigation aid. However, the influence of map-matching as a navigation aid is investigated in this research, and the manner in which this investigation is carried out was described. Finally, a map-matching algorithm developed by the author was described in detail.

# Chapter 5

## Analysis Details

### 5.1 Introduction

This chapter ties together the material of the previous chapters, bringing the theories established in Chapter 2 to bear on the Kalman filter presented in this chapter. This chapter begins with a discussion of several important assumptions that have been made to simplify analysis. Next, the Kalman filter equations and the equations of sensitivity analysis are given. Most of these equations were already presented in Chapter 3 as sensor error models. However, in this chapter, the equations are grouped more conveniently for readers interested in the details of the Kalman filter and the sensitivity analysis mechanization. Readers who are not interested in these details may skip Sections 5.3 and 5.4 without loss of continuity.

### 5.2 Simplifications

A general analysis of an automobile navigation system bears certain complications that must be simplified in order to obtain meaningful results efficiently. In this section, four important simplifications that are made in the analysis in this research are introduced and justified.

### 5.2.1 Dealing with Trajectory Dependencies

The first such issue arises in connection with the effect that the vehicle's trajectory has on the performance of a navigation system. Simply stated, the vehicle's motion—e.g. how often the vehicle turns, how fast the vehicle moves, how often the vehicle changes lanes, etc.—will have an impact on the positioning accuracy of its navigation system. This fact is readily seen when one realizes that certain sensor errors will change as the vehicle moves. For example, a rate gyro scale factor error contributes more to heading error when the vehicle turns, and the magnitude of the errors caused by the odometer depends on the vehicle's speed. Because the magnitude of some sensor errors vary with the vehicle's motion, analysis results are trajectory-dependent and therefore specific to the chosen trajectory. This makes it more difficult to perform a general analysis.

In addition to trajectory-dependent sensor errors, there are also map-matching errors that are trajectory-dependent. For example, as was discussed in Section 4.2.3, a map-matching algorithm produces an accurate estimate of a vehicle's position perpendicular to the road, but not parallel to it. A navigation system utilizing a successful map-matching algorithm will therefore have accurate knowledge of the vehicle's location *perpendicular* to the vehicle's heading. Immediately following a 90-degree turn, then, the location of the vehicle should be known accurately *in 2 dimensions*. Hence, the impact that map-matching has on positioning accuracy depends on the vehicle's trajectory—if the vehicle turns frequently, a successful map-matching algorithm will cause the positioning error to drop at each turn. On the other hand, if the vehicle travels on a straight road over a long distance, map-matching will restrain position error only in the cross-track direction, not the along-track direction.

These trajectory dependencies impede efficient analysis because they tie analysis results to a specific trajectory—any analysis results apply only for the assumed trajectory. One obvious analysis methodology, then, is to examine the navigation system's performance over a wide variety of representative trajectories. This type of analysis would require an examination of many trajectories that the vehicle would be likely to take. However, it is immediately clear that this method would be tedious and almost certainly incomplete.

A much more efficient alternative can be found if one recognizes that vehicles spend most of their time traveling on straight roads. Simple experience tells us that this is true and that turns are infrequent. For this reason, most of the analyses in this research assume that the vehicle is traveling on a straight road at a constant speed, thereby eliminating trajectory-dependent effects associated with turns and vehicle acceleration. Trajectory dependencies are not ignored but will, instead, be examined with individual simulation runs. While this solution is not a truly general solution, it reduces the analysis to its simplest form and will produce results that are general except during those occasions when the vehicle turns.

### 5.2.2 Simplifying Map-matching

A second issue arises in connection with the complexities of analyzing the effects of map-matching on navigation system performance. Because map-matching is a key element in many existing automobile navigation systems, its influence on navigation system performance should be included in this research. However, quantifying the influence that map-matching has on a navigation system's performance is complicated for two reasons. First, a map-matched position "measurement" is fundamentally different from sensor measurements—a sensor measurement consists of "the truth plus systematic and random errors," while a map-matched position could be incorrect altogether (containing no truth) if the map-matching algorithm fails. If a map-matching algorithm fails to produce a correct "measurement" of position, the Kalman filter's position estimate may diverge from the true vehicle position. Treating data from a map-matching algorithm as true position measurements is therefore risky because a divergent map-matching algorithm could ruin the navigation system's performance. Unfortunately, whether a map-matching algorithm will fail is nearly impossible to predict because it depends on the specific workings of the algorithm, the quality of the sensor data, and the geometry of the local road network. Second, analysis results would be valid only for the particular map-matching algorithm and trajectory at hand because the effects of map-matching on navigation system performance are

determined by details specific to the algorithm— e.g. the manner in which the map-matching data is weighed against the sensor data, how a map-matched position is derived, and the particular weaknesses of the map-matching algorithm.

For these reasons, we seek a simpler approach to examining the influence of map-matching on navigation system performance. In particular, we seek a solution that is independent of any particular map-matching algorithm, but, at the same time, reveals the salient influence of map-matching on navigation system performance. To this end, it has been assumed in this research that a *perfect* map-matching algorithm is in place—i.e. one that *always* produces a cross-track position fix on the correct road. Also, the navigation Kalman filter has been designed to utilize the map-matched position as a cross-track position measurement. This approach avoids having to implement a map-matching algorithm and having to deal with the attendant complications of map-matching divergence. It also avoids tying all simulation results to a single implementation of a map-matching algorithm and its particular pathologies. Finally, since this approach treats the map-matched position as a “measurement” of cross-track position, the entire system analysis is kept within a single framework—no exceptions need to be made in the analysis to treat map-matching input.

### 5.2.3 Choosing a Vehicle Trajectory

A third simplification that is closely related to the previous two arises in connection with the actual vehicle trajectory chosen for the simulations. As the first simplification has established, the vehicle is assumed to be traveling on a straight road in most simulations. Another way to state this simplification is, obviously, that the vehicle’s nominal heading is always constant. In most simulations, then, the vehicle’s heading was simulated to be constant and identically zero. (In simulations involving map-matching, however, the performance of the system is explored as the vehicle moves laterally on the road. In these simulations, then, the heading is not constant. The reader is referred to Section 4.3 for details.) Furthermore, the vehicle’s nominal path

is assumed to *coincide* with a local North-pointing axis. Therefore, in the simulations, the vehicle is always moving due North on the North-pointing axis of a local East-North-Up reference frame. There is no loss of generality in making these assumptions, but because the vehicle moves along one axis in a local reference frame, certain convenient simplifications arise. For example, under these assumptions, the vehicle's cross-track position and its x-position (i.e. East-position) are equivalent; similarly, its along-track position and its y-position (i.e. North-position) are equivalent. Therefore, the cross-track position "measurement" provided by the map-matching algorithm is a "measurement" of the vehicle's x-position. Hence, in the measurement equations in Section 5.4.2, the map-matching "measurement" is treated as a measurement of the vehicle's x-position.

### 5.2.4 Linearizing the Kalman Filter

One last simplification to be noted arises in connection with the assumptions of sensitivity analysis. One important prerequisite for a sensitivity analysis is that the reference system equations be *linear*. However, certain equations in the navigation Kalman filter are nonlinear. The filter is an extended Kalman filter, and therefore sensitivity analysis cannot be directly applied. However, this issue can be avoided by linearizing the nonlinear equations about a known nominal state trajectory. The filter presented in this chapter is therefore a *linearized* Kalman filter. This linearization is done only for the sake of analysis, since a filter for which the nominal state trajectory is known *a priori* could not be used in a real navigation system. As a result of the linearization, the filter estimates the *deviation* of the true states from the nominal state trajectory. The states in the filter are therefore *perturbation* states, not the "full" state. Results have shown that the performance of the linearized Kalman filter is virtually identical to that of the extended Kalman filter; therefore, it can be argued that the results of the sensitivity analysis (applied to the linearized filter) apply to the extended Kalman filter, as well. In this chapter, only the linearized filter's equations are given.

## 5.3 The Kalman Filter Equations

### 5.3.1 The Kalman Filter Model Equations

In this section, the model equation for the navigation Kalman filter will be presented. Before presenting the equations, however, there are several important introductory remarks that need to be made.

First, it is important to note that the states of the Kalman filter represent *perturbations* from a nominal state trajectory because the filter is linearized. To remind the reader that the states are perturbations, a  $\delta$  symbol has been included in the notation of certain perturbation quantities. Elements of the nominal state are denoted with an overbar (e.g.  $\bar{x}$ ), and the “full” state is given by

$$\mathbf{x} \equiv \bar{\mathbf{x}} + \delta\mathbf{x} \quad (5.1)$$

For individual elements of  $\delta\mathbf{x}$ , the  $\delta$  symbol is used only for those states whose nominal state value is non-zero. For example, the vehicle’s nominal speed is denoted  $\bar{V}$  and its perturbation speed is denoted  $\delta V$ , since  $\bar{V} \neq 0$ . Most of those states that represent sensor errors, however, have nominal values that are defined to be identically zero. Defining the nominal states in this way is perfectly legitimate as long as the assumptions of the linearization are not violated. The perturbation states are therefore identically equal to the “full” states, and the  $\delta$  notation is dropped for these states. Dropping the  $\delta$  notation avoids needless clutter in the equations and maintains notational consistency with the sensor error models presented in Chapter 3.

Second, the equations that follow are presented as continuous-time differential equations, but the equations of sensitivity analysis (in Chapter 2) were discrete-time equations. The equations are presented in their respective domains for convenience only.

Third, with regard to notation, the subscript “f” which appears in the following equations denote quantities associated with the filter, while the subscript “r” denotes quantities associated with the reference system. (This notation is consistent with that of Chapter 2.) However, certain elements of the nominal trajectory do *not* have

an “f” or “r” subscript. The subscript has been dropped to emphasize that many elements in the reference system’s and filter’s nominal trajectory vectors are identical (i.e.  $\bar{\Theta}_r \equiv \bar{\Theta}_f \equiv \bar{\Theta}$ ).

Finally, the model equations for the linearized filter can be divided into two groups: equations modeling the vehicle’s kinematic motion, and equations modeling sensor errors. The equations that model the kinematic motion of the vehicle remain the same no matter what set of sensors is available to the filter, while the equations that model the sensor errors will vary, depending on which sensors are available to the filter. (Recall that we are interested in examining the performance of a number of different navigation systems, each of which utilizes a different set of navigation sensors.) Therefore, some of the equations to be shown are not present in a given filter mechanization if the corresponding sensor is not part of the navigation system—e.g. the equation for the compass’ bias is not necessary if the navigation system does not include a compass.

Having made these preliminary remarks, the filter equations will now be presented. The perturbation quantities that describe the vehicle’s kinematic motion are its x-position ( $\delta p_{x,f}$ ), y-position ( $\delta p_{y,f}$ ), speed ( $\delta V_f$ ), acceleration ( $\delta a_f$ ), and heading ( $\delta \Theta_f$ ). The manner in which these quantities are related depends on certain elements of the nominal state vector. The equations that follow, for example, depend on the vehicle’s nominal speed ( $\bar{V}$ ) and nominal heading ( $\bar{\Theta}$ ). The quantities that model the sensor errors include the rate gyro’s bias ( $\omega_{b,f}$ ), the compass’ bias ( $\Theta_{b,f}$ ), the odometer scale factor bias ( $S_{b,f}$ ), and the error in the x- and y-components of the GPS position fixes ( $\lambda_{x,f}$ ,  $\lambda_{y,f}$ ). The filter’s state vector is defined as

$$\delta \mathbf{x}_f \equiv \left[ \delta p_{x,f} \quad \delta p_{y,f} \quad \delta V_f \quad \delta a_f \quad \delta \Theta_f \quad \omega_{b,f} \quad \Theta_{b,f} \quad S_{b,f} \quad \lambda_{x,f} \quad \xi_{x,f} \quad \lambda_{y,f} \quad \xi_{y,f} \right]^T \quad (5.2)$$

With the exception of the equations for the bias in the GPS position fixes, the model equations in the Kalman filter are given by

$$\dot{\delta p}_{x,f} = \sin(\bar{\Theta})\delta V_f + \bar{V} \cos(\bar{\Theta})\delta \Theta_f \quad (5.3)$$

$$\dot{\delta p}_{y,f} = \cos(\bar{\Theta})\delta V_f - \bar{V} \sin(\bar{\Theta})\delta \Theta_f \quad (5.4)$$

$$\dot{\delta V}_f = \delta a_f \quad (5.5)$$

$$\dot{\delta a}_f = -\frac{1}{\tau_{a,f}}\delta a_f + u_{a,f} \quad (5.6)$$

$$\dot{\delta \Theta}_f = \delta \omega_{meas} - \omega_{b,f} \quad (5.7)$$

$$\dot{\omega}_{b,f} = u_{\omega_{b,f}} \quad (5.8)$$

$$\dot{\Theta}_{b,f} = u_{\Theta_{b,f}} \quad (5.9)$$

$$\dot{S}_{b,f} = -\frac{1}{\tau_{S_{b,f}}}S_{b,f} + u_{S_{b,f}} \quad (5.10)$$

where all quantities denoted with a  $u$  represent zero-mean Gaussian white noise. The quantity  $\delta \omega_{meas}$  is treated as an exogenous input to these model equations. This quantity can be thought of as a measurement of the perturbation in the vehicle's heading rate and is defined as

$$\delta \omega_{meas} \equiv \omega_{meas} - \bar{\omega} - \bar{\omega}_b \quad (5.11)$$

where  $\omega_{meas}$  is the measured heading rate (i.e. the raw output of the rate gyro),  $\bar{\omega}$  is the nominal heading rate of the vehicle, and  $\bar{\omega}_b$  is the nominal rate gyro bias. Injecting a raw measurement into a model equation as an exogenous input is convenient when the measured quantity is a derivative of one of the filter's states. Formulating the filter in this way allows it to track rapid changes in the state. In this case, injecting the raw rate gyro output into the model equation allows the filter to track rapid changes in the vehicle's heading. If a more conventional approach were taken, in which the rate gyro measurement were part of the measurement vector ( $z$ ), then it would be necessary to include a mathematical model of the vehicle's heading in the model equations. This approach was attempted and was found to result in poorer filter performance when the vehicle turned and, in general, produced poor estimates of the vehicle's heading.

As was mentioned in the introductory remarks, the states that represent sensors errors— $\omega_{b,f}$ ,  $\Theta_b$ , and  $S_{b,f}$ —are, technically, perturbation states. However, the nominal values for these states are *defined* to be identically zero. The perturbation states

are therefore identically equal to the “full” states. The models for these perturbation states are therefore identical to the models for  $\omega_b$  (Equation 3.9),  $\Theta_b$  (Equation 3.20), and  $S_b$  (Equation 3.32) in Chapter 3. Also, since  $\bar{\omega}_b \equiv 0$ , Equation 5.11 can be rewritten as

$$\delta\omega_{meas} \equiv \omega_{meas} - \bar{\omega} \quad (5.12)$$

The equations for the bias in the GPS position fixes were not included in Equations 5.3 through 5.10 because the models for the x- and y-components of the bias vary depending on whether SA is on, SA is off, or differential corrections are available. In all cases, the models for the position bias add four more equations to those above. Also, as with the other sensor errors, the nominal values for the biases and their derivatives are *defined* to be identically zero, and the  $\delta$  notation is dropped. If SA is on, the x-component of the error in the position fixes ( $\lambda_{x,f}$ ) is given in Equations 3.34 and 3.35:

$$\dot{\lambda}_{x,f} = \xi_{x,f} \quad (5.13)$$

$$\dot{\xi}_{x,f} = -\beta_{x,f}^2 \lambda_{x,f} - 2\beta_{x,f} \xi_{x,f} + u_{x,f} \quad (5.14)$$

(The model for the y-component of the bias is identical and is therefore not given.) This is a second-order Gauss-Markov process whose behavior is controlled by the constant  $\beta_{x,f}$  and the Gaussian white noise  $u_{x,f}$ . If SA is off, the error equations have the same form as these equations, but the numerical values for the parameters that govern the equations (i.e. the initial conditions for  $\lambda_{x,f}$  and  $\xi_{x,f}$ ,  $\beta_{x,f}$ , and the RMS value of  $u_{x,f}$ ) differ.

If DGPS position fixes are available, then the meaning of  $\xi_{x,f}$  and  $\xi_{y,f}$  change. Each component of the position error is modeled as the sum of two biases (i.e.  $\lambda_{x,f} + \xi_{x,f}$ ). Each bias is modeled as a first-order Gauss-Markov process (with time-constants denoted  $\tau_{\lambda,f}$  and  $\tau_{\xi,f}$ ) that is driven by white noise (denoted  $u_{\lambda,f}$  and  $u_{\xi,f}$ ). The equations for the biases were originally given in Equations 3.41 and 3.43:

$$\dot{\lambda}_{x,f} = -\frac{1}{\tau_{\lambda,f}} \lambda_{x,f} + u_{\lambda,f} \quad (5.15)$$

and

$$\dot{\xi}_{x,f} = -\frac{1}{\tau_{\xi,f}}\xi_{x,f} + u_{\xi,f} \quad (5.16)$$

The model for the y-component of the bias is identical and is therefore not given.

