

## **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



---

**PROBABILISTIC ENGINEERING DESIGN  
OPTIMIZATION: APPLICATIONS TO  
SPACECRAFT AND NAVIGATION SYSTEMS**

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

**By  
Samuel Phillip Pullen**

**June 1996**

---

**UMI Number: 9630375**

**Copyright 1996 by  
Pullen, Samuel Phillip**

**All rights reserved.**

---

**UMI Microform 9630375  
Copyright 1996, 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**

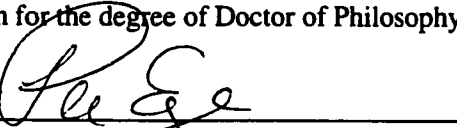
---

**© Copyright by Samuel Pullen 1996**  
**All Rights Reserved**

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

  
Bradford W. Parkinson  
(Principal Adviser)

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

  
Per K. Enge


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

  
N. Jeremy Kasdin

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

  
Jonathan P. How

Approved for the University Committee on Graduate Studies:

  
\_\_\_\_\_

## ABSTRACT

In addition to designing engineering systems for optimal performance, developing systems that are *robust* to possible failures and/or non-ideal operating conditions is of great importance. In modern engineering practice, this is normally accomplished by incorporating conservative performance margins such that projected “worst-case” outcomes can be accommodated. In aerospace, safety or robustness considerations often dominate the design process, but this can lead to over-designed systems, lengthy development programs, and expensive final products.

Provided that decision makers accept the unavoidable possibility of failure, a superior approach based on *system uncertainty* and *user utility* modeling exists. System performance uncertainty, including unknown parameters and possible unit failures, is modeled using the best available information. The user’s utility function, of arbitrary mathematical form, expresses the relative “goodness” of all possible outcomes. Once the axioms of decision theory are met, a maximum-utility search among the design space determines the optimal solution. Because these problems do not conform to standard mathematical assumptions, there is no guarantee of finding the best possible answer, but modern computer search techniques now provide the capability to converge toward the global optimum in reasonable time. As with traditional systems engineering, the optimal-decision process is *iterative*, since the computer search results are reviewed by designers who can further develop their risk and utility models.

This approach has been successfully applied to several system design tasks in this thesis. A tutorial aircraft landing control problem is used to illustrate the basic procedure and to demonstrate the safety improvements that are possible when controllers are designed with system failure modes in mind. Applications to spacecraft design are then developed. New models for spacecraft reliability prediction have been combined with mission utility functions to predict the overall mission reliability of the *Gravity Probe-B* (GP-B) spacecraft and to find improved redundancy architectures for space vehicles.

Uncertainty-based optimization has also been demonstrated to significantly improve the process of *Receiver Autonomous Integrity Monitoring* (RAIM) for Global Positioning System (GPS) navigation users. Similar uncertainty models applied to augmented Differential GPS (DGPS) systems can predict overall performance and

integrity for large regions of users. Combining these with top-level objective models allows augmented GPS architectures to be optimized iteratively, as the latest experimental data *updates* the risk model and motivates additional system improvements.

## ACKNOWLEDGMENTS

There are many people I must thank for their help in my thesis research and in my graduate life in general. First, of course, is my advisor, Prof. Brad Parkinson, who gave me the freedom to pursue my own course of research and encouraged me to push forward in new directions. Prof. Per Enge also gave critical guidance and opened new doors for my research. Many people in the GPS research group helped me adapt my work to navigation problems by availing me of their incredible combined knowledge in GPS navigation systems. Within this group, I would especially like to thank Boris Pervan, Dr. Todd Walter, and Dr. Changdon Kee.

When I first came to Stanford, Prof. Emery Reeves motivated me to think in terms of new approaches to spacecraft design. I learned a great deal from him. After I came to work on the GP-B spacecraft project, Gaylord Green and Jeremy Kasdin provided key guidance, as did Raffi Atzmon and Ben Taller who were responsible for Quality Assurance during my research tenure.

Many people outside the Aero/Astronautics Department encouraged and supported my work on non-traditional optimization approaches. In particular, I would like to thank Prof. Ross Shachter, Prof. M.E. Pate-Cornell, and Prof. Peter Glynn. Many friends of mine also provided insight into software approaches, including Jeremy Levitt and Chris Wilson. Many other friends of mine, including Ken Chin, Ken Honer, Grace Chen, and Dave Shen, gave me ideas from time to time, often without knowing it, and they were there during both the pleasant and the difficult times I faced during my graduate study. I would also like to thank the people I met during my study of Japanese and my time in Japan, including Dr. Shuichi Sasa and Dr. Yoshikazu Miyazawa of the National Aerospace Laboratory, Dr. Takashi Shimomura of Osaka University. In addition, the Japanese *senseis* at Stanford encouraged my study of Japanese language and culture despite the time constraints of my thesis research.

Finally, I would especially like to thank my parents, Allen and Roz Pullen, my brother Adam, and my friends Sean Bodkin and Brian Zucker, just for being the best

family and friends a person could ask for. My girlfriend, Peggy Brister, has recently come into my life and has made all the difference as I completed my research. I hope this thesis is a fitting tribute to their constant support of me.

## **Editorial Notes**

This thesis attempts to use gender-neutral phrasing where possible. However, in some cases, the masculine pronoun may be used in the traditional grammatical sense to refer to both sexes. Please note that there is no intent to limit the discussion to males in any aspect of this research.

The quotes that are used to begin each chapter are taken from a variety of sources. They are intended to provide a flavor for each successive topic from the wider world in which the concepts presented here are meant to be applied. The opinions voiced within these quotes do not always agree with the points raised in the text. In some cases, they are included to illustrate the source of the disagreements about which the thesis is concerned and to provide a global perspective on what is at stake.

Unlike most engineering theses, much of the text in this document is argumentative because it draws support for its theories from technical results that could be interpreted in several ways. There are no “proofs” to be found here; the uncertainty models constructed and simulated for each application illustrate that no deterministic bounds or conclusions about system performance can be drawn when the underlying mathematical models and parameters contain questionable assumptions. Simulation results are evaluated to determine if uncertainty-based methods offer significant improvements over deterministic “worst case” design approaches.

However, the problems addressed in this thesis are subsets of the overall development process, and the probability models and objective functions used in each case are also open to question. The conclusions presented here thus require significant qualitative discussion to bolster their case while fairly considering alternative viewpoints. Therefore, the results of this thesis are intended to be taken as a demonstration of both the feasibility of the new decision-based design paradigm introduced here and its potential to be applied to an entire development program in a unified way. More work will be required before the limits of this potential are well understood. In the meantime, this will remain a controversial topic.

## **Table of Contents**

<b>Abstract .....</b>	<b>iv</b>
<b>Acknowledgments.....</b>	<b>vi</b>
<b>Editorial Notes.....</b>	<b>viii</b>
<b>Table of Contents .....</b>	<b>ix</b>
<b>List of Figures.....</b>	<b>xvi</b>
<b>List of Tables .....</b>	<b>xix</b>
<b>List of Acronyms .....</b>	<b>xxi</b>
<b>1.0 INTRODUCTION.....</b>	<b>1</b>
<b>1.1 The Development of Systems Engineering Practice .....</b>	<b>1</b>
<b>1.2 Decision Making Under Uncertainty .....</b>	<b>3</b>
<i>1.2.1 Uncertainty Modeling .....</i>	<i>4</i>
<i>1.2.2 User Preference Function Modeling.....</i>	<i>10</i>
<b>1.3 Applications to Engineering Systems Design.....</b>	<b>13</b>
<b>1.4 The Gravity Probe-B Experiment.....</b>	<b>16</b>
<b>1.5 The Global Positioning System (GPS) .....</b>	<b>18</b>
<b>1.6 A Survey of the Relevant Literature.....</b>	<b>19</b>
<i>1.6.1 Decision-Making under Uncertainty.....</i>	<i>20</i>
<i>1.6.2 Spacecraft Reliability Modeling .....</i>	<i>20</i>
<i>1.6.3 Global Optimization.....</i>	<i>21</i>
<i>1.6.4 Spacecraft Optimization under Uncertainty.....</i>	<i>22</i>
<i>1.6.5 Control System Optimization under Uncertainty.....</i>	<i>22</i>
<i>1.6.6 Global Positioning System: Introduction and RAIM Concepts.....</i>	<i>23</i>
<i>1.6.7 Global Positioning System: Probability Models and User Value             Functions .....</i>	<i>24</i>
<i>1.6.8 Global Positioning System: WAAS .....</i>	<i>25</i>
<b>1.7 Contributions of this Research .....</b>	<b>26</b>

<b>2.0 DECISION ANALYSIS TUTORIAL: A CANONICAL AIRCRAFT</b>	
<b>CONTROL PROBLEM</b>	29
<b>2.1 Linear Quadratic Control Theory and Limitations</b>	30
2.1.1 <i>State-Space Dynamic Models</i>	30
2.1.2 <i>Linear Quadratic Gaussian (LQG) Controller Formulation</i>	33
2.1.3 <i>Robust Control Techniques</i>	34
<b>2.2 Aircraft Dynamic Model and LQG Controller</b>	36
2.2.1 <i>Longitudinal 747 Aircraft Dynamic Model</i>	36
2.2.2 <i>LQG Optimal Controller for Aircraft Precision Approach</i>	36
<b>2.3 Performance Uncertainty Model</b>	39
2.3.1 <i>LQG Controller Sensitivity Analysis</i>	39
2.3.2 <i>Prior Probability Model for WAAS/Wind Uncertainty</i>	41
2.3.3 <i>Simulation of Prior Probability Model</i>	43
<b>2.4 Value Model and Optimal Controller Search</b>	43
2.4.1 <i>Accident Risk Cost Assessment</i>	43
2.4.2 <i>Combined Risk and LQR Cost Function</i>	45
2.4.3 <i>Optimal Search using Simulated Annealing</i>	47
<b>2.5 Baseline and Optimal Controller Results</b>	48
2.5.1 <i>Optimal Controller for PPM and Risk Model</i>	48
2.5.2 <i>Use of LQR Terminal Controllers</i>	49
2.5.3 <i>Simulated Annealing Solution for the Ideal Case</i>	50
<b>2.6 Conclusions and Future Work</b>	53
<b>3.0 SPACECRAFT RELIABILITY MODELING</b>	56
<b>3.1 A Working Definition of “Reliability”</b>	56
<b>3.2 Current Reliability Engineering Practice</b>	58
<b>3.3 Weaknesses in Traditional Reliability Analysis</b>	59
<b>3.4 New Concepts: Time-Dependent Failure Rates and Parameter Uncertainty</b>	63

3.4.1	<i>Global Optimization</i> .....	63
3.4.2	<i>Failure Rate Parameter Uncertainty</i> .....	64
3.4.3	<i>Application to Gravity Probe-B Spacecraft Bus</i> .....	66
3.5	<b>New Concepts: FMECA-based Reliability Predictions</b> .....	72
3.5.1	<i>Introduction to FMECA Methodology</i> .....	72
3.5.2	<i>FMECA Reliability Sampling Approach</i> .....	73
3.5.3	<i>Results for GP-B Payload Dewar</i> .....	76
3.6	<b>Conclusions and Further Work</b> .....	77
4.0	<b>SPACECRAFT PROBABILISTIC DESIGN OPTIMIZATION</b> .....	80
4.1	<b>Deterministic Reliability Optimization</b> .....	81
4.2	<b>Gravity Probe-B Spacecraft Bus Optimization</b> .....	81
4.2.1	<i>Optimization Problem Overview</i> .....	81
4.2.2	<i>Probabilities of Non-Controllable Events</i> .....	82
4.2.3	<i>Decision-Maker Utility Functions</i> .....	84
4.2.3.1	<u>LMSC Value Model</u> .....	84
4.2.3.2	<u>Stanford Value Model</u> .....	86
4.2.4	<i>Constraint Modeling</i> .....	87
4.2.5	<i>Final Objective Function Evaluation</i> .....	89
4.2.6	<i>Value Models for the Risk-Averse Case</i> .....	90
4.3	<b>New Design Optimization Approaches</b> .....	91
4.3.1	<i>Simulated Annealing (SA) Algorithm</i> .....	92
4.3.1.1	<u>Methodology</u> .....	92
4.3.1.2	<u>Problem Encoding and SA Perturbations</u> .....	94
4.3.2	<i>Genetic Algorithms</i> .....	97
4.3.3	<i>Optimization Process Flow Chart</i> .....	102
4.4	<b>Spacecraft Bus Optimization Results</b> .....	103
4.4.1	<i>Baseline Design and Problem Formulation</i> .....	103
4.4.2	<i>Design Optimization Results</i> .....	105
4.4.3	<i>Optimal Design Value Evolutions</i> .....	106

4.4.4	<i>Optimal Spacecraft and Mission Reliability .....</i>	108
4.4.5	<i>Objective Function Evaluation Uncertainty.....</i>	110
4.4.6	<i>Variant Solution with Two Batteries .....</i>	112
4.5	<b>Spacecraft Design: Conclusions and Significance .....</b>	114
5.0	<b>GPS SYSTEM INTEGRITY MONITORING.....</b>	117
5.1	<b>Global Positioning System (GPS) Performance Simulation.....</b>	117
5.1.1	<i>GPS Satellite Constellation Almanac.....</i>	117
5.1.2	<i>User Locations and Position Calculations .....</i>	120
5.1.3	<i>SPS Ranging and Position Accuracies.....</i>	124
5.1.4	<i>Performance Evaluation Concepts and Terms .....</i>	124
5.1.4.1	<u>Performance.....</u>	125
5.1.4.2	<u>Availability.....</u>	126
5.1.4.3	<u>Continuity .....</u>	126
5.1.4.4	<u>Integrity .....</u>	127
5.2	<b>GPS RAIM Integrity Monitoring Approaches.....</b>	129
5.2.1	<i>Mathematical Background .....</i>	129
5.2.2	<i>Choosing Thresholds for RAIM Statistics .....</i>	131
5.2.3	<i>Weaknesses of Traditional RAIM .....</i>	133
5.3	<b>A New RAIM Approach Using Cost-Based Optimization.....</b>	134
5.3.1	<i>GPS System Faults: Prior Probability Model (PPM).....</i>	135
5.3.2	<i>Simulation of GPS System under PPM .....</i>	137
5.3.3	<i>Non-Precision Approach using SPS.....</i>	139
5.3.4	<i>Risk-Based RAIM Cost Model .....</i>	139
5.3.5	<i>Cost-Based Threshold Optimization .....</i>	142
5.4	<b>Simulation and Optimal Threshold Results .....</b>	144
5.4.1	<i>Spacecraft and Receiver Reliability.....</i>	144
5.4.2	<i>GPS Geometry Results .....</i>	145
5.4.3	<i>Optimal RAIM Results for SPS Users .....</i>	146
5.4.4	<i>RAIM Optimal Result Summary.....</i>	151

5.4.5	<i>Results for Improved SPS Ranging Accuracy</i> .....	153
5.5	<b>Comparisons with Traditional RAIM</b> .....	155
5.6	<b>Bayesian Updating and GIC Potential</b> .....	158
5.7	<b>Summary and Future Developments</b> .....	161
6.0	<b>WIDE AREA AUGMENTATION SYSTEM IONOSPHERE INTEGRITY ..</b>	163
6.1	<b>Introduction to the Wide Area Augmentation System (WAAS)</b> .....	163
6.2	<b>Rare-Event Ionosphere Decorrelation Models</b> .....	166
6.2.1	<i>Introduction</i> .....	166
6.2.2	<i>Breakdown of WAAS Error Sources</i> .....	167
6.2.3	<i>Spatial Decorrelation Error Models</i> .....	168
6.2.3.1	<u><i>Ionosphere Sampling Model</i></u> .....	170
6.2.3.2	<u><i>WAAS Ionospheric Sampling Procedure</i></u> .....	173
6.2.3.3	<u><i>Troposphere Sampling Model</i></u> .....	175
6.2.4	<i>Simulation of DGPS Corrections</i> .....	176
6.3	<b>Ionosphere Decorrelation: Simulation Results</b> .....	177
6.3.1	<i>Spatial Decorrelation Error Results</i> .....	177
6.3.2	<i>Position Error Results</i> .....	179
6.4	<b>WAAS Integrity Methods and Results</b> .....	181
6.4.1	<i>Residuals Threshold Test Algorithm</i> .....	181
6.4.2	<i>WAAS User Cost Model</i> .....	181
6.4.3	<i>WAAS User Integrity Results</i> .....	182
6.5	<b>WAAS Isolation/Correction Algorithms</b> .....	184
6.6	<b>Rare-Event Ionosphere: Conclusions</b> .....	188
7.0	<b>WIDE AREA AUGMENTATION SYSTEM COVERAGE PREDICTION...</b>	190
7.1	<b>Introduction</b> .....	190
7.2	<b>Covariance Analysis of Normal WAAS Performance</b> .....	191
7.2.1	<i>Overview of WAAS Simulation Approach</i> .....	191
7.2.2	<i>RMTSA Ranging Error Model</i> .....	193

7.2.3	<i>Clock/Ephemeris Covariance Prediction</i>	196
7.2.3.1	<u>Covariance Prediction Equations</u>	196
7.2.3.2	<u>Computer Implementation Procedure</u>	200
7.2.4	<i>Ionospheric Grid Error Projection Model</i>	201
7.2.4.1	<u>WRS Covariance</u>	202
7.2.4.2	<u>WMS Covariance</u>	204
7.2.4.3	<u>User Covariance</u>	205
7.2.4.4	<u>Numerical Challenges</u>	206
7.3	<b>WAAS Coverage Prediction Results</b>	208
7.3.1	<i>Output Histograms and Computed Results</i>	208
7.3.2	<i>Results for Stanford WAAS Network</i>	210
7.3.3	<i>Stanford WAAS Plus Hawaii WRS</i>	213
7.3.4	<i>FAA NSTB Results</i>	214
7.3.4.1	<u>Original NSTB Network Results</u>	215
7.3.4.2	<u>Revised CONUS NSTB Results</u>	218
7.3.4.3	<u>NSTB Plus “Four Corners”</u>	220
7.3.4.4	<u>NSTB Plus Canada and Mexico</u>	222
7.3.5	<i>Summary of Results</i>	224
7.4	<b>Conclusions and Further Work</b>	225
8.0	<b>WAAS OVERALL SYSTEM EVALUATION: A EUROPEAN EXAMPLE</b>	227
8.1	<b>Introduction</b>	227
8.2	<b>Preliminary Design for a European WAAS</b>	230
8.2.1	<i>European WAAS Developments</i>	230
8.2.2	<i>Results for Baseline European WAAS</i>	231
8.2.3	<i>Single-Frequency WRS Option</i>	233
8.3	<b>Coverage-Based Integrity Simulations</b>	234
8.3.1	<i>Background and User Cost Model</i>	234
8.3.2	<i>Revised WAAS Rare-Event Probability Models</i>	235
8.3.3	<i>Integrity Simulation Procedure</i>	237

8.3.4	<i>Integrity Results for 4-WRS European WAAS</i> .....	239
8.3.5	<i>Integrity Simulation "Max Slope" Geometry Variant</i> .....	240
<b>8.4</b>	<b>Genetic Algorithm Optimization Model</b> .....	244
8.4.1	<i>WAAS Network Design Encoding</i> .....	244
8.4.2	<i>GA Population Evolution Operators</i> .....	246
8.4.3	<i>GA Optimization Procedure</i> .....	247
<b>8.5</b>	<b>WAAS Network Objective Function</b> .....	249
8.5.1	<i>Population Multiplier</i> .....	250
8.5.2	<i>Network Acquisition Costs</i> .....	250
8.5.3	<i>User Benefit Estimates</i> .....	251
8.5.4	<i>Accuracy and Integrity Evaluation</i> .....	252
8.5.5	<i>Value of 4-WRS Baseline European WAAS</i> .....	253
<b>8.6</b>	<b>"First-Generation" WAAS Results</b> .....	253
<b>8.7</b>	<b>Conclusions and Further Work</b> .....	257
<b>9.0</b>	<b>SUMMARY AND RECOMMENDATIONS</b> .....	260
9.1	<b>Systems Design and the Decision Paradigm</b> .....	260
9.2	<b>Reliability-Based Spacecraft Design Optimization</b> .....	262
9.3	<b>GPS Performance Prediction and Optimization</b> .....	263
9.4	<b>The Future of Aerospace Design</b> .....	267
<b>APPENDIX A:</b>	<b>Axioms of Decision Theory and Required Assumptions</b> .....	270
<b>APPENDIX B:</b>	<b>Values and Tools for Longitudinal Aircraft Control Model</b> .....	276
B.1	<i>Matrix Values for 747 Longitudinal Control Problem</i> .....	276
B.2	<i>LQR Cost Determination from Covariance Matrix Propagation</i> .....	278
<b>BIBLIOGRAPHY</b> .....		279

## List of Figures

<b>Figure 1.1: Simple Probability Tree for Possibility “B”</b> .....	5
<b>Figure 1.2: Probability Tree for Relevant Distinctions A/B</b> .....	7
<b>Figure 1.3: Decision Tree for Problem of Distinctions A/B</b> .....	9
<b>Figure 1.4: Master Decision Tree for Spacecraft Design Process</b> .....	14
<b>Figure 1.5: GP-B Spacecraft</b> .....	17
<b>Figure 1.6: NAVSTAR Constellation Orbital Geometry</b> .....	19
<b>Figure 2.1: Terminal Deviation Accident Risk</b> .....	45
<b>Figure 2.2: Failure Approach Simulation Results</b> .....	49
<b>Figure 2.3: Closed-Loop Pole Location Comparison</b> .....	51
<b>Figure 3.1a: LMSC GP-B Reliability Block Diagram</b> .....	67
<b>Figure 3.1b: GP-B Reliability Block Diagram for Science/Control</b> .....	68
<b>Figure 3.2: GP-B Spacecraft Bus Reliability Uncertainty</b> .....	70
<b>Figure 3.3: GP-B Dewar FMECA Reliability Uncertainty</b> .....	77
<b>Figure 4.1: Simulated Annealing Flow Chart</b> .....	92
<b>Figure 4.2: Solution Perturbation Distribution</b> .....	96
<b>Figure 4.3: Canonical GA Operator Example</b> .....	99
<b>Figure 4.4: Spacecraft Bus Optimization Flow Chart</b> .....	102
<b>Figure 4.5a: LMSC Optimal Value Evolution</b> .....	107
<b>Figure 4.5b: Stanford Optimal Value Evolution</b> .....	107
<b>Figure 4.6: Penalty Costs for SA Runs</b> .....	108
<b>Figure 4.7a: Spacecraft Bus Reliability Density</b> .....	109
<b>Figure 4.7b: GP-B Mission Reliability Density</b> .....	109
<b>Figure 5.1: NAVSTAR Constellation Orbital Geometry</b> .....	118
<b>Figure 5.2: Optimal Threshold Selection Process</b> .....	143

<b>Figure 5.3</b>	<b>Distribution of Satellites in View.....</b>	<b>145</b>
<b>Figure 5.4:</b>	<b>HDOP Cumulative Probability .....</b>	<b>145</b>
<b>Figure 5.5:</b>	<b>Ranging and Position Error Densities.....</b>	<b>146</b>
<b>Figure 5.6:</b>	<b>Best RAIM Threshold Results.....</b>	<b>147</b>
<b>Figure 5.7:</b>	<b>Optimal RAIM Threshold Results .....</b>	<b>148</b>
<b>Figure 5.8:</b>	<b>Best RAIM Cost Results.....</b>	<b>150</b>
<b>Figure 5.9:</b>	<b>Chi-Square Comparison Results.....</b>	<b>157</b>
<b>Figure 6.1:</b>	<b>WAAS Network Layout Overview.....</b>	<b>164</b>
<b>Figure 6.2:</b>	<b>WAAS Integrity Simulation Flow Chart .....</b>	<b>174</b>
<b>Figure 6.3:</b>	<b>User Pierce Point and WMS Grid Geometry .....</b>	<b>176</b>
<b>Figure 6.4:</b>	<b>LADGPS Spatial Decorrelation Errors.....</b>	<b>178</b>
<b>Figure 6.5:</b>	<b>WAAS Ionosphere Decorrelation Error .....</b>	<b>178</b>
<b>Figure 6.6:</b>	<b>VDOP Cumulative Distributions .....</b>	<b>179</b>
<b>Figure 6.7:</b>	<b>LADGPS Position Error Distribution.....</b>	<b>179</b>
<b>Figure 6.8:</b>	<b>WAAS Position Error Distributions .....</b>	<b>180</b>
<b>Figure 6.9:</b>	<b>WAAS RAIM Optimal Thresholds.....</b>	<b>183</b>
<b>Figure 6.10:</b>	<b>WAAS Spatial Decorrelation with Isolation .....</b>	<b>186</b>
<b>Figure 6.11:</b>	<b>WAAS Position Errors with Isolation.....</b>	<b>186</b>
<b>Figure 7.1:</b>	<b>WAAS Covariance Overview .....</b>	<b>193</b>
<b>Figure 7.2:</b>	<b>Ionosphere Error Spatial Decorrelation .....</b>	<b>204</b>
<b>Figure 7.3:</b>	<b>Stanford WAAS 95% Position Accuracy.....</b>	<b>210</b>
<b>Figure 7.4:</b>	<b>Stanford WAAS 99% Position Accuracy.....</b>	<b>211</b>
<b>Figure 7.5:</b>	<b>Stanford WAAS User Availability wrt 4.1-m ILS Cat. I Requirement ..</b>	
	<b>.....</b>	<b>211</b>
<b>Figure 7.6:</b>	<b>Stanford WAAS 95% UIVE.....</b>	<b>212</b>
<b>Figure 7.7:</b>	<b>Stanford/Hawaii Position Error (<i>unsmoothed</i>) .....</b>	<b>213</b>
<b>Figure 7.8:</b>	<b>Stanford/Hawaii Position Error (<i>WRS carrier smoothing</i>).....</b>	<b>214</b>
<b>Figure 7.9:</b>	<b>FAA 18-WRS NSTB 95% Accuracy .....</b>	<b>215</b>

<b>Figure 7.10: FAA NSTB 3-D 95% Accuracy Plot .....</b>	<b>216</b>
<b>Figure 7.11: FAA NSTB 3-D 95% VDOP Plot .....</b>	<b>216</b>
<b>Figure 7.12: FAA NSTB 95% UIVE.....</b>	<b>217</b>
<b>Figure 7.13: Revised NSTB 95% Accuracy .....</b>	<b>218</b>
<b>Figure 7.14: Revised NSTB 95% UIVE.....</b>	<b>219</b>
<b>Figure 7.15: Revised NSTB 95% UDRE.....</b>	<b>219</b>
<b>Figure 7.16: 22-WRS Variant 95% Accuracy .....</b>	<b>220</b>
<b>Figure 7.17: 22-WRS Variant 3-D Accuracy Plot .....</b>	<b>221</b>
<b>Figure 7.18: NSTB + Canada/Mexico 95% Accuracy.....</b>	<b>222</b>
<b>Figure 7.19: Canada/Mexico 3-D Accuracy Plot .....</b>	<b>223</b>
<b>Figure 8.1: European WAAS 95% Accuracy .....</b>	<b>231</b>
<b>Figure 8.2: European WAAS 95% UIVE .....</b>	<b>232</b>
<b>Figure 8.3: European WAAS 95% UDRE.....</b>	<b>232</b>
<b>Figure 8.4: European WAAS Locations and Options.....</b>	<b>245</b>
<b>Figure 8.5: Genetic Algorithm Optimization Procedure.....</b>	<b>248</b>
<b>Figure 8.6: Accuracy of 5-WRS European Network .....</b>	<b>254</b>
<b>Figure 8.7: Accuracy of 6-WRS + GEO Network .....</b>	<b>254</b>
<b>Figure 8.8: UIVE for 6-WRS + GEO Network.....</b>	<b>255</b>
<b>Figure 8.9: UDRE for 6-WRS + GEO Network .....</b>	<b>255</b>
<b>Figure 9.1: A Future Aerospace Design Firm.....</b>	<b>268</b>
<b>Figure A.1: Probability Tree for Weather Problem .....</b>	<b>271</b>
<b>Figure A.2: Weather Problem Preference Probability Assignment.....</b>	<b>273</b>
<b>Figure A.3: Completed Decision Tree for Weather Problem.....</b>	<b>274</b>

## List of Tables

<b>Table 2.1:</b>	<b>LQG Sensitivity Analysis Results.....</b>	<b>39</b>
<b>Table 2.2:</b>	<b>WAAS Failure Model Summary .....</b>	<b>42</b>
<b>Table 3.1:</b>	<b>Prediction Factors by Mission Type .....</b>	<b>64</b>
<b>Table 3.2:</b>	<b>Failure Rate Variation Cases .....</b>	<b>66</b>
<b>Table 3.3:</b>	<b>FMECA Risk Uncertainty Classifications .....</b>	<b>74</b>
<b>Table 4.1:</b>	<b>SA Algorithm Parameters.....</b>	<b>93</b>
<b>Table 4.2:</b>	<b>GA Parameters .....</b>	<b>99</b>
<b>Table 4.3a:</b>	<b>LMSC Optimal Design Results .....</b>	<b>104</b>
<b>Table 4.3b:</b>	<b>Stanford Optimal Design Results .....</b>	<b>104</b>
<b>Table 4.4:</b>	<b>Reliability and Cost Improvement for Optimal Design.....</b>	<b>109</b>
<b>Table 4.5:</b>	<b>Stanford Optimal Result with Redundant Battery .....</b>	<b>112</b>
<b>Table 5.1:</b>	<b>Errors for Normal SPS Operation.....</b>	<b>124</b>
<b>Table 5.2:</b>	<b>Prior Probability Model Parameters.....</b>	<b>136</b>
<b>Table 5.3:</b>	<b>SPS RAIM Cost Parameters .....</b>	<b>140</b>
<b>Table 5.4:</b>	<b>Observed Failure Probabilities.....</b>	<b>144</b>
<b>Table 5.5:</b>	<b>HDOP Availability Thresholds.....</b>	<b>149</b>
<b>Table 5.6:</b>	<b>RAIM Simulation Output Summary .....</b>	<b>151</b>
<b>Table 5.7:</b>	<b>Revised Errors for Normal SPS Operation.....</b>	<b>153</b>
<b>Table 5.8:</b>	<b>RAIM Output Summary for no-SA Case .....</b>	<b>154</b>
<b>Table 6.1:</b>	<b>WAAS Error Source Summary .....</b>	<b>168</b>
<b>Table 6.2:</b>	<b>Ionosphere Deviations by Season .....</b>	<b>171</b>
<b>Table 6.3:</b>	<b>WAAS User RAIM Cost Parameters .....</b>	<b>182</b>
<b>Table 6.4:</b>	<b>WAAS RAIM Results Summary .....</b>	<b>183</b>
<b>Table 6.5:</b>	<b>SDIA Results Summary .....</b>	<b>187</b>

<b>Table 7.1:</b>	<b>One-Sigma RMTSA Errors.....</b>	<b>194</b>
<b>Table 8.1:</b>	<b>Baseline European WAAS User Integrity.....</b>	<b>239</b>
<b>Table 8.2:</b>	<b>Baseline User Integrity Using Max Slope.....</b>	<b>242</b>
<b>Table 8.3:</b>	<b>European WAAS Augmentation Options.....</b>	<b>245</b>
<b>Table 8.4:</b>	<b>Integrity for European WAAS Variants.....</b>	<b>256</b>
<b>Table 8.5:</b>	<b>European WAAS Requirements Comparison .....</b>	<b>258</b>

## **List of Acronyms**

CE	Certain Equivalent (decision theory)
CI	Confidence Interval (statistical)
CONUS	Continental United States
DEM	Design, Environment, and Manufacturing (class of spacecraft failures)
DGPS	Differential Global Positioning System
DoD, DOD	Department of Defense (U.S.)
DOP	Dilution of Precision (integrity monitoring)
FA	False Alarm
FAA	Federal Aviation Administration
FMECA	Failure Modes, Effects, and Criticality Analysis
FOC	Final Operational Capability
GA	Genetic Algorithm
GDOP	Geometry Dilution of Precision
GEO	Geosynchronous satellite
GP-B	Gravity Probe-B (Mission)
GPS	Global Positioning System
HDOP	Horizontal Dilution of Precision
ILS	Instrument Landing System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
LAAS	Local Area Augmentation System
LADGPS	Local Area DGPS
LMSC	Lockheed Missiles and Space Company (now part of Lockheed-Martin)
MCS	Master Control System (DoD)
MD	Missed Detection (integrity monitoring)
MOPS	Minimum Operational Performance Standards

<b>MTBF</b>	<b>Mean Time Between Failures</b>
<b>MTTR</b>	<b>Mean Time To Repair</b>
<b>NA</b>	<b>Non-Availability</b>
<b>NSTB</b>	<b>National Satellite Test Bed (WAAS)</b>
<b>OCS</b>	<b>Operational Control Segment (DoD)</b>
<b>ORD</b>	<b>Operational Requirements Document</b>
<b>PDOP</b>	<b>Position Dilution of Precision</b>
<b>PICA</b>	<b>Performance, Integrity, Continuity, Availability</b>
<b>PPM</b>	<b>Prior Probability Model</b>
<b>PPS</b>	<b>Precise Positioning Service (GPS)</b>
<b>PRA</b>	<b>Probabilistic Risk Analysis</b>
<b>RAIM</b>	<b>Receiver Autonomous Integrity Monitoring</b>
<b>RF</b>	<b>Random Failure (class of spacecraft failures)</b>
<b>RMTSA</b>	<b>Receiver Noise, Multipath, Troposphere, and Selective Availability (GPS ranging errors)</b>
<b>RNP</b>	<b>Required Navigation Performance</b>
<b>RPE</b>	<b>Required Position Error (limit)</b>
<b>RTCA</b>	<b>Radio Technical Commission for Aeronautics</b>
<b>SA</b>	<b>Selective Availability (GPS)</b>
<b>SA</b>	<b>Simulated Annealing (optimization)</b>
<b>SPS</b>	<b>Standard Positioning Service (GPS)</b>
<b>UDRE</b>	<b>User Differential Range Error</b>
<b>UIVE</b>	<b>User Ionosphere Vertical Error</b>
<b>VDOP</b>	<b>Vertical Dilution of Precision</b>
<b>WAAS</b>	<b>Wide Area Augmentation System</b>
<b>WADGPS</b>	<b>Wide Area DGPS</b>
<b>WMS</b>	<b>Wide Area Master Station</b>
<b>WRS</b>	<b>Wide Area Reference Station</b>
<b>WR</b>	<b>Wearout Failure (class of spacecraft failures)</b>

## Chapter 1: Introduction

*The sciences do not try to explain, they hardly even try to interpret, they mainly make models. By a model is meant a mathematical construct which with the addition of certain verbal interpretations, describes observed phenomena. The justification of such a mathematical construct is solely and precisely that it is expected to work.*

- John von Neumann (1903-1957)

*You cannot apply mathematics as long as words still becloud reality.*

- Hermann Weyl (1885-1955)

*There are three roads to ruin; women, gambling and technicians. The most pleasant is with women, the quickest is with gambling, but the surest is with technicians.*

- Georges Pompidou (1911-1974)

### 1.1 The Development of Systems Engineering Practice

The modern art of systems engineering developed in the latter half of the 19th Century, as a “critical mass” of technical expertise developed in several fields to support the cooperative work of engineers across various specialties. Prior to this, most technical innovations were the product of inventors, working alone or in small teams, who independently developed and publicized their specific inventions. The Industrial Revolution changed this by bringing people trained in various complementary technical specialties to work together to build and operate machines. Soon it became possible to make incremental progress that fed on itself; taking advantage of new breakthroughs and constantly refining accepted techniques. At the same time, the design and manufacture of ever-more-complex systems required specialists to learn to work together to coordinate and set standards for the work of each branch of the project.

This system-development developed into its modern form in the 20<sup>th</sup> Century and has become a worldwide standard since World War II. *Specifications* are its skeleton: they define the system-level parameters that must be achieved by each subsystem. Related to this are *budgets*, which allocate total system cost, weight, power, etc. to each subsystem in a more flexible way (for example, cost overruns in one area may be combated by adopting a cheaper design in another subsystem). Within these limits, the specialists working on each subsystem then have considerable freedom to design their

piece of the puzzle. Their ingenuity within their areas of expertise is thus indispensable to the success of the project. As a result, systems engineers may appear to be merely “bean counters” who limit innovation by citing budgetary or compatibility constraints.

As is true for any methodology that has worked well and has universal acceptance, modern systems engineering has become enshrined in the aerospace design community. Hierarchies of specifications developed over the years just after World War II while aerospace technology development boomed. These provided (and still provide) a clear framework for laying out a design effort in its initial stages and flowing down requirements to each of the subsystem engineering departments. As aerospace technology innovation leveled off by the early 1970’s (corresponding to a time when both NASA and defense funding dropped precipitously), this structure solidified itself and became the dominant development paradigm throughout the aerospace industry. Much of what is taught in systems engineering classes at the university level implicitly assumes that this structure naturally underlies all development efforts and teaches students how best to operate within it.

In the world of rapid technological change in which we live today, this system design framework increasingly appears out of date. In its current incarnation, it is built upon large sections of engineers responsible for developing and enforcing arbitrary requirements at every level of design detail. Entire organizations exist primarily to ensure that ongoing development programs are held to the letter of these specifications and budgets. In addition, specifications are inherently inflexible, making it difficult to adapt a development project to new developments in the underlying technologies. All of this can smother the spirit of innovation and the healthy tolerance for risk that leads to dramatic technology invention and productivity improvement on almost a yearly basis in electronics-related industries.

The research in this thesis takes the first step away from reliance on the traditional framework still used in practically all aerospace companies. The basis of a new approach is outlined in this section, and then each chapter of this thesis demonstrates its applicability to a specific subset of aerospace design practice. The theory of decision making under uncertainty lies at the core of each chapter; it provides a global means of analyzing the

uncertainty inherent in possible courses of action and develops theorems by which the optimal choice can be made. It is introduced in Section 1.2. Section 1.3 shows how these concepts can be applied to the aerospace systems design process by transforming it into a series of decisions made at discrete points in the development cycle. A “master decision tree” is presented which illustrates how the decision perspective is applied to a spacecraft design, development, and test program. This top-level tree is easily broken down into separate subsystem projects that normally are done independently, but its integrated structure allows design optimization to be carried out as a cohesive whole. An integrated approach thus has the potential to avoid the administrative overhead that comes with detailed specifications to which lower-level design efforts are slavishly held.

This thesis covers a great deal of territory; thus a complete summary of the work on which it is based is impossible to construct. Section 1.4 instead introduces the research fields that are most relevant and cites the most important sources for each, giving brief descriptions of what they contain. Section 1.5 summarizes the research contributions of this thesis and draws the separate threads from each chapter together to demonstrate how the tools developed here make fully integrated optimal aerospace design under uncertainty possible.

## **1.2 Decision Making under Uncertainty**

Decision making under uncertainty is based on a fundamental idea: all non-trivial decisions made in the real world are best evaluated by building a model of the uncertain consequences of each alternative and then choosing the one with the highest implicit probability of the most desirable result. This requires much explanation to be applicable to practical cases, but it helps to remember that it all boils down to this single concept.

A simple illustration helps to illustrate this applicability. Many people who are said to be indecisive have trouble with simple decisions such as what to order in a restaurant. If they knew that the dish that they would enjoy most can be prepared quickly and is inexpensive, the choice becomes trivial. If the first two were known to be true but the dish is very expensive relative to what the patron is willing to pay, a deterministic value

judgment must be made to trade off the taste value versus price for the alternatives on the menu. However, if the patron is unsure of the value of each dish because he or she does not know the quality of ingredients, preparation, etc., he or she must make a decision based on a qualitative best guess as to which outcome is most likely for each choice. Thus, uncertainty has been introduced, and even if it is not modeled quantitatively, it tends to be the primary source of hesitation as our indecisive gourmand tries to form a fuzzy uncertainty picture from which he or she can make up his or her mind.

The key element that decision analysis adds to this picture is a mathematical demonstration that, given the assumption of a “rational” objective function in a number-line sense, an optimal decision exists within this domain of user preference and uncertainty. Along with this comes a conceptual method for deriving quantitative uncertainty models in cases where quantitative input data is insufficient to allow the use of statistical inference. Decisions can be formalized into a generic pattern where a decision maker, at a given point separating the past (and what is known about it) from the future (and the uncertainties it holds), faces a set of *alternatives*, each of which presents the decision maker with a *deal*. A “deal” can be thought of as a “bet” or “lottery” in which the decision maker’s “payoff” depends on various future *prospects* whose realization (and likelihood of realization) are uncertain [1-2]. The decision maker must evaluate the deal presented by each alternative based on these uncertainties and a function which measures the value of each prospect, or *outcome*.

The following two subsections further detail the decision analysis approach to the two apparently unrelated issues of uncertainty modeling and user preference modeling. Once the probability model (Section 1.2.1) is in place, a method for integrating preference functions into a probability format for seamless evaluation is introduced in Section 1.2.2 and Appendix A.

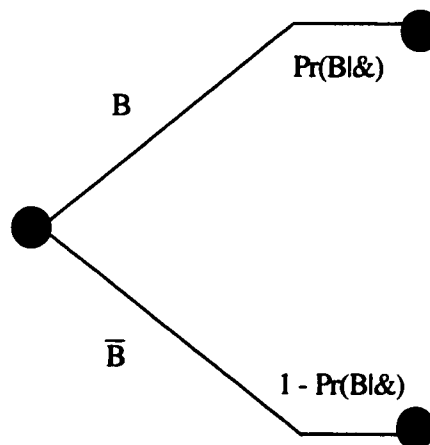
### *1.2.1 Uncertainty Modeling*

Occasionally, there is a well-respected and accepted method of formulating probability estimates for uncertain future events in a given field. Or it may be possible to come up with representative probabilities from controlled experiments. However, when

probabilities of rare events are required, these approaches tend to break down. One conventional response to this problem in engineering is to design a system according to the “worst” conceivable prospect and be satisfied if it can be accommodated while maintaining satisfactory operation. This has proved successful in the past, but it can lead to a deterministic design focus that allows unforeseen possible circumstances to fall by the wayside. Once a “worst case” is defined, it is easy to focus on the peculiarities of that situation in one’s attempts to mitigate it, and the design one then comes up with can easily be non-robust to other threatening circumstances.

Decision analysis provides a means for formulating uncertainty models that at least have the potential to avoid this problem. First, it is important to clearly define the possible alternatives into unambiguous categories so that they pass the so-called “Clarity Test”. Otherwise, confusion as to what outcome falls where could corrupt the relative likelihoods that are assigned. Once this is done, probabilities can be assigned (without necessarily using any data) for each of these possibilities based on whatever information is available. A common notation for this is  $\Pr(B|\&)$ , where  $B$  is a specific *distinction* that represents a given possibility and the “&” refers to all information generically available at the time the decision must be made. A *probability tree* can be constructed to represent this as shown in Figure 1.1, where  $\bar{B}$  represents all outcomes other than  $B$  [1-3].

Setting the probability for  $B$  given “&” is simply based on the best estimate that the decision maker armed with knowledge “&” can form. There is no rule on how to do this, but a guideline for intuitive probability assessment exists which is best demonstrated by example [1-3]. Let us say that a person with a reasonable knowledge of American history (from high school and college classes) is asked to determine his uncertainty about how many separate strikes (events of labor unrest) occurred in the U.S. during World War 2. This



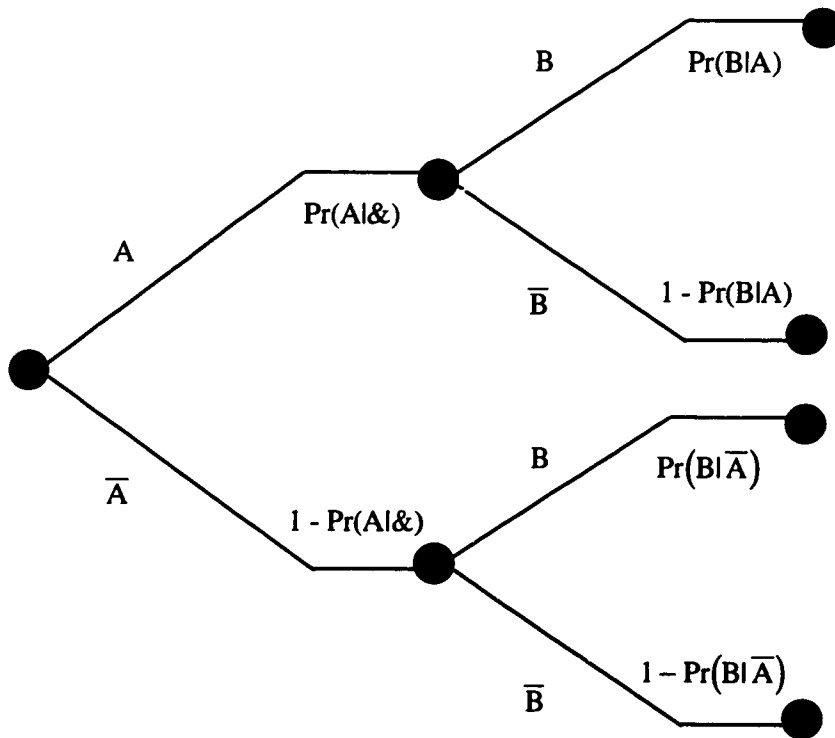
**Figure 1.1: Simple Probability Tree for Possibility “B”**

number requires a clear definition: we are referring to the period when the U.S. was at war (December 1941 - August 1945) and statistics tabulated by the U.S. Department of Labor. Lacking detailed knowledge of this subject, one is tempted to just guess a most-probable number and call that the mean (50th percentile) of one's uncertainty distribution. Then one is faced with wondering just how much uncertainty exists about that number.

A better approach in cases where one is substantially uncertain is to instead estimate the 25-75 percentile limits [1-3]. In other words, one tries to place upper and lower bounds on the number in question that make it equally likely that the number falls inside or outside the chosen interval. Once one is thinking in that frame of mind, it is easier to next choose uncertainty bounds of 5-95%, 1-99%, and so on before deciding on one's mean estimate of the number. Interpolating between these intervals can be done using Gaussian curves or whatever one feels is most suitable to reflect the right amount of low-probability "tail expansion." By the way, the correct answer to the World War 2 labor unrest question is on the order of 41,000, which tends to be much higher than most people guess. By starting with an uncertain interval rather than a point guess, one's frame of mind is shifted to think about what key information is not included in "&" and, as a result, just how wide one's range of uncertainty is.

Although one may be hesitant to place any weight on a probability distribution arrived at in this fashion, decisions based on whatever information is available are optimal under this theory, provided that the resulting probability assessment truly represents the decision maker's "best guess." In this sense, assessments based on the information contained in "&" are in the same domain as any set of statistically significant data -- there is only a difference in level of "relevance" to the uncertain outcome. A key point is that there is nothing to be gained by wishful thinking about these likelihoods. Any amount of willful or unintentional self-deception, such as goes on (it can be argued) with traditional spacecraft reliability models, can lead to a sub-optimal decision. In effect, this is because the resulting decision problem was evaluated based on something less than the set (&) of available information.

While making decisions based a limited amount of knowledge in “&” is valid if a decision must be made now, often alternatives exist which allow the decision maker to obtain more information before making an irreversible choice. In a probability tree, this can be demonstrated from a case of two distinctions B and A which are “relevant” to each other. “Relevant” in this context refers to statistical dependence: knowledge of A or  $\bar{A}$  influences one’s probability distribution on B. We can present this situation in the probability tree shown in Figure 1.2.



**Figure 1.2: Probability Tree for Relevant Distinctions A/B**

Often it is easier to assign probabilities in a certain direction. In the case shown in Figure 1.2, it was probably easier to choose probabilities for A and then choose those for B in the two separate cases of A or  $\bar{A}$ . In some cases, however, the decision maker’s outcome is most dependent on A, and he or she has the option of obtaining, for a cost, information about whether B or  $\bar{B}$  is the case. If this information regards knowledge of a

distinction C that is relevant to B, C could be added to Figure 1.2 in whatever order is easiest to assess probabilities. For this illustration, we will simply assume that the decision maker can find out whether B or  $\bar{B}$  holds for some cost that has value in his or her preference function. In this case, the decision maker can reverse the order of A and B in Figure 1.2. The marginal probability distribution for B is given by basic probability theory:

$$\Pr(B) = \Pr(B|A)\Pr(A) + \Pr(B|\bar{A})\Pr(\bar{A}) \quad (1.1)$$

Then, Bayes' Rule is used to compute the dependent probabilities of A given knowledge of B:

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B)} \quad (1.2)$$

Now we can represent these two options in an augmented probability tree that we will call a *decision tree*. This is shown in Figure 1.3. The black square on the left indicates that a choice is to be made between paying to obtain knowledge of B or  $\bar{B}$  versus not getting this information but going ahead with a project whose outcome is dependent on A. In this case, a third option is not doing the project at all. The utilities  $U$  on the right hand side give the value to the user of each possible outcome.