The reason for presenting Equations 5.3 through 5.16 is, of course, to benefit the reader interested in the specifics of the sensitivity analysis mechanization. Several matrices that appear in the formulation of the equations of sensitivity analysis can be identified from Equations 5.3 through 5.16. Specifically, the matrices that can be extracted from these equations are  $\Phi_{f,k}$ ,  $B_k$ ,  $Q_{f,k}$ ,  $\Xi_k$ ,  $\Gamma_k$  and  $\Psi_k$ ; the meaning of each of these matrices can be found by examining Equations 2.49 and 2.50. These matrices are particularly important because they are required for the time-update equation in the sensitivity analysis (i.e. Equation 2.72). The continuous-time form of the first two matrices in this list (i.e.  $\Phi_{f,k}$  and  $B_k$ ) can be formed by inspection Equations 5.3 through 5.16 with reference to Equation 2.49. The matrix  $Q_{f,k}$  is required by the Kalman filter algorithm (see Equation 2.10) and is the discrete-time form of the spectral density matrix for the additive white noise in the model equations.

Some clarifying remarks are needed in connection with the last three matrices (i.e.  $\Xi_k$ ,  $\Gamma_k$  and  $\Psi_k$ ). These matrices define the input  $\mathbf{u}_k$  in accordance with Equation 2.50, which is repeated here:

$$\mathbf{u}_k = \Xi_k \bar{\mathbf{x}}_{r,k} + \Gamma_k \delta \mathbf{x}_{r,k} + \psi_k \quad (5.17)$$

The quantity  $\mathbf{u}_k$  appears in the model equations as an exogenous input to Equation 5.7. Because there is only 1 input to the filter's model equations,  $\mathbf{u}_k$  is actually a scalar quantity that is given by

$$u_k = \delta \omega_{meas} \equiv \omega_{meas} - \bar{\omega} \quad (5.18)$$

Since  $u_k$  is a scalar, the matrices  $\Xi_k$  and  $\Gamma_k$  have dimensions  $1 \times n$  (where  $n$  is the number of states in the reference system), and  $\Psi_k$ , the covariance of  $\psi_k$ , is also a scalar.

We can develop Equation 5.18 further by making use of Equation 3.5. Equation

3.5 defines the output of the gyro in terms of the vehicle's true heading rate, the true rate gyro bias, the true value of the noise in the gyro's output, and the true rate gyro scale factor. Substituting from Equation 3.5 into Equation 5.18 for  $\omega_{meas}$ , we arrive at

$$u_k = \left(1 + \frac{\delta K}{K_T}\right) (\omega_r + \omega_{b,r} + v_{\omega,r}) - \bar{\omega} - \bar{\omega}_b \quad (5.19)$$

where  $\omega_r$  is the true vehicle heading rate,  $\omega_{b,r}$  is the true rate gyro bias, and  $v_{\omega,r}$  is uncorrelated Gaussian noise in the output of the gyro. The quantities  $\omega_r$  and  $\omega_{b,r}$  are elements of the reference system state vector, and  $v_{\omega,r}$  is also a reference system quantity. Noting that

$$\delta\omega_{r,k} \equiv \omega_r - \bar{\omega} \quad (5.20)$$

$$\delta\omega_{b,r,k} \equiv \omega_{b,r} - \bar{\omega}_b = \omega_{b,r} \quad (5.21)$$

Equation 5.19 can be reformulated to

$$u_k = \frac{\delta K}{K_T} \bar{\omega} + \left(1 + \frac{\delta K}{K_T}\right) (\delta\omega_{r,k} + \delta\omega_{b,r,k}) + \left(1 + \frac{\delta K}{K_T}\right) v_{\omega,r} \quad (5.22)$$

Equation 5.22 has the same form as the general expression for  $u_k$  given in Equation 5.17. Hence, the exact values for the elements of the matrices  $\Xi_k$ ,  $\Gamma_k$ , and  $\Psi_k$  can be identified by inspection of Equation 5.22 together with the definition of the reference system state vector (to be given in Section 5.4.1).

With the equations given thus far, the matrices  $\Phi_{f,k}$ ,  $Q_{f,k}$ ,  $B_k$ ,  $\Xi_k$ ,  $\Gamma_k$  and  $\Psi_k$  that appear in the equations of sensitivity analysis can be formed. The remaining matrices that appear in the formulation of the equations of sensitivity analysis (in Chapter 2) can be found by inspecting equations that appear in the following sections.

### 5.3.2 The Kalman Filter Measurement Equations

In this section, the measurement equations for the Kalman filter are presented. The purpose of this section is to give the reader enough information to form the filter's observation matrix ( $H_f$ ) and the filter's measurement noise covariance matrix ( $R_f$ ).

The equations are slightly complicated by the fact that the filter is a linearized

Kalman filter. As was described in Chapter 2, the measurement vector in a linearized filter is actually a linear combination of perturbation states. The quantities measured from the navigation sensors are modified according to Equation 2.55, which is repeated here:

$$\delta \mathbf{z}_{r,k} \equiv \mathbf{z}_{r,k} - H_{f,k} \bar{\mathbf{x}}_{f,k} \quad (5.23)$$

where  $\mathbf{z}_{r,k}$  is a vector of quantities measured from the sensors,  $H_{f,k}$  is the filter's observation matrix, and  $\bar{\mathbf{x}}_{f,k}$  is the value of the nominal state trajectory. Performing this subtraction puts the measurement vector in terms of the filter's perturbation states,  $\delta \mathbf{x}_f$ . (For a more detailed derivation of this equation, the reader is referred to Section 2.4.)

To arrive at an expression with the same form as Equation 5.23, we shall begin with the “full” measurement vector,  $\mathbf{z}_{r,k}$ . For the filter at hand, the elements of the “full” measurement vector are the vehicle's x-position ( $p_{x,meas}$ ) and y-position ( $p_{y,meas}$ ), the distance traveled between measurements ( $D_{meas}$ ), the vehicle's heading ( $\Theta_{meas}$ ), and the cross-track position as determined by the map-matching algorithm ( $p_{m,meas}$ ). The position measurements are taken from a GPS receiver, the distance traveled is measured by the odometer, and the heading measurement is obtained from a compass or an attitude-capable GPS receiver. In matrix form, the “full” measurement vector is given by

$$\mathbf{z}_{r,k} = \begin{bmatrix} p_{x,meas} \\ p_{y,meas} \\ D_{meas} \\ \Theta_{meas} \\ p_{m,meas} \end{bmatrix} \quad (5.24)$$

The nominal distance measured by the odometer is defined as

$$D_{meas} \equiv S_o N \quad (5.25)$$

where  $S_o$  is the nominal odometer scale factor (in meters/pulse) and  $N$  is the number

of pulses generated by the odometer between measurements.

The filter design assumes that the measured quantities are corrupted with white noise and represent a linear combination of the filter's "full" states. If stand-alone GPS position fixes are being used by the filter, then the filter's *model* for the relationship between its measurements and its "full" states is given by

$$p_{x,meas} = p_{x,f} + \lambda_{x,f} + v_{p_x,f} \quad (5.26)$$

$$p_{y,meas} = p_{y,f} + \lambda_{y,f} + v_{p_y,f} \quad (5.27)$$

$$D_{meas} = V_f T + \frac{1}{2} a_f T^2 - S_{b,f} N - K_{S_{b,f}} V_f N + v_{odom,f} \quad (5.28)$$

$$\Theta_{meas} = \Theta_f + \Theta_{b,f} + v_{\Theta,f} \quad (5.29)$$

$$p_{m,meas} = p_{x,f} + v_{m,f} \quad (5.30)$$

where  $T$  is the time between measurements,  $K_{S_{b,f}}$  is a constant, and  $\Theta_{b,f}$  is the compass' bias. Also, every term denoted with a  $v$  represents white noise. The subscript "f" on these terms indicates that the RMS value of each noise term is *assumed* by the filter and is not necessarily equal to the *actual* RMS value; the actual noise terms will be distinguished with the subscript "r" in subsequent equations. The remaining symbols represent filter states that are defined in Section 5.3.1. If DGPS position fixes are available, then Equations 5.26 and 5.27 are replaced with

$$p_{x,meas} = p_{x,f} + \lambda_{x,f} + \xi_{x,f} + v_{p_x,f} \quad (5.31)$$

$$p_{y,meas} = p_{y,f} + \lambda_{y,f} + \xi_{y,f} + v_{p_y,f} \quad (5.32)$$

The perturbation measurement for the filter is computed by subtracting  $H_f \bar{\mathbf{x}}_{f,k}$  from  $\mathbf{z}_{r,k}$  in accordance with Equation 5.23:

$$\delta \mathbf{z}_{r,k} = \mathbf{z}_{r,k} - H_{f,k} \bar{\mathbf{x}}_{f,k} \quad (5.33)$$

$$\delta \mathbf{z}_{r,k} = \begin{bmatrix} p_{x,meas} \\ p_{y,meas} \\ S_o N \\ \Theta_{meas} \\ p_{m,meas} \end{bmatrix} - \begin{bmatrix} \bar{p}_x + \bar{\lambda}_x \\ \bar{p}_y + \bar{\lambda}_y \\ \bar{V}T + \frac{1}{2}\bar{a}T^2 - \bar{S}_b N - K_{S_b,f}\bar{V}N \\ \bar{\Theta} + \bar{\Theta}_{b,f} \\ \bar{p}_x \end{bmatrix} \quad (5.34)$$

This equation has the same form as Equation 5.23, and the vector  $\delta \mathbf{z}_{r,k}$  is the perturbation measurement.

The filter implementation *assumes* that the perturbation measurement vector has this structure:

$$\delta \mathbf{z}_{f,k} = H_{f,k} \delta \mathbf{x}_{f,k} + \mathbf{v}_{f,k} \quad (5.35)$$

and the Kalman gain is computed based on this assumption. In reality, however, the measurement vector  $\mathbf{z}_{r,k}$  contains errors that are not modeled in the filter. The elements of the matrix  $H_{f,k}$  can be identified by comparison of Equation 5.33 with Equation 5.34. The matrix  $R_{f,k}$  is a diagonal matrix whose diagonal elements are the mean-square values of the noise terms appearing in Equations 5.26 through 5.30.

## 5.4 The Equations of Sensitivity Analysis

### 5.4.1 The Reference System Model Equations

Like the filter's model equations, the model equations for the reference system can be divided into two groups: those states which model the vehicle's kinematics, and those which model sensor errors. Naturally, some of the model equations in the reference system differ from those in the Kalman filter. The notation used in the following equations is nearly identical to that used in the filter's model equations, with the principle difference being the presence of the subscript "r" to distinguish variables as

reference system quantities. The reference system state vector is given by

$$\delta \mathbf{x}_r \equiv \begin{bmatrix} \delta p_{x,r} \\ \delta p_{y,r} \\ \delta V_r \\ \delta a_r \\ \delta \Theta_r \\ \delta \omega_r \\ S_{b,r} \\ \varphi_{b,r} \\ \varpi_{b,r} \\ \alpha_{b,r} \\ \theta_{o,r} \\ \phi_r \\ \vartheta_r \\ \eta_r \\ \delta p_{m,r} \\ \lambda_{x,r} \\ \xi_{x,r} \\ \lambda_{y,r} \\ \xi_{y,r} \end{bmatrix} \quad (5.36)$$

The equations that model the vehicle's kinematics are given by

$$\dot{\delta p}_{x,r} = \sin(\bar{\Theta})\delta V_r + \bar{V} \cos(\bar{\Theta})\delta \Theta_r \quad (5.37)$$

$$\dot{\delta p}_{y,r} = \cos(\bar{\Theta})\delta V_r - \bar{V} \sin(\bar{\Theta})\delta \Theta_r \quad (5.38)$$

$$\dot{\delta V}_r = \delta a_r \quad (5.39)$$

$$\dot{\delta a}_r = 0 \quad (5.40)$$

$$\dot{\delta \Theta} = \delta \omega_r \quad (5.41)$$

$$\dot{\delta \omega}_r = 0 \quad (5.42)$$

The quantities  $\dot{\hat{\alpha}}_r$  and  $\dot{\hat{\omega}}_r$  are zero because the vehicle's actual trajectory is defined to coincide exactly with its nominal trajectory; hence, the reference system's perturbation states associated with the vehicle's kinematic motion are identically zero.

The odometer's scale factor bias model (Equation 3.32) is

$$\dot{S}_{b,r} = -\frac{1}{\tau_{S_{b,r}}} S_{b,r} + u_{S_{b,r}} \quad (5.43)$$

If a rate gyro is available, its bias is modeled as the sum of a constant ( $\varphi_{b,r}$ ) plus a time-varying quantity described by a second-order Gauss-Markov process ( $\varpi_{b,r}$ ), as given in Equations 3.11 through 3.13:

$$\omega_{b,r} \equiv \varphi_{b,r} + \varpi_{b,r} \quad (5.44)$$

where

$$\dot{\varphi}_{b,r} = 0 \quad (5.45)$$

$$\dot{\varpi}_{b,r} = \alpha_{b,r} \quad (5.46)$$

$$\dot{\alpha}_{b,r} = -\beta_{\alpha_{b,r}}^2 \varpi_{b,r} - 2\beta_{\alpha_{b,r}} \alpha_{b,r} + u_{\alpha_{b,r}} \quad (5.47)$$

If a fluxgate compass is available, the bias in the compass' reading is given by Equation 3.16 as the sum of a constant ( $\theta_{o,r}$ ), a heading-dependent term ( $\phi_r$ ), and a time-varying term ( $\vartheta_r$ ):

$$\Theta_{b,r} = \theta_{o,r} + \phi_r + \vartheta_r \quad (5.48)$$

where

$$\dot{\theta}_{o,r} = 0 \quad (5.49)$$

$$\dot{\vartheta}_r = \eta_r \quad (5.50)$$

$$\dot{\eta}_r = -\beta_{\eta,r}^2 \vartheta_r - 2\beta_{\eta,r} \eta_r + u_{\eta,r} \quad (5.51)$$

The heading-dependent bias ( $\phi_r$ ) is a nonlinear function of heading ( $\Theta$ ) given by

$$\phi_r = A \sin(\Theta) + B \cos(\Theta) + C \sin(2\Theta) + D \cos(2\Theta) \quad (5.52)$$

In order for this to be used as a model equation in the sensitivity analysis, it must be differentiated with respect to time and linearized about the nominal state trajectory. The general expression for this quantity is lengthy and complicated; however, when the vehicle is traveling in a straight line (as is the case in most simulations), the nominal heading rate is zero, and the expression is greatly simplified:

$$\dot{\phi}_r = A \cos(\bar{\Theta})\delta\omega_r - B \sin(\bar{\Theta})\delta\omega_r + 2C \sin(2\bar{\Theta})\delta\omega_r - 2D \cos(2\bar{\Theta})\delta\omega_r \quad (5.53)$$

The nominal value of  $\phi_r$  (denoted  $\bar{\phi}_r$ ) is a function of  $\bar{\Theta}$  and  $\bar{\omega}$ .

If GPS-based heading measurements are available instead of the fluxgate compass, the bias in the measured heading is modeled as a constant. The model for  $\Theta_b$  therefore simplifies to

$$\Theta_{b,r} = \theta_{o,r} \quad (5.54)$$

If GPS fixes are available, then four additional states are added to the model equations. The equations for these states are the same as those used in the filter's model equations. If SA is on, then the x-component of the error in the position fixes is given by Equations 3.34 and 3.35:

$$\dot{\lambda}_{x,r} = \xi_{x,r} \quad (5.55)$$

$$\dot{\xi}_{x,r} = -\beta_{x,r}^2 \lambda_{x,r} - 2\beta_{x,r} \xi_{x,r} + u_{x,r} \quad (5.56)$$

(The model for the y-component of the bias is identical and is therefore not given.) If SA is off, the error equations have the same form as these equations, but the numerical values for the parameters that govern the equations (i.e. the initial conditions for  $\lambda_{x,r}$  and  $\xi_{x,r}$ ,  $\beta_{x,r}$ , and the RMS value of  $u_{x,r}$ ) differ. If DGPS position fixes are available,

then the models are given by

$$\dot{\lambda}_{x,r} = -\frac{1}{\tau_{\lambda,r}}\lambda_{x,r} + u_{\lambda,r} \quad (5.57)$$

and

$$\dot{\xi}_{x,r} = -\frac{1}{\tau_{\xi,r}}\xi_{x,r} + u_{\xi,r} \quad (5.58)$$

and the position error is modeled as the sum of these two biases (i.e.  $\lambda_{x,r} + \xi_{x,r}$ ).