In this example, one has the option to obtain information on B which is relevant to the key outcome A. The value of information in this case has a natural upper bound, which is the value of *perfect information* on A or  $\bar{A}$ . If the outcome A were known, the decision maker's choice and his resulting value outcome would be obvious (if not, more uncertain distinctions should be added to Figure 1.3). Let us assume that the project would only be worth doing if A is the result. Then the value of perfect information on A is equivalent to the weighted value from the probability tree (where the value of doing nothing holds if  $\bar{A}$  is the result) minus the value that would result if no information on A were obtained (this is based on the best alternative in Figure 1.3, which will depend on the specific values and probabilities involved):

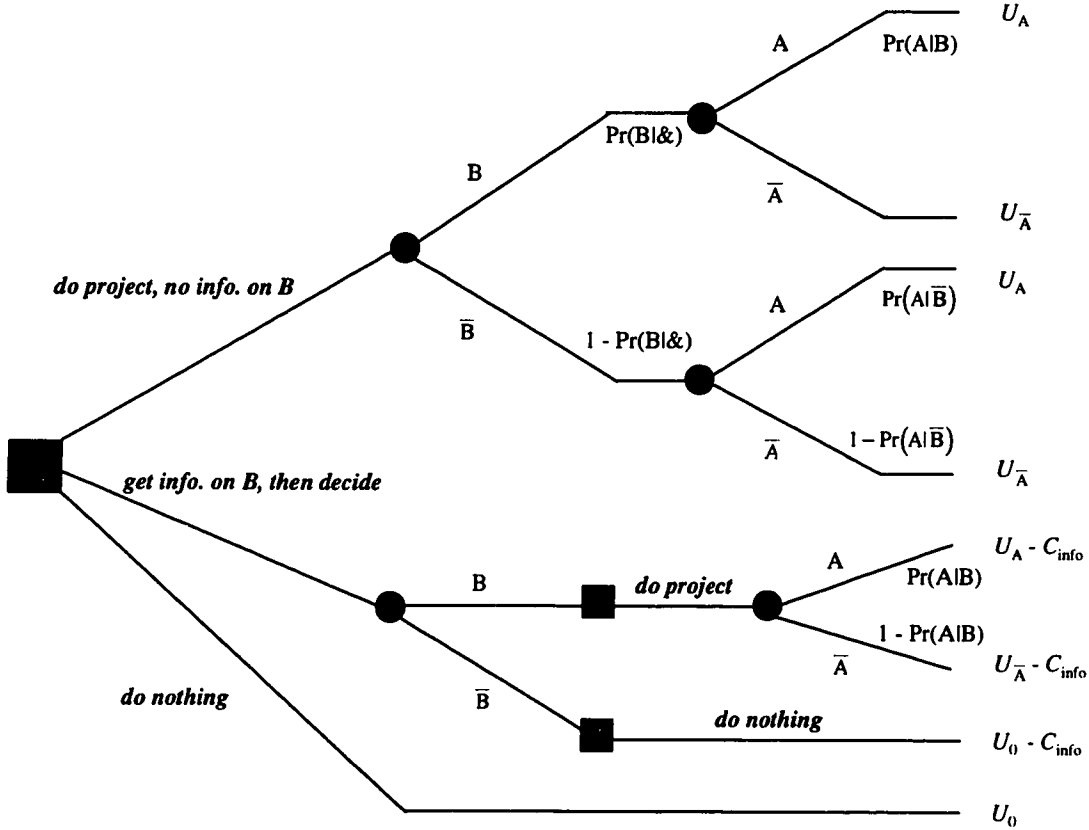


Figure 1.3: Decision Tree for Problem of Distinctions A/B

$$Value(PI,A) = U_A \Pr(A) - U_0 \Pr(\bar{A}) - Value(\text{best choice, Fig. 1.3}) \quad (1.3)$$

Given the value of perfect information on A, we know that the value of *imperfect* information on A, which is given by perfect information on the relevant distinction B in Figure 1.3 (and is assigned the cost  $C_{\text{info}}$ ), must be no greater than the value of perfect information on the key distinction A given in (1.3). Otherwise, it would never be optimal to pay for that information. This is not as rare as one might think; as the value of knowing more about what is likely to happen may not be significantly higher than accepting the lottery one is faced with based only on the prior information “&”.

Note that this simple demonstration captures the whole realm of value-of-information issues. In engineering design, it often comes up in the context of whether expensive experiments on prototype hardware are worth the cost in time and money.

Traditional development processes have a standard set of experiments of increasing levels of complexity that must be completed before development is considered successful. In many cases where time or cost pressures are greater, the optimal decision may be to proceed with fewer tests, which means less information and perhaps greater project uncertainty, rather than to pass on the opportunity altogether.

At lower levels of detail, the value of information issue arises frequently in controller and filter design. For example, Kalman filters begin with a prior covariance model and propagate it forward in time, arriving at updated covariance results from a best sequential least-squares fit. After-the-fact smoothing may use both a forward and a backward sweep through the data to produce the posterior covariance [9-3,4]. This approach, which is widely used in modern engineering design, is simply a Bayesian update to a prior information (&) matrix such that the posterior covariance represents the best uncertainty model given the dynamic information available up to the present time. A control law or failure monitor will use that information to make an “optimal” decision on the best action to take (size of the control input, or whether or not to warn of a failure).

Finally, the decision analysis perspective easily allows controller models to be generalized to non-Gaussian, non-linear models where the model equations, order, and/or parameters are uncertain. If a decision maker is uncertain about the validity of the Kalman filter assumptions for a given application, its use would be inferior to the use of a model that applies whatever is necessary to evaluate a more flexible uncertainty model in real time. This is true even if the results are non-intuitive from the standpoint of the traditional assumptions. The significance of this conclusion will become clearer as the decision-analysis philosophy is applied to each of the applications in this thesis.

### *1.2.2 User Preference Function Modeling*

To fit within a valid decision-theoretic framework, the decision maker’s preferences for each of the possible outcomes must fit the “Rules of Actional Thought” which are detailed in Appendix A. In general, these rules simply ensure that the user’s preferences among the possible outcomes follow a logical order from best to worst with no circularity. Almost all decision makers intend that their preferences and evaluation

methods be logical and follow the intent of these rules. Thus, once decision makers agree to take a quantitative approach, it becomes feasible to build a valid decision model for their problem.

In addition to requiring the use of probabilities to assess uncertainty, the equivalence rule has the decision maker choose a “preference probability”  $p$  for each intermediate outcome. This number is chosen so that he or she would be equally willing to accept either that intermediate outcome or a lottery in which he or she receives the best possible outcome with probability  $p$  and the worst outcome with probability  $1-p$ . Doing this for each intermediate outcome accomplishes two things. First, all of the intermediate outcomes are given a value between 0 and 1 that decreases as the user preference decreases. Thus, a suitable objective function has been created. Where the “delta property” holds (see [1-3]), the relationship will be linear, but any function that satisfies the rules in Appendix A is fine. Second, because the preference probabilities behave as real probabilities in a mathematical sense, a probability tree along the lines of Figure 1.1 can be substituted for each intermediate outcome to express all possibilities as functions of the best and worst outcomes. The problem can then be “solved” by working out the probability of receiving the best outcome for each alternative and choosing the one with the highest probability of that most desirable result.

The preference probability approach allows the decision maker to choose any weighting of the intermediate outcomes that he desires. This choice of weighting may implicitly include a measure of *risk aversion*, which places higher values for certain equivalents (C.E. = a certainty of getting a particular intermediate outcome -- see Appendix A) than would be suggested from the expected value of the equivalent lottery. For example, a person who is indifferent between a certain receipt of \$80 and a 50-50 lottery between zero and \$200 is risk averse because the expected value of the 50-50 chance is \$100. Someone whose C.E. for this lottery is \$100 would be said to be *risk neutral*, and someone whose C.E. is greater than \$100 would be *risk seeking*. For significant decisions, the latter category is rare, of course.

In practice, it is normally convenient to represent a decision maker's implied preferences in a mathematical function that can be evaluated either analytically,

numerically, or by simulation (in decreasing order of convenience). Traditionally, objective functions are chosen first and foremost by their ability to fit a standard optimization format that has a well-known globally optimal solution. Good examples of this are linear functions that can be optimized using Linear Programming and linear-quadratic functions for which an analytical solution is given by Linear Quadratic Regulator (LQR) theory [9-4]. However, it is often hard to translate these measures into the decision maker's system-level preferences. Furthermore, analytical functions are often artificially constrained in non-realistic ways. For example, the LQR method requires that the plant be "controllable" by the actuator in terms of linear system theory [9-8]. In this case, LQR can always stabilize the system and ensure that the objective function has a smooth global optimum. Robust control theory [9-6,11] extends the LQR formulation to include bounded parameter variation such that guaranteed stability is always possible. In reality, this just does not hold water -- where significant uncertainty is present, a non-zero probability of instability will remain for any feasible design. In marked contrast, the decision analysis approach has the flexibility to handle a direct modeling of the user's true preferences expressed over the full range of envisioned uncertainty. Controller design based on probability models that encompass all known sources of uncertainty provides "robustness" automatically, since the solution will be an optimal tradeoff between nominal performance and the entire space of off-nominal conditions.

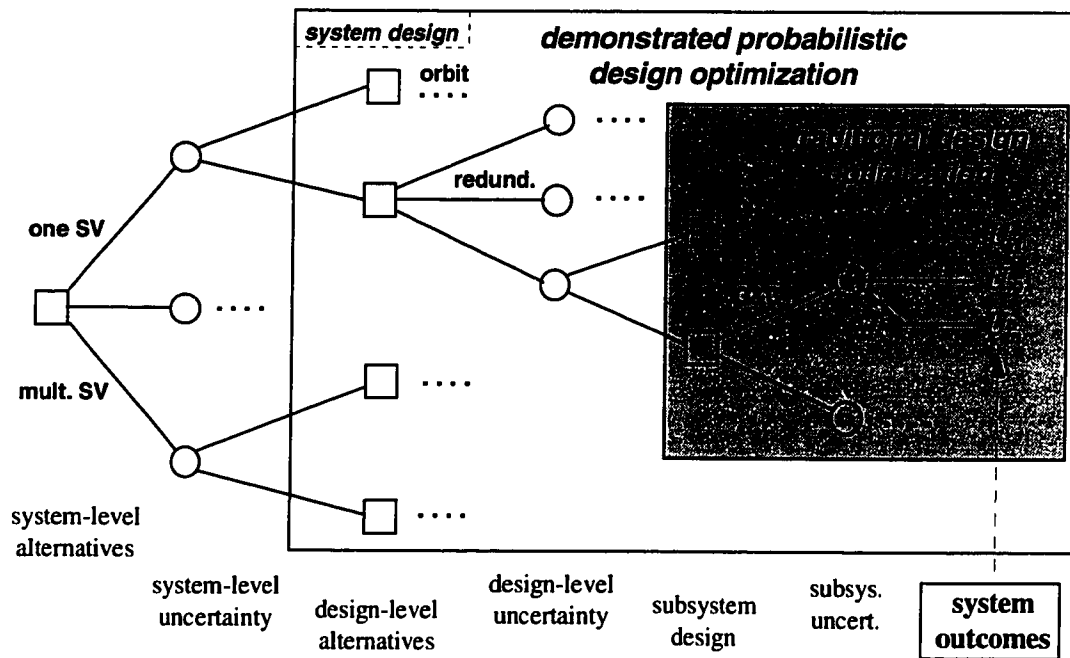
Normally, preference probabilities or values assigned for possible outcomes can be more conveniently represented as mathematical functions, but for complicated cause-and-effect probability trees, no analytical representation of the outcome probabilities is feasible. Monte Carlo simulation on modern personal computers and workstations now makes it possible to evaluate *any* uncertain system behavior model and its associated preference function. Therefore, a fair decision-theoretic representation of engineering design problems is possible. Simplifying assumptions and analytical calculations remain useful, of course, but the presence of a means of complete uncertainty modeling and consequence evaluation opens up a new window of opportunity for system developers whose domain of activity does not easily fit into the traditional format.

### **1.3 Applications to Engineering System Design**

In principle, the decision-analysis structure introduced in the last section can be applied to any problem in which the decision maker(s) are willing to both formulate the problem in the probabilistic framework of Section 1.2.1 and follow the “Rules of Actional Thought” described in Appendix A. Translating this to an aerospace problem is not necessarily trivial, but all of the necessary elements are present in Section 1.2. In particular, a means of building a reasonable “best guess” uncertainty model is required. Traditional models that do not fit the precepts of decision theory (normally because they are known to be inaccurate simplifications) can often be revised to include these “non-ideal” sources of uncertainty. This is what is done for spacecraft performance uncertainty modeling in Chapter 2. In the chapters on GPS, prior uncertainty models are built from published data that is itself highly uncertain. In both cases, Monte Carlo simulation provides the means to translate these models into uncertainty projections from which decisions can be made according to user preference functions that measure top-level performance (degree of mission success) from the simulation outputs. No formal decision trees are shown, but each chapter explains how the solution it constructs is based on the fundamental decision principles and analysis steps introduced in this chapter.

Figure 1.4 presents a conceptual decision tree that attempts to capture the complete scope of a spacecraft design and development program. It demonstrates that the entire process can be viewed as a single integrated decision analysis problem in which many different levels of decisions are made over time. It shows an interlocking relation of decision authority and uncertainty at each level of detail. While it is common for higher-level decisions to be made earlier, resolution of the higher-level uncertainty (such as one’s ability to maintain a reliable constellation instead of just a single satellite) is not necessarily resolved before uncertainty at lower levels. In fact, each of the nodes in this tree may imply a smaller encapsulated decision problem that is resolved over time as more information becomes available at all levels.

On the left-hand side, basic architecture decisions on how to accomplish a specific satellite-based mission, such as how many vehicles to build and launch, are shown as



**Figure 1.4: Master Decision Tree for Spacecraft Design Process**

“system-level alternatives.” Other decision points at this level may include the overall schedule for development and deployment. One level further down in detail are the “design-level alternatives” for a given satellite. Nominal orbit decisions may take place here, but they are likely heavily influenced by the architecture decisions. A key decision at this level is the layout of the spacecraft subsystems and the amount of redundancy applied in each area. That decision problem is addressed in detail in Chapter 3.

Each subsystem has its own design decisions as shown further to the right in Figure 1.4. Once the basic elements are decided (source of power and how much should be available at beginning of life, for example), each subsystem must be designed to meet the implied requirements of the higher-level decisions. Traditionally, subsystems are designed to meet detailed lists of specifications decided at higher levels. In this case, however, no such presumption needs to be made. Of course, higher-level decisions will bound the space of acceptable choices, but the subsystem design format suggested here only needs to exchange information about user preferences within the overall decision tree as optimal searches take place at several levels simultaneously. Using the evolutionary

global optimization techniques introduced in Chapter 3, which easily make use of simulation based evaluations, it is now feasible to do just that, although computer speed is still a limiting factor.

While the complete problem as outlined here is not solved in this thesis, Chapters 2 and 3 on spacecraft reliability prediction and optimal design present a solution (for the Gravity Probe-B spacecraft) to the part of the decision tree contained in the “system design” box in Figure 1.4. This is a considerable advance in integrated optimization capability over the traditional subsystem optimization problems shown in the smaller gray-filled box. Note that almost all subsystem optimization methods in use today are either deterministic or based on bounded-uncertainty, as in the case of robust control design. The decision-theoretic approach developed here has been shown to handle a much larger problem in reasonable time and to produce results. Admittedly, the optimization carried out in Chapter 3 makes decisions only at the system (and not subsystem) level, but no further leap is needed to integrate more-detailed subsystem simulations into the system-level simulation model presented in Chapter 3 and to simultaneously optimize system redundancy and subsystem design variables.

The chapters on GPS network and algorithm design demonstrate this by linking the setting of thresholds for user integrity verification algorithms to the top-level design of augmented GPS network architectures. In Chapters 4 and 5, prior probability models (PPM) with substantial parameter uncertainty are constructed for both stand-alone GPS and for part of the Wide Area Augmentation System (WAAS). A user value model is then constructed based on a fundamental risk allocation for passenger aircraft contained in the Required Navigation Performance (RNP) set of proposed requirements [6-10]. By simulating GPS geometries and sampling from the uncertainty space of the relevant PPM, a simple post-processed search algorithm can choose thresholds for Receiver Autonomous Integrity Monitoring (RAIM) which minimize this top-level value model (and hence optimize a fundamental cost vs. safety tradeoff). Both PPM simulation and user risk optimization are different from the dominant requirements-based methods for setting thresholds as described in the literature on the subject [7-10,11]. As a result, the optimal thresholds and resulting user performance predictions vary widely (and are generally more

pessimistic, as would be expected from an increased-uncertainty model) from previously published results, suggesting that the traditional method is sub-optimal from a decision-theoretic standpoint.

Using the covariance propagation model (which assumes Gaussian error distributions) for normal WAAS conditions as developed in Chapter 6, a combined accuracy and safety evaluation for WAAS networks is possible and is described in Chapter 7. Optimal RAIM threshold selected for a given network is included in this process. A top-level value model for a governmental agency is then built to translate these WAAS results into the domain of monetary costs and benefits. It then becomes possible to select the optimal elements of a WAAS network designed for a particular airborne precision approach application using a simple genetic algorithm, which is limited only by effective computer processing speed.

In the context of Figure 1.4, the WAAS network optimization of Chapter 7 would be a design-level decision that incorporates optimized integrity thresholds for each network alternative (a subsystem-level decision problem) from Chapter 5. There is no reason why this integrated decision framework could not be expanded further to include system-level decisions, such as what sort of augmented GPS network (WAAS, LAAS, or a combination of both) should be used to provide Category I precision approaches to the commercial and private aviation communities.

## **1.4 The Gravity Probe-B Experiment**

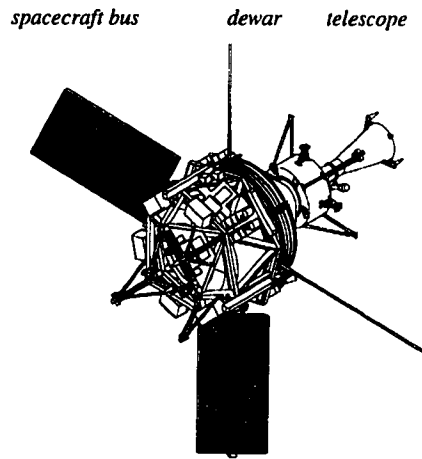
Chapters 2 and 3 use the *Gravity Probe-B* (GP-B) spacecraft as the example for demonstrating the new reliability calculation models and redundancy optimization procedures introduced in this thesis. This spacecraft is the result of a long-running cooperative arrangement between Stanford University and NASA to develop a spacecraft that will orbit experiments to verify Einstein's Theory of General Relativity. By orbiting a spacecraft in polar low-earth orbit and using drag-free control to remove disturbances caused by particle impacts, gravity gradients, and the like, it is possible to monitor two relativistic effects on bodies in orbit around a massive object such as the Earth [5-3]. One

is the *geodetic effect*, which is caused by the curved space-time that results from Earth's mass. The second is the *frame-dragging* effect, which is due to the "dragging" of local space-time by the Earth's rotation on its axis. These effects (6.6 and 0.042 arc-seconds per year respectively for a 650 km polar orbit) are tiny compared to Newtonian disturbances. Therefore, extremely precise gyros and readout sensors, a science telescope for precise inertial reference, and an extremely accurate drag-free attitude controller are all required [5-3,4,6,7].

Figure 1.5 shows a drawing of the GP-B spacecraft.

The GP-B satellite is divided into two sections. The *experimental payload* is built around the *probe*, which contains the gyros, sensors, proof mass, telescope, gas lines, and electronics, and the *dewar*, which surrounds the probe with superfluid helium to keep its temperature in the cryogenic range needed by the sensors. The payload is responsible for measuring the measure the gyroscopic spin axis direction to better than 0.1 milliarc second over one year [5-3,7]. Much of this equipment has never flown before; so its reliability is uncertain. Methods for modeling payload uncertainty will be discussed in Section 2.5.

The *spacecraft bus*, which supports payload operations in space, is being developed separately by Lockheed Missiles and Space Company (LMSC) [5-5]. It is based on more conventional assemblies, many of which have been developed for use by previous LMSC spacecraft. In conjunction with the payload, the spacecraft bus has four key tasks which must be successfully achieved for the duration of the primary mission: (1) maintain the gyros in a purely gravitational orbit, (2) minimize Newtonian disturbance torques on the gyros by means of drag-free control, and (3) keep the satellite and probe pointed at the guide star (which serves as an inertial reference) [5-4,6]. All of these tasks have associated allowable error tolerances that are very demanding. These specifications and the many more-detailed ones that flow from them represent the primary definition of



**Figure 1.5: GP-B Spacecraft**

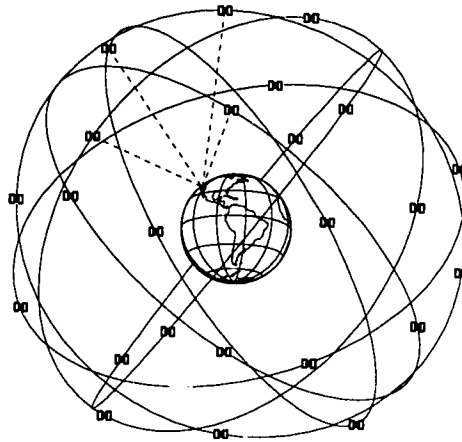
success of the mission in terms of orbital operations. Simple value models that capture these goals are discussed in Section 3.2.3.

## **1.5 The Global Positioning System (GPS)**

The *Global Positioning System* (known as GPS) is a product of several decades of development of spacecraft-based navigation systems by the U.S. Department of Defense (DoD). Originated in the early 1970's, it provides accurate position and velocity-determination capability known as the Precise Positioning Service (PPS) to military users all over the world. In 1983, President Reagan guaranteed the availability of a degraded single-frequency GPS navigation capability known as the Standard Positioning Service (SPS) to all civilian users. This degradation of the basic GPS Signal-in-Space (SIS), known as Selective Availability (SA), is applied by the DoD to prevent civilian users from obtaining the full 21-meter (95%) horizontal positioning accuracy specified for the PPS. The SPS instead specifies 100-meter (95%) accuracy, but research into Differential GPS (DGPS) has made it possible to correct for both the DoD-induced and ambient system errors to provide horizontal accuracies well within 10 meters (95%). This level of precision promises to revolutionize many aspects of airborne, ground, and marine navigation [6-13].

The current GPS system, which reached Final Operational Capability (FOC) in 1995, contains 25 NAVSTAR satellites (all but one are now of the Block-II and IIA varieties) in 12-hour (sidereal) orbits at 55° inclinations, arranged into six orbital planes. Three of the 24 satellites are “active” spares. Figure 1.6 shows the orbits of these six planes around the Earth. Note that this constellation of orbits repeats itself every (sidereal) day from the perspective of a user fixed on Earth, a fact which simplifies the simulation of GPS satellite orbits. The NAVSTAR constellation is controlled by the U.S. Air Force 2nd Space Operations Squadron at Falcon AFB. The associated Operational Control Segment (OCS) includes the Falcon AFB Master Control Station (MCS), a set of worldwide DoD monitor stations to observe the health of each satellite, and ground antenna sites which send uplinks to the satellites.

Each GPS satellite broadcasts its position (along with other almanac information) and a precise time stamp to users on Earth or in space. A user receiver can determine his or her range to each visible satellite, and if four or more satellites are in view, it is then possible to compute location in three dimensions along with an estimate of the time bias in the user receiver [6-17]. Although Selective Availability degrades user accuracy to the 100-meter range, that is still sufficient for many applications



**Figure 1.6: The GPS Constellation**

which now make do with much less accuracy available only in certain regions. It is also possible to reduce GPS errors down to and below the meter-level if ground-based augmentations are present to help correct for SA and other errors that are correlated over a wide region.

## **1.6 A Survey of the Relevant Literature**

Because this thesis draws on so many different areas, it is very difficult to completely account for all of the key references in each one. Instead, I have listed a few references in each specialty that I have used as starting points in my research. These and the references that they contain may also serve as the starting point for a more in-depth study of a specific topic. The bibliography included at the end of this thesis breaks up all references into these categories as well.

### ***1.6.1 Decision-Making Under Uncertainty***

The key starting points in this relatively new field are the work of Professors Howard and Matheson in [1-2]. The course notes for the Stanford class on decision theory, EES 231, are also useful [1-3]. Prof. M.E. Pate-Cornell, an expert in the application of decision theory to probabilistic risk analysis, has applied it to the aerospace

field by studying the risk to the Space Shuttle from debonding of the thermal protection tiles [1-7]. She has also contributed greatly to the top-level study of safety-critical warning systems and the tradeoff issues that they pose [1-5,6]. Finally, the most authoritative exposition of modeling user preferences into mathematical objective functions is [1-4]. An important discussion of the acceptability of assigning numerical values to human life for the purposes of risk cost/benefit evaluation is in [1-1].

### *1.6.2 Spacecraft Reliability Modeling*

MIL-HDBK-217F [4-12] details the traditional methodology used by almost all spacecraft developers in making their reliability predictions, as well as serving as a source of exponential failure rates. NPRD-95 [4-8] provides failure-rate data for non-electrical assemblies. MIL-STD-1629A [4-7] is the corresponding government document laying out the format of Failure Modes and Effects Analysis studies. Textbooks by Dhillon [3-5], Billinton and Allan [3-2], and Leemis [3-10] are three of the many books that give a good survey of system reliability, including traditional redundancy optimization.

The work of Herbert and Myron Hecht on fitting new probability distributions to spacecraft reliability data [4-3] is a critical foundation of the sampling-based models developed in this thesis. Also, studies published by TRW in the 1970's and early 1980's have influenced the approach taken here, especially the "weak sister" concept [4-1,13,14]. RAND has studied the effect of post-failure improvised "work-arounds" on mission reliability of military spacecraft [4-6].