Finally, the reference system includes a state to model the bias in the “measurement” of cross-track position from the map-matching algorithm. This “measurement” is biased because the actual location of the vehicle generally does not coincide with the centerline of the road that is stored in the map database. At the beginning of this chapter, it was pointed out that a simplifying assumption was made in which the centerline of the road was assumed to coincide with a local north-pointing axis. Under this simplifying assumption, the “measurement” of the vehicle’s cross-track position from the map database is always zero. Therefore, the bias in the map “measurement” is the negative of the vehicle’s true x-location:

$$\dot{p}_{m,r} = -\sin(\bar{\Theta})\delta V_r - \bar{V} \cos(\bar{\Theta}) \quad (5.59)$$

The complete set of reference system model equations consists of those preceding equations that model the time histories of the elements of  $\delta \mathbf{x}_r$  (defined in Equation 5.36). Certain matrices that appear in the time-update equation in the sensitivity analysis (i.e. Equation 2.72) can be found by inspection of these equations. The matrices of particular interest are  $\Phi_{r,k}$ , the state transition matrix, and  $Q_{r,k}$ , the covariance matrix for the white noise terms driving the differential equations. These matrices can be found by putting the reference system model equations into matrix form and discretizing the resulting system of continuous-time differential equations.

### 5.4.2 The Reference System Measurements Equations

Recall that the “full” measurement vector,  $\mathbf{z}_r$ , is

$$\mathbf{z}_{r,k} = \begin{bmatrix} p_{x,meas} \\ p_{y,meas} \\ D_{meas} \\ \Theta_{meas} \\ p_{m,meas} \end{bmatrix} \quad (5.60)$$

This measurement vector can be rewritten as a linear combination of the nominal trajectory and reference system states plus additive noise:

$$\mathbf{z}_{r,k} = H_{r,k}(\bar{\mathbf{x}}_{r,k} + \delta\mathbf{x}_{r,k}) + v_{r,k} \quad (5.61)$$

If stand-alone GPS fixes are used, then the measurement equations are given by

$$\mathbf{z}_{r,k} = \begin{bmatrix} p_{x,r} + \lambda_{x,r} + v_{p_x,r} \\ p_{y,r} + \lambda_{y,r} + v_{p_y,r} \\ \frac{1}{2}a_r T^2 + V_r T - S_{b,r}N - K_{S_{b,r}}V_r N + v_{odom,r} \\ \Theta_r + \theta_{o,r} + \phi_r + \vartheta_r + v_{\Theta,r} \\ p_{x,r} + p_{m,r} \end{bmatrix} \quad (5.62)$$

$$= \begin{bmatrix} \bar{p}_x + \bar{\lambda}_x \\ \bar{p}_y + \bar{\lambda}_y \\ \frac{1}{2}\bar{a}T^2 + \bar{V}T - \bar{S}_bN - K_{S_{b,r}}\bar{V}N \\ \bar{\Theta} + \bar{\theta}_{o,r} + \bar{\phi}_r + \bar{\vartheta}_r \\ \bar{p}_x + \bar{p}_{m,r} \end{bmatrix} + \begin{bmatrix} \delta p_{x,r} + \lambda_{x,r} \\ \delta p_{y,r} + \lambda_{y,r} \\ \frac{1}{2}\delta a_r T^2 + \delta V_r T - S_{b,r}N - K_{S_{b,r}}\delta V_r N \\ \delta\Theta_r + \theta_{o,r} + \delta\phi_r + \vartheta_r \\ \delta p_{x,r} + \delta p_{m,r} \end{bmatrix} + \begin{bmatrix} v_{p_x,r} \\ v_{p_y,r} \\ v_{odom,r} \\ v_{\Theta,r} \\ v_{m,r} \end{bmatrix} \quad (5.63)$$

Note that, for clarity, the time index  $k$  has been dropped from the individual elements of the matrices above. Also, recall that the nominal values for most of the sensor errors (i.e.  $\bar{\lambda}_x$ ,  $\bar{\lambda}_y$ ,  $\bar{\xi}_x$ ,  $\bar{\xi}_y$ ,  $\bar{S}_b$ ,  $\bar{\theta}_o$ , and  $\bar{v}$ ) are *defined* to be identically zero for both the filter and reference system's nominal trajectories. The reader will note, however, that these symbols are included in the preceding equation for completeness. Finally, if DGPS position fixes are used instead of stand-alone GPS position fixes, then the first two equations in Equation 5.63 should be replaced with

$$p_{x,meas} = p_{x,r} + \lambda_{x,r} + \xi_{x,r} + v_{p_x,r} \quad (5.64)$$

$$p_{y,meas} = p_{y,r} + \lambda_{y,r} + \xi_{y,r} + v_{p_y,r} \quad (5.65)$$

Equation 5.63 has the form

$$\mathbf{z}_{r,k} = H_{r,k}\bar{\mathbf{x}}_{r,k} + H_{r,k}\mathbf{w}_{r,k} + \mathbf{v}_{r,k} \quad (5.66)$$

from which the matrices  $H_{r,k}$  (the reference system observation matrix) and the diagonal matrix  $R_{r,k}$  (the covariance matrix of  $\mathbf{v}_{r,k}$ ) can be identified.

## 5.5 Summary

This chapter opened with a discussion of several assumptions which were justified and which simplify the analysis considerably. After this, the equations for the Kalman filter and the sensitivity analysis were presented in detail. Many of these equations were presented as sensor error models in Chapter 3. In this chapter, however, they were grouped so that the interested reader could more easily identify the matrices that appear in the equations of sensitivity analysis. Having justified the assumptions made in the analysis and having presented the Kalman filter and sensitivity equations, the results of the sensitivity analysis be presented.

# Chapter 6

## Results

### 6.1 Introduction

The main goal of this research is to gain an understanding of low-cost navigation systems by quantifying the contribution that individual error sources make to a navigation system's performance. The most natural measure of a navigation system's "performance" is its positioning accuracy. However, a great deal can be learned about a navigation system's behavior by examining other quantities, as well. Therefore, the results and discussion which follow are not limited to an examination of positioning error only. Instead, several performance parameters are examined and various influences on each of these parameters are discussed.

The results which will be shown were obtained by applying sensitivity analysis to several navigation systems, each of which utilized a different set of navigation sensors. Most results pertain to a system utilizing GPS position fixes, a rate gyro, and an odometer. This set of sensors will be referred to as the "baseline" system configuration, and the performance of this system will generally serve as the benchmark to which other systems will be compared. The discussions which follow are organized on a topical basis and include the key results that best convey to the reader an understanding of the error mechanisms in the navigation Kalman filter.

GPS Positioning Type	SA On	SA Off	DGPS
Steady-state RMS Cross-track Position Error (m)	19.3	7.7	1.1
RMS Cross-track Bias Error in GPS Fixes (m)	20.6	8.0	1.07

Table 6.1: Comparison of cross-track position error and GPS position error

## 6.2 The Roles of Various Sensors While GPS Fixes Are Available

We shall first examine the ability of the Kalman filter to estimate the vehicle's position while GPS position fixes are available. To do so, results will be shown for 3 navigation systems: each of the 3 systems utilized the baseline sensor set (i.e. GPS fixes, a rate gyro, and an odometer) but utilized a *different* type of GPS position fix (GPS with SA on, GPS with SA off, and DGPS). For these 3 navigation systems, the cross-track and along-track components of the position error will be examined separately, beginning with the cross-track error.

Table 6.1 shows the RMS error in the GPS position fixes and the RMS error in the cross-track position estimate after the filter reached steady-state. The first row in the table shows the RMS value of the error in the cross-track position estimate after the Kalman filter reached steady-state. The data in the 3 columns in this row show results for each of the 3 navigation systems at hand. The second row in the table shows the RMS value of the bias in the GPS position fixes that were available to the navigation system. The data in this (second) row represent the accuracy of the cross-track position measurements that were available to the filter. The data in the second row serves as a benchmark to which the data in the first row can be compared to determine how much the filter was able to improve upon its position measurements.

As the data show, the filter is not able to reduce the positioning error significantly below that of the RMS value of the bias error in the GPS position fixes. In fact, when DGPS position fixes are utilized, the RMS error in the cross-track position estimate is greater than the RMS bias in the position fixes. This occurs because the RMS value of the uncorrelated noise in the DGPS position fixes is large (1.4 meters) in

GPS Positioning Type	SA On	SA Off	DGPS
GPS Bias Drift	95.3%	97.4%	92.7%
GPS Measurement Noise	0.9%	1.8%	6.4%
Rate Gyro Bias Drift	3.5%	0.5%	0.1%
Rate Gyro Measurement Noise	0.3%	0.3%	0.8%
Odometer Errors	0.0%	0.0%	0.0%

Table 6.2: Relative contributions to mean-square error in cross-track position estimate

comparison to the RMS bias error and, therefore, figures prominently into the cross-track position error. The data in Table 6.1 suggest that, while GPS position fixes are available, the accuracy of the positioning system is dominated by the accuracy of the GPS position fixes.

Sensitivity analysis results support this conclusion and provide further insight into the filter's error mechanisms. Table 6.2 shows the percent contribution that various error sources make to the steady-state mean-square error in the estimate of the vehicle's cross-track position. The error sources listed in the table have been categorized according to their sensor of origin and their time-correlation. For example, "GPS Bias Drift" refers to the time-correlated errors in the GPS position fixes. The percentages listed in this row of the table represent the fractional contribution that random bias drift in the GPS position fixes makes to the total mean-square cross-track position error. "GPS Measurement Noise" refers to errors in the GPS position fixes that are uncorrelated in time. "Rate Gyro Bias Drift" refers to time-correlated bias errors in the rate gyro's output, and "Rate Gyro Measurement Noise" refers to errors in the rate gyro's output that are uncorrelated in time. Finally, the influence of uncorrelated and correlated errors in the odometer readings are grouped together in the row labeled "Odometer Errors." As the data show, the bias in the GPS position fix contributes more than 90% of the total error in the cross-track position estimate, demonstrating that the positioning error is dominated by the accuracy of the GPS position fix.

The data in this table reveal that the error sources consisting of uncorrelated noise—the GPS measurement noise and the rate gyro measurement noise—become increasingly significant as the accuracy of the position fixes improves. The fractional

GPS Positioning Type	SA On	SA Off	DGPS
Steady-state RMS Position Error (m)	13.9	6.5	1.1
RMS Bias Error in GPS Position Fix (m)	20.6	8.0	1.07

Table 6.3: Comparison of along-track position error and GPS positioning error

contribution of the uncorrelated error sources increases because, as the GPS position accuracy improves, the filter is better able to mitigate the effects of the *correlated* error sources than the *uncorrelated* error sources. In other words, the filter estimates the correlated errors with increasing accuracy as the GPS measurement accuracy improves, but it is less able to reduce the effects of the uncorrelated noise sources. This occurs because the information in an uncorrelated signal contains no information that can be utilized to predict its future value—the value at one timestep is unrelated to the value at every other timestep. In contrast, the value of a correlated signal at any point in time contains information about the signal at a later point in time. The filter is therefore better able to take advantage of the improved accuracy in the GPS position fixes when estimating correlated errors.

Results for the along-track error are slightly better. Table 6.3 shows results similar to those of Table 6.1, and Table 6.4 shows results similar to those of Table 6.2. Table 6.4, however, includes separate rows for the contributions of the correlated and uncorrelated errors in the measurement from the odometer. As Table 6.3 shows, the steady-state along-track error is smaller than the steady-state cross-track error when SA is on or off. When DGPS position fixes are utilized, the steady-state along-track error is the same as the steady-state cross-track error. Table 6.4 shows the percent contribution that each error source makes to the mean-square along-track position error. As the data in this table show, the position error is dominated by the bias in the GPS position fix.

The data presented in Tables 6.4 exhibit trends that are similar to those in Table 6.2—uncorrelated errors (from the odometer and position measurements) contribute an increasing fraction of the total error as the GPS position measurements become more accurate. Not all of the correlated error sources, however, exhibit the same trends as those in Table 6.2. Specifically, the fractional contribution of the GPS

GPS Positioning Type	SA On	SA Off	DGPS
GPS Bias Drift	85.8%	90.3%	91.9%
GPS Measurement Noise	0.4%	0.8%	4.9%
Rate Gyro Bias Drift	0.0%	0.0%	0.0%
Rate Gyro Measurement Noise	0.0%	0.0%	0.0%
Odometer Bias Drift	13.3%	8.1%	0.4%
Odometer Quantization Noise	0.5%	0.8%	2.8%

Table 6.4: Relative contributions to mean-square error in along-track position estimate

bias drift increases as the position measurements become more accurate. This trend appears because the absolute contribution of the odometer bias error drops rapidly. The filter effectively takes advantage of the improved positioning accuracy to estimate the odometer bias, and, as a result, the error in the odometer bias decreases rapidly. Because the odometer bias is estimated more accurately, the odometer bias drift contributes less to the error in the along-track position estimate. The absolute contribution of the GPS bias drift to the along-track position error also decreases as the position measurements improve, but it does not decrease as rapidly as the contribution of the odometer's bias drift. As a result, the *fractional* contribution of the GPS bias drift increases.

The data in Tables 6.2 and 6.4 also show that the rate gyro errors contribute nothing to along-track position error and that the odometer errors contribute nothing to cross-track position error. (Other results have shown that a compass, like a rate gyro, contributes only to cross-track position error.) Hence, heading sensors are tied to cross-track position and the odometer is tied to along-track position. This simple observation is, perhaps, not surprising when one recognizes that a heading measurement is kinematically related to changes in cross-track position and the odometer measurement is kinematically related to changes in along-track position. As a consequence of these kinematic relationships, each dead-reckoning sensor error is “orthogonal” to a component of position error. This result reveals a fundamental error mechanism of the filter and leads to important conclusions. For example, changes in sensor quality will have little or no impact on “orthogonal” components of

the positioning error—e.g. using a high-quality gyro will not improve the along-track positioning error. Furthermore, improvements in one component of position accuracy will improve calibration of only the associated dead-reckoning sensors—e.g. the improvement in cross-track position accuracy afforded by map-matching will improve gyro calibration, but not odometer calibration.

## 6.3 The Influence of GPS Positioning Type

The results shown thus far demonstrate that the dead-reckoning sensors do not significantly reduce the position error while GPS position fixes are available. However, if GPS position fixes become *unavailable*, the subsequent performance of the navigation system is determined by two factors: the sensors' drift characteristics and the accuracy with which dead-reckoning sensor errors were calibrated *before* the GPS fixes became unavailable. In this section, these two issues will be examined. First, results will be presented which quantify the influence that each type of GPS positioning (i.e. SA on, SA off, and DGPS) has on dead-reckoning sensor calibration. Next, the performance of each navigation system without GPS will be examined, and the major contributors to positioning error will be identified.

### 6.3.1 The Influence of GPS on Sensor Calibration

As was mentioned, a navigation system's performance without GPS is determined partly by the accuracy with which the dead-reckoning sensor errors are calibrated *before* the GPS fixes become unavailable. For the "baseline" navigation system, the calibration of the rate gyro's bias, the odometer scale factor bias, and the vehicle's heading are the key parameters that govern the position error growth rate after GPS position fixes become unavailable. Therefore, in this section, we shall examine the Kalman filter's ability to calibrate these 3 dead-reckoning parameters. Results will be shown for 3 navigation systems, each of which utilizes a different type of GPS positioning.

We begin by examining the filter's ability to estimate the vehicle's heading. The

GPS Positioning Type	SA On	SA Off	DGPS
RMS Error (deg)	0.76	0.42	0.12
GPS Bias Drift	59.4%	67.1%	8.1%
GPS Measurement Noise	0.9%	2.8%	40.0%
Rate Gyro Bias Drift	34.1%	21.2%	5.6%
Rate Gyro Measurement Noise	5.6%	9%	46.2%

Table 6.5: Relative contributions to mean-square error in heading estimate

first row of Table 6.5 shows the RMS steady-state heading error for the same 3 navigation systems described in Section 6.2. The remaining rows show the percent contribution that various error sources make to the mean-square error in the steady-state heading estimate.

There are several interesting points that can be made from the data in this table. First, the RMS error in the heading estimate is a strong function of the positioning accuracy, decreasing by a factor of more than 6 over the range of position fix accuracy. Second, when stand-alone GPS position fixes are utilized (i.e. SA on or off), the position fixes contribute the largest fraction of the total heading accuracy. This may be surprising because one might guess that the rate gyro errors would contribute more to heading error than the GPS bias. This result underscores the strong influence of position error on heading error. Third, the contributions that the uncorrelated error sources make become much more significant and the correlated errors become less significant as the accuracy of the position fixes improves. For example, when DGPS is used, the sources of uncorrelated noise—GPS measurement noise and rate gyro measurement noise—contribute almost 90% of the total error. When SA is on, the correlated errors—SA-induced position errors and the rate gyro bias drift—contribute more than 90% of the total error.

The trends observed in the relative contributions of the correlated and uncorrelated noise sources were also observed in the results in Section 6.2. The explanation for this trend is similar to the one given in Section 6.2—the improved positioning afforded by DGPS allows the correlated dead-reckoning sensor errors to be estimated more accurately than when stand-alone GPS (with SA is on) is utilized. As a result, the accuracy of the gyro bias estimate is better, and the gyro’s bias drift contributes

GPS Positioning Type	SA On	SA Off	DGPS
RMS Error (deg/s)	0.0034	0.0027	0.0018
GPS Bias Drift	23.6%	23.6%	1.2%
GPS Measurement Noise	0.5%	1.1%	6.3%
Rate Gyro Bias Drift	71%	65.2%	53.5%
Rate Gyro Measurement Noise	4.9%	10.1%	39.1%

Table 6.6: Relative contributions to mean-square error in rate gyro bias estimate

correspondingly less to the heading error. The noise sources, on the other hand, cannot be calibrated because they are uncorrelated random quantities. Therefore, their influence on heading is affected less by the improved positioning accuracy. When SA is on, the position fixes are used less effectively by the filter to calibrate the dead-reckoning sensor errors. Since the position bias (when SA is on) cannot be estimated accurately, the rate gyro bias is not calibrated well and bias drift figures more prominently into the total heading error.

Next, we examine the filter's ability to estimate the rate gyro's bias. Table 6.6 shows the RMS steady-state error in the rate gyro bias estimate for each navigation system and the percent contribution that various error sources make to the mean-square error in the steady-state rate gyro bias estimate. As the table shows, the trends in individual sensor contributions are similar to those observed in the data of Table 6.5: bias drift in the stand-alone GPS position fixes contributes a large fraction of the error, and the contributions of the correlated and uncorrelated error sources exhibit similar trends. Unlike the data in Table 6.5, however, the rate gyro's error parameters (i.e. noise and bias drift) contribute a much larger fraction of the total error.

Finally, we examine the ability of the filter to estimate the odometer scale factor bias. Table 6.7 shows the RMS error in the odometer scale factor bias estimate and the percent contribution that various error sources make to the mean-square error in the odometer scale factor bias estimate. As the data show, the contribution of the drift of the bias in the position fixes drops off quickly when DGPS position fixes are utilized, and the odometer's uncorrelated quantization noise becomes much more significant. The random drift in the odometer bias is the dominant error source in all

GPS Positioning Type	SA On	SA Off	DGPS
RMS Error (m/pulse $\times 10^{-4}$ )	4.91	3.98	2.15
GPS Bias Drift	28.1%	32.0%	3.0%
GPS Measurement Noise	0.2%	0.5%	4.1%
Odometer Bias Drift	71.1%	65.5%	59.9%
Odometer Quantization Noise	0.6%	2.0%	33.0%

Table 6.7: Relative contributions to mean-square error in the odometer scale factor bias estimate

cases, but its contribution decreases as the position fixes become more accurate.

In Table 6.7, the fractional contribution of the GPS bias drift exhibits a seemingly peculiar trend—it does not decrease monotonically with the accuracy of the position fixes. Instead, it is largest when SA is off. This trend appears simply because the calculation of the fractional contribution of each error source reflects the contributions that the other error sources make. A simple calculation can show that the *absolute* contribution of the GPS bias drift decreases monotonically, as expected. However, when the *fractional* contribution of the GPS bias drift is computed, it increases for the “SA Off” case because contribution of the odometer bias drift drops more rapidly. Hence, even though the absolute contribution of the GPS bias drift decreases, it does not decrease as rapidly as the contribution of the odometer’s bias drift. As a result, the *fractional* contribution of the GPS bias drift increases for the “SA Off” case.

These results were obtained assuming a “worst-case” odometer bias drift rate. Because the results in Table 6.7 show that the odometer bias drift is the most significant contributor to the error in the estimate of the odometer’s scale factor bias, reduction of the odometer bias drift rate would have a very significant impact on these results. In the discussion of Chapter 3, the point was made that the parameters in the model for the odometer bias drift are uncertain. Therefore, it is important to explore the likely *range* of odometer bias drift rates. Table 6.8 shows the results obtained when the RMS bias drift used to obtain the results in Table 6.7 was decreased by a factor of 3.

GPS Positioning Type	SA On	SA Off	DGPS
RMS Error (m/pulse $\times 10^{-4}$ )	2.98	2.57	1.47
GPS Bias Drift	76.4%	76.6%	6.4%
GPS Measurement Noise	0.5%	1.2%	8.8%
Odometer Bias Drift	21.5%	17.4%	14.2%
Odometer Quantization Noise	1.6%	4.8%	70.6%

Table 6.8: Mean-square error in the odometer scale factor bias estimate for low bias drift

### 6.3.2 System Performance Without GPS Position Fixes

In Chapter 1, the point was made that a GPS receiver may not be able to provide a position fix under certain circumstances. In urban or heavily-foliated environments, for example, buildings or foliage may prevent a sufficient number of GPS satellite signals from reaching the GPS receiver. As a result, the GPS receiver may be incapable of providing a position fix for an indefinitely long period of time, and the navigation system would have to produce a position estimate based solely on its dead-reckoning sensor data. The results presented in the previous section provide clues to the performance to be expected from various navigation systems if GPS position fixes become unavailable. Naturally, one would expect that those systems for which the rate gyro bias, the heading, and the odometer scale factor bias are calibrated more accurately would perform better after GPS position fixes become unavailable. However, this is not conclusive because the performance of each system without GPS also depends on the drift characteristics inherent in each dead-reckoning sensor. Therefore, in this section we shall examine the influence of sensor drift on system performance by examining the performance of each system without GPS position fixes. The navigation systems to be examined are the same as those of the previous section.

Figures 6.1 and 6.2 show the RMS cross-track and along-track positioning error versus time for a system in which the GPS position fixes were corrupted with SA. The uppermost curve in each plot represents the total RMS position error; the other curves in each figure represent the RMS contributions that each error source makes to the position error. After the filter reached (nearly) steady-state (after 1300 seconds), the GPS position measurement was denied to the Kalman filter, causing the filter to

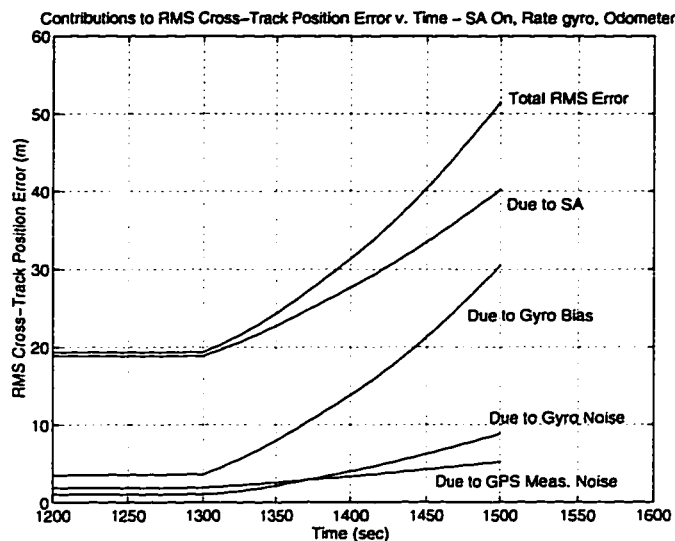


Figure 6.1: RMS error in cross-track position estimate after a GPS loss (SA on)

continue with measurements from only the rate gyro and odometer. Note that the time axis (i.e. the ordinate) begins at 1200 seconds because we are interested only in the results after that time.

As both figures show, the GPS positioning errors (bias and measurement noise together) make the largest absolute contribution to the position error *after* the GPS measurements have become unavailable. At first glance, this may not appear to make sense. After all, how can GPS measurement errors contribute to positioning error if the GPS position fixes are not available? The answer to this question lies in the realization that the GPS position error contributes to errors in the *calibration* of the heading, rate gyro bias, and the odometer scale factor bias during the first 1300 seconds. (This fact was demonstrated in the previous section.) Part of the position error that accrues *after* GPS becomes unavailable is due to the calibration error induced by GPS errors *before* GPS becomes unavailable. Therefore, the GPS measurement errors cause the position error to increase with time even after the GPS fixes become unavailable because they induce errors in the estimates of the vehicle's heading, the gyro's bias, and the odometer scale factor bias. Hence, the plot demonstrates that position error grows rapidly as a result of the calibration error

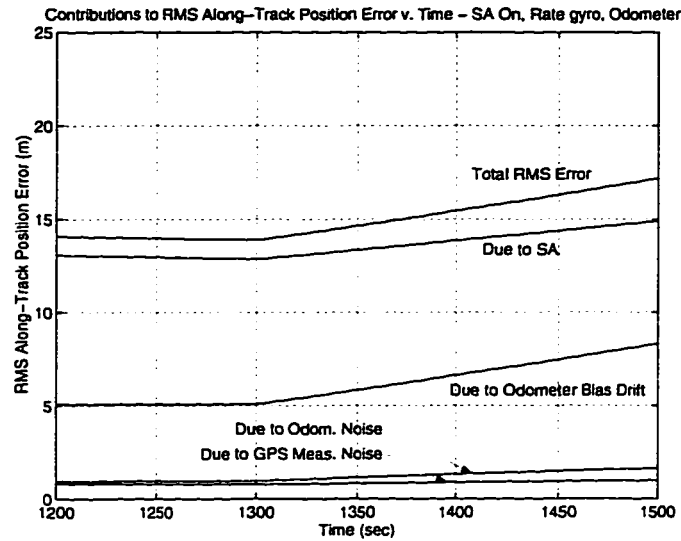


Figure 6.2: RMS error in along-track position estimate after a GPS loss (SA on)

induced by SA *while* the GPS position fixes are available.

Comparison of the figures shows that the cross-track position error grows much more rapidly than the along-track position error. The reason for this is revealed by examining the contributions that the individual sensor errors make to the position error. As Figure 6.1 shows, the growth rate of the cross-track position error is dominated by the contribution of the rate gyro bias' random drift. Figure 6.2 shows that the growth rate of the along-track position error is dominated by the contribution of SA and the odometer scale factor bias drift. Part of the reason that the cross-track position error grows more rapidly than the along-track position error is that the rate gyro bias is two integrations removed from position; this implies that a constant rate gyro bias will cause position error to grow roughly as the *square* of time. The odometer scale factor bias, on the other hand, is only one integration removed from position, implying that a constant odometer scale factor bias error will cause position error to grow *linearly* with time. Hence, the magnitude of the sensor errors *and* their kinematic relationship to position determine the growth rate of the position error.

Because the rate gyro's bias drift dominates the growth rate of the cross-track

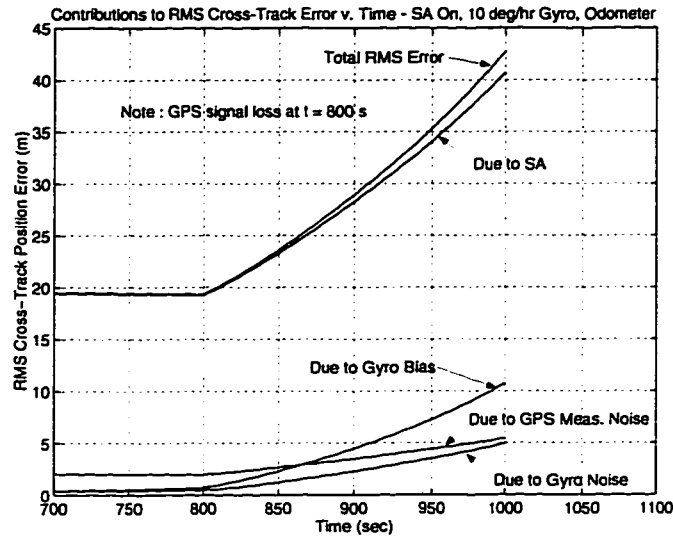


Figure 6.3: RMS error in cross-track position when using a 10-degree/hour gyro

position error, using a gyro with a lower bias drift rate should improve system performance. However, even when a better gyro is used, the total cross-track position error is not reduced significantly. Figure 6.3 shows the total RMS cross-track position error versus time for a system in which the rate gyro (with a bias drift of approximately 30 degrees/hour) was replaced with a better one (having a bias drift of approximately 10 degrees/hour). (This approximates the replacement of the Murata rate gyro with a fiber-optic rate gyro.) The uppermost curve in the figure is the total RMS cross-track position error, and the remaining curves represent the contributions that individual sensor errors make to the total cross-track position error. As the data show, the position error induced by SA is by far the largest contributor. When compared with Figure 6.1, one can see that the contribution of the 10-degree/hour rate gyro is significantly less than that of the 30-degree/hour gyro. However, the contribution of SA remains nearly unchanged, and, ultimately, using the better gyro is of little consequence.

For a system utilizing GPS position fixes with SA off, the relative contributions of the various sensor errors are similar to those when SA is on. Figures 6.4 and 6.5 show data similar to Figures 6.1 and 6.2. As the data show, the growth rate of the error in

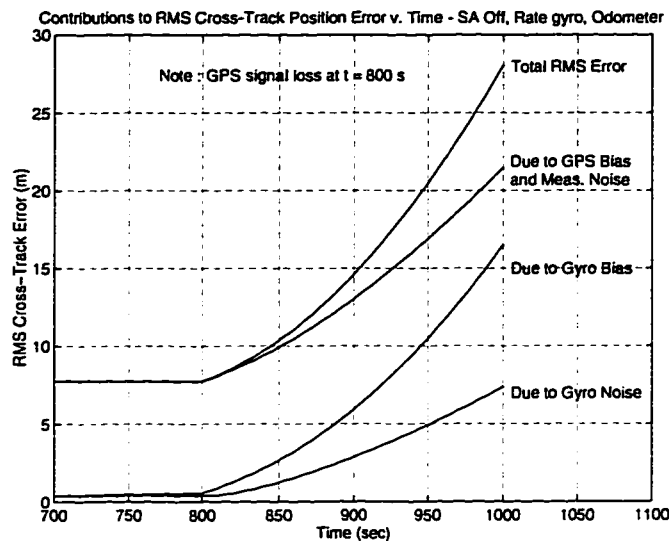


Figure 6.4: RMS error in cross-track position estimate after a GPS loss (SA off)

the cross-track direction is dominated by the rate gyro's bias drift and is higher than that for the along-track direction.

For a system utilizing DGPS position fixes, results are roughly similar—the rate gyro's bias and noise dominate the cross-track position error growth rate, and the odometer's bias drift dominates the along-track position error growth rate. Figures 6.6 and 6.7 show data similar to Figures 6.1 and 6.2. As the data show, the total contribution of the gyro bias drift to the cross-track position error is substantially less than when stand-alone GPS position fixes (with SA on) are utilized. This occurs because the rate gyro's bias is calibrated more accurately when DGPS is utilized. Also, the relative contribution of the noise in the gyro's output is larger in this case than when SA is on. Interestingly, the contribution of the gyro bias drift and the noise in the gyro output are commensurate. Nevertheless, the contribution of the drift eventually exceeds that of the noise, and the growth rate of the total position error is still dominated by the gyro bias drift.