The use of Monte Carlo simulation to evaluate reliability models is now commonplace. A detailed mathematical development is contained in [3-13]. Henley and Kumamoto in [3-7] give a good application-oriented overview and reference some of their more-detailed papers there. Law and Kelton, in [3-9], provide comprehensive coverage of simulation, including sampling from a variety of probability distributions, statistical inference on simulation results, and techniques for variance reduction. William Press, *et.al.*, in [3-12], give a series of helpful C functions for this purpose. In fact, all of the C simulations performed in this thesis use the numerical routines given here.

### 1.6.3 Global Optimization

Theoretical aspects of Simulated Annealing (SA) are well-covered in [2-15], including the foundations of the proof that the algorithm will converge to the global optimum given infinite time. Practical aspects of the implementation used here, including the variation operators from one solution to the next and the resulting evaluation covariance, are contained in [2-10,16]. Chapter 10 of Press, *et.al.* [3-12] also has useful information on this topic, along with sample codes for the Traveling Salesman Problem.

The literature on Genetic Algorithms (GA) has expanded greatly in the past few years. Two key texts written by Goldberg [2-5] and Davis [2-2] explain the basic philosophy and search operators well and show how they can be applied to a variety of real-world problems. In particular, Davis discusses the use of integer and real-number solution encodings. Techniques for choosing optimal parameters for GA operators are discussed in [2-7]. Several papers have applied genetic algorithms to the design of control systems, see [2-10,12]. The latter shows how a genetic algorithm can use a flexible encoding to handle solutions of widely varying complexity. Finally, the handling of statistical noise as a result of simulation-based evaluation is discussed in [2-1,8]. In [2-1], Aizawa and Wah show how the expected amount of uncertainty in fitness evaluations affects the optimal GA control parameter choices and how these can be adapted during the GA evolution.

In the aerospace field, Prof. Ilan Kroo and his students at Stanford are one of several groups working on integrating top-level global optimization with subsystem-specific optimization codes that incorporate gradient search and domain-specific knowledge. The gradient-based approach to this task, which passes partial-derivative information back and forth between the overall optimization and the subsystem searches, is described in [2-18]. Peter Gage, in his Ph.D. thesis at Stanford [2-3], developed a system-level GA for aircraft design optimization that has a variable-complexity encoding -- it incorporates solutions with varying vector lengths to allow certain members of the population to include design items that others completely lack.

#### *1.6.4 Spacecraft Optimization under Uncertainty*

This thesis represents, to the author's knowledge, the first extensive study of spacecraft optimization based on reliability models. Reliability-based optimization has typically focused on maximizing a deterministic reliability measure under maximum cost/size constraints or minimizing cost/size under a minimum acceptable reliability constraint. Algorithms to solve this problem using NLP and/or dynamic programming are discussed in [2-8,3-2]. The use of design heuristics to aid optimal redundancy allocation is discussed in [2-11]. Recently, SA and GA's have been applied to optimal redundancy or reliability design; see [2-13].

#### *1.6.5 Control System Optimization under Uncertainty*

Although not specifically covered in this thesis, the application of design-under-uncertainty techniques can be extended to control system optimization. Control system design for robustness is a very active research field. In the time domain, students of Prof. Bryson at Stanford have developed SANDY for multiple-model (with probability weighting) controller gain optimization [9-11] and, more recently, methods to optimize these parameters to handle the worst-case deviation of the parameter uncertainty vector [9-13]. These approaches rely on linear system theory and assumptions to evaluate the possible system models, and they also have to bound the extent of possible parameter variation so that a guaranteed-stable design is possible (this is also necessary for  $H_\infty$ -based robustness-design methods).

As is the case for spacecraft optimization, the use of simulation to evaluate control systems removes the need to adhere to these limiting models. Simulation-based evaluations can handle nonlinearities and cases where stability is not guaranteed with ease, and a higher-level evaluation function could model the performance/risk/cost tradeoffs involved. Prof. Stengel and his students at Princeton University have recently published their approach to simulation-based control system evaluation, known as "stochastic robustness," in which simulation outputs such as probability of instability and probability of meeting time-domain performance specifications are used as bottom-line robustness metrics [9-14,16]. Stengel and Marrison [9-12] have also applied GA's to do top-level

optimization for well-known “benchmark” problems. This is an extremely significant area for further research, as it has the potential to revolutionize robust controller design by linking it directly to real-world applications and objectives (as discussed in Section 1.3).

#### *1.6.6 Global Positioning System: Introduction and RAIM Concepts*

The body of literature regarding the intricacies and applications of satellite-based navigation is growing rapidly. A comprehensive resource on the Global Positioning System has recently been published [6-14]. The signal processing, delay-lock loop, and correlator equations used to compute standard GPS positioning are given in [6-17]. An analysis and explanation of the baseline GPS satellite constellation is given in [6-6].

The traditional assumptions and equations for traditional Receiver Autonomous Integrity Monitoring (RAIM) are given by [7-8,10]. Sturza and Brown in [7-11] used these models to derive practical means of setting residual statistic thresholds for snapshot integrity checks as well as geometry criteria thresholds for availability determination. Pervan [7-9] has demonstrated that for a very accurate system known as IBLS, RAIM provides excellent capability to screen out a wide range of system failures if the traditional RAIM assumptions are made. Walter [7-13] has introduced measurement weighting matrices into the residuals decision statistic for the Wide Area Augmentation System (WAAS) application.

In addition to single-epoch RAIM, general research into time-based filters to perform Fault Detection and Isolation (FDI) is an extensive research field in and of itself. Willsky [7-14] and Frank [7-5] give detailed surveys of this field and of the various Kalman-type filters developed for this purpose. An attempt to derive a similar approach for GPS integrity monitoring is described in [7-2]. Bancroft and Chen [7-1] have worked out Bayesian-update equations from the traditional RAIM equations that provide the framework optimal fault detection using a “loss function.” Finally, Specht [3-14] discusses the use of probabilistic neural networks as classifiers which in effect update a Bayesian network with current data, resulting in time-updated posterior fault probabilities from which to make isolation decisions.

### *1.6.7 Global Positioning System: Probability Models and User Value Functions*

Although “official” GPS Space Segment failure modes and probabilities are not available from the Department of Defense (DoD), several papers have published estimates based on previous satellites with navigation-related missions and the limited amount of available GPS orbital experience. Durand and Caseau [6-4] developed ten sets of “long-term” and “short-term” mean failure and repair times which are intended to encompass the range of failure parameter uncertainty. The resulting Markov models were used to evaluate satellite availability throughout the GPS constellation. Phlong and Elrod [6-15] further develop this Markov process approach using the Durand/ Caseau “most probable” set data, and they show the results of updated simulations which include geostationary satellite augmentations.

Statistical studies of observed GPS performance to date are contained in [7-4] and [6-3]. The latter is based on results of the FAA Performance Analysis Network (PAN) which has begun monitoring Standard Positioning Service (SPS) performance from three sites in CONUS. Another important study on these lines is the IBM/DoD GPS Integrity Study [7-3] which was summarized by Gower [7-6]. This study focuses on integrity of the Precise Positioning Service (PPS) for military users and studies the effects of proposed improvements to the DoD Operational Control System (OCS) architecture on PPS user integrity. Because it includes a fairly comprehensive Failure Modes and Effects Analysis (FMEA), albeit in a non-standard format, it is a useful resource for civilian GPS user system failure models as well.

One of the motivations for the RAIM user value model developed in this thesis is the Required Navigation Performance (RNP) tunnel requirements development methodology, described in [6-10]. In this effort, the probabilities of allowed penetration of specified navigation boundaries are motivated by a top-level risk analysis that allocates a small part of the overall aircraft fatal-accident risk to the navigation system. This analysis provides a basis for trading off continuity and integrity risk in the optimal-RAIM approach developed in Section 4.0 of this thesis. The RNP framework is controversial because of its focus on Total System Error (TSE) instead of simply specifying allowed Navigation System Error (NSE), but its key contribution is the setting of navigation

system requirements within an overall risk/performance context. It provides an alternative to the requirements for precision approach and landing developed for the Instrument Landing System (ILS) in ICAO Annex 10 [6-9]. Recently, the FAA Satellite Program Office has developed an Operational Requirements Document (ORD) [6-11] for GPS-based approach and landing systems that borrows from both the RNP and ICAO Annex 10. These requirements are still in a state of flux, however, especially for Category III.

#### *1.6.8 Global Positioning System: Wide Area Augmentation System (WAAS)*

Many of the ideas behind Wide Area DGPS (WADGPS) were developed in Dr. Changdon Kee's Ph.D. thesis [8-9]. More recently, Enge and Van Dierendonck [8-7] describe the layout of a practical WAAS network in detail, focusing on the design of the 250-bps digital signal to be transmitted via geosynchronous satellite to users. FAA plans for the establishment of an operational WAAS to allow Category I precision approaches are detailed in [8-11]. Performance requirements for the use of WAAS for Category I are given in both the RTCA Minimum Operational Performance Standards (MOPS) for user equipment [8-13] and the FAA WAAS Specification [8-16].

Much of the development of real-time algorithms for the computation of WAAS corrections at the Wide-Area Master Station (WMS) has taken place at Stanford. Tsai [8-12] explains the use of prior information in the batch least-squares solution for clock/ephemeris corrections, and Chao [8-3] details the method for computing ionospheric corrections in the MITRE grid format, which is itself described in [8-6]. Chao [8-2] also has developed algorithms for calibrating the slowly changing spacecraft and receiver interfrequency biases as part of the computation of WMS ionospheric corrections.

Dr. Jack Klobuchar at Phillips Research Laboratories is the acknowledged expert on the effects of ionospheric delay on GPS ranging performance. Information of ionospheric spatial decorrelation based on comparisons between fixed dual-frequency receivers stationed hundreds of kilometers apart is given in [8-10]. This data, along with similar results published by MITRE [8-6], forms the basis for the ionospheric error models constructed in Sections 5.0 and 6.0 of this thesis. Other attempts to build models of this type for WAAS user coverage prediction (as described in Section 6.0) have been reported

by Dr. Walter Poor of MITRE [8-15] and Dr. Tysen Mueller of Teal Consulting [8-14]. Recent ionospheric observations processed using the grid algorithm in real time are given in [8-4], which estimates the obtainable grid ionosphere accuracy from the results.

## **1.7 Contributions of this Research**

The primary top-level contribution of this thesis research is the application of the general decision-analysis framework to aerospace system design problems. At a fundamental level, this research illustrates how traditional approaches to the design of spacecraft are sub-optimal and how decision analysis has the potential to reshape them. For application to spacecraft and other aerospace systems, this thesis develops more flexible probability models that better capture uncertainty due to both inherent randomness and limited designer knowledge for a system under development. Monte Carlo simulation is applied to evaluate resulting the resulting performance uncertainty, and flexible optimization techniques that can tolerate evaluation uncertainty are applied to search for optimal design parameters.

None of these methodologies are fundamentally new; references to key sources for all of these categories are included in the preceding section. What is new is their combined adaptation to specific aerospace design problems along with comparisons of the results to those obtainable by more traditional deterministic methods. Because of the presumption of uncertain behavior, it is impossible to prove that probabilistic solutions are superior, but the results of this thesis strongly suggest that they are better tailored to real problems and user value models. The five most significant contributions of a specific nature are listed below:

1. *Uncertainty-Based Engineering Reliability Models*: Exponential failure rates are converted to Weibull distributions whose scale (life) parameters are assigned probability distributions based on the weight of evidence used to generate the original failure rates. A new FMECA-based reliability uncertainty prediction model is also presented and applied to Gravity Probe-B Unlike traditional

reliability approaches, these algorithms are suitable for use in a decision-theoretic framework to make optimal design choices.

2. *Simulation-Based Optimization using Global Search Methods*: For most real-world design problems in which the best possible probability model is derived, no algorithm exists that is guaranteed to find a globally optimal solution in finite time. This research develops and demonstrates the use of new approaches such as Simulated Annealing and Genetic Algorithms for this purpose. These algorithms require many iterations and can be time-consuming, but they can handle very unstructured problems in which Monte Carlo simulation is required to give a evaluation of the mission value function.
3. *GP-B Spacecraft System Design under Uncertainty*: Revised reliability models are combined with simulation-based optimization to demonstrate the application of decision theory to spacecraft design optimization for the first time. The result contains surprises (single-point failure modes) which conflict with traditional design philosophy. A revised solution in which component redundancy is required is derived, but some of the value-function benefit of the first solution is sacrificed.
4. *GPS RAIM Optimization using Prior Uncertainty Models*: This thesis demonstrates in several contexts that a fully general model for user navigation uncertainty can lead to better accuracy and integrity algorithms than those given by traditional “Gaussian i.i.d.” error models. The decision-theoretic method used is flexible enough to be applied to any GPS navigation application, regardless of the prior state of knowledge about specific GPS system failures.
5. *WAAS Performance Prediction and WRS Optimization Approaches*: The use of linear covariance propagation under normal performance conditions is derived in this thesis to predict accuracy and availability for a widespread set of WAAS users. This “coverage prediction” code is very useful in examining the effects on user performance of changing the proposed WRS layout or geostationary satellite augmentation. The predictions generated for a grid of WAAS users can be used as the basis for rare-event simulations that determine optimal integrity performance. Combined with a cost/benefit model, these results can be used by a

genetic algorithm to search for the optimal set of network augmentations, making end-to-end WAAS network design possible for the first time.

The applications of decision theory to aerospace design problems undertaken here for the first time illustrate that the philosophy of decision making under uncertainty has much to offer the field of engineering design. Methods that provide deterministic answers under limiting assumptions will always have a key role in the initial stages of design (as well as in education). However, the tools now exist to go beyond this paradigm where necessary and produce solutions that are robust to an unrestricted picture of system uncertainty and user preferences. Technology is rapidly coming along to support this vision (see Chapter 8), but the fundamental question is the applicability of unrestricted uncertainty models to problems that once had cut-and-dried solutions. Although this thesis clearly demonstrates the feasibility of uncertainty approaches and illustrates its important advantages, it is only a first step in motivating the gradual acceptance of decision theory for the solution of real-world engineering problems.

## **Chapter 2: Decision Analysis Tutorial -- A Canonical Aircraft Control Problem**

*As far as the laws of mathematics refer to reality, they are not certain, and as far as they are certain, they do not refer to reality.*

- Albert Einstein (1879-1955)

The decision-theoretic approaches discussed in Chapter 1 can complement more traditional design methods for problems with arbitrary performance uncertainty, but their underlying philosophy is in conflict with the assumptions of deterministic procedures. It is thus very difficult to directly compare the results of a deterministic procedure (under its normal assumptions) with those of a decision-analysis approach with arbitrary (and flexible) uncertainty and value models. For example, in Chapter 4, a new method for computing thresholds for GPS integrity warning algorithms (based on new failure probability models and airborne user cost functions) is developed and contrasted to the traditional method, which relies on a chi-square distribution to compute the results. This new approach will always do better once one applies the best available uncertainty and value models, but it cannot be proven to be superior in terms of the performance measures of the traditional method. Integration of deterministic and decision-theoretic approaches for a single application is therefore non-trivial.

This chapter uses a fairly simple problem in aircraft control theory to illustrate the differences between the two approaches and how they can be used cooperatively. It also serves as a translation of the theory described in Chapter 1 to the engineering design domain. The chosen problem starts as a textbook example of linear-quadratic controller design for a passenger aircraft conducting a Category I precision approach using the Global Positioning System (GPS - see Section 1.5) augmented by the Wide Area Augmentation System (WAAS), which is the subject of Chapters 5-7. It is then expanded to encompass a simplified model of both WAAS and wind disturbance uncertainty along with a model of the accident risk inherent in wandering too far off the reference approach path. Simulation-based optimization is used to refine the initial LQG controller/estimator design in the presence of this uncertainty. The ability of the simulation approach to

converge on the correct LQG answer (under traditional assumptions - no failures or accident risk) is then demonstrated. Finally, the sensitivity of the optimal design to changes in the parameters of the uncertainty model is examined to demonstrate the connection between deterministic sensitivity analysis and the all-encompassing approach to uncertainty modeling required by a decision-theoretic approach.

## 2.1 Linear-Quadratic Control Theory and Limitations

### 2.1.1 State-Space Dynamic Models

Given a deterministic linear model of plant dynamics for a system to be controlled by state feedback, linear-quadratic regulator (LQR) theory allows the designer to choose feedback gains that minimize state deviations from zero (or a desired trajectory). A brief summary will be given here; complete details can be found in [9-4,8]. The control problem examined here is first defined by the *continuous* time-domain linear plant equations expressed in state-space form:

$$\begin{aligned}\dot{x}(t) &= \mathbf{A} x(t) + \mathbf{B} u(t) + \mathbf{B}_w w(t) \\ y(t) &= \mathbf{C} x(t) + \mathbf{D} u(t) + v(t)\end{aligned}\tag{2.1}$$

where  $x(t)$  is an  $n \times 1$  vector of plant states,  $y(t)$  is the system output vector, which is some linear combination of these states,  $w(t)$  is a vector of disturbances, and  $v(t)$  is a vector of measurement errors. The corresponding *discrete* set of equations is:

$$\begin{aligned}x(k+1) &= \mathbf{A}_d x(k) + \mathbf{B}_d u(k) + w_d(k) \\ y(k) &= \mathbf{C} x(k) + \mathbf{D} u(k) + v(k)\end{aligned}\tag{2.2}$$

where a zero-order hold has been applied. In this Gauss-Markov process model, the states at discrete step  $k+1$  are only dependent on the states at step  $k$ . The equations (involving a matrix exponential) for transforming from continuous to discrete models

depend on the sampling time  $T_s$  and (along with MATLAB code) are detailed in [9-2,3,7]. The discrete format is preferred here because actual controllers in aircraft, as well as computer simulations of control response, are based on the discrete models.

State feedback control allows the designer to choose control inputs based on the states at the last time step in order to improve controller response [9-7]:

$$\begin{aligned} u(k) &= -\mathbf{K}x(k) \\ \therefore x(k+1) &= (\mathbf{A}_d - \mathbf{B}_d \mathbf{K})x(k) + w_d(k) \end{aligned} \quad (2.3)$$

where  $\mathbf{K}$  is the feedback gain matrix. If all unstable modes in (2.1) are *controllable* by state feedback (see [9-8]),  $\mathbf{K}$  can be chosen to give stable steady-state response for the deterministic case ( $w(k) = 0$ ). In the absence of random disturbances  $w(k)$  (and with all other matrix entries in (2.3) known with certainty), the *closed-loop* response of (2.3) is guaranteed to be stable if the closed-loop *eigenvalues*  $\lambda_i$  satisfy  $\lambda_i(\mathbf{A} - \mathbf{BK}) < 0$ , i.e., if all roots  $s$  of the characteristic closed-loop equation [9-7]:

$$\det(s\mathbf{I} - \mathbf{A} + \mathbf{BK}) = 0 \quad (2.4)$$

have their real parts in the left half of the  $s$ -plane (i.e.,  $\text{Re}(s) \leq 0$  for all  $s$  satisfying (2.4)). In the discrete domain, the analogous equation is [9-8]:

$$\det(z\mathbf{I} - \mathbf{A}_d + \mathbf{B}_d \mathbf{K}) = 0 \quad (2.5)$$

and stability is guaranteed if all roots  $z$  of (2.5) are within a unit circle in the complex  $z$ -plane, i.e.,  $\text{Re}(z)^2 + \text{Im}(z)^2 \leq 1$ .

State feedback requires measurements of each of the state variables to be fed back in the form of control inputs. If all states are not directly measured, an estimator loop can be designed to feed back approximations of these variables in real-time. The estimator equations have the form (in discrete space) [9-8]:

$$\hat{x}(k+1) = \mathbf{A}_d \hat{x}(k) + \mathbf{B}_d u(k) + \mathbf{L}[z(k) - \mathbf{C}_s \hat{x}(k)] \quad (2.6)$$

where  $\hat{x}$  is the estimate of the true state vector  $x$  at a given step,  $z$  is the vector of raw measurements from the available sensors, and  $\mathbf{L}$  is the estimator feedback gain matrix. Note that the distribution of states to measurements is defined by the matrix  $\mathbf{C}_s$ , which differentiates the measurements  $z$  from the system outputs  $y$  (governed by  $\mathbf{C}$ ). Defining the estimation error as:

$$\tilde{x}(k) \equiv \hat{x}(k) - x(k)$$

and differencing (2.3) from (2.6) gives a relation which propagates the estimation error forward in time as a function of the plant model and estimator gains [9-8]:

$$\tilde{x}(k+1) = [\mathbf{A}_d - \mathbf{L}\mathbf{C}_s] \tilde{x}(k) \quad (2.7)$$

This estimate has the same format as the controller model; thus similar conditions for stability apply. Note that the estimator uses the knowledge of plant behavior presumed in the controller model (2.3) to filter the raw measurements  $z(k)$ . It may thus be used even when all states can be measured as a means of "smoothing" noisy raw inputs. However, it is dependent on the quality of the underlying physical model and is sensitive to uncertainty in the plant parameters contained in the system matrices  $\mathbf{A}_d$  and  $\mathbf{B}_d$ .

The controller in (2.3) is implemented by feeding back the estimated state  $\hat{x}(k)$  rather than the true state  $x(k)$ . According to the *separation principle* derived in [9-8], the characteristic equation of the combined controller-estimator is simply the product of the controller and estimator model determinants. As a result, the closed-loop poles of the combined system are the same as if the controller and estimator were designed separately and merged as the final step. This greatly eases the task of designing controllers with estimator loops so long as the plant parameters are known precisely or are estimated in real time by an adaptive controller (the former is usually assumed to be the case, but substantial uncertainty often exists and should be accounted for).

### 2.1.2 Linear Quadratic Gaussian (LQG) Controller Formulation

Given that a system is stabilizable by  $(\mathbf{A}_d, \mathbf{B}_d)$  and detectable by  $(\mathbf{A}_d, \mathbf{C}_s)$ , the controller and estimator gains  $\mathbf{K}$  and  $\mathbf{L}$  can be chosen to give desirable closed-loop performance in several ways [9-10]. Pole placement allows the designer to select desirable closed-loop poles (which dictate system response) and find the gain matrices that produce the desired pole locations [9-7]. A more general approach is *optimal control*, which produces system responses that are optimal according to a pre-defined objective function. Discrete time-invariant linear-quadratic design, which can be applied to both controller (regulator - LQR) and estimator (LQE) problems, chooses  $\mathbf{K}$  to minimize a cost function of the form [9-8]:

$$J_{\text{LQR}} = \frac{1}{2} \sum_{k=0}^{\infty} [x^T(k) \mathbf{Q} x(k) + u^T(k) \mathbf{R} u(k)] \quad (2.8)$$

where the process is evaluated at steady-state ( $k \rightarrow \infty$ ).  $\mathbf{Q}$  and  $\mathbf{R}$  are constant weighting matrices that express the relative costs of both off-nominal state responses ( $\mathbf{Q}$ ) and use of controller authority ( $\mathbf{R}$ ). For example, if the baseline  $\mathbf{Q}$  for a four-state system is the  $4 \times 4$  identity matrix ( $\mathbf{I}_4$ ), more weight could be placed on limiting the excursions of state #1 by increasing the first diagonal entry of  $\mathbf{Q}$ . The solution for  $\mathbf{K}$  is based on the steady-state solution to the algebraic Riccati equation [9-4] and can be easily evaluated by computer applications such as MATLAB.

Optimal estimator (LQE) design is performed separately to select  $\mathbf{L}$  to minimize the mean-square error state  $\tilde{x}(k)$  [9-3]:

$$J_{\text{LQE}} = \frac{1}{2} \sum_{k=0}^{\infty} [\tilde{x}^T(k) \mathbf{W}^{-1} \tilde{x}(k) + (z - \mathbf{C}\hat{x})^T \mathbf{V}^{-1} (z - \mathbf{C}\hat{x})] \quad (2.9)$$

where  $\mathbf{W}$  and  $\mathbf{V}$  are the covariances of the random disturbances  $w$  (wind) and  $v$  (sensor error) from (2.2), which are assumed to be white Gaussian noise processes. Note that the solution that minimizes mean-square state estimator error weights the input data based on the best available probabilistic measure (white Gaussian covariance matrices).

### 2.1.3 Robust Control Techniques

In general, robust design techniques attempt to choose controller designs - in this case the feedback gain matrices  $\mathbf{K}$  and  $\mathbf{L}$  - such that the outputs are not highly sensitive to variations in the model form, plant parameter, and/or disturbance parameter values. Several methods have been developed to do this. For the linear time-domain models of Section 2.1.1, the SANDY multiple-model algorithm extends the LQR solution method to a set of plant models  $(\mathbf{A}_i, \mathbf{B}_i)$ , where each model is assigned a probability weight  $w_i$  of being the "true" model [9-11]. A more recent variant of this approach chooses the controller that gives the best performance for the worst-case (normalized) variation from the nominal model [9-13].  $H_\infty$  control methods have been developed which essentially parameterize plant uncertainty in the frequency domain and select controllers that suppress the response to these variations [9-6]. These methods provide considerable theoretical insight into the effects of system uncertainty because they model it using traditional control design tools (linear models, Gaussian noise, transfer-function or state-space descriptions, etc.). They thus share the assumptions and limitations of the theories on which they are based.