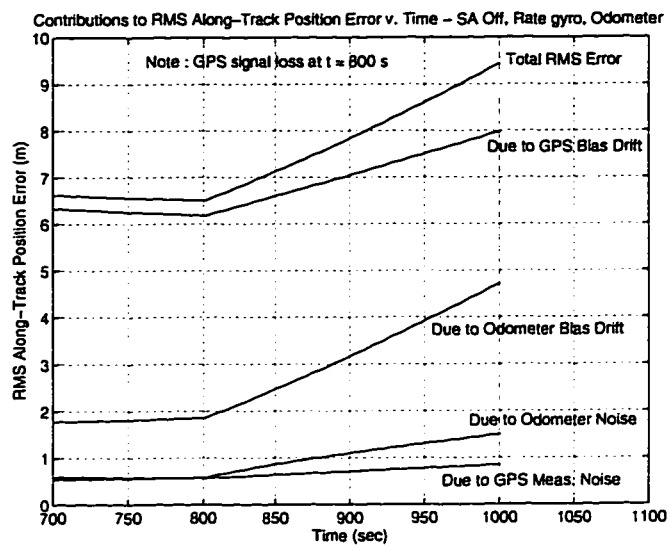


Figure 6.5: RMS error in along-track position estimate after a GPS loss (SA off)

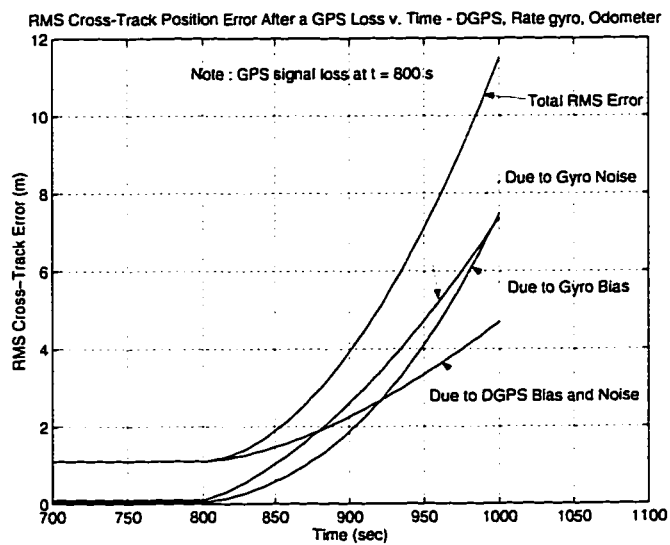


Figure 6.6: RMS error in cross-track position estimate after a GPS loss (DGPS)

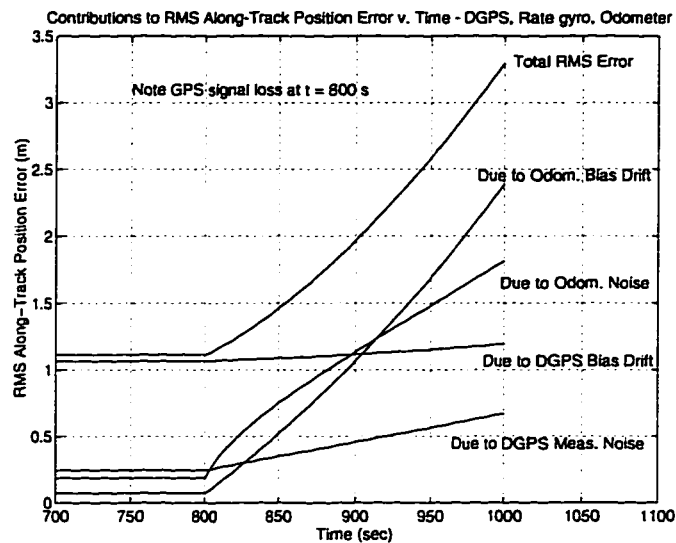


Figure 6.7: RMS error in along-track position estimate after a GPS loss (DGPS)

## 6.4 The Influence of a Heading Measurement

The data presented in Section 6.3.2 demonstrated that the growth rate of the position error (without GPS fixes) is dominated by heading errors induced by the rate gyro's bias drift and noise. This fact suggests that the growth rate of the position error could be reduced if an absolute heading measurement were added to the system because a heading measurement would allow the filter to continuously calibrate the rate gyro's bias even without GPS fixes and would also bound the total heading error. In this section, results will be shown for two systems utilizing a heading measurement in addition to GPS, a rate gyro, and an odometer. The difference between the systems lies in the source of the heading measurement—one uses a compass and the other uses a GPS-based heading measurement.

### 6.4.1 Using a Fluxgate Compass

First, results will be shown for a system in which the heading measurement is taken from a fluxgate compass. As was discussed in detail in Chapter 3, a fluxgate compass is susceptible to magnetic disturbances and, as a result, its output suffers from errors

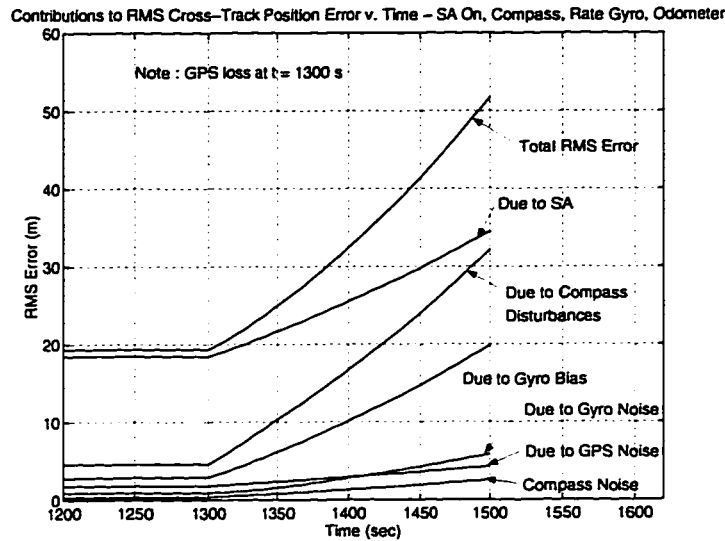


Figure 6.8: RMS error in cross-track position estimate when using a compass

that occur at unpredictable times and with unpredictable magnitude. Therefore, because “typical” compass behavior defies accurate definition, this analysis examines system performance when subjected to a *range* of compass errors. With this approach, the results here should, at least, *bound* the expected system performance. In the results that follow, magnetic disturbances to the compass reading were simulated as having a minimum value of 0.0 degrees and a maximum RMS value of 8.0 degrees. Although the errors in the compass’ output can far exceed an RMS value of 8.0 degrees (see Figure 3.6), it is assumed that any real navigation system would employ a fault detection algorithm, and that the compass data would be rejected if the errors exceeded 8.0 degrees (RMS).

Having established the grounds for the analysis, we proceed by examining Figure 6.8. This figure shows the error in the cross-track position estimate versus time. Notice that the time axis begins at 1200 seconds, when the filter is nearly in steady-state, and that the GPS position fixes become unavailable beginning at 1300 seconds. As the figure shows, the simulated magnetic disturbances dominate the growth rate of the cross-track position error. Comparison of Figure 6.8 with Figure 6.1 shows that the total RMS cross-track position error is approximately 52 meters at the end of

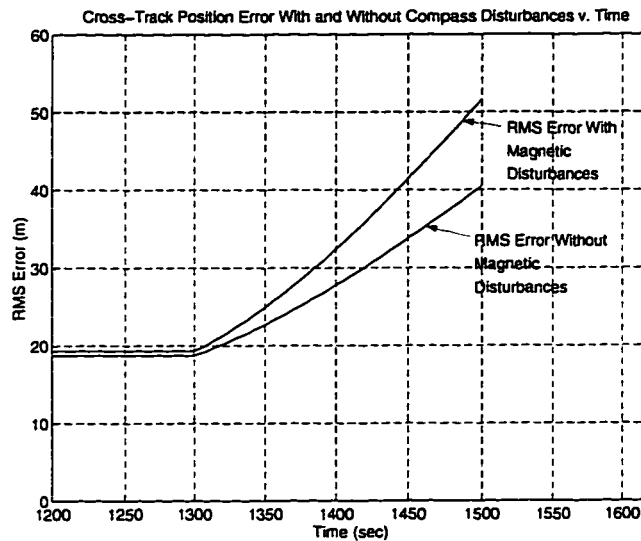


Figure 6.9: Cross-track position error using a compass with and without magnetic disturbances

both simulations. The data shown in Figure 6.1 were obtained for a system in which a compass was not utilized. Therefore, the addition of the fluxgate compass did not cause the cross-track position error to decrease at all. If one assumes a “best-case” scenario in which there are *no* magnetic disturbances whatsoever, then the cross-track position error will be lower. Figure 6.9 shows the total RMS cross-track position error obtained when the magnetic disturbances are eliminated. Also shown in the figure, for reference, is the total RMS cross-track position error that is shown in Figure 6.8.

The space between the two curves in Figure 6.9 represents the range of performance that could be expected from the navigation system if the RMS value of the disturbances in the compass measurements were between 0.0 and 8.0 degrees. Note that, when there are *no* magnetic disturbances, the cross-track position error at the end of the simulation is approximately 40 meters, and when the magnetic disturbances are maximum (8.0 degrees RMS), the cross-track position error is approximately 52 meters. The key observation to be made here is that a very substantial improvement in the performance of the compass improved the cross-track position error (at the end of the simulation) by only 23%. The improvement in the cross-track position error

is not larger because the Kalman filter has been designed conservatively with regard to the compass measurements. In other words, because the compass measurements can be very inaccurate, the filter has been designed so that it does not “trust” the compass measurements heavily. The net result is that the filter’s estimate of heading (and, therefore, its estimate of cross-track position) contains larger errors than would be obtained if the compass measurements could be considered as more reliable. In sum, then, significant decreases in the errors in the compass measurements do not result in correspondingly large decreases in cross-track position error because the filter design is conservative.

To close this section, it should be mentioned that the results in Figures 6.8 and 6.9 should be accepted with caution. As was mentioned, the magnetic disturbances appearing in the compass measurement were modeled as a first-order Gauss-Markov process with a time-constant of 3.0 seconds and an RMS value of 8.0 degrees. This model was chosen based on heuristic observations of actual compass disturbances similar to those found in Figure 3.6. If such disturbances are not well-modeled as a first-order Gauss-Markov process with a time-constant of 3.0 seconds, then the influence of the disturbances on the RMS cross-track position error will likely be different than that suggested by Figure 6.9. If the time-constant of the disturbances is less than 3.0 seconds, the disturbances are more nearly approximated by uncorrelated noise and their deleterious effects on system performance will probably decrease because they will be smoothed. On the other hand, if the time-constant is greater than 3.0 seconds, the negative impact of the disturbances on the cross-track position error will likely increase.

### 6.4.2 Using a GPS-based Heading Measurement

As the results in the previous section demonstrated, heading measurement error strongly affects system performance. Improving heading sensor performance may therefore improve overall system performance significantly. In this section, the benefit of using a GPS-based heading measurement is examined. (See Section 3.5.4 for a discussion of GPS-based heading measurements.)

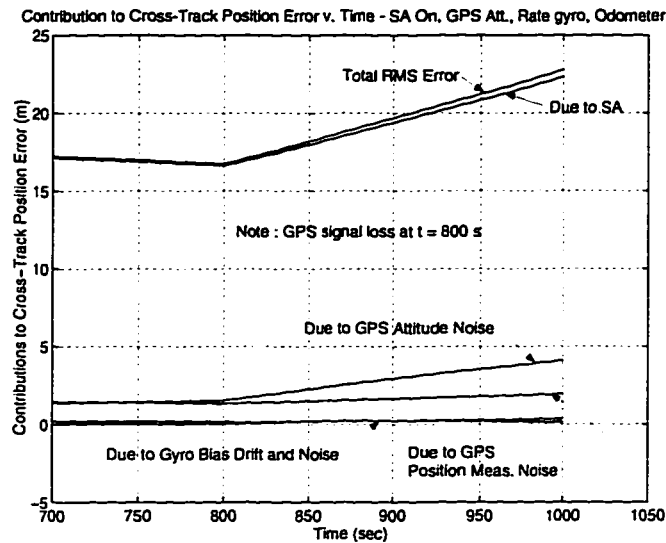


Figure 6.10: RMS error in cross-track position estimate when using GPS attitude

Figure 6.10 shows the RMS cross-track position error (versus time) for a system utilizing a GPS-based heading measurement. Notice that the time axis begins at 700 seconds, when the filter is nearly in steady-state, and the GPS position fixes become unavailable beginning at 800 seconds. As the data show, the RMS cross-track position error grows much more slowly when the GPS attitude measurement is used than when the compass is used.

For purposes of comparison, Figure 6.11 shows the RMS cross-track position error for 3 systems: a system without a heading measurement, a system with a compass (with no magnetic disturbances), and a system utilizing a GPS-based heading measurement. The curves in Figure 6.11 are taken from Figures 6.1, Figures 6.9, and 6.10. As the data show, the RMS cross-track position error is much smaller when the GPS attitude measurement is used than when the compass is used, even when there are no magnetic disturbances in the compass' reading. This difference in performance arises because the Kalman filter design is more conservative when the compass is utilized. The reason that the filter is designed more conservatively is so that it can track the heading-dependent changes in the compass' bias. (Recall from Equation 3.16 that the model for the compass' bias includes a heading-dependent error term.) Unlike the

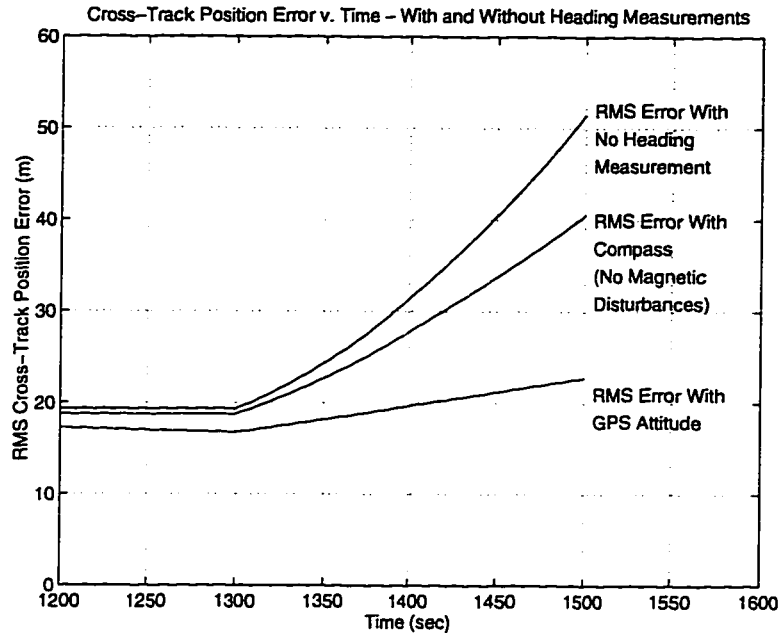


Figure 6.11: Comparison of RMS cross-track position error for 3 heading measurements

compass, the heading measurement from the GPS receiver is assumed to be corrupted with a *constant* bias and uncorrelated noise. Because the bias in the GPS heading measurement is constant, the filter parameters were chosen to take advantage of this knowledge. The performance of the more-conservative filter is worse because it weighs its heading measurement less and therefore reaps less benefit from this measurement.

Figure 6.11 also shows that the benefits of utilizing a GPS-based heading measurement are substantial, particularly when the performance of this system is compared with that of the system without a heading measurement. With the heading measurement, the contribution of the rate gyro bias drift to the cross-track position error is reduced enormously, and the cross-track position error grows approximately 6 meters while the GPS position fixes are unavailable. In contrast, without the heading measurement, the rate gyro bias is the dominant contributor to the cross-track position error growth rate, and the cross-track position error grows by more than 30 meters while the GPS position fixes are unavailable.

## 6.5 The Influence of Vehicle Speed

It was assumed in all of the simulation results shown thus far that the vehicle was moving at a constant speed of 10 meters/second (22 MPH). This is a typical speed for a vehicle traveling on residential roads. However, the performance of a navigation system will depend on the vehicle's speed, and the extent to which the performance is influenced by vehicle speed should be examined. This section compares results for two navigation systems utilizing the same set of sensors (i.e. GPS with SA on, a rate gyro, and an odometer). In each simulation, however, the vehicle's speed was different—10 m/s in one simulation and 30 m/s (67 MPH) in the other.

### 6.5.1 The Estimate of Cross-track Position

Results show that, while GPS position fixes are available, the error in the steady-state estimate of the vehicle's cross-track position is nearly the same at both speeds. This occurs because the cross-track position error is dominated by the accuracy of the GPS position fixes, regardless of the vehicle's speed. When GPS position fixes become unavailable, however, the growth *rate* of the cross-track position error is lower when the vehicle moves faster. Table 6.9 shows the growth rate of the cross-track position error after the GPS position fixes become unavailable. The growth rates in Table 6.9 were calculated by taking the ratio of the change in cross-track position error that occurred while the GPS position fixes were unavailable to the total distance traveled during that time. The reader should note that the cross-track position error grows more rapidly as a function of *time* when the vehicle moves faster; however, it grows more slowly as a function of the *distance traveled*. The reader should also note that, because the cross-track position error is not a linear function of time, the growth rates given in Table 6.9 depend on the length of time that the GPS position fixes are unavailable; the data in Table 6.9 were computed from results in which the position fixes were unavailable for 200 seconds.

The cross-track position error grows more slowly when the vehicle moves faster because the Kalman filter estimates the vehicle's heading and calibrates the rate gyro's bias more accurately *while* the GPS position fixes are available. Table 6.10 shows the

Vehicle Speed	10 m/s	30 m/s
Cross-track Position Error Growth Rate (cm error/m traveled)	1.6	0.93

Table 6.9: Cross-track position error growth rate at 2 vehicle speeds

Vehicle Speed	10 m/s	30 m/s
RMS Heading Error (deg)	0.76	0.37
RMS Rate Gyro Bias Error (deg/s)	$3.4 \times 10^{-3}$	$2.5 \times 10^{-3}$

Table 6.10: RMS error in heading and rate gyro bias estimates at 2 vehicle speeds

RMS error in the steady-state heading and rate gyro bias estimates at each vehicle speed. As the data in the table show, the estimates improve when the vehicle moves faster. This occurs because, as the vehicle moves faster, the distance that the vehicle travels between consecutive measurements increases (because the sampling rate is fixed). Because the position fixes move farther apart as the vehicle moves faster, the accuracy of the heading inferred by the filter from the position measurements improves.

### 6.5.2 The Estimate of Along-track Position

The error in the estimate of the vehicle's along-track position is slightly worse when the vehicle moves faster. After 800 seconds, for example, the RMS error in the estimate of the vehicle's along-track position is 15.0 meters when the vehicle's speed is 10 m/s and 16.8 meters when the vehicle's speed is 30 m/s. Note that this is opposite to the effect observed with heading. The along-track position error increases with the vehicle's speed because of the bias in the odometer scale factor. The contribution of the odometer scale factor bias to the error in the odometer measurement increases in proportion to the vehicle's actual speed; as the vehicle travels faster, then, the error in the odometer measurement becomes larger, and the error in the filter's estimate of along-track position increases as a result. For example, if an odometer is biased by 1%, a measurement of speed obtained from this odometer will be in error by 10 cm/s when the vehicle is traveling at 10 m/s and 30 cm/s when the vehicle is traveling at 30 m/s. This error in the estimate of the vehicle's speed will figure into the error

in the filter's along-track position estimate. Hence, for a given odometer scale factor bias, the measurement error increases as the vehicle's speed increases, and the error in the filter's estimate of along-track position increases as a result.

## 6.6 The Influence of Odometer Resolution

Simulations in this section were run to examine the influence that the odometer's resolution has on the estimates of various quantities. "Odometer resolution" is the distance that a vehicle moves forward between consecutive pulses generated by the odometer. This quantity is measured in units of centimeters/pulse (or meters/pulse). The number of pulses generated by the odometer on an anti-lock braking system varies between 12 and 60 pulses per wheel revolution. The distance traveled between pulses depends on the radius of the wheel, but a rough calculation suggests that, for a typical compact car with a wheel circumference of 2.0 meters, this corresponds to resolutions between 3 and 17 centimeters per pulse.

The odometer's resolution defines the magnitude of its quantization error—the finer the resolution, the smaller the quantization error. The quantization error, in turn, appears in the odometer measurement as an uncorrelated zero-mean random error (see Section 3.4). Examination of Figures 6.2, 6.5, and 6.7 suggest that the odometer's quantization error is not a significant contributor to along-track position error unless DGPS position fixes are utilized *and* those position fixes become unavailable. (This result is consistent with results shown previously that demonstrated that uncorrelated sensor errors affect position accuracy more when DGPS position fixes are used than when stand-alone GPS position fixes are used.) The results shown in Figures 6.2, 6.5, and 6.7 could be considered nearly "worst-case" because the odometer resolution was relatively coarse (20.0 cm/pulse).

Because the odometer resolution can play a significant role in positioning accuracy when DGPS position fixes are utilized, it is worth examining the benefits of using an odometer with a finer resolution in a system utilizing DGPS position fixes. Figure 6.12 shows how a decrease in the odometer resolution will cause the along-track position error to grow at a slower rate when DGPS position fixes become unavailable. The

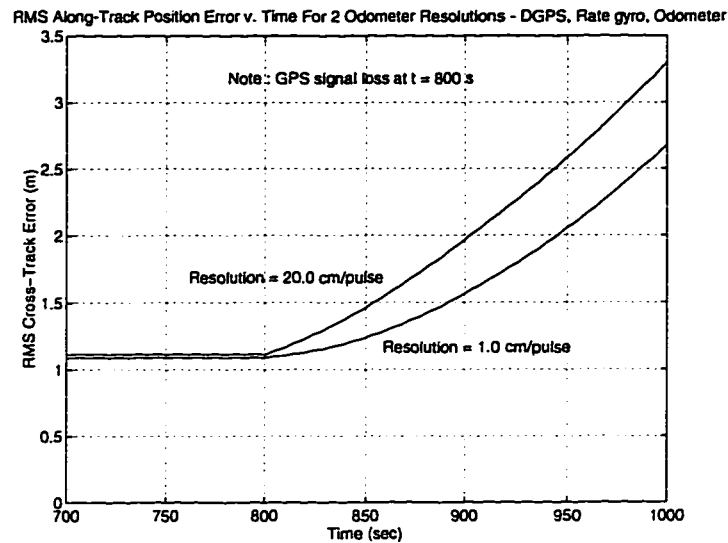


Figure 6.12: Along-track position error with various odometer resolutions

figure shows the RMS along-track position error for two navigation systems, each of which utilized DGPS position fixes, a rate gyro, and an odometer. The odometer resolution was 20 cm/pulse in one simulation and 1.0 cm/pulse in the other. The DGPS position fixes were denied to the Kalman filter beginning at 800 seconds to simulate a loss of GPS positioning capability. As the figure shows, the improvement in the odometer resolution has some effect on the growth rate of the cross-track position error. Ultimately, the growth rate of the along-track position error is dominated by the odometer's scale factor bias drift, which is the same for both systems shown in the figure. As a result, the position error eventually grows at approximately the same rate in both systems.

It is worth noting that the resolution of the odometer has a strong effect on the filter's estimate of speed. Table 6.11 shows the contributions that various error sources make to the mean-square error in the estimate of the vehicle's speed. Data in the table are shown for 3 navigation systems, each of which utilized GPS with SA on, a rate gyro, and an odometer with a different resolution (1.0 cm/pulse, 10.0 cm/pulse, and 20.0 cm/pulse). As the data show, the contribution that quantization noise makes to

Odometer Resolution	1.0 cm/pulse	10.0 cm/pulse	20.0 cm/pulse
RMS Error (m/s)	0.03	0.054	0.093
GPS Bias Drift	47.4%	14.2%	2.0%
GPS Measurement Noise	0.5%	0.2%	0.2%
Odometer Bias Drift	49.7%	14.9%	4.9%
Odometer Quantization Noise	2.4%	70.7%	92.9%

Table 6.11: Relative contributions to mean-square error in speed estimate for 3 odometer resolutions

the mean-square estimate error increases as the odometer resolution becomes increasingly coarse. The interesting point to be made is that the fractional contribution of the quantization error dominates the total error when the resolution is 20.0 cm/pulse but is nearly insignificant when the resolution is 1.0 cm/pulse. Odometer resolution is therefore a key parameter if the navigation system operator is interested in obtaining accurate estimates of the vehicle's speed.

## 6.7 The Influence of Map-matching on Sensor Calibration

As described in Chapter 4, the use of map-matching to calibrate dead-reckoning sensors is not always beneficial. Errors in the map-matched position can skew sensor calibration, causing the navigation software to produce erroneous estimates of certain sensor parameters. Specifically, map-matching can degrade heading sensor calibration as a result of the vehicle's lateral motion on a road. The reader is referred to Sections 4.3 and 5.2.2 for more detailed explanations of the manner in which map-matching is treated in this research.

In this section, results for two simulations will be presented. Both navigation systems included map-matching, GPS (with SA on), a rate gyro, and an odometer. The difference between the simulations lies in the motion of the vehicle: in the first simulation, the vehicle moved in a straight line (parallel to the centerline of the road) and made a single lane-change; in the second simulation, the vehicle's motion is more realistic, constantly wandering laterally approximately 30 centimeters. The purpose

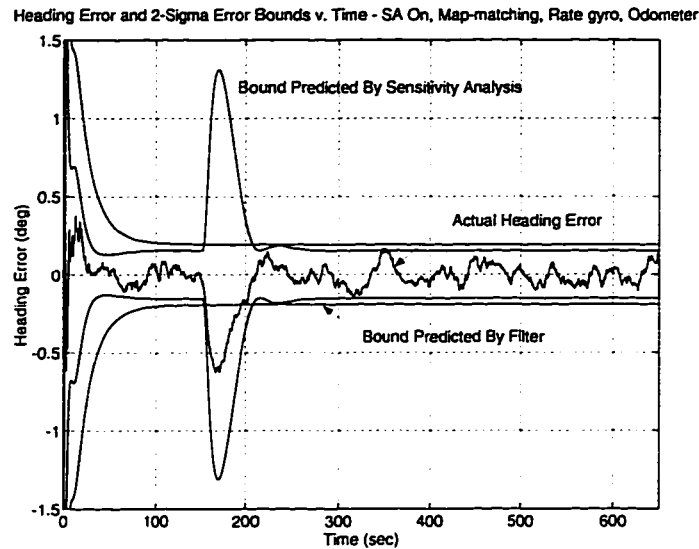


Figure 6.13: Heading error induced by map-matching during a lane-change

of examining the effects of a lane-change and side-to-side motion with two separate simulations is to isolate the individual effects of these two types of motion.

### 6.7.1 The Effects of a Lane-change

Figure 6.13 shows the influence that a lane-change has on the filter's heading estimate when map-matching is in use. The figure shows the  $2\text{-}\sigma$  (i.e. 95%) error bounds predicted by the sensitivity analysis and by the filter itself. For reference, the figure also shows the actual error in the heading estimate for a single run. The lane-change began at 150 seconds and lasted for 10 seconds; in that time, the vehicle moved laterally 3 meters. As the figure shows, the filter, being unaware of the bias in the cross-track position "measurements," optimistically predicts that the error in the heading estimate remains unchanged from its steady-state value during the lane-change. The sensitivity analysis results, however, indicate that the heading error will reach approximately 1.3 degrees ( $2\text{-}\sigma$ ). The actual heading error in the filter's heading estimate confirms the sensitivity analysis results. Figure 6.14 shows the influence that a lane-change has on the filter's estimate of the rate gyro's bias. The figure shows the

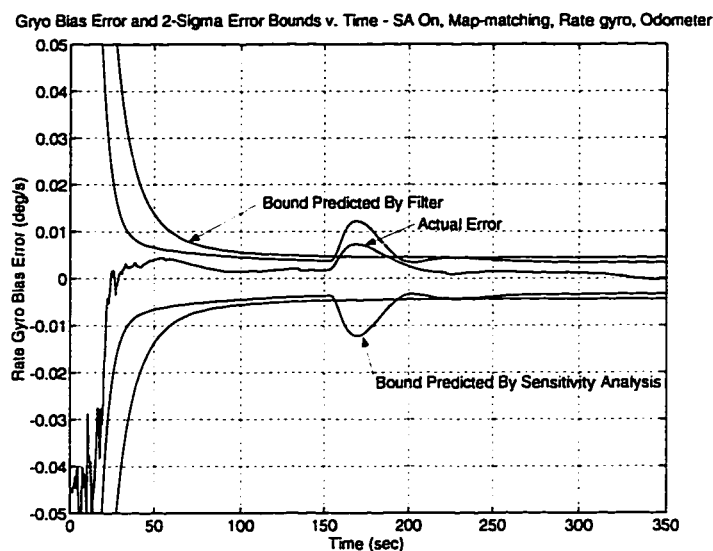


Figure 6.14: Gyro bias error induced by map-matching during a lane-change

$2\text{-}\sigma$  (i.e. 95%) error bounds predicted by the sensitivity analysis and by the filter itself. For reference, the figure also shows the actual error in the rate gyro bias estimate for a single run. The results are similar to those shown for the heading estimate, but the error induced in the gyro bias estimate does not appear to be as severe as that induced in the heading estimate. This result is consistent with results presented in Tables 6.5 and 6.6, in which it was demonstrated that position fixes influence the heading estimate more heavily than the rate gyro bias estimate. This is probably due in part to the fact that the rate gyro bias is two integrations removed from position, while heading is only one integration removed from position. Another cause for this result may be the fact that a heading rate measurement is available, but a heading measurement is not. The presence of a heading rate measurement implies that this measurement will figure prominently into the estimate of heading rate; the absence of a heading measurement implies that the filter must infer its heading estimate from other measurements. The position measurement will likely influence the heading estimate more heavily than the heading rate estimate.

Figures 6.13 and 6.14 show one other result that may not be obvious at first glance—even though the lane-change lasted only 10 seconds, the errors in the heading

and gyro bias estimates that were induced by the lane-change lasted much longer (over 100 seconds). This happened because, loosely speaking, the Kalman filter was “unaware” of the error in the estimates. When the lane-change occurred, the Kalman gain in the filter had reached its steady-state value and was weighing the sensor data as if the errors in the heading and gyro bias estimates were small. The filter therefore did not weigh the data from the other sensors appropriately to eliminate these errors quickly. This idea may be understood more clearly if one examines the beginning of the simulation, when the filter “knew” that the errors in the heading and gyro bias estimates were large. When the filter “knew” that the errors were large, the Kalman gain was computed in such a way that the GPS and map-matching measurements were utilized to quickly reduce the heading and gyro bias errors. In steady-state, the filter gain was calculated assuming that the errors in the heading and gyro bias had reached their steady-state values, and the errors induced by the lane-change decrease much more slowly.

A lane-change can therefore induce calibration errors that last much longer than the lane-change itself. One way that the effects of lane-changes on calibration errors can be mitigated is to increase the RMS measurement noise associated with the map-matching “measurement.” The Kalman filter would then attribute to the map-matched position a larger fraction of the discrepancy between the heading inferred from the map-matched positions and the heading rate measured from the gyro. However, increasing the RMS measurement noise associated with the map-matching “measurement” would have a detrimental effect on the cross-track positioning accuracy. This would occur because the filter would weigh the GPS position fixes more heavily, and the bias in the GPS position fixes would figure more prominently into the position estimate.

### 6.7.2 The Effects of Lateral Motion Within a Lane

In this section, we shall examine the calibration errors that are caused by map-matching as a result of a vehicle’s lateral motion within a lane. In the simulation from which the following results were taken, the vehicle wandered laterally very slowly,

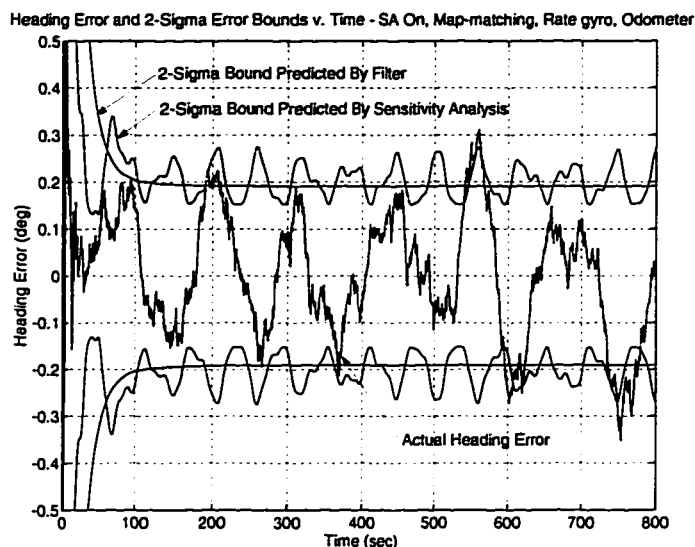


Figure 6.15: Heading error induced by map-matching as a result of motion within a lane

with an amplitude of approximately 30 centimeters. Results are shown in Figure 6.15. The figure shows 3 curves: the  $2\text{-}\sigma$  (i.e. 95%) heading error predicted by the filter, the  $2\text{-}\sigma$  heading error predicted by the sensitivity analysis, and the true heading error obtained from one simulation run. As the figure shows, the filter does not predict the heading error induced by map-matching. However, the heading error predicted by the sensitivity analysis does not substantially exceed that predicted by the filter.

The reason that the motion does not cause the heading error to increase significantly can probably be understood in light of the fact that the vehicle's actual heading varies only very slightly from the heading of the road. In other words, the vehicle's average heading *must* be the same as the road's (constant) heading, otherwise the vehicle would head off the road. The vehicle's *actual* heading deviates only a very small amount from this average. For the simulation at hand, for example, the changes in heading required to cause the vehicle to wander are very small, reaching a maximum of approximately 0.3 degrees. Therefore, during the simulation, the *actual* heading of the vehicle was never more than 0.3 degrees from the heading that would be inferred from consecutive map-matched position "measurements." It should not, therefore,

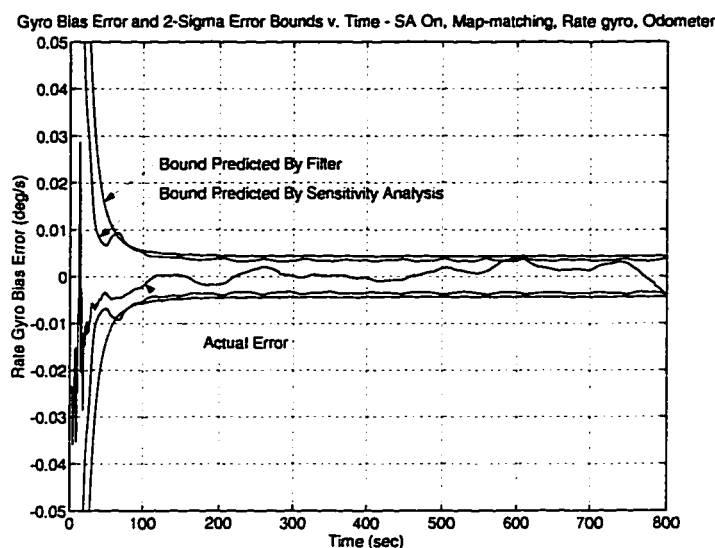


Figure 6.16: Rate gyro bias error induced by map-matching as a result of motion within a lane

be too surprising that the motion of the vehicle within the lane does not increase the heading error significantly.