In addition to these limitations, practical use of robust design tools is often limited by the assumption that system uncertainty is arbitrarily bounded such that, under controllability and other restrictions, systems can be designed which guarantee stable, satisfactory performance for all possible outcomes. For robustness to relatively small variations about a well-known nominal model, this should suffice, but in a more general sense, robust controller design should acknowledge that in some cases, a guaranteed bound on variations is unrealistic. Assuming bounded uncertainty works fine for many problems as long as sensitivities to parameter errors are carefully studied. However, in practice, there will often be a non-zero (and non-trivial) probability of unacceptable results because it is impossible to protect against everything that can go wrong.

Many bounded uncertainty models focus on parameter variations due to imperfectly identified, simplified, and/or linearized system models, neglecting to some degree the prospect of rare but disabling *system failures* that can radically alter performance. Decision theory demonstrates in very general terms that solving a problem

(and making a decision) based on an uncertainty model that ignores these possibilities will produce a sub-optimal design. Once it is acceptable to use models with non-zero failure probabilities, the issue becomes finding the design that provides the optimal trade-off between nominal (or near-nominal) performance and risk of failure (defined as not meeting the original system requirements). This task is the focus of the design optimization projects carried out in this thesis.

The potential gap between theoretical robust control and actual uncertain behavior has motivated research on using more flexible tools to evaluate this performance-risk tradeoff. Monte Carlo simulation of controller behavior under uncertainty was used in [9-16] to evaluate a probabilistic LQG objective function. Genetic algorithms (discussed in Section 3.3.2) were used in [9-12] to search for optimal gains, demonstrating that simulation-based optimization under unbounded uncertainty is feasible. This research coined the term "stochastic robustness" and applied it to aircraft control systems, the subject of this chapter, in [9-14]. Because simulation can evaluate *any* performance measure and any set of model equations, controllers can be optimized with respect to top-level system requirements rather than the more limiting linear-quadratic formulations of Section 2.1.2.

Using probabilistic simulation and evolutionary search techniques, it is thus possible to apply the principles of decision theory to controller optimization under uncertainty. Preference models can be designed based on the principles of Section 1.2.2. and Appendix A, which weigh all possible outcomes (including various levels of system failure) against the value of successful performance. Monte Carlo simulation of controller performance under an *unbounded* navigation and wind failure model allows this value model to be evaluated for each trial solution produced by an evolutionary search algorithm. In addition to showing the result achieved for a linear autopilot problem, this chapter aims to demonstrate the application of decision theory to a fairly simple problem and to identify the value of applying such an approach in addition to (or instead of) traditional robustness techniques for problems where bounded uncertainty is an unrealistic assumption.

## 2.2 Aircraft Dynamic Model and LQG Controller

### 2.2.1 Longitudinal 747 Aircraft Dynamic Model

A dynamic model for the longitudinal motion of a Boeing 747 near sea level in a low-air-speed ( $V = 221$  ft/s) configuration on final approach is given in [9-4] based on a linearization about steady level flight. The six states in this model are expressed as perturbations about the nominal condition:  $x = [\text{airspeed } (u_a), \text{ vertical velocity } (w_a), \text{ pitch rate } (q), \text{ pitch angle } (\theta), \text{ glidepath deviation } (d), \text{ throttle } (\delta_t)]$ . Two longitudinal control inputs are available:  $u = [\text{elevator } (\delta_e), \text{ throttle command } (\delta_{tc})]$ . Note that the throttle includes a first-order model of 4-second time delay between command  $\delta_{tc}$  and response  $\delta_t$ . This details of this model (including values for the continuous and discrete plant parameter matrices and nominal covariance matrices) are contained in Appendix B (pp. 276-278).

In the LQG problem,  $w(t)$  is a random input vector expressing wind disturbances in both horizontal and vertical axes, and  $v(t)$  is a vector of random errors in the three sensor inputs: airspeed, pitch angle, and vertical glideslope deviation. An inertial measurement unit (IMU) provides the first two measurements, while WAAS provides a measure of absolute vertical position that is compared to the ideal  $3^\circ$  glideslope. Disturbance and measurement covariance models are also given in Appendix B, along with the diagonal matrix  $X_0$  of initial state covariances.

### 2.2.2 LQG Optimal Controller for Aircraft Precision Approach

As outlined in Section 2.1.2, discrete LQR design minimizes the following index chosen for the precision approach problem:

$$J_{LQR} = \frac{1}{N_t} \sum_0^{N_t} \left[ u^2 + 10 d^2 + 10 \delta_t^2 + 10 (\delta_e^2 + 0.333 \delta_{t_i}^2) \right] \quad (2.10)$$

where in this case, the multiplier  $(1/2)$  from (2.8) has been replaced by  $(1/N_t)$  - the result does not change. Note that  $J_{LQR}$  is dependent on both aerodynamic control inputs, the lagged throttle state  $\delta_t$ , as well as the airspeed and glideslope deviation states (these three are output states; thus the output matrix  $C = \text{diag}[1, 0, 0, 0, 1, 1]$ ). The implied state and

control weighting matrices are  $\mathbf{Q} = \text{diag}[1, 0, 0, 0, 10, 10]$  and  $\mathbf{R} = \text{diag} [10, 10/3]$ . The glideslope deviation ( $d$ ) has the highest weight because it is the most critical parameter in the RNP tunnel evaluation of system performance, but the weights also limit controller authority and place some importance on "smooth" aircraft motion down the ideal glideslope. Note that the weight on glideslope deviation was increased from 1 (as used in [9-2]) to 10 in order to minimize terminal glideslope deviations, which translate into accident risk according to the value model developed in Section 2.4. Optimal LQR controllers for higher weights on  $d$  were also examined by simulation with the ideal noise model, but these did not provide substantially better glideslope deviation suppression.

This example illustrates a key difficulty of LQR design for real-world problems: the weights  $\mathbf{Q}$  and  $\mathbf{R}$  are not directly translatable into top-level performance preferences; thus "trial and error" is often applied to search for the weights that give the "best-looking" performance. Guidelines for setting weights (or solving for them in an "inverse" fashion) exist [9-2], but this disconnect raises a key question: an LQR solution can be proved to be optimal, but what is it optimal for? This issue is particularly important for "risk-sensitive" LQG methods that attempt to find robust optimal controllers by minimizing a risk-averse variation of the LQR/LQE performance indices. This risk-sensitive index is an exponential of the basic LQ functions (2.8-9) that follows the "delta neutral" property discussed in Section 4.2.6 [9-17]. Risk-sensitive LQG controllers should be more robust than standard ones, but the key problem is that the basic LQ objective function is not a direct measure of top-level user preferences. This makes it difficult to model risk aversion according to the decision-analysis axioms of Appendix A.

Time-invariant LQR design carries a further restriction: it assumes that the system is optimized based on steady-state performance as time (or the number of steps) goes to infinity. The LQR cost (2.8) can then be computed directly based on known properties of the steady-state response [9-9]. However, in this case, we are only interested in glideslope tracking from 1500 to 200 ft altitude, which is lightly damped and highly oscillatory. Thus, the steady-state LQR cost is not exactly the quantity of interest. Section B.2 of Appendix B details a method for numerically computing the LQR cost at each step by propagating state and state-error covariance matrices forward in time (for pre-defined

controller and estimator gains). As explained in Section 2.4, Monte Carlo simulation is required once non-ideal failures are introduced, and it can be used in the ideal case as well at the cost of incurring statistical uncertainty. This is demonstrated and discussed further in Section 2.5.

Based on the separation principle of Section 2.1.1, the associated Kalman estimator is based on the weights defined by the uncertainty covariances  $\mathbf{W}_e$  and  $\mathbf{V}$ . Unlike the LQR case, the linear-quadratic estimation solution has a fundamental basis: if  $\mathbf{W}_e$  and  $\mathbf{V}$  truly represent the best possible model of uncertainty, the Kalman estimator solution to (2.9) gives the *minimum error variance* (or *maximum likelihood*) estimator in a statistical sense [9-3,4, 3-3]. However, in many cases, the white Gaussian model of uncertainty is known to be flawed, implying (in decision-theoretic terms) that the resulting estimator is sub-optimal. One approach in this case is simply to increase the Gaussian variances to obtain a "conservative" design. Time correlation in a given error channel can be added to the plant model for cases where the randomness is not truly white, and correlation between different errors can be modeled by non-zero off-diagonal terms in the covariance matrices. But if the basic Gaussian error model is incorrect (or incomplete) or if there is substantial uncertainty in the covariance entries and correlation coefficients, the LQG estimator is not optimal in the most general sense.

For the 747 longitudinal approach problem and performance indices given here, the LQG-optimal controller and estimator gains are as follows:

$$\begin{aligned} \mathbf{K} &= \begin{bmatrix} -.3448 & .9842 & -1.9095 & -3.8728 & -.4649 & -1.0952 \\ .6558 & -.7426 & 1.5684 & 2.8563 & .3634 & 2.4937 \end{bmatrix} \\ \mathbf{L} &= \begin{bmatrix} .0312 & -.0211 & .0010 & -.0063 & -.0106 & 0 \\ -.0106 & -.0462 & .0010 & .0148 & .3601 & 0 \\ -.0002 & .0014 & 0 & .0008 & .0009 & 0 \end{bmatrix}^T \end{aligned} \quad (2.11)$$

This optimal solution has an expected cost for the ideal case of  $J_{LQR} \cong 3400$  (the overbar that denotes "expected value" is deleted for clarity from this point on) and RMS glideslope deviation  $\cong 18.1$  ft at the terminal altitude (200 ft). This result will be compared to what is obtained once a non-ideal uncertainty model is applied in Section 2.5.1.

## 2.3 Performance Uncertainty Model

### 2.3.1 LQG Controller Sensitivity Analysis

The disturbance and noise models assumed for the LQG controller derived in the last section hold reasonably well for normal conditions (which are assumed to exist at least 95% of the time for well-designed systems). For many systems whose rare-event performance is well documented and understood, Gaussian error models adequately represent all foreseeable performance uncertainty. However, for this problem, insufficient knowledge exists to fully characterize WAAS as a navigation sensor in this simple way. Because WAAS has not yet been fully implemented over CONUS and has not yet experienced ionospheric errors at the peak of the 11-year solar cycle, there is reason to believe that WAAS performance in rare cases is worse than that predicted by a simple Gaussian distribution. Data in [8-10] supports this suspicion (see the detailed discussion in Section 5.2.3). Furthermore, wind gusts in rare cases are likely to be worse than indicated by the white-noise Gaussian model in  $W_c$  even though this model may fit well within 2 or 3 $\sigma$  of the mean.

Faced with this uncertainty, one can examine the sensitivity of the LQG controller to changes in the variances in the matrices  $W_c$  and  $V$ . Table 2.1 shows to what degree the performance index (2.8) and root-mean-square (RMS) terminal glideslope deviation degrade when each of the diagonal entries in these noise covariances is (separately) *multiplied by four* (i.e., the standard deviation is *doubled*), assuming that the optimal LQG solution (2.11) is unchanged. The results show that increased noise variance has little

Noise Parameter	Increased $\sigma$	Increase in $J_{LQR}$	Inc. in d(200 ft)
airspeed	4 (ft/sec)	2.1 %	6.1 %
pitch angle	4 (crad)	4.2 %	7.8 %
WAAS glideslope dev.	16 (ft)	1.9 %	6.2 %
wind gusts (hor./vert.)	1.9 (ft/sec)	282.9 %	120.2 %

**Table 2.1: LQG Sensitivity Analysis Results**

effect on the final result, but increased wind disturbance variance has a major effect on both steady-state ( $J_{LQR}$ ) and terminal ( $d(200 \text{ ft})$ ) performance. Wind model uncertainty thus dominates the problem and is a key issue for autopilot design, as indicated by Boeing in informal discussions (also see [9-15]).

If we could ensure that the underlying model is white and Gaussian and can be bounded by the increased variances evaluated here, we could conclude that the optimal solution is robust to any conceivable variation in a single sensor noise parameter, since the results for increased sensor noise in Table 2.1 are acceptable. The robust control approaches mentioned in Section 2.1 can expand on this by changing the ideal LQG solution and finding a revised solution that is robust to the entire (bounded) range of noise (or plant) parameter variances [9-2,6]. In this case, however, when the LQG estimator is redesigned with the altered wind model of row 4 of Table 2.1 to give:

$$\mathbf{L}_{\text{worst-case}} = \begin{bmatrix} .0860 & -.0750 & .0008 & -.0025 & -.0096 & 0 \\ -.0096 & -.1388 & .0016 & .0091 & .4912 & 0 \\ -.0016 & .0053 & 0 & .0028 & .0006 & 0 \end{bmatrix}^T \quad (2.12)$$

$J_{LQR}$  is still increased by about 140%, while the RMS terminal glideslope deviation increases by 62% (to 29.4 ft). These results are better than those for the unmodified estimator in Table 2.1, but they are still difficult to tolerate. Design effort would then turn on finding a tighter upper bound on wind variance so that acceptable performance could be guaranteed within a tighter envelope (this is what is done in autopilot certification simulations [9-15]).

In the decision analysis framework, sensitivity studies of this type help identify the *value of (additional) information* (introduced in Section 1.2.1) on the uncertainty of the parameters in the disturbance/noise model. It is clear from Table 2.1 that information that would allow the wind uncertainty to be reduced has high value in terms of the objective function  $J_{LQR}$ . New information that increases the wind uncertainty would also be very valuable, but in a negative fashion, as it would illustrate the limitations of even the optimal feedback control law in such an environment. However, the value of information cannot

be evaluated properly unless the decision-maker begins with a valid uncertainty model. Since the ideal model ( $W, V$ ) for the LQG controller artificially constrains the uncertainty space, one cannot extrapolate the "true" value of information directly from it. Sensitivity analysis can thus help isolate the key effects of uncertainty on nominal performance, but it is difficult to examine the full space of uncertainty from an idealized noise model or deal with cases where uncertainties cannot be satisfactorily bounded.

### 2.3.2 *Prior Probability Model for WAAS/Wind Uncertainty*

Given the current state of information on WAAS and wind performance, there is no reason to have high confidence that we can bound all of the uncertain variables to support a guarantee of acceptable performance. In this situation, it is difficult to evaluate LQG system response under uncertainty if we are limited to white Gaussian noise models with fixed variances. Decision making under uncertainty suggests that we instead attempt to model both the likelihood of abnormal noise and our uncertainty about the degree to which the noise variance will increase. Table 2.2 shows a simplified uncertainty model that affects WAAS noise and wind gusts under rare conditions. It is approximated from the studies of Chapters 5-7 and includes:

1. *Sudden GPS/Inmarsat satellite outages:* The loss of a single satellite ranging source should not force the aircraft to abort its approach, but it will increase the resulting WAAS vertical position error by an amount that depends on the difference between the original and reduced (or "subset") Vertical Dilution of Precision as defined in Section 4.1.2.
2. *Ionosphere/Troposphere disturbances:* As detailed in Sections 5.2 and 5.3, atmospheric disturbances that are not well-modeled by the WAAS correction for a given satellite may result in substantially increased ranging error. If the disturbance is sufficiently localized, the WAAS network may not realize that its broadcast corrections are erroneous.

3. *Missed WAAS correction messages*: The most likely rare event to affect WAAS users is missing one or two message updates which will be sent out every 6-10 seconds. Each message contains a “fast” update to allow the user to correct for Selective Availability (SA - see Sections 1.5 and 5.2.2) if it is present. If messages are missed, the SA error contribution will increase according to the following equation (derived from (5.1)):

$$\sigma_{\text{add,SA}} = 0.5 a (10 + 10n)^2 (m) \quad (2.13)$$

where  $n$  is the number of consecutive missed messages and  $a$  = SA acceleration  $\cong 0.004 \text{ m/s}^2 \cong 0.013 \text{ ft/s}^2$ . If  $n > 2$ , the user must perform a *missed approach*.

4. *User receiver/processing errors*: Since user aircraft have multiple receivers, they should be protected against local errors due to malfunctions in a single hardware or software element. However, in rare cases, it may be possible for the same error to manifest itself in all user channels, leading to significantly higher errors.

The effects of these WAAS performance uncertainties can be included in the broader system uncertainty context by also modeling the wind gust variance as an uncertain parameter. In operational autoland simulations, a wind gust model is assumed beforehand and taken as gospel. Autopilot performance is tested in two cases: the normal case (given by  $W_c$ ) and a limit case where the wind variances are increased and degraded performance is expected. As noted in Section 2.3.1, the results turn out to be very sensitive to the wind model used [9-15]. For a decision-analysis approach, it is better not

Rare Event	Probability	Effect
GPS satellite outage	0.0001	$N(1.5,.5) \times \text{error } \sigma$
iono/troposphere disturbance	0.005	$N(2.5,1) \times \text{error } \sigma$
missed WAAS correction	0.009	+ error function( $n$ ) (2.13)
user receiver/processing failure	0.001	$N(3,1) \times \text{error } \sigma$
high wind gusts	0.005	$N(1.5,.5) \times \text{wind } \sigma$

**Table 2.2: WAAS Failure Model Summary**

to artificially bound the possibilities in this way; thus Table 2.2 includes a random multiplier for wind along with a probability of worse-than-normal wind behavior.

### *2.3.3 Simulation of Prior Probability Model*

The total probability of the events in Table 2.2 (which are assumed to be mutually exclusive for the purposes of this application) is 0.02. Normal conditions thus hold about 98% of the time and do not require sampling from the failure distributions. Normal-case and failure-case performance are therefore simulated separately. When simulating failure cases, one (and only one) rare event is assumed to occur just after the initial condition at 1500 ft altitude, avoiding the need to sample repeatedly over the course of the approach (but also limiting the scope of possible failure effects). A uniform random variable  $U(0,1)$  is sampled at the beginning of each failure approach simulation and is divided by the total failure probability (0.02) to determine which of the five failure classes in Table 2.2 is present. A Gaussian sample is then taken to determine the error multiplier that is applied to the standard deviation of the WAAS entry (row 3, col. 3) in  $V$  or to  $W_c$  for increased wind gusts.

Once revised error matrices are determined (and discretized as done in Appendix B), the time-step simulation of a given approach is conducted in one-second steps. The nominal descent from 1500 to 200 ft altitude takes 110 seconds; thus  $N_t = 110$  samples of Gaussian distributions are generated for  $w(k)$  and for each of the three elements of  $v(k)$ . The discrete state-space equations (2.2) are then propagated forward from a Gaussian initial condition sampled from the covariance  $X_0$ . The state and control histories  $x(k)$  and  $u(k)$  are then determined and used to evaluate the revised performance indexes described in the next section.

## **2.4 Value Model and Optimal Controller Search**

### *2.4.1 Accident Risk Cost Assessment*

For this Category I study, it is assumed that precision guidance is relied upon until the minimum ceiling of 200 ft is reached, although in many cases the pilot will reach a

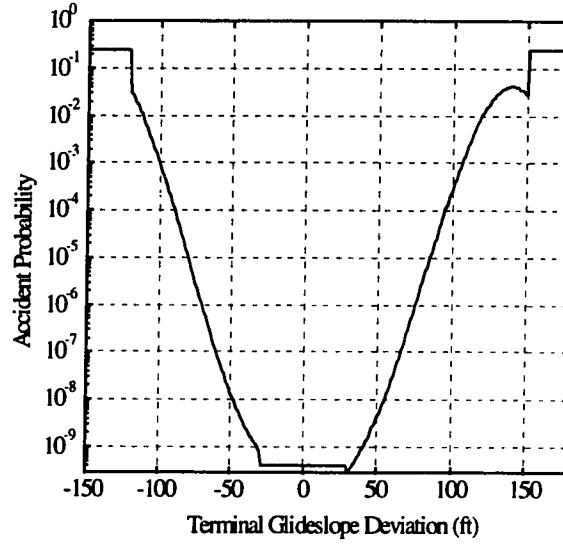
zone of adequate visibility above this altitude. The RNP tunnel concept (discussed in more detail in Section 4.3) defines a  $10^{-7}$  *outer tunnel* for Category I approaches of total system error (TSE), which combines navigation sensor error (NSE - error in the navigation signal relative to the ideal “true” glideslope) and flight technical error (FTE - the error in the autopilot flying the glideslope deviation signal). This tunnel gradually narrows to a vertical limit of  $\pm 110$  ft error at 200 ft (the horizontal dimension is much wider and is much less of a concern for WAAS - see Section 5.2). Penetrating this surface is assumed to result in a 1% chance of an accident [6-10].

Based on this limited information, a cost model which measures the risk of an accident can be built under the premises of decision theory. This risk is measured based on the deviation at the terminal point  $d(200 \text{ ft})$  only. Total system errors of less than  $\pm 30$  ft are assumed to carry a base risk of  $5 \times 10^{-10}$  per approach (other failure modes form the lower limit of risk for small errors). Accident risk then increases faster than linearly until a saturation point is reached. Negative errors (plane below glideslope) leave less room for recovery and are assumed to be 3 times worse than positive ones of the same magnitude. Accident risk for negative errors is evaluated for  $d(200 \text{ ft})$  based on the following cubic function which is valid for  $30 \leq -d \leq 120$  ft (simply divide the result by 3 for  $d > 0$ ):

$$\begin{aligned} \text{Log}_{10} \{ \text{Pr}(acc) \} = & -9.37 \times 10^{-6} |d^3| + 2.23 \times 10^{-3} |d^2| \\ & - 7.40 \times 10^{-2} |d| - 8.59 \end{aligned} \quad (2.14)$$

As explained above, for  $|d(200 \text{ ft})| \leq 30$  ft, the base risk of  $5 \times 10^{-10}$  is assessed, and errors outside the range  $[-120, 150]$  ft have a saturation risk of  $\text{Pr}(acc) = 0.25$ . This function is plotted on semilog scale for both positive and negative glidepath deviations in Figure 2.1. This function is plotted on semilog scale for both positive and negative glidepath deviations in Figure 2.1. Note that the slightly non-smooth corners of this plot are due to fitting the cubic function (2.14) onto the actual risk model for ease of computation -- they do not cause problems during optimization.

Note that this is a simplified model for the purposes of this example problem. It is clear that the resulting accident risk for a given controller will vary as the risk parameters in this model are changed. For example, if the base risk were  $10^{-10}$  instead of  $10^{-9}$ , small error cases would be more advantageous relative to those with errors in the 30-50 ft range. Nevertheless, until more information is obtained, the minimum-cost controller based on this model is optimal under the precepts of decision theory.



**Figure 2.1: Terminal Deviation Accident Risk**

#### 2.4.2 Combined Risk and LQR Cost Function

The final cost of a given controller balances accident risk with near-nominal LQG performance. A linear scale is used for  $\Pr(acc) \leq 10^{-4}$  in which a risk of  $10^{-8}$  (which is 25% of the ORD requirement of  $4 \times 10^{-8}$  for Category I precision approach [6-11]) is assigned to be equivalent to an LQR cost of 3000, which roughly represents optimal nominal performance. Once again, an arbitrary cost parameter is applied based on the user's projected preference for risk reduction versus off-nominal performance. In this case, a doubling of the risk to  $2 \times 10^{-8}$  is equivalent to an LQR cost of 6000, thus users will not stomach increases in risk beyond the ORD requirement (since they would overwhelm any possible nominal performance gain). The total cost is then:

$$J_{\text{total}} = J_{\text{LQR}} + J_{\text{risk}} = J_{\text{LQR}} + 3000 \Pr(acc) / 10^{-8} \quad (2.15)$$

While it may appear that this cost model was derived arbitrarily, it was in fact done according to the precepts of decision theory, which poses very few restrictions on the

preference functions that are "valid" for quantitative analysis. As explained in Appendix A, it requires that:

1. *Probability Rule*: Uncertainty must be completely and honestly expressed in terms of probabilities. Within the limits of this example problem, this is provided by the PPM in Section 2.3.2. Note that uncertainty in that model is unbounded such that there is always non-zero probability of an accident.
2. *Order Rule*: All possible outcomes are clearly ordered from best (nominal performance with minimal  $J_{LQR}$  and zero terminal glideslope deviation) to worst (aircraft accident).
3. *Equivalence Rule*: This rule is satisfied by equating a given level of nominal performance with a given level of accident risk. In this case, the "certain equivalent" of a lottery in which the plane crashes with probability 1 in  $10^{-8}$  and is fine otherwise is simply  $J_{LQR} = 3000$ .
4. *Substitution Rule*: This requires that we be willing to substitute a certain equivalent (nominal performance in this case) for the equivalent lottery in the process of decision evaluation. This is done by  $J_{total}$  in (2.15), which combines nominal and risk-based cost into a single performance index, and all other possibilities are ranked (in consistent order) according to this function.
5. *Choice Rule*: This is satisfied by the evolutionary optimization process which selects the best design according to the lowest cost (2.15) found.

It should be clear that decision theory allows any preference function that is mathematically consistent and is based on the best-possible probabilistic rendering of the underlying system uncertainty. Note that the goal of optimization is now not just optimal LQR controller performance but a broader measure of user acceptance. The intervening step of setting specifications which must be met by lower-level design is avoided, along with the inefficiency that results (as demonstrated in the rest of this thesis).

### 2.4.3 Optimal Search using Simulated Annealing

*Simulated Annealing* (SA) is an evolutionary optimization technique, so called because the optimal solution "evolves" in a series of gradual permutations of a current "best" solution. It is described more fully in Section 3.3.1. Here, SA is used to search for the controller design that minimizes the sum of LQG and accident risk costs as shown in (2.15). SA operates in several iterative loops within which new solutions are generated and evaluated, then a probabilistic decision is made on whether or not to accept each new solution.

Beginning with the ideal LQG solution in (2.11), a new solution is generated in each trial by perturbing the gains from the last solution  $\mathbf{KL}_0$  by a  $\text{Normal}(\mathbf{KL}_0, 0.1\mathbf{KL}_0)$  random sample. Only the elements of  $\mathbf{K}$  and the 3<sup>rd</sup> column of  $\mathbf{L}$  (which uses WAAS measurements) are varied, giving 18 design variables. The other two columns are not varied for this problem because they affect the airspeed and pitch angle measurements, which are not part of the failure model. Note that this *does not* mean that the optimal solution to this problem has the LQG gains for airspeed and pitch angle, but this assumption is made here to simplify the search.

For each new solution, 100 approach simulations are run in which at least one failure occurs. The failure state is sampled at the initial approach point (1500 ft) out of the 2% total failure probability from Table 2.2. No-failure results can be computed from the deviation covariance given by the discrete Gauss-Markov process [9-3]. The no-failure and failure costs (both cases are evaluated using (2.15)) are given 98% and 2% weights, respectively, to get the final cost,  $J_{\text{eval}}$ .

Simulated Annealing always adopts a new solution with lower cost than the old one. If the new solution cost is higher, it will still be randomly accepted with probability:

$$\Pr(\text{accept}) = -\left(J_{\text{eval}}^{\text{new}} - J_{\text{eval}}^{\text{old}}\right) / T \quad (2.16)$$

where  $T$  is a "temperature" that starts at 1000 (accepting most early changes whose cost increase from the prior solution is 500 or less) and decreases by 10% after each set of 30 new solutions. While the high early temperature helps the search avoid becoming trapped

in local minima, it becomes less tolerant of changes that do not give a lower cost as the number of trials increases.

## 2.5 Baseline and Optimal Controller Results

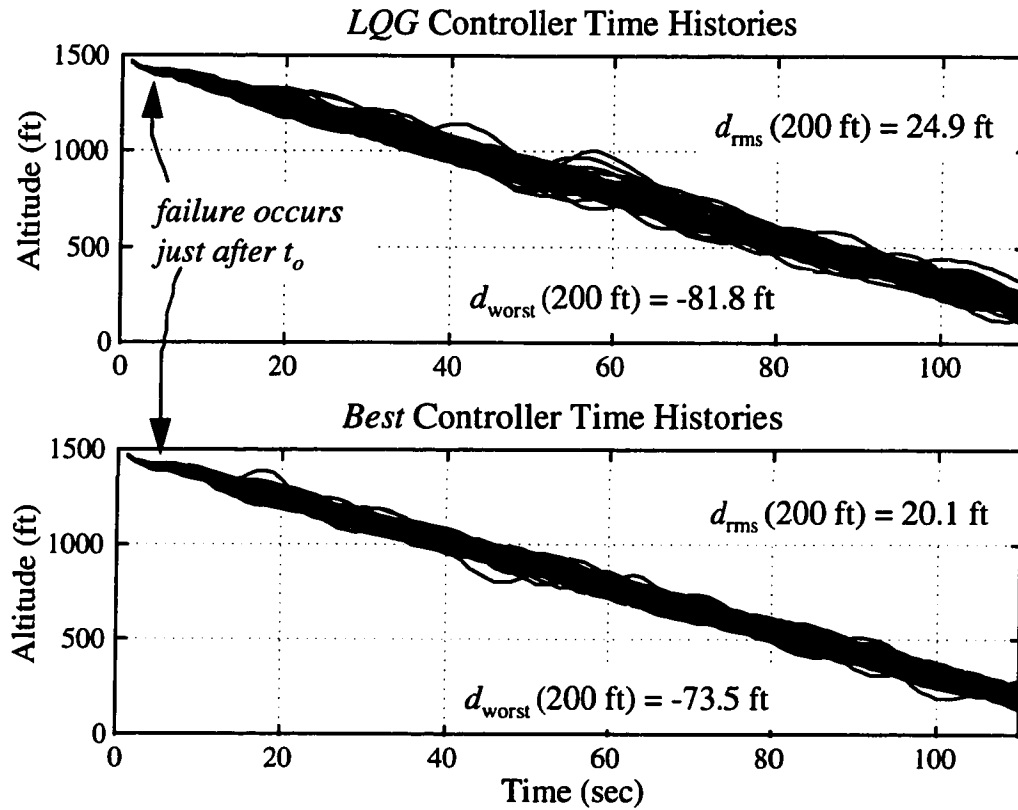
### 2.5.1 Optimal Controller for PPM and Risk Model

The LQG solution from (2.11) has  $J_{\text{eval}} \cong 4520$ ;  $\text{Pr}(\text{acc}) \cong 3.3 \times 10^{-9}$  when evaluated by simulation. The best solution found by Simulated Annealing is:

$$\begin{aligned} \mathbf{K}_{\text{opt}} &= \begin{bmatrix} -0.3026 & 1.3216 & -1.0311 & -3.7337 & -0.8176 & 1.4158 \\ 0.7803 & 0.1187 & 3.6723 & 1.0848 & 0.1848 & 4.8164 \end{bmatrix} \\ \mathbf{L}_{\text{opt}}(\text{col. 3}) &= [-0.0002 \quad 0.0025 \quad 0 \quad 0.0011 \quad 0.0029 \quad 0]^T \end{aligned} \quad (2.17)$$

This result has  $J_{\text{eval}} \cong 4030$ ;  $\text{Pr}(\text{accident}) \cong 1.2 \times 10^{-9}$ , which is a substantial improvement over the steady-state LQR solution. The SA solution results in a decreased cost due to accident risk of 630: 990 (for LQR) - 360 (for SA), while it only gives up 140 in nominal performance: 3670 (for SA) - 3530 (for LQR).

Figure 2.2 shows a set of 500 *failure-case* glideslope time histories for both solutions and shows that the best controller does a better job of avoiding large deviations. This figure indicates both the RMS terminal glideslope deviations and the highest-risk deviations found in the 500 simulations. For failure cases, the optimal controller improves RMS terminal deviation by about 20% when compared to the LQG solution. Worst-case terminal deviation improves about 10%, but this understates the risk reduction achieved, as modeled by (2.14). The accident risk in the LQG worst case is  $1.4 \times 10^{-5}$ , which is *seven times* the worst-case accident risk of the best design found by simulated annealing ( $2.0 \times 10^{-6}$ ). Note that in both cases, the "worst-case" for accident risk is a negative glideslope deviation.



**Figure 2.2: Failure Approach Simulation Results**

### 2.5.2 Use of LQR Terminal Controllers

Note that better terminal performance could also be obtained by using time-varying gains, as described in [9-2,3,4]. This makes the controller and estimator gains variable at each time step. In terminal LQG design, a separate quadratic cost is applied to the deviation at the terminal point. Once one decides to apply time-varying gains, the terminal LQR controller will do a much better job of suppressing terminal deviations. However, the terminal LQG design is still dependent on noise and cost model assumptions that are inaccurate; thus a better design (in decision-theoretic terms) would again result from application of the PPM and the LQR/risk value model along with SA optimal search.

In this case, however, the resulting controller gain matrices are functions of time, giving many hundreds of gain entries over the 110 one-second time updates in the Category I approach modeled here. While evolutionary search has demonstrated the

ability to handle problems with hundreds of design variables (see [2-2,5]), a shortcut is available here: use the terminal-LQG solution calculation but make the entries in the weighting matrices  $\mathbf{Q}$  and  $\mathbf{R}$  the design variables for optimal search (a total of 40 variables for the precision approach problem). The PPM and cost model (2.15) can then be used (with modification for the changed definition of  $J_{LQR}$ ) with SA to search for the optimal time-varying solution given by the "optimal" weighting matrices.

### 2.5.3 *Simulated Annealing Solution for the Ideal Case*

Evolutionary optimization approaches such as simulated annealing can search for the best solution to a problem of arbitrary mathematical form, but they cannot guarantee that the global optimum will be found in finite time. As a result, they are relatively "expensive" tools for optimal design which should be used only when traditional approaches (and their mathematical assumptions) do not fit the problem at hand. This is usually true for real-world design, but it is also useful to examine how well the general search approach handles an ideal problem for which a known solution exists.

As noted in Section 2.2.2, the LQR controller and LQE estimator are known to globally minimize the unmodified linear-quadratic objective functions (2-8,2-9) under the ideal error assumptions (i.e., no failures, white Gaussian errors as modeled by  $\mathbf{W}$  and  $\mathbf{V}$ ). Starting from an arbitrary set of gain matrices, simulated annealing search can be applied to this problem, and the resulting controller can be compared to the known optimum derived using LQG in (2.11). Although a deterministic evaluation of  $J_{LQR}$  is possible from the equations in Appendix B and could be used by SA, simulation will be used instead (without sampling from the PPM of Section 2.3.2) in order to preserve the uncertainty characteristic of the general decision-theoretic approach. The problem is simplified by only optimizing the controller gains; the correct LQE estimator is used throughout the search. This assumes that stochastic separation is maintained as in Section 2.1.1, but there is no reason why the SA design space could not be expanded to refine controller and estimator gains at the same time.

For this problem, the SA search was initialized with the "wrong" LQR solution. The ideal LQR controller for a different state weighting matrix  $\mathbf{Q}_b = \text{diag}[1, 0, 0, 0, 2, 2]$

was chosen (recall that the "correct"  $\mathbf{Q} = \text{diag}[1, 0, 0, 0, 10, 10]$ ). The baseline controller gains are then:

$$\mathbf{K}_{\text{base}} = \begin{bmatrix} -1.1886 & .5830 & -1.2990 & -2.2964 & -.2330 & -.8588 \\ .6035 & -.5624 & 1.3601 & 2.1650 & .2394 & 2.2532 \end{bmatrix} \quad (2.18)$$

These gains are all of lower magnitude than the optimal result of (2.11); they range between 50-92% of the correct values on an element-by-element basis (all signs remain the same). Simulated annealing begins with this solution, which has an LQR cost (using the correct weights) of  $J_{\text{LQR}} \cong 4385$ , and searches for the known optimal result using the parameters given in Section 2.4.3 (but without failure sampling). After 26 outer-loop iterations (the search temperature being reduced after each), the search converged to the following solution:

$$\mathbf{K}_{\text{opt}} = \begin{bmatrix} -.2725 & 1.1689 & -2.2955 & -5.1491 & -.8073 & -.3350 \\ .4703 & -.8484 & 1.4550 & 1.9885 & .2431 & 2.7172 \end{bmatrix} \quad (2.19)$$

Figure 2.3 plots the z-plane closed-loop controller poles (the eigenvalues of  $\mathbf{A}_d - \mathbf{B}_d\mathbf{K}$ ) of both the LQR controller (2.11) as 'x' and the new SA controller as '+'. At first glance, these two solutions are a little different. The two real-axis roots are very similar, but the SA controller has asked for faster response from the glideslope deviation pole-pair (the furthest left pair of poles) at the expense of another pair which has become a little more oscillatory (moved further right), signifying that slightly more control

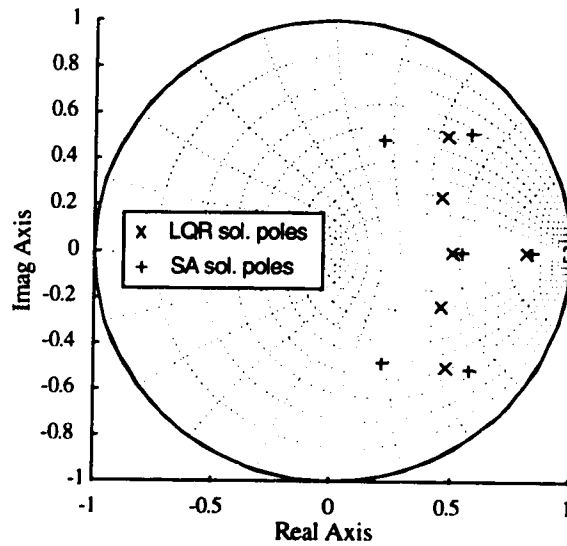


Figure 2.3: Closed-Loop Pole Location Comparison

authority is used in by the *SA* controller to make the glideslope deviation decay faster. The average difference between the gain entries in  $\mathbf{K}$  for the two solutions is 30%, although the largest difference is 73% for the [1,5] entry.

An interesting result emerges: the ideal (no-failure) cost of this controller is  $J_{\text{LQR}}(\text{SA}) \cong 2840$ , which is somewhat *lower* than the ideal cost for the steady-state LQR controller (2.16) of about 3400. This is caused by the fact that simulation over a finite set of points in time produces a cost evaluation that is different from the steady-state result, for which steady-state LQR is the optimal solution. The analytical LQR solution considers that time-invariant LQR cost (2.8) is based only on steady-state conditions (since early costs are negligible over an infinite time scale), while the simulation which evaluates LQR cost for *SA* search terminates at 110 seconds and only considers the cost to that point. Thus, initial response and control usage is important to the simulation. In this case, the LQR cost breakdown between state deviation cost  $x^T \mathbf{Q}_d x$  and control usage cost  $u^T \mathbf{R}_d u$  shows that the controller designed by LQG (2.11) suffers an average state cost of 3320 and an average control cost of 80 over the 110-second simulation, compared to 2705 and 135 respectively for the *SA* controller (2.19) (the effect of the cross-term  $2x^T \mathbf{N}_d u$  that results from the discretization of Section 2.2.1 is negligible). RMS glideslope deviation at the terminal point (200 ft altitude) reflects this as well: it is 17.7 ft for the steady-state LQR controller and 15.6 ft for the *SA* controller (both of these are better than the failure-state results presented in Section 2.5.1). This confirms that the *SA* controller uses a little more controller authority to improve state performance in the short run, whereas the LQR controller is optimal at steady-state only and has less incentive to respond quickly.

It turns out that the LQR cost and RMS glideslope deviation performance are very sensitive to the initial state covariance  $\mathbf{X}_0$ . The diagonal matrix for  $\mathbf{X}_0$  given in Appendix B (and taken from [9-2,3]) has a very optimistic initial variance for glideslope deviation (the fifth diagonal entry), representing total system error, of  $(1.32 \text{ ft})^2$ . This is probably too good even for a Category III ILS and autopilot, and it is certainly better than the achievable steady-state performance with WAAS. The *SA* controller exploits this very good initial performance with increased state feedback, resulting in better RMS state response. The ideal controller has no knowledge of it and cannot do the same. However,

if higher initial deviation variances are used, the ideal controller becomes superior. In fact, if it is assumed that the aircraft uses unaugmented GPS (affected by Selective Availability) before acquiring the WAAS glideslope and thus has an initial variance of  $(125 \text{ ft})^2$  (the other elements of  $\mathbf{X}_0$  remain the same as before), the LQR costs over the approach increase to 10,250 for the steady-state LQG solution and 11,400 for the SA solution [6-14]. The ideal result is now better, as expected, since there is no reason to base feedback on the poor initial state information. These cases with alternate initial covariances were run with the original LQE estimator; SA should find further improvement (for a given  $\mathbf{X}_0$ ) if it were allowed to alter the estimator gains as well.

Note that time-varying LQR design does take the initial state covariance into account and can choose time-varying controller gains accordingly [9-3]. It thus provides the best possible controller response over a given finite-time window and update rate as long as the (Gaussian) error model assumptions are correct. But as noted in the previous section, simulation-based optimization has the potential to improve on even this solution when non-ideal models (such as failure modes) are included.

## 2.6 Conclusions and Future Work

These examples demonstrate that simulation-based optimization can augment LQG design by adapting a baseline solution to optimize an arbitrary value function with no bounds on model uncertainty. More fundamentally, however, this approach illustrates that decision theory can be applied at a practical level to optimize controllers and estimators for arbitrary models of uncertainty and user value. The problem studied here, a simplified linear controller/estimator for aircraft using WAAS for Category I precision approaches, illustrates that this more general approach can add substantial value to control design.

This chapter serves as a brief tutorial on the application of decision theory to a problem that is traditionally solved by linear-quadratic optimal design. Although this problem is simplified for illustrative purposes and does not represent the best possible model, several key differences have been discussed in detail. First of all, no assumptions of the mathematical form of the noise or uncertainty models are required. Because

uncertainty is not bounded in an arbitrary fashion, decision theory states that the resulting design will be the best possible given the available information. Note that another designer with a different view of reality may use a different uncertainty model, and the design that results will still be optimal for that person. Decision theory also clarifies the value of gaining additional information with which to update the uncertainty models.

In order to reflect user preferences in a flexible way, the decision-theoretic approach allows for any objective function that meets the “rules of actional thought” outlined in Appendix A. In this case, the standard LQR performance index  $J$  is augmented by a penalty on accident risk at the terminal point of the approach. Designers can thus trade nominal performance for risk of the worst possible result based on the uncertainty model. The equivalence of the two defines the risk tolerance of the decision maker along the lines of Section 1.3.2. No guarantee of acceptable performance can be obtained, but the result is a more realistic guide for decision making under uncertainty than deterministic robust control approaches that must arbitrarily bound worst-case performance.

The results of the decision theoretic approach for this sample problem are encouraging because they demonstrate that simulation-based risk-sensitive controller optimization is both practical and beneficial. The optimal controller found by Simulated Annealing in Section 2.5.1 is a significant improvement over the nominal steady-state LQR controller in that accident risk is reduced by two-thirds while nominal (fault-free) performance is only slightly affected. In Section 2.5.3, SA with simulation-based evaluation is shown to improve on time-invariant LQR design over a finite time window even in the fault-free case, although of course time-varying LQR design (Section 2.5.2) would provide the best possible performance for a given sampling interval.

A key feature of simulation-based optimization is raised by this chapter. While it is clear that simulation-based optimization will converge to a known optimal result when the model is fixed to the assumptions of the underlying theory, it will adapt itself to the actual problem presented to it by the simulation model if it differs from those assumptions, as it does here. The disadvantage of this is that the optimal result is sensitive to errors in the mathematical model. But in the context of decision-making under uncertainty, this is a major enabling advantage with significant implications for the execution of engineering

design. In this context, decision theory states that it is better to find the design that is optimal for the best possible model rather than one that is based on theoretical assumptions that are known to be somewhat non-applicable. This is true even when the "best possible model" is not itself correct due to limited information, because simulations can be built that sample from the "information" or "assumption" uncertainty present at the time a decision must be made. A result that is truly optimal according to the precepts of decision theory can be derived, and it can be shown to be superior to the theoretical result when the prevailing uncertainty is taken into account. This is in fact what is done in Chapters 4 and 5 of this thesis.

The above findings do not imply that LQG design techniques cannot be adapted to provide controllers that are robust to uncertainty. LQG design certainly provides a good starting point as well as an important analytic foundation, and both structured-uncertainty and ad-hoc models can be created to derive "sufficiently robust" controllers from the ideal design. Sensitivity analysis was carried out in Section 2.3.1, but more could be done to modify the ideal (2.11) or worst-case (2.12) LQG controllers to handle the failure model of Section 2.3.2. The key difference is that simulation-based design has the ability to meet all the requirements of decision-making under uncertainty for any reasonable problem definition. This is important where failure mode effects that cannot be bounded exist and where the risk-reward preferences of the user go far beyond what LQG performance indices can handle.

Operationally, the result of Section 2.5 demonstrates that WAAS, even with considerable performance uncertainty, provides a reasonable margin of safety for Category I precision approach. WAAS performance will be subject to substantial rare-event uncertainty for the foreseeable future, and the design approach introduced here allows integration of the control law, estimator, and fault detection and isolation (FDI) routines into a combined optimal design for any risk model (tailored to a given state of knowledge) and user preference function. Chapters 6-8 of this thesis will demonstrate the capabilities of WAAS, decision-theoretic optimal integrity monitoring algorithms for the Category I application, and a means to integrate aircraft user performance models within a larger optimization of an entire WAAS network architecture.

## Chapter 3: Spacecraft Reliability Modeling

*It was said that the Saturn V (rocket) had a reliability of 0.9999. In the story, a group from headquarters goes down to Marshall and asks Wernher von Braun how reliable the Saturn is going to be. Von Braun turns to four of his lieutenants and asks, "Is there any reason why it won't work?" to which they answer (in German): "Nein." "Nein." "Nein." "Nein." Von Braun then says to the men from headquarters, "Gentlemen, I have a reliability of four nines.*

- Charles Murray, *Apollo: The Race to the Moon*. Footnote, p. 102.

### 3.1 A Working Definition of "Reliability"

Before beginning a detailed discussion of spacecraft reliability prediction, it is helpful to clarify precisely what we mean by the term "reliability." Reliability is generally thought of as the "probability of success" up to a given time of operation. More specifically, a recent book gives the following definition [3-10]:

**Definition:** *The reliability of an item is the probability that it will adequately perform its specified purpose for a specified period of time under specified environmental conditions.*

Each of the italicized words in this definition must be specified as necessary. For example, "adequate performance" for a spacecraft must be defined by the system-level mission requirements, and the "environmental conditions" are given by the orbital or interplanetary setting under which the spacecraft is expected to perform. Note that there is a related concept called "availability" which is used for repairable systems on Earth. Systems that can normally be repaired after a failure occurs may be modeled as being "operational" with availability  $a$  over a long period of time and "down" or "unworkable" with probability  $1-a$ . Availability is a more general concept which we will encounter again in Chapters 6 and 7 when discussing GPS-based aircraft landing systems.

Returning to "reliability," there is an important distinction to be made when using the standard definition above. In practically all applications, it is understood that reliability predictions presume that the system in question "should" work. In other words, reliability prediction is considered to be separate from the more basic question of *feasibility* -- whether the system is designed in such a way that it should be expected to perform

adequately. After all, the determination of basic feasibility is critical to the continuation of any project past its initial stages. However, as will be discussed in detail later, observations from spacecraft orbital history show that many failures result from an unseen design or manufacturing flaw that resulted in failure being likely. This fact has not gone unnoticed by system engineers; a common view is that traditional reliability predictions (which often seem optimistic) only apply to “random” failures of otherwise-healthy components. More likely causes of failures are thus separated from reliability calculations, making the latter almost useless.

This result of the traditional reliability engineering perspective (detailed in Section 3.2) preserves the distinction between well-designed subsystems (which rarely fail) and design or manufacturing errors that can be blamed on someone. However, to be useful as a top-level predictor of system success (and for probabilistic optimization), a reliability model should incorporate all sources of failure. Therefore, the probability models developed in this thesis will presume a weakened assumption of original system feasibility. In some cases, reliability will be contingent on “a reasonable belief of system feasibility” despite the possible (unknown) presence of design flaws. In other (more risky) cases, no preliminary assumption of feasibility will be made at all -- in essence, the question of feasibility will be included in the model as a random variable. This might apply, for example, to the design of a bare-bones nonredundant “lightsat” which itself serves as a feasibility test for a new technology development effort.

The following sections develop these basic ideas. We start with the traditional model and the assumptions which underlie it. After discussing its weaknesses and its unsuitability for probabilistic optimization, we turn to new models which modify the standard model by substituting mixed Weibull/Exponential time-to-failure distributions and which include the sources of parameter uncertainty that the traditional model ignores. The utility of these improved models will become clear as we turn to the following chapter (Chapter 4) on reliability-based system optimization.

### 3.2 Current Reliability Engineering Practice

Reliability modeling for spacecraft is currently conducted using a single standard approach mandated by the U.S. Department of Defense (DoD), NASA, and practically all commercial customers. The analysis method and relevant failure rates for electrical parts are specified in MIL-HDBK-217F [4-11] and related requirements documents, and supporting data for non-electrical parts is contained in NPRD-95 [4-7]. These methods are based on the *exponential* failure distribution in which reliability over time  $t$  is given by:

$$R(t) = e^{-\lambda t} \quad (3.1)$$

where  $\lambda$  is a constant failure rate calculated using data and environmental factors published in [4-11]. These handbooks of failure-rate data attempt to fit observed failure data from many sources into simple equations that are used to compute  $\lambda$  for a given application based on temperature, environment, electrical package technology, etc. For spacecraft orbital operations, the space flight environmental factor  $S_f$  in [4-11] (which multiplies tabulated data to produce a failure rate for a given application) was reduced from 0.9 to 0.45 in the 1980's to account for the Hecht report [4-3] observation that the predictions using the older factor were too pessimistic. This is a crude representation of the Hecht findings which are modelled more accurately in the models in Section 3.4.

The exponential distribution is *memoryless*; the probability of failure over a given interval of time is independent of the length of time that has already passed. This assumption is often questionable, as spacecraft have been observed to fail more often early in their design life, when design and manufacturing flaws are most likely to become apparent. After a certain amount of successful operation in orbit, the failure probability decreases toward a baseline level, which pertains until system wearout takes effect. This behavior is well-known to spacecraft engineers, but the exponential distribution continues to be used because of its simplicity.