Finally, we examine the influence of lateral in-lane motion on the estimate of the rate gyro bias. Figure 6.16 shows the  $2\text{-}\sigma$  rate gyro bias error predicted by the filter, the  $2\text{-}\sigma$  rate gyro bias error predicted by the sensitivity analysis, and the true error in the rate gyro bias estimate obtained from one simulation run.

As the figure shows, the actual error in the rate gyro bias estimate is well within the filter's predicted  $2\text{-}\sigma$  boundary. The filter is more "pessimistic" about the rate gyro bias estimate because the Kalman filter design is conservative. There are two different models for the rate gyro bias drift: one in the Kalman filter equations (a random walk) and one in the reference system equations (a second-order Gauss-Markov process). The parameters of the random walk model were conservatively chosen so that the filter could track *worst-case* drift in the rate gyro's bias. Therefore, the filter has been designed to "expect" bias drift that is worse than the true nominal bias drift. As a result, the filter's prediction of the error in the bias estimate is larger than the actual error.

In sum, the influence of lateral motion within a lane has a greater effect on the estimate of heading than on the estimate of the rate gyro's bias. However, the largest effects of motion within a lane are not nearly as large as the effects of a lane-change. The reason that lateral motion within a lane has less effect than a lane-change is simply due to the nature of the vehicle's motion. Lateral motion within a lane involves heading changes that are smaller than those encountered in a lane-change, and the changes in heading associated with in-lane motion take place over time periods that are much longer than those associated with a lane-change.

## 6.8 The Influence of Rate Gyro Scale Factor Errors and Turns

In this section, the effect of rate gyro scale factor error is examined. As discussed in Chapter 3, the most significant influence of rate gyro scale factor error arises when the vehicle makes turns that sweep through large angles. Since many roads intersect at a right angle, a simulation was run in which the vehicle made a 90-degree turn. The simulation was run two times, each time with a different rate gyro scale factor error (1% and 5%). Figure 6.17 shows the heading error as a function of time for both simulations. The turn started at 1000 seconds and lasted approximately 7 seconds.

As the data show, the peak heading error is greater than 4 degrees when the rate gyro scale factor is in error by 5%. Also, although the turn lasted only 7 seconds, the heading error induced by the turn lasts much longer than 7 seconds. Finally, when the scale factor error is 5%, the RMS heading error exhibits peculiar behavior in which it decreases after the turn, reaches a minimum of approximately 0.8 degrees, then increases again to almost 2 degrees at the end of the simulation. This result may not be terribly significant, however, because empirical testing showed that the scale factor error for two particular low-cost rate gyros was on the order of 1% (see Figures 3.4 and 3.5). The results in Figure 6.17 suggest that a 1% scale factor error is probably not significant.

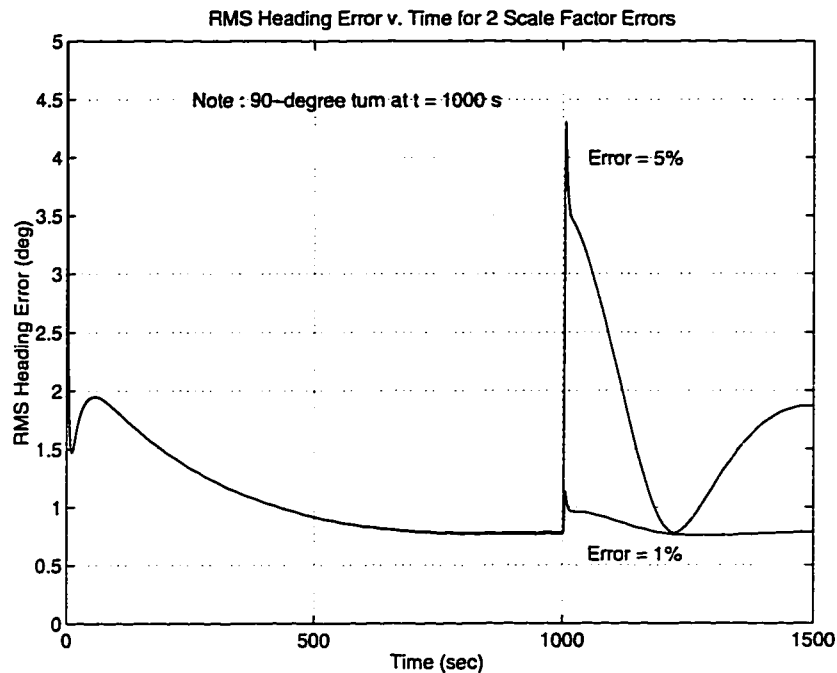


Figure 6.17: Heading error induced by rate gyro scale factor error

## 6.9 Summary

In this chapter, the techniques of sensitivity analysis derived in Chapter 2 were applied to a number of navigation systems. Each navigation system was represented by a specific Kalman filter implementation, the mathematical structure for which was described in detail in Chapter 5. The analysis was used to quantify the contributions that various error parameters make to navigation system performance. These quantitative results provided insight into the fundamental error mechanisms in each Kalman filter and identified key parameters that dominate navigation system accuracy.

Key results show that, while GPS position fixes are available to the filter, the navigation system's positioning accuracy is approximately the same as the accuracy of the GPS position fixes. When GPS position fixes are unavailable, the growth rate of the position error is dominated by the rate gyro's bias drift. Also, results showed that the relative contributions of uncorrelated noise sources become increasingly significant as the accuracy of the GPS position fixes improves.

Results further showed that the type of GPS position fix available to the filter strongly influences dead-reckoning sensor calibration and, therefore, strongly influences system performance if the GPS position fixes become unavailable. Performance of the navigation systems without GPS was worst when stand-alone GPS position fixes (with SA on) were used by the filter. However, this performance was greatly improved when the navigation system was augmented with a GPS-based heading measurement.

Map-matching was shown to have a significant influence on heading sensor calibration. However, the magnitude of the error introduced into the calibration is largely determined by the nature of the vehicle's lateral motion on the road. Lane-changes introduce much more error into the estimate of heading and the estimate of the rate gyro bias than lateral motion within a lane, because lane-changes involve relatively rapid changes in vehicle position, while lateral motion within a lane involves small heading changes over much longer time periods.

The rate gyro's scale factor error can cause very significant heading errors if the vehicle turns and if the scale factor error is relatively large. A constant rate gyro scale factor error of 5% was shown to cause the RMS heading error to increase by approximately 3.5 degrees as the vehicle made a 90-degree turn. Furthermore, the heading error induced by the vehicle's turn decreased at a very slow rate. This result implies that, if GPS position fixes become unavailable immediately after the vehicle makes a turn, position error will accrue rapidly. This result should be tempered, however, with the realization that empirical test data showed scale factor errors to be on the order of 1% for two low-cost rate gyroscopes. In addition, analysis results indicate that a scale factor error of 1% is probably not significant.

Finally, as the vehicle's speed increases, the error in the heading estimate decreases. Consequently, after position fixes become unavailable, the ratio of the increase in cross-track position error to the distance traveled decreases as the vehicle's speed increases. In contrast, the error in the estimate of the vehicle's along-track position increases. Also, the odometer's quantization error has little impact on positioning accuracy while position fixes are available; however, it does have a significant impact on the growth rate of the error in the along-track position estimate when

DGPS position fixes are utilized and subsequently become unavailable. Furthermore, odometer resolution can be the dominant error source in the estimate of the vehicle's speed.

# Chapter 7

## Conclusions and Closing Remarks

In this chapter, broad implications of the results of the previous chapter are drawn. The primary goal is to draw implications for navigation system design and sensor selection. An attempt is also made to keep an eye to the future since dead-reckoning sensor performance and the accuracy of GPS positioning will likely improve.

### 7.1 Conclusions

#### 7.1.1 Conclusions Drawn From Research Results

The most glaring implications for sensor selection and system design arise in connection with the type of GPS positioning utilized in the navigation system. The choice of GPS positioning type (if a choice is available) has strong implications for navigation system performance while GPS position fixes are available and during those times when GPS position fixes are unavailable. The positioning accuracy of a system utilizing GPS position fixes (but *not* utilizing map-matching) is virtually the same as that of the position fixes themselves. Hence, statistically, the greatest positioning accuracy that could be expected from a system is roughly that of the GPS position fixes. If, at some point in time, the GPS position fixes become unavailable, the position error will, statistically, always increase beyond that of the error in the position fixes. The rate at which the position error increases will depend not only on the type

of GPS positioning utilized but also on which sensors comprise the dead-reckoning unit and on the drift characteristics of the individual dead-reckoning sensors. Dead-reckoning sensor calibration is substantially better if DGPS position fixes are utilized than if stand-alone GPS position fixes (with SA on) are utilized. In fact, SA has such a strong influence on parameter calibration that improving gyro bias drift by a factor of 3 (from 30-degree/hour to 10-degree/hour) was shown to be of little benefit. Hence, if DGPS positioning is utilized, lower-quality dead-reckoning sensors can be employed to achieve a given level of system performance when the position fixes become unavailable for a given duration. Alternately, for a given dead-reckoning unit, positioning error will grow at a much slower rate when GPS position fixes become unavailable, *if* DGPS position fixes are available to calibrate the dead-reckoning sensors in advance.

Once GPS position fixes become unavailable, which sensors comprise the dead-reckoning unit has a strong effect on navigation system performance and on which error parameters dominate the positioning error growth rate. Assuming a dead-reckoning unit comprised of a rate gyro and an odometer, for example, then the rate gyro's bias drift will dominate the position error growth rate (if the rate gyro has the characteristics assumed in this research). The contribution that the gyro's bias drift makes to the cross-track position error is always greater than the contribution made by the random noise in the gyro's output, *unless* DGPS position fixes are utilized to calibrate the gyro's bias. If a heading measurement from a compass is added to this system, then system performance depends strongly on the compass' magnetic environment. If a relatively high-quality heading measurement (as might be obtained from an attitude-capable GPS receiver) replaces the compass, the performance of the navigation system is improved substantially: the growth rate of the cross-track position error (after GPS position fixes become unavailable) is substantially reduced and the contribution of the rate gyro's bias drift to position error is greatly reduced. Under these circumstances, the growth rate of the cross-track position error is dominated by the errors in the heading measurement and calibration error induced by SA before the position fixes become unavailable.

Another important result showed that individual dead-reckoning sensors affect

each component of position error differently. As one might guess, the heading rate and heading measurement errors contribute only to the cross-track position error, and the odometer errors contribute only to the along-track position error. This simple observation indicates that changes in sensor quality will have little or no impact on “orthogonal” components of the positioning error—e.g. using a high-quality gyro will not improve the along-track positioning error. Furthermore, improvements in one component of position accuracy will improve calibration of only the associated dead-reckoning sensors.

Finally, the presence of a successful map-matching algorithm can greatly reduce positioning error. The implicit caveat here, however, is that the algorithm must be *successful*. Whether a map-matching algorithm will be successful depends very much on the strengths and weaknesses of the particular algorithm. It is therefore difficult to make statements about how successful map-matching algorithms are in general, and how much map-matching improves navigation accuracy. However, even if the map-matching algorithm is successful in the sense that the map-matched location does not diverge from the vehicle’s true location, map-matching is still not always beneficial to sensor calibration. Errors in the map-matched position can skew the sensor calibration, causing the navigation software to produce erroneous estimates of certain sensor errors. Specifically, map-matching can induce heading sensor calibration errors when a vehicle moves laterally on a road. (Typically, lateral motion is produced by lane-changes and by a driver’s natural tendency to wander within a lane.)

In summary, the key conclusions of this research are

- Positioning accuracy is dominated by GPS position fixes while the fixes are available. Utilizing dead-reckoning sensors that are much better than existing low-cost dead-reckoning sensors does not decrease positioning error significantly below the bias error in the GPS position fixes.
- When stand-alone GPS fixes (with SA on) are utilized, the calibration error induced by SA when the position fixes are available has very significant deleterious effects on system performance *after* the position fixes become unavailable.

Improving gyro quality significantly does not improve performance substantially because heading error is strongly affected by SA. For this system, then, calibration error plays a more significant role than sensor quality in determining system performance after position fixes become unavailable.

- DGPS position fixes improve parameter calibration significantly. The improvement in sensor calibration means that lower-quality (and therefore less expensive) dead-reckoning sensors can be used to achieve a given level of system performance when the position fixes become unavailable for a given duration.

### 7.1.2 Suggestions for Navigation System Design

This section summarizes a few of the author's recommendations for navigation system design. The recommendations that follow are based partly on results that can be found in this thesis and partly on speculations that the author believes to be true, based on the experience gained in the pursuit of this research.

To utilize the results in this thesis as a basis for a navigation system design, one must first choose a performance specification for the navigation system at hand. For the sake of this discussion, let us consider a navigation system that must continuously position a vehicle with 10 to 15 meters of accuracy.

Given this specification, it is probably true that any navigation system that utilizes stand-alone GPS position fixes (with SA on) would have to rely entirely on its map-matching algorithm to achieve the specified positioning accuracy. This would be true regardless of which low-cost dead-reckoning sensors were used to augment the GPS position fixes. Even if SA were off, the navigation system would probably have to rely heavily on its map-matching algorithm, especially when GPS position fixes were unavailable. Whether or not such a navigation system could successfully meet the specification, then, would probably depend almost entirely on the performance of its map-matching algorithm.

If DGPS position fixes are utilized, the prospects for achieving continuous positioning to an accuracy of 10 meters are much better. In a system using DGPS position fixes, dead-reckoning sensor calibration can be improved enough that the

navigation system could meet its performance specification even if the position fixes become unavailable for several minutes and even if no map-matching algorithm were employed. Utilizing a dead-reckoning system that included a low-quality gyroscope and a GPS-based heading measurement *or* a relatively high-quality gyroscope (and *no* heading measurement) would probably prove to be a very good system.

Furthermore, utilizing DGPS would probably make it possible to employ a map-matching algorithm that is far simpler than existing algorithms. This could be very significant because manufacturers of commercial navigation systems (which use GPS with SA on) generally devote years of effort to fine-tuning the performance of their map-matching algorithms. Eliminating the need for a complex map-matching algorithm could greatly reduce the development time of a navigation system. Even though it would probably still be necessary to make use of map-matching when position fixes became unavailable for several minutes, one could have greater confidence that the map-matching algorithm would start on the right road and also that the dead-reckoning sensor calibration would be accurate.

## 7.2 Closing Remarks

The point of the research in this thesis has been to provide insight into the nature of land-vehicle navigation system performance through quantitative analysis. The information produced by this research will hopefully help today's navigation system designers understand tradeoffs in various candidate system designs, and also help navigation system designers in the future, if/when Selective Availability is turned off or DGPS becomes widely available. Finally, it is hoped that this research can also reveal to sensor designers those sensor error parameters that contribute most to positioning error and guide them into a design with appropriate performance tradeoffs.

# Appendix A

## Numerical Values for Various Parameters

This appendix contains numerical values for various parameters which appear in the equations of Chapters 3 and 5, including time constants, the spectral densities of process noise parameters, and RMS values of measurement noise parameters. Not all of the values given in the following tables are numerical. For example, the value for  $v_{odom,f}$  is given as  $\frac{S_o}{3}$ . This indicates that  $v_{odom,f}$  depends on  $S_o$ , the nominal odometer scale factor. In most simulations, the value of  $S_o$  was 0.20 meters/pulse. Finally, certain parameters are statistical quantities and are described by their distribution in the following tables. Normally-distributed quantities with mean  $m$  and variance  $\sigma^2$  are denoted  $N(m, \sigma^2)$ ; quantities with a uniform distribution in the interval  $[a, b]$  are denoted  $U(a, b)$ .