Most spacecraft contracts use (3.1) and the data in [4-7,11] to compute reliability predictions for the components of their design. Redundancy is usually built in to avoid

*single-point failure modes*, which are events that by themselves cause mission failure. Using (3.1) to compute component reliability, series, parallel, and binomial ( $k$  out of  $n$  units need to work) network reliability can be computed using the standard equations in [3-5] which assume that all failure modes are *independent*:

$$\begin{aligned}
 \text{series:} & R_s(t) = R_1(t) R_2(t) \\
 \text{parallel:} & R_p(t) = 1 - (1 - R_1(t)) (1 - R_2(t)) \\
 \text{binomial:} & R_{n-k}(t) = \sum_k^n \binom{n}{k} [R(t)]^k [1 - R(t)]^{n-k}
 \end{aligned} \tag{3.2}$$

These equations also extend to the multiple-component case. For components which have a “standby” backup unit rather than one that is always operating, a correction factor  $q = 0.10$  is multiplied into the failure rate of the standby unit (this will not apply where passive radiation or temperature exposure limits device lifetime). This is supposed to reflect the much lower stress over time on a component that is not powered up. The result of these calculations is a system reliability prediction over the mission time line that must meet user specifications.

Since spacecraft are to a large degree unrepairable after launch, reliability is a key concern, but most systems engineers distrust handbook data and the assumptions present in the traditional model despite being obligated to do the computations. As a result, spacecraft tend to be overdesigned to “ensure” adequate reliability. This guarantees that the reliability specifications are met, but it does not help engineers make informed risk-based tradeoff decisions.

### 3.3 Weaknesses in Traditional Reliability Analysis

As mentioned in the previous section, the body of spacecraft failure data collected over the past 30 years tends to disprove the assumptions of the exponential reliability model. This fact has been known for some time [4-1], but the ease of analytical reliability calculations using (3.1) has dissuaded systems engineers from adopting various more-

detailed models that have been proposed since the late 1960's. In engineering, there has always been a preference for equations that give analytical, direct results; thus the continuation of the flawed traditional reliability model is not surprising. However, the fact that the exponential distribution is a convenient mathematical operator is partially just an excuse for the failure to update our analysis methods.

Clearly, one motivation for doing a spacecraft reliability analysis is to illustrate the risks inherent in a planned mission. Early in the history of spaceflight, the dominance of manned missions gave engineers strong motivation to classify and predict the extent of the risk using the best tools possible. This period (the early 1960's) coincided with the development of *Probabilistic Risk Analysis* (PRA) as a tool to project the risks of nuclear power station operations [1-5]. Attempts were made to apply reliability models to the Apollo program, and these analyses gave what many NASA engineers thought were very pessimistic estimates of overall mission success probability. A key difficulty was the lack of observed life test data for many of the Apollo components. Since mass testing over several years was impossible, it became easier to ignore the prediction of risk and instead focus on intensive searches for design flaws, a procedure which motivated FMECA analysis (to be discussed later). Although Apollo 13 (one out of seven missions to attempt to land on the moon) suffered a mission failure (a plan was improvised which brought the crew back home), the great success of the Apollo program as a whole suggested to many engineers that reliability predictions in general were misleading and a waste of design resources [4-4]. These observations fed a common preconception among engineers that the goal of systems development was to achieve "determinism". If a system were designed, developed, and manufactured properly, it would succeed with probability one, and if not, someone was to blame.

Although data with which to create better reliability models was available by the early 1970's [4-1,12], the fact that spacecraft reliability analysis was already suspect removed any strong motivations to update the methodology. This state of affairs continues to this day, and along the way, spacecraft engineers grew to distrust the results of standard reliability analyses. They were necessary from a contract-compliance point of view, but the results were not seen as aiding the design-for-robustness development

process. Current-day reliability analyses for spacecraft routinely predict mission-lifetime reliabilities well above 95%, and customers now expect such a result as part of the system requirements. Beyond this, few people pay much heed to reliability numbers. This can be viewed as “Engineers pretend to compute reliability predictions, and customers pretend to believe them.” As a result, current reliability prediction methods cannot be used for design optimization under uncertainty, and in fact they serve little practical purpose.

This is not to say that current design methods do not attempt to provide high reliability. While numerical reliability predictions tend to be ignored, certain basic design philosophies have become universally applied to spacecraft. Foremost among these is the dictum: “no credible *single-point failures*.” As mentioned earlier, a *single-point failure* is defined as a specific failure event (breakdown of the flight computer, for example) that directly leads to overall mission failure. Given sufficient design budgets, this is conceptually easy to avoid: simply have at least *two* of each mission-critical component type. Most modern spacecraft thus have redundancy at the subsystem and/or component levels to accomplish this. Spacecraft designs that include redundancy in this fashioned are assumed to be “relatively” reliable in a deterministic sense.

In addition to the application of component-level redundancy, spacecraft systems at all levels are normally designed with substantial *performance margin*. Since it is clearly recognized that performance in orbit has a random element, engineers often look for what they consider to be a “worst-case performance” event and design the system to operate successfully even under such conditions. If a design can be found that can handle this worst case, then the underlying risk factor is considered to have been removed, even though it is usually impossible to bound a random event with certainty. If such a design is not possible, the system in question gets special treatment as a source of risk that cannot be eliminated. Mission designers will pay careful attention as more is learned through the prototype-and-test process, hoping that new information will allow an acceptable worst-case bound to be derived.

From a decision-theoretic perspective (see Section 1.2), this process is sub-optimal in several respects. First, although NASA and commercial customers have good reason to be risk-averse, the application of margin everywhere in an attempt to meet worst-case

requirements leads to systems that are overdesigned for their objective; making them non-cost-effective solutions from the user's perspective. In trying to achieve a guarantee of mission success, system resources are arbitrarily allocated to each risk factor as needed to meet worst-case requirements that do not accurately correspond to mission risk. For example, a subsystem that cannot satisfy an arbitrarily conservative requirement receives an inordinate amount of attention compared to a system that does possess adequate margin, even though the latter system may yet fail due to design, manufacture, hardware, or the presence of environments outside the supposed "worst case." The effort to achieve a risk-free design is futile; thus resources are consumed for little performance gain.

A separate evaluation of likely failure causes, known as a *Failure Modes, Effects, and Criticality Analysis* (FMECA), provides a means of highlighting the physical failure sources that appear to pose the greatest risk to the mission and focusing risk-reduction effort in these areas [4-6]. Details of FMECA analysis and a related reliability prediction approach will be covered in Section 3.5. Note that FMECA is also a deterministic type of analysis. Failure modes of sufficiently high "criticality" are intolerable; thus modes identified to be unacceptable receive further development and/or testing until the apparent risk has been reduced below the criticality criterion.

As mentioned above, deterministic design heuristics and failure-mode studies certainly are important, but they often have the unfortunate side effect of leading engineers to assume that systems which pass these criteria should be practically guaranteed to perform successfully in orbit. If a failure should occur during the mission, a partial solution can often be improvised (known as a *work-around*), but if mission failure results, someone must be blamed for an error in design or judgement. Much of the apparent risk aversion present in the spacecraft development process seems to be a result of exaggerated expectations of flawless orbital performance promoted by NASA over the years (to justify the cost of space exploration) and passed on to the public by the media. Traditional reliability calculations continue to be made because of their analytical convenience and their ability to be massaged to support these claims and expectations.

### 3.4 New Concepts: Time-Dependent Failure Rates and Parameter Uncertainty

#### 3.4.1 Time-Dependent Failure Model

The first step toward improving the traditional reliability model is to use a *Weibull* failure distribution that allows failure rates to vary with time. It is a generalization of the exponential distribution, and its success probability is given by:

$$R(t) = \exp\left[-\frac{t^\beta}{\alpha}\right] \quad (3.3)$$

Here,  $\alpha$  is a *scale* factor that expresses mean time-to-failure, and  $\beta$  is a *shape* factor that varies the effective failure rate over time ( $\beta = 1$  gives the exponential distribution). TRW studies using their own spacecraft anomaly databases in 1974-75 [4-1,12] recommended the use of Weibull distributions in place of reliance on (3.1). In [4-3], Herbert and Myron Hecht utilized their spacecraft failure database (consisting of 300 satellite missions over the period 1961-1984) as the basis for a model which divides overall reliability into two *statistically independent* parts represented by (3.1) and (3.3). Basically, this approach separates so-called “random” component failures (RF) from those due to design, environment, and/or manufacture (DEM) that occur early in the mission. This corresponds to the discovery from the Hecht data that almost 50% of all spacecraft on-orbit failures are due to early-mission DEM causes, and these were usually not foreseen by the risk analyses that were conducted during spacecraft development. The Hecht method places a rough dividing line at  $t = 2$  years, which represents the crossover point in time between the 50% of failures assumed to be DEM (Weibull) and the other 50% due to RF (Exponential) events.

For the RF term, the exponential distribution (3.1) is used with a revised base failure  $\lambda_r$ , which is computed by dividing the tabulated (or sampled) failure rate by 1.41 (or 2.82 for rates computed using the old  $S_r$  of 0.9). This new rate is then used to compute  $\alpha$  and  $\beta$  for the related Weibull distribution in (3.3) used to model DEM, based on the type of spacecraft mission being conducted, as shown in Table 3.1. Where

applicable, a separate Weibull distribution with  $\beta = 2$  is used to model wearout failures (WR) at the end of the spacecraft design life.

<b>Mission Type</b>	<b><math>\beta</math></b>	<b><math>\alpha \lambda_{rf}</math></b>
General (default)	0.12	0.54
Communication	0.4	0.66
Navigation	0.9	0.93
Observation	0.13	0.55
Scientific	0.09	0.53

**Table 3.1: Prediction Factors by Mission Type [4-3]**

Given  $\alpha \lambda_{rf}$  and  $\beta$ , the computation of  $R(t)$  for a component proceeds as follows:

- (1) Obtain “official” exponential failure rate  $\lambda$  for given component
- (2) Obtain random failure rate  $\lambda_{rf}$  by dividing by 1.41 (2.82 for pre-1985 data from previous versions of [4-11])
- (3) Select parameters  $\beta$  and  $\alpha \lambda_{rf}$  from Table 3.1 ( $\beta = 2$  for wearout)
- (4) Compute Weibull parameter  $\alpha$  using:  $\alpha = \alpha \lambda_{rf} / \lambda_{rf}$
- (5) Compute reliabilities  $R_{RF}(t)$  from (3.1) and  $R_{DEM}(t)$  from (3.3)
- (6) Compute  $R(t)$  by assuming independence between RF and DEM failures:

$$R(t) = R_{RF}(t) R_{DEM}(t) \quad (3.4)$$

#### 3.4.2 Failure Rate Parameter Uncertainty

The failure rates tabulated in [4-7,11] are the result of statistically inferring failure model parameters from databases, which may be extensive but still contain considerable statistical uncertainty. In particular, the authors of [4-11] merge observed failure rates into a system of multiplicative factors based on unit complexity, operating environment, design format, and other details. This system of multiplying inferred factors adds more uncertainty to the final result. For its part, [4-7] must rely on much smaller sample sizes

for the vast array of non-electrical components it surveys and groups into somewhat arbitrary categories.

However, an effort was made during the compilation of [4-7] to estimate the likely variance in its failure rate predictions, given that its data is a generic compilation from a variety of sources. It was concluded (conservatively) that the *natural logarithm* of the tabulated failure rate is Normally distributed with a mean ( $\mu$ ) of  $\lambda_{tr}$  (the unknown "true" failure rate) and a standard deviation ( $\sigma$ ) of 1.5. This translates into a 68% probability that  $\lambda_{tr}$  lies within the one-sigma confidence interval  $[0.22\lambda, 4.5\lambda]$  and a 95% probability that it lies within the  $2\sigma$  interval  $[0.08\lambda, 11.9\lambda]$ , where  $\lambda$  is the tabulated failure rate as before. This level of variance is very high, but it at least gives us a guide as to the variability of tabulated failure rates. A  $\sigma$  of 1.5 represents the maximum variability in our model, since the data in [4-7] is very uncertain given its small sample spaces and imprecise component classifications. Data from [4-11], as well as vendor-supplied component data from life tests and operational experience, is given a smaller variance in this model. The variances for different components are thus assigned using the "fuzzy" categories and guidelines in Table 3.2, although specific circumstances may cause the reliability evaluator to choose a different variability for the base  $\ln(\lambda)$ . For example, he may be aware that the latest model of a rate gyroscope has had limited testing or operational pedigree and that its predecessor models have recently experienced problems in orbit. A  $\sigma$  of 1.25 or even 1.5 may be appropriate here. Other components which historically have given no problems over many years may deserve a  $\sigma$  lower than 0.5 (note that even this small number represents significant uncertainty about the mean failure rate). Finally, a combination of data from [4-7,11] and operational experience could combine to lower the  $\sigma$  assessed.

Source of Data	Standard Dev. of $\ln(\lambda)$
MIL-HDBK-217E	0.75
Specific component history	0.75
if based on $\geq 5$ yr. orbit ops.	0.5
info. from vendor w/out orbit exp.	1.0
NPRD-95	1.0 - 1.5
Default	
no significant information	1.5
substantial, varied information	0.75

**Table 3.2: Failure Rate Variation Cases**

The uncertainty inherent in the handbook failure rates is an important concern. To resolve this underlying uncertainty without requiring limiting mathematical assumptions, an algorithm for *Monte Carlo simulation-based* reliability predictions has been developed. For each trial, an exponential failure rate for each component in the reliability model is sampled from a Normal distribution with the mean published failure rate and the variance assigned to that component from Table 3.2. The result becomes the “official” failure rate for Step 1 of the conversion procedure in the previous section (for that trial only). The conversion process of the previous section is then carried out to get  $R(t)$  for that component type. Finally, once the samples and conversions for all components are completed, the output reliabilities for a given time-to-failure can be combined based on the overall system redundancy layout using (3.2). Because of the combined system uncertainty posed by many interacting components, a significant number of trials are needed to obtain the resulting reliability uncertainty distributions.

### 3.4.3 Application to Gravity Probe-B Spacecraft Bus

This approach has been applied to several designs for the Gravity Probe-B spacecraft bus. The spacecraft bus is designed to support the experimental payload by handling the normal orbital maintenance tasks, including power generation and regulation, attitude control, thermal control, and overall coordination through the flight computer. Unlike the experimental payload, the components that make up the spacecraft bus

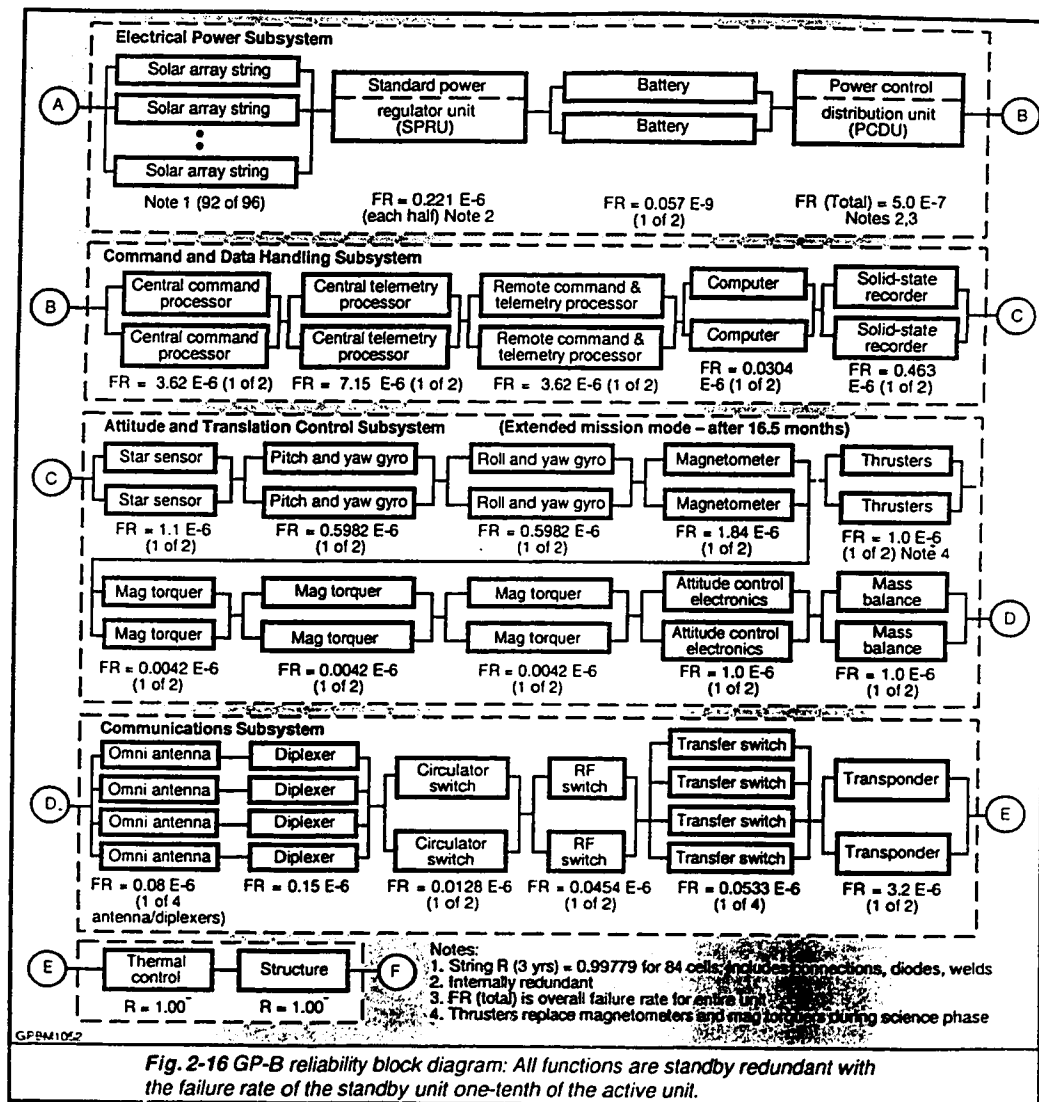
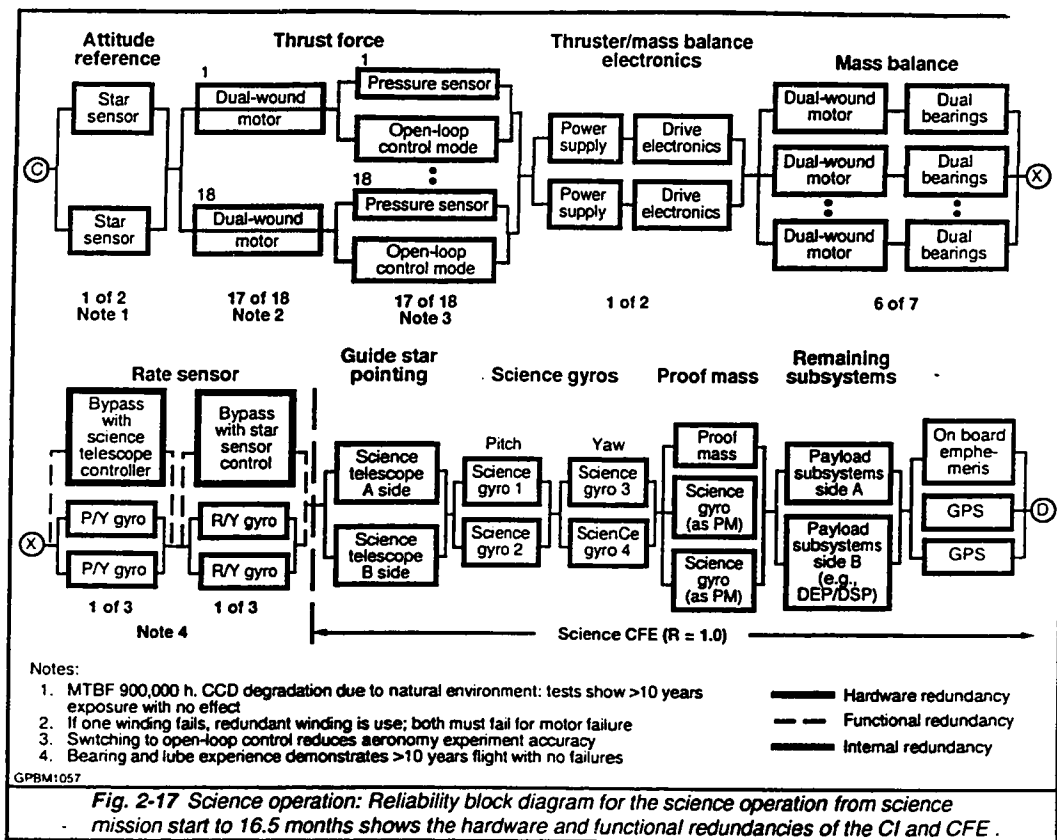


Fig. 2-16 GP-B reliability block diagram: All functions are standby redundant with the failure rate of the standby unit one-tenth of the active unit.

Figure 3.1a: LMSC GP-B Reliability Block Diagram



**Figure 3.1b: GP-B Reliability Block Diagram For Science/Control**

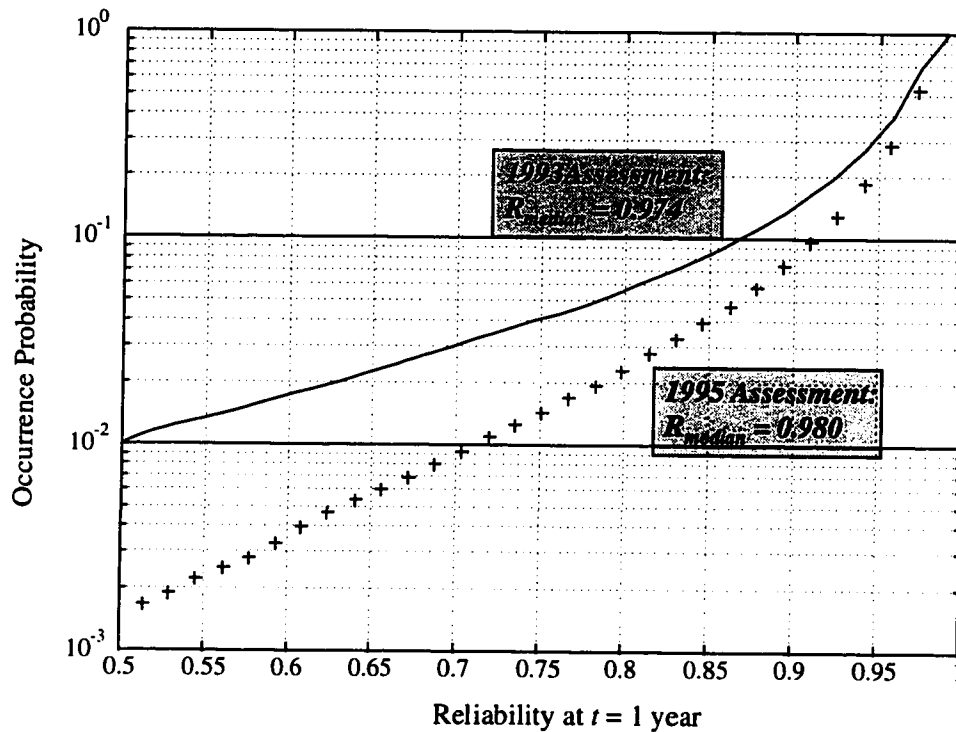
generally have some orbital history, allowing us to compute or extrapolate reasonable failure rate estimates for them. Lockheed Missiles and Space Company (LMSC; now Lockheed-Martin) won the contract for the spacecraft bus in 1993; thus the LMSC component-redundancy design is considered the “baseline” for our study.

Figure 3.1 is a functional block diagram of the LMSC spacecraft bus design, taken from Figures 2-16 and 2-17 of the LMSC management proposal [5-5]. It is broken into two parts: Figure 3.1a (2-16) for the spacecraft bus and Figure 3.1b (2-17) for the guidance and control subsystem and experimental payload interface. The block diagram is broken down into six subsystems plus the components that are “shared” with the science payload. Numbers of each component needed to work for mission success (e.g., “1 of 2”) are given along with the failure rates used in the LMSC traditional-style reliability calculation (expressed in failures per hour). Based on these figures, LMSC has predicted a spacecraft bus reliability for the 1-year primary science mission of 99.0%.

Our analysis of the spacecraft bus reliability begins with the failure rates given by LMSC (many of which were taken from subcontractor data). In most cases, the numbers are reasonable enough to use as the tabulated mean failure rates  $\lambda$  in the failure-rate-sampling procedure. However, several subsystems, such as the SPRU, battery, and magnetic torquer, seem to have been given overly optimistic failure rates. These were changed to better agree with more realistic numbers published in competing proposals.

Figure 3.2 is a plot of reliability uncertainty (for  $t = 1$  year) for the LMSC design. Both curves were separately generated using 50,000 Monte Carlo trials in MATLAB. This graph has “true” reliability, meaning the unknown actual reliability of the spacecraft bus, on the  $x$ -axis. The  $y$ -axis represents the “probability of occurrence,” or the relative likelihood of realizing a given actual spacecraft reliability, in logarithmic scale. Both curves are *cumulative distribution function (cdf)* plots that give the probability, for a given level of reliability  $R$ , that the unknown “true” reliability is lower than  $R$ .