Symbol	Definition	Spectral Density
$u_{a,f}$	Acceleration model process noise $(\text{m/s}^2)^2/\text{s}$	1.0
$u_{\omega_b,f}$	Rate gyro bias model process noise $(\text{rad/s})^2/\text{s}$	$2.0821 \times 10^{-11}$
$u_{\Theta_b,f}$	Compass bias process noise $(\text{rad}^2/\text{s})$	$3.046 \times 10^{-6}$
$u_{\Theta_b,f}$	GPS-based heading bias process noise $(\text{rad}^2/\text{s})$	$3.046 \times 10^{-10}$
$u_{S_b,f}$	Odometer scale factor bias process noise $(\text{m/pulse})^2/\text{s}$	$(0.0033S_o)^2 \frac{2}{\tau_{S_b,f}}$
$u_{x,f}$	GPS position error process noise with SA on $(\text{m}^2/\text{s})$	$7.63 \times 10^{-3}$
$u_{x,f}$	GPS position error process noise with SA off $(\text{m}^2/\text{s})$	$1.15 \times 10^{-3}$
$u_{\lambda,f}$	DGPS bias process noise $(\text{m}^2/\text{s})$	$8.4 \times 10^{-4}$
$u_{\xi,f}$	DGPS bias process noise $(\text{m}^2/\text{s})$	$1.85 \times 10^{-4}$

Table A.1: Spectral densities of process noise parameters in Kalman filter model equations

Symbol	Definition	Value
$\tau_{a,f}$	Acceleration model time constant (s)	3.0
$\tau_{S_b,f}$	Odometer scale factor bias time constant (s)	1800
$\beta_{x,f}$	Stand-alone GPS error parameter (1/s)	0.0165
$\tau_{\lambda,f}$	DGPS bias time constant (s)	321
$\tau_{\xi,f}$	DGPS bias time constant (s)	10800
$K_{S_b}$	Odometer speed error rate (m/(pulse-m/s))	$9.0 \times 10^{-5}$

Table A.2: Values for various constant parameters in Kalman filter model equations

Symbol	Definition	RMS Value
$v_{\omega,f}$	Rate gyro measurement noise (volts)	$4.71 \times 10^{-4}$
$v_{p_x,f}$	X-position measurement noise (m)	1.4
$v_{p_y,f}$	Y-position measurement noise (m)	1.4
$v_{odom,f}$	Odometer quantization noise (m/pulse)	$\frac{S_o}{3}$
$v_{\Theta,f}$	Compass measurement noise (rad)	0.0349
$v_{\Theta,f}$	GPS-based heading measurement noise (rad)	0.0349
$v_{m,f}$	Additive noise in map "measurement" (m)	1.0

Table A.3: RMS values for measurement noise parameters in Kalman filter measurement equations

Symbol	Definition	Spectral Density
$u_{S_b,r}$	Odometer scale factor bias process noise (m/pulse) <sup>2</sup> /s	$\frac{2(0.001S_o)^2}{\tau_{S_b,r}}$ to $\frac{2(0.0033S_o)^2}{\tau_{S_b,r}}$
$u_{\alpha_b,r}$	Rate gyro bias process noise (rad/s <sup>2</sup> ) <sup>2</sup> /s	$2.94 \times 10^{-17}$
$u_{\eta,r}$	Compass bias process noise (rad <sup>2</sup> /s)	0.0285
$u_{x,r}$	GPS position error process noise with SA on (m <sup>2</sup> /s)	$7.63 \times 10^{-3}$
$u_{x,r}$	GPS position error process noise with SA off (m <sup>2</sup> /s)	$1.15 \times 10^{-3}$
$u_{\lambda,r}$	DGPS bias process noise (m <sup>2</sup> /s)	$8.4 \times 10^{-4}$
$u_{\xi,r}$	DGPS bias process noise (m <sup>2</sup> /s)	$1.85 \times 10^{-4}$

Table A.4: Spectral densities of process noise parameters in reference system equations

Symbol	Definition	Value
$\tau_{S_b,r}$	Odometer scale factor bias time constant (s)	1800
$K_{S_b}$	Odometer speed error rate (m/(pulse-m/s))	$9.0 \times 10^{-5}$
$\varphi_{b,r}$	Constant component of rate gyro bias (rad)	$N(0, 1.39 \times 10^{-6})$
$\beta_{\alpha_b,r}$	Rate gyro bias drift parameter (1/s)	$2.98 \times 10^{-4}$
$\theta_{o,r}$	Constant component of compass bias (rad)	$N(0, 0.00762)$
$\theta_{o,r}$	Constant component of GPS-based heading bias (rad)	$N(0, 0.00762)$
$A, B, C, D$	Compass bias parameters (rad)	$U(0, 0.0872)$
$\beta_{\eta,r}$	Compass bias error parameter (1/s)	0.715
$\beta_{x,r}$	Stand-alone GPS error parameter (1/s)	0.0165
$\tau_{\lambda,r}$	DGPS bias time constant (s)	321
$\tau_{\xi,r}$	DGPS bias time constant (s)	10800
$K_T$	True rate gyro scale factor (rad/s/volt)	0.7854

Table A.5: Values for various constant parameters in reference system model equations

Symbol	Definition	RMS Value
$v_{\omega,r}$	Rate gyro measurement noise (rad/s)	$4.71 \times 10^{-4}$
$v_{p_x,r}$	X-position measurement noise (m)	1.4
$v_{p_y,r}$	Y-position measurement noise (m)	1.4
$v_{odom,r}$	Odometer quantization noise (m/pulse)	$\frac{S_o}{3}$
$v_{\Theta,r}$	Compass measurement noise (rad)	0.0349
$v_{\Theta,r}$	GPS-based heading measurement noise (rad)	0.0349
$v_{m,r}$	Additive noise in map “measurement” (m)	0.0

Table A.6: RMS values for measurement noise parameters in reference system measurement equations

# Bibliography

- [1] D. Alsip et al. The Coast Guard's differential GPS program. *Navigation: Journal of the Institute of Navigation*, 39(4):345–361, 1992.
- [2] Murata Erie North America. Gyrostar piezoelectric vibrating gyroscope, 1993. Specification sheets for Part no. ENX-0011.
- [3] Robert B. Asher and Richard M. Reeves. Performance evaluation of suboptimal filters. *IEEE Transactions on Aerospace and Electronic Systems*, AES-11(3):400–405, 1975.
- [4] B. Barshan and H. Durrant-Whyte. Inertial navigation systems for mobile robots. *IEEE Transactions On Robotics and Automation*, 11(3):328–342, 1995.
- [5] C. Beatty. Land vehicle navigation—from concept to production. In *Proceedings of the First International Symposium on Land Vehicle Navigation*, pages 9.1–9.17, 1984.
- [6] R. Borcherts et al. Database accuracy effects on vehicle positioning as measured by the certainty factor. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 291–294, 1993.
- [7] F. Chenavier and J. Crowley. Position estimation for a mobile robot using vision and odometry. In *The Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2588–2593, 1992.

- [8] Ronald R. Clark and Marvin A. Needler. Performance and adjoint sensitivity analysis of the navstar global positioning system. In *The NAECON Record*, pages 834–840, 1976.
- [9] W.C. Collier. In-vehicle route guidance systems using map matched dead reckoning. In *The Proceedings of the IEEE Position, Location and Navigation Symposium*, pages 359–363, 1990.
- [10] Joseph A. DAppolito et al. Error analysis of hybrid aircraft inertial navigation systems. In *The Proceedings of the AIAA Guidance and Control Conference*, pages 1–10, Aug 1972. AIAA paper no. 72-848.
- [11] Dee Ann Divis. GLONASS emerges; change in ISNS game plan. *GPS World Magazine*, 7(5):12, May 1996.
- [12] Systron Donner. Gyrochip horizon solid state rate gyroscope, 1994. Specification sheets for Part no. QRS14-00100-102.
- [13] U.S. DOT and ITS America. *National Program Plan for Intelligent Transportation Systems, Final Draft*. U.S. Department of Transportation, 1994.
- [14] Per Enge et al. Coverage of DGPS/radiobeacons. *Navigation: Journal of the Institute of Navigation*, 39(4):363–381, 1992.
- [15] Per Enge et al. DGPS/radiobeacon field trials comparing type 1 and 9 messaging. *Navigation: Journal of the Institute of Navigation*, 40(4):395–408, 1993.
- [16] R. L. French. The evolution of automobile navigation systems in Japan. In *The 49th Annual Meeting of the Institute of Navigation*, Jun 1993.
- [17] R. L. French and G. M. Lang. Automatic route control system. *IEEE Transactions on Vehicular Technology*, VT-22(2):36–41, 1973.
- [18] Ran Gazit. Aircraft surveillance and collision avoidance using GPS. Phd thesis, Stanford University, Stanford, California, 1996. Department of Aeronautical and Astronautical Engineering.

- [19] Arthur Gelb et al. *Applied Optimal Estimation*. MIT Press, Cambridge, Massachusetts, 1974.
- [20] Glen Gibbons. GPS and rail safety; wireless 911; budget helper. *GPS World Magazine*, 7(4):12, Apr 1996.
- [21] Robert E. Griffin and Andrew P. Sage. Large and small scale sensitivity analysis of optimum estimation algorithms. *IEEE Transactions on Automatic Control*, AC-13(4):320–328, Aug 1968.
- [22] U.S. Coast Guard. Differential gps, 1993. Brochure.
- [23] U.S. Coast Guard. Radiobeacon facts and figures, 1993. Brochure.
- [24] C. Harris. Prototype for a land based automatic vehicle location and navigation system. Master's thesis, University of Calgary, Calgary, Alberta, Canada, 1989. Report No. 20033, Department of Surveying Engineering.
- [25] C. Harris and E. Krakiwsky. AVLN 2000 automatic vehicle location and navigation system. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 126–130, 1989.
- [26] S. Honey et al. Vehicle navigational system and method, Jan 1989. U.S. Patent 4,796,191. Assigned to Etak, Inc.
- [27] Liang-Jong huang and Wei-Wen Kao. Fuzzy logic based algorithm for displaying travel path, 1994. Zexel U.S.A. internal report.
- [28] T. Hunter and J. Ashjaee. Land navigation and fleet management with gps, loran, and dead reckoning sensors. In *The Proceedings of the IEEE Position, Location and Navigation Symposium*, pages 54–60, 1988.
- [29] H. Ikeda et al. Sumitomo electric's navigation systems for private automobiles. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, volume 1, pages 451–462, 1991.

- [30] K. Ishikawa et al. Map navigation software of the electro-multivision of the '91 toyota soarer. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, volume 1, pages 463–473, 1991.
- [31] F. Iwaki et al. Recognition of vehicle's location for navigation. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 131–138, 1989.
- [32] R. Janc. Consideration of the various error sources in a practical automatic vehicle location system. In *The Proceedings of the IEEE Vehicular Technology Conference*, pages 277–284, 1984.
- [33] M. Kakihara et al. Vehicle navigation apparatus employing node selection, comparison and elimination techniques, Apr 1992. U.S. Patent 5,109,344. Assigned to Mazda Motor Corporation.
- [34] D. King. Landfall, a high resolution automatic vehicle-location system. *GEC Journal of Science and Technology*, 45(1):34–44, 1978.
- [35] D. Klimmek. Valuation of land vehicle navigation systems. In *Proceedings of the First International Symposium on Land Vehicle Navigation*, pages 21.1–21.14, 1984.
- [36] Wei-Wen Koa. Apparatus for correcting vehicular compass heading with the aid of the global positioning system, Aug 1994. U.S. Patent 5,339,246. Assigned to Zexel Corporation Diahatsu-Nissan.
- [37] F. Komura et al. Navigation system and method using map data, May 1994. U.S. Patent 5,311,173. Assigned to Hitachi, Ltd.
- [38] E. Krakiwsky et al. The intelligent vehicle-highway navigation systems database. Software program from Intelligent Databases International Ltd, Version 4.6.
- [39] E. Krakiwsky, C. Harris, and R. Wong. A kalman filter for integrating dead reckoning, map matching and gps positioning. In *The Proceedings of the IEEE Position, Location and Navigation Symposium*, pages 39–46, 1988.

- [40] K. Kumar et al. Emerging low(er) cost inertial sensors. In *The 2nd St. Petersburg International Conference on Gyroscopic Technology and Navigation*, May 1995.
- [41] J. Last and C. Scholefield. The combined use of low frequency radio and dead reckoning for automatic vehicle location. In *Proceedings of the First International Symposium on Land Vehicle Navigation*, pages 14.1–14.19, 1984.
- [42] D. Leber. A systems analysis of the impact of navigation instrumentation on-board a mars rover, based on a covariance analysis of navigation performance. Master's thesis, MIT, 1992. Department of Aeronautics and Astronautics.
- [43] T. Lezniak, R. Lewis, and R. McMillen. A dead reckoning/map correlation system for automatic vehicle tracking. *IEEE Transactions on Vehicular Technology*, VT-26(1):47–60, 1977.
- [44] S. Liu, Z. Zhang, and J. Hung. A high accuracy magnetic heading system composed of fluxgate magnetometers and a microcomputer. In *The Proceedings of the IEEE National Aerospace and Electronics Conference*, volume 1, pages 148–152, 1989.
- [45] Arther Louis. Sony device keeps your bearings. *The San Francisco Chronicle*, page C4, 1996. Oct. 1.
- [46] S. Matsuzaki. Apparatus for estimating current heading using magnetic and angular velocity sensors, Aug 1993. U.S. Patent 5,235,514. Assigned to Sumitomo Electric Industries, Ltd.
- [47] F. Ken Ikeda Morisue. Evaluation of map-matching techniques. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 23–28, 1989.
- [48] J. Muraszko. Pathfinder—an automatic vehicle location and status monitoring system. In *Proceedings of the Second International Symposium on Land Vehicle Navigation*, pages 8.1–8.12, 1989.

- [49] D. Neculescu, J. Sasiadek, and D. Green. Fusion of inertial and kinematic navigation systems for autonomous vehicles. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 462–465, 1993.
- [50] N. Oki et al. Portable vehicle navigation system (nv-1): Its features and operability. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 482–485, 1993.
- [51] H Oshizawa and C. Collier. Description and performance of navmate, an in-vehicle route guidance system. In *The Proceedings of the American Control Conference*, pages 782–787, 1990.
- [52] M. Ostertag. Improved localisation for traffic flow control. Submitted to Periodica Polytechnica for publication, 1994.
- [53] B.W. Parkinson et al., editors. *Global Positioning System : Theory and Applications*, volume 2. American Institute of Aeronautics and Astronautics, Inc., Washington, D.C., 1996.
- [54] B.W. Parkinson et al., editors. *Global Positioning System : Theory and Applications*, volume 1. American Institute of Aeronautics and Astronautics, Inc., Washington, D.C., 1996.
- [55] J. Ritland and K. Spalding. Impact of inertial system quality on gps-inertial performance in a jamming environment. In *AIAA Conference*, pages 1459–1467, 1987. AIAA Paper 87-2594.
- [56] F. Sausen. Position location methods for road bound mobiles by dead reckoning with support of stored digital city-map information. In *Proceedings of the First International Symposium on Land Vehicle Navigation*, pages 6.1–6.18, 1984.
- [57] Craig Scott. Improved gps positioning for motor vehicles through map-matching. In *The Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation*, pages 1391–1400, 1994.

- [58] Jan Soderkvist. Piezoelectric beams and vibrating angular rate sensors. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 38(3):271–280, May 1991.
- [59] Jan Soderkvist. Micromachined gyroscopes. *Sensors and Actuators A*, 43:65–71, 1994.
- [60] British Aerospace Systems and Equipment. The micropulse rate gyroscope. Specification sheets from manufacturer.
- [61] H. Tamai and T. Abe. Navigation system for guiding a vehicle along a pre-computed optimal route, Mar 1994. U.S. Patent 5,291,414. Assigned to Zexel Corporation Diahatsu-Nissan.
- [62] M. L. G. Thoone. Carin, a car information and navigation system. *Philips Technical Review*, 43(11/12):317–327, 1987.
- [63] H. Tsuji et al. Evaluation of location system combining a gps receiver with inertial sensor. In *The Proceedings of the Vehicle Navigation and Information Systems Conference*, pages 645–649, 1991.
- [64] Y. Ueyama and N. Yasui. Vehicle in-situ locating apparatus, Oct 1991. U.S. Patent 5,060,162.
- [65] H. von der Hardt et al. A method of mobile robot localisation by fusion of odometric and magnetometric data. *International Journal of Advanced Manufacturing Technology*, 9:65–69, 1994.
- [66] C. Wang. Location estimation and uncertainty analysis for mobile robots. In *The Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1230–1235, 1988.
- [67] L. Whitcomb. Using low cost magnetic sensors on magnetically hostile land vehicles. In *The Proceedings of the IEEE Position, Location and Navigation Symposium*, pages 34–38, 1988.

- [68] S. Yokoyama and A. Nanba. Vehicular navigation apparatus, Aug 1991. U.S. Patent 5,043,902. Assigned to Aisin Aw Company, Ltd.
- [69] E. Zeevi. Vehicle navigation system, Oct 1989. U.S. Patent 4,878,170.
- [70] R. Zickel and N. Nehemia. Gps aided dead reckoning navigation. In *The Proceedings of the ION National Technical Meeting*, pages 577–586, 1994.