This plot can be quite confusing at first because most engineers are not used to discussing the probability of a probability as shown here. Normally, a single probability value is given when discussing reliability. However, the parameter uncertainty introduced in Section 3.4.2 makes the true spacecraft reliability an uncertain value which must be



**Figure 3.2: GP-B Spacecraft Bus Reliability Uncertainty**

characterized by its own probability distribution. This representation of spacecraft performance uncertainty is necessary for the decision-theoretic optimization to be carried out in Chapter 3 because it is a more honest depiction of the true uncertainty faced by the spacecraft designer.

The two separate curves shown on the plot represent the same analysis done at two separate points during the GP-B spacecraft development program. The upper curve (plotted as a line) represents the reliability uncertainty at the time of the LMSC proposal evaluation in mid-1993. Based on updated data and component-specific information, the simulation was redone in mid-1995. New information is incorporated by revising the mean failure rates and especially the variances from Table 3.2 for components which have relevant new information. These later results are shown by the “+” curve. As expected, the increased amount of information available for the 1995 study tends to lower the variances for many components, producing the improved “+” results in Figure 3.2. The 1995 result is seen to be a clear improvement over the earlier 1993 study because the

reliability curves visible in Figure 3.2 are primarily the 50% of cases worse than the median (because the medians are close to 1.0). The 1995 curve lies below the 1993 curve and falls off faster, indicating that the updated reliability prediction has better worst-case tail performance in addition to having a slightly higher median reliability. Note that in the 1993 prediction, the likelihood of a true spacecraft reliability below 0.5 was 1%, while in the 1995 prediction, this likelihood falls to 0.15%.

Figure 3.2 demonstrates the importance of reliability uncertainty and its relationship to deterministic reliability predictions. Reliability uncertainty is obviously bounded by one on the upside, but the downside reliability can be quite bad in rare cases. Thus, the *median* (or the midpoint of an ordered list of samples) of the reliability histogram is taken to be the closest equivalent of a single reliability prediction. The medians for the 1993 and 1995 evaluations are 97.4% and 98.0% respectively. Primarily because of the modeling of *infant mortality* as described in the last section, these results are lower than the LMSC reliability prediction of 99% cited above. Note that infant mortality gives a more pessimistic reliability result than the traditional approach for  $t < 2$  years; thus our revised predictions will be more pessimistic throughout the duration of the GP-B primary mission.

Also note that the reliability uncertainty was significantly reduced between the 1993 and 1995 evaluations, as the cumulative distribution curve for the 1995 case lies well under that for the original evaluation. Looking at Figure 3.2, we can determine the 95% lower *confidence intervals* (CI's) for 1-year spacecraft reliability from the point at which the cdf's cross the  $y = 0.025$  axis (the upper CI's would be at the  $y = 0.975$  axis crossing - they are clearly 1.0). They are 67% for the 1993 case and 81% for the 1995 case -- a substantial improvement. Still, the presence of a 2.5% chance that the true reliability is below 81% is still worrisome. As the development-and-test process proceeds, new information should continue to give reduced uncertainty each time the reliability evaluation is updated. This "best estimate" of the remaining performance uncertainty satisfies the requirements of decision theory, making it possible to use this reliability model in the probabilistic optimization methodology to be presented in Chapter 4.

### 3.5 New Concepts: FMECA-based Reliability Predictions

#### 3.5.1 Introduction to FMECA Methodology

Unlike the spacecraft bus, the Gravity Probe-B spacecraft experimental payload is a fundamentally new instrument, although it is built using space-qualified parts. The functionality of many payload systems has not been fully tested in a space environment; thus its subsystems face a performance risk that is greater than the sum of their piece-parts. Without any significant statistics from which to derive representative failure rates, it is difficult to apply the reliability prediction methods outlined in Section 3.4. Lacking a better method, failure rate guesses would have to be used, and the large uncertainty that results would require failure rate sigmas much greater than the maximum of 1.5 for substantiated failure rates. Although this representation of performance risk would still be valid (as it would be the best we can do with the available information), its uncertainty would dominate the overall mission risk evaluation.

A better representation of the underlying uncertainty has been derived based on the FMECA analysis tool discussed in Section 3.2. FMECA classifies risks by type of failure event as opposed to failure source component, making it easier to perform a risk breakdown without a complete component performance model. Failure events are classified into three categories [4-6]:

(1) *Occurrence probability*: a rough estimate of the relative probability that the failure mode will occur, ranked from A (most likely) to E (least likely)

(2) *Severity*: a measure of the seriousness of the resulting threat to the mission given that the failure mode has occurred, ranked from 1 (highest threat) to 4 (lowest threat)

(3) *Conditional Probability*: a measure of the likelihood of suffering the most severe threat to the mission given that the failure mode has occurred, also ranked from 1 (highest probability) to 4 (lowest probability)

The goal of FMECA analysis is to compute a final value, or *criticality*, for each failure mode that ranks its overall risk at the system level. The process for making this computation is discussed in [4-6]. Basically, criticality is computed from approximate

probability measures for each of the FMECA classifications. These three numbers are multiplied together, and the result is then mapped into a ranking from 1 (highest risk) to 24 (lowest risk). Although it may differ for a particular application, a general guideline lists criticalities of 7 or lower as “unacceptable,” 8-12 as “undesirable,” 13-18 as “acceptable with review,” and 19-24 as “acceptable without review.” Once a FMECA analysis of a particular subsystem is available, an evaluation of the worst-criticality failure modes can be made. They will either be “re-evaluated” to qualify, or specific (deterministic) risk-reduction steps will be taken, such as additional prototype testing, to support later reevaluation and qualification.

### 3.5.2 FMECA Reliability Sampling Approach

The “fuzzy” probability set that is used to compute criticality serves as the foundation of a method to convert FMECA assessments into distributions of reliability uncertainty. Clearly, there will be a great deal of variation among true (unknown) failure probabilities; thus the variances about the approximate mean probabilities will be large. However, we do obtain a reasonably comprehensive model of the failure space expressed in terms of probability (as needed for PRA). The mapping of FMECA categories to reliability uncertainty is given in Table 3.3. Note that the P(M),  $\beta$ , severity, and assigned “meaning” in the first two columns of Table 3.3 are generally accepted FMECA concepts (although numerals 1-5 may replace A-E in some studies), whereas the probability distributions shown in the third column are introduced by this thesis based on the fuzzy connection of “meaning” to implied probability orders of magnitude in [4-6].

Given this mapping to probability space, we can conduct Monte Carlo sampling of overall subsystem reliability just as was done in the failure-rate model described in Section 3.3.2. An overall mission loss probability can be approximated from samples of probability uncertainty from the resulting dependent distributions:

$$P(MF) = P(M \Rightarrow \text{occur}) P(S_{ev} | \text{occur}) P(\beta | S_{ev}) \quad (3.5)$$

### OCCURRENCE PROBABILITY

<b>P(M)</b>	<b>Meaning</b>	<b>Failure Prob. Dist.(<math>\mu, \sigma</math>)</b>
A	frequent	Gamma(0.2, 0.125)
B	probable	Gamma(0.1, 0.07)
C	occasional	Lognormal(2, 0.5)
D	remote	Lognormal(3, 0.75)
E	improbable	Lognormal(5, 1.5)

### SEVERITY

<b>Sev.</b>	<b>Meaning</b>	<b>Failure Prob. Dist.(<math>\mu, \sigma</math>)</b>
1	catastrophic	Uniform[0.5, 1.0]
2	critical	Gamma(0.1, 0.05)
3	minor	Lognormal(2, 0.5)
4	other	Lognormal(3, 0.75)

### CONDITIONAL PROBABILITY

<b><math>\beta</math></b>	<b>Meaning</b>	<b>Failure Prob. Dist.(<math>\mu, \sigma</math>)</b>
1	actual loss	Uniform[0.7, 1.0]
2	probable loss	Normal(0.5, 0.15)
3	possible loss	Gamma(0.05, 0.03)
4	no effect	Lognormal(5, 1.5)

**Table 3.3: FMECA Risk Uncertainty Classifications**

Ideally, assuming that each mission failure cause is statistically independent, we can multiply all the sampled  $P(MF)$ 's for a given Monte-Carlo sampling trial together to get the overall failure probability for the subsystem in question. In practice, many other difficulties exist.

First, FMECA assessments of failure modes do not represent mission time-to-failure. Failure modes are broken down by which phase of the mission they are applicable to, such as ground testing, launch and ascent, and orbital operations. In converting to spacecraft reliability, we normally ignore pre-flight events, segregate launch events into a

separate category, and focus on orbital events, but our sampled reliabilities do not have a time dependence built into them, as do those of Section 3.4. Generally, it can be assumed that the FMECA-based risk of a given failure mode is evaluated to the end of the mission, which is about 1 year in the case of GP-B. This can then be “back-converted” to a reliability prediction over time by using (3.1), (3.3), and (3.4) in reverse, but significant additional uncertainty is added by this time-mapping procedure.

Also, since FMECA assessments only exist for specific, already-known failure modes, the question of measuring the risk of failure due to unforeseen causes is still open. The Hecht method discussed in Section 3.4.2 bases its 50-50 division of failure sources on the historical record of spacecraft failure events. Strictly, 18.9% of the Hecht database failures were due to “unknown” causes. Furthermore, most of the “design” failures (24.8%) were due to an unforeseen occurrence that invalidated the pre-launch design model [4-3]. One could thus assume that the FMECA-listed failure modes represent only  $100 - 18.9 - 24.8 = 56.3\%$  of the total failure likelihood. However, whether this is really a fair judgement is not clear. Conservatively, though, multiplying the output FMECA reliabilities  $R$  by  $1.0 / 0.563 = 1.776$  might be a prudent course.

Since the implied failure event probabilities given in Table 3.3 are uncertain, Monte Carlo simulation must again be used to resolve them in a method similar to that given in Section 3.4.2. For a given sample trial, “true” failure probabilities are sampled from the above distributions, and then  $P(MF)$  is computed for each listed failure cause using (3.5). Failure causes that are similar in physical origin, such as failure of a sensor type that exists in several places on the apparatus, use the same sampled failure probabilities for each cause of that type. In this way, failure rate dependencies are modeled for events where the underlying failure likelihoods are correlated.

To compute the overall subsystem mission failure probability for a given trial, the mission failure probabilities for each event are multiplied together as in a series network. Note that at this level, probabilistic *independence* is indeed assumed. While much of the correlation between different failure modes is captured by the common failure probability sampling just described, the assumption of independence at the top level (failure *occurrences* are independent even though failure *likelihoods* are correlated) may be

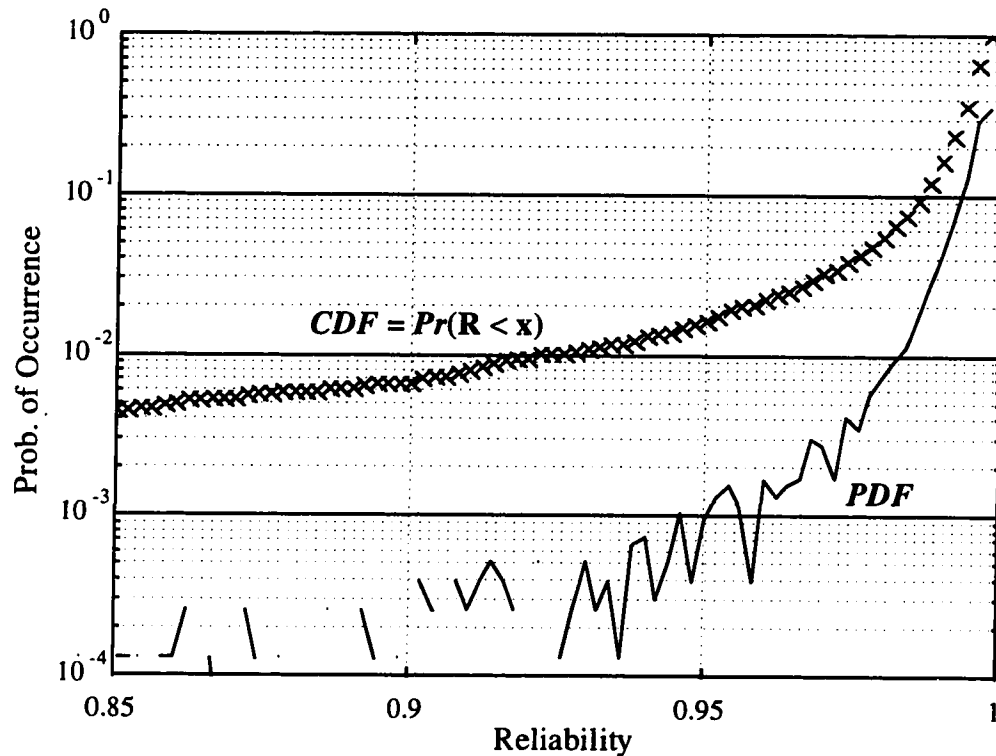
questionable in some cases. Modeling of top-level dependencies can be accomplished by building a tree of dependent events known as an *event tree* [1-5]. In the future, it may be possible to link FMECA-based reliability sampling with an event-tree that directly represents events as they occur in the mission timeline. This linkage would also directly handle the question of representing mission time in the FMECA reliability predictions.

As in the spacecraft bus sampling procedure, top-level reliabilities for each trial are stored in histograms which are plotted and analyzed after all trials are completed. For the GP-B payload development program, draft FMECA's are presented at the PDR for each subsystem, and revised ones are completed by CDR. Thus, we will conduct reliability conversion analyses for each subsystem separately as its FMECA is completed. A subsequent Monte Carlo sampling procedure will be used as needed to combine subsystem reliability histograms into an overall payload reliability prediction.

### 3.5.3 Results for GP-B Payload Dewar

An example result of this FMECA reliability prediction method for the Gravity Probe-B spacecraft *dewar subsystem* is shown in Figure 3.3. It is the result of 4 million Monte Carlo samples (in C) of the 82 separate failure modes (69 on-orbit, 13 launch/ascent) of relevance to the mission itself (rather than testing and preparing for launch) listed in the dewar FMECA released by Lockheed. Note that a much greater number of trials are desirable for FMECA reliability evaluation (compared to the failure rate method) because of the much greater number of separate failure sources, each of which requires three samples as per Table 3.3.

Two curves are shown on the plot. The lower one (a line) simply plots the output histogram of dewar reliability observed over the 4 million trials. Below a reliability of 0.925, the curve becomes jagged and incomplete because of the scarcity of trials that result in such low success probabilities. The upper one ('x') shows the resulting cumulative distribution (cdf) of dewar system reliability. From this curve, we see that given the current FMECA information, the likelihood that the true dewar reliability (to the end of the on-orbit mission) is below 95% is about 0.018, or less than 2%. In fact, the median reliability of the histogram is 99.7%, which is very good. However, the cdf levels



**Figure 3.3: GP-B Dewar FMECA Reliability Uncertainty**

off below 90% reliability, suggesting that our uncertainty makes it difficult to rule out reliabilities lower than 85% beyond a baseline likelihood of 0.003.

Compared to methods using failure rate data, the FMECA reliability approach gives very uncertain results. Lacking more definitive information, it does, however, provide a representation of performance uncertainty suitable for probabilistic decision-making. Furthermore, it provides a starting point for uncertainty models that evolve with time, as they can be updated using Bayesian techniques [3-3] periodically as new information becomes available.

### 3.6 Conclusions and Further Work

The key question to be asked about the new reliability models discussed here is: what significant improvements in mission-level risk modeling do they provide? It is clear that the traditional reliability method of Section 3.2 is of limited use. However, both the

failure-rate and FMECA models developed here require extensive computer simulation, and they only give reliability *uncertainty* results which many find hard to digest.

The most fundamental benefit of these new methods is the proper representation of failure-rate uncertainty. The practical impact of “fuzzy” reliability predictions may be unclear to traditional systems engineers, but its importance becomes evident when one analyzes a system from the perspective of decision making under uncertainty (see Section 1.2). Even without this perspective, one can examine the 90% or 95% lower confidence intervals of the resulting reliability uncertainty distributions in addition to the median reliabilities to get a better picture of which subsystems pose the largest mission performance risk. This information should help to refine the process of deterministic risk reduction efforts.

In addition, several interesting phenomena become apparent from the use of simulations to evaluate reliability uncertainty. The best example is so-called *weak sister* subsystems, which are discussed in the 1974-5 TRW studies [4-1,12]. This refers to subsystems or components that, unknown to the developers, contain critical weaknesses that make them the key drivers in the overall (unknown) spacecraft reliability. These flaws are more likely to become apparent early in the mission; thus missions that survive this early phase are more likely to contain no such weak sisters. The TRW studies suggest a complex model that requires indentifying the “weak sister” risk for each component by fitting specific curves to failure databases that have been broken down into “weak sisters,” “normal populations,” etc.

A detailed study of weak-sister likelihoods would be useful, but it turns out that Monte Carlo failure rate sampling reveals the same phenomenon naturally, without needing a separate mathematical construct. In each failure-rate sampling trial, there is a chance that one of the component classes will “get unlucky” and receive a very high failure rate that translates into, for example, only a 70% likelihood of surviving one year in space. This causes the overall system reliability for that trial to be bounded by 70% on the upside, even if every other component is sampled to be very reliable. Different component classes will contain weak sisters in separate sampling trials; thus the result of a large set of

samples automatically incorporates the effects of weak sisters into the overall system reliability uncertainty evaluation.

In addition, simulations naturally represent some of the *dependence* between the failure likelihoods of identical or related components. In a given trial, the same random sample is applied to all systems judged to be similar, then the reliability for the affected component classes is determined from (3.2) by assuming independence of actual failure events. In many cases, even the latter assumption is questionable, but the inclusion of dependence of failure probabilities goes a long way toward defeating the optimistic assumption of complete probabilistic independence. For example, a particular component class with one primary and one (standby) backup unit will have the following reliability if it receives its median failure rate, which we will assume to be  $1 \times 10^{-6}$  failures/hour. After one year, the overall reliability for that class, using a standby “*q*” factor of 0.1, is 0.99995; a good result. However, if a sample of one order of magnitude worse,  $1 \times 10^{-5}$ , is obtained for a given trial, the reliability becomes only 0.995, or *two orders of magnitude* worse. The modeling of dependence, even in a limited fashion, helps to more realistically evaluate the true benefit gained from redundancy, which is a primary tool of the system designer.

Finally, the need for computers to conduct Monte Carlo simulations is not as onerous as it might seem. First of all, the sampling and reliability computation is simple and requires only floating-point arithmetic operators. Coded in C, millions of samples can be conducted in only a few hours on the newest personal computers. Furthermore, the traditional method of Section 3.2 is now usually carried out using computer software which can numerically transform a functional block diagram of the type (including the applicable failure rate information) shown in Figure 3.1 into system reliability numbers. Since the “Rubicon” of reliance on computers has already been crossed, the analytical ease of (3.1) is no longer as significant as it used to be. Instead, the rapidly-increasing processing power of PC’s and workstations allows us to build very flexible models and evaluate them in whichever way best represents the information we possess.

## Chapter 4: Spacecraft Probabilistic Design Optimization

*The concept of design was born the first time an individual created an object to serve human needs. Today, design is still the ultimate expression of the art and science of engineering. From the early days of engineering, the goal has been to improve the design so as to achieve the best way of satisfying the original need, within the available means.*

- Panos Papalambros, Douglass Wilde, *Principles of Optimal Design*, p. 12.

### 4.1 Deterministic Reliability Optimization

While it is not generally true for spacecraft, many fields can take advantage of deterministic reliability predictions built upon comprehensive failure databases to devise methods for optimizing a given design from a reliability standpoint. More specifically, there exist algorithms which can maximize system reliability under total system cost and weight constraints. Conversely, they can minimize cost or weight given a lower-limit reliability constraint. The design variables may include continuous variables and integers that represent the amount of redundancy applied to each component class. Due to the presence of nonlinear state equations (that compute reliability given a specific design alternative) and discrete integer encodings, nonlinear programming (NLP) methods are required to find the optimal design with certainty [2-8]. Typical problem statements of this type are [3-5]:

$$\begin{array}{ll} \text{Maximize Reliability} & \text{Minimize Cost} \\ \text{Max} \quad R_{\text{sys}} = \prod_{i=1}^M R_i & \text{Min} \quad C_{\text{sys}} = \sum_{i=1}^M c_i \\ \text{such that } \sum_{i=1}^M p_{ki}(R_i) < a_k & \text{such that } R_{\text{sys}} = \prod_{i=1}^M R_i(Y_i) \geq R_{\text{req}} \end{array} \quad \text{or} \quad (4.1)$$

where  $R$  represents reliability,  $C$  and  $c$  represent (monetary) cost,  $p_k$  is the consumption of resource  $k$  to obtain reliability  $R_i$  for component  $i$ ,  $a_k$  is the constraint on that resource, and  $Y_i$  is the unit redundancy chosen for that component. While NLP or dynamic programming approaches can be applied to a general problem, the use of design *heuristics*

in combination with a more-specific optimization approach tends to be more successful for individual cases [2-11].

Note that despite the goal of design optimization for reliability, what is being optimized here is a *deterministic* measure of risk performance. In other words, the optimization method assumes that reliability can be directly computed from the model for a given design, and it assumes that reliability, in and of itself, is the goal. This may be the case for simple systems, but the design of a spacecraft is coupled with other issues that do not necessarily fall within the designer's control, such as the launch vehicle and orbital insertion. The use of reliability as a system objective is a proxy for a more complete model of mission-level user utility which, in part, expresses the effects of failures at certain times on the overall objective, be it commercial, scientific, or otherwise. Therefore, a better optimization can be done with a higher-level objective function that expresses a tradeoff between probabilistic performance and design cost and effort.

In the spacecraft case, however, it is the significant uncertainty about mean reliability predictions that limits the usefulness of deterministic reliability optimization. If these mean values include the uncertainty demonstrated in the analyses of Chapter 3, the deterministic approach would ignore the greater part of the true performance uncertainty. This is an important reason why spacecraft design optimization using traditional reliability methods is not seen as worthwhile.

## **4.2 Gravity Probe-B Spacecraft Bus Optimization**

### **4.2.1 Optimization Problem Overview**

As described in Chapter 1, the Gravity Probe-B spacecraft is designed to support a physics experiment in orbit. As such, it can be subdivided into the *experimental payload*, which includes the probe with its extremely precise gyroscopes to measure relativistic drifts, and the *spacecraft bus*, which includes the support equipment needed to operate the experiment in space. Because the payload is composed of new systems which have not been extensively tested in space, reliability predictions for it are done using the FMECA method described in Section 3.5. The design of the payload is controlled by scientific

imperatives as well as the need for system-level redundancy. In contrast, the spacecraft bus design uses well-understood subsystems to supply and regulate power, maintain the required spacecraft position and attitude, etc. Thus, (uncertain) failure rates are available for the spacecraft bus components, and the methods of Section 3.4 have been applied (as shown in Figure 3.2 for the current LMSC spacecraft design).

Using these performance uncertainty models, we can optimize the layout of the spacecraft bus design not with respect to its own reliability but instead to the overall mission objective. Our basic decision vector will be the amount of component redundancy to provide. In other words, we will choose the optimal number of each type of component to provide to the spacecraft bus, relative to the minimum number of each component class required to accomplish the mission. For a given design choice, the simulation-based reliability method of Section 3.4 will be applied, and simulations of other relevant performance models will be conducted as needed. The simulation outputs give the objective function evaluation for that design choice.

Recall from Section 3.2 that the traditional approach insists on avoiding single-point failures. Thus, every component class will have at least one more than the minimum assumed necessary for mission success. This dictum is visible in the LMSC baseline design shown in Figure 3.1 of the last chapter. In particular, it is thought that the GP-B spacecraft bus should be made very simply and reliably so as not to add unnecessary failure modes to the risks presented by the relatively untested payload system. This design logic will be tested by the probabilistic optimization to be carried out here.

#### *4.2.2 Probabilities of Non-Controllable Events*

In the spacecraft bus optimization, the spacecraft bus reliability will be computed using the failure-rate simulation method of Section 3.4. However, the unknown reliabilities of the launch vehicle (including orbit insertion) and the experimental payload are clearly critical to the mission value function. Although the spacecraft bus designer cannot control these external risks, the marginal benefit of added spacecraft reliability will be affected by the reliability of these other systems. For example, if the payload reliability is below 0.90, the value (in terms of the overall mission) of increasing spacecraft reliability

to end-of-mission from 0.98 to 0.99 will be lower than if the payload reliability is also in the 0.98-0.99 range. This is true simply because the overall system probability of success is dominated by its "weakest link" (recall the related "weak sister" effect on spacecraft reliability described in Section 3.6); thus improvements to higher-reliability subsets of the mission will have limited effect on the likelihood of overall success.

While payload subsystem reliabilities can be predicted from the FMECA method of Section 3.5, it will not be possible to compute the end-to-end payload reliability until all subsystem FMECA's have been completed. For now, simple uncertainty distributions will be assessed for both the launch/orbit insertion and for the payload. For the *Delta launch vehicle* and *orbit insertion*, the assumed mean success reliability is a point probability of 93%. The standard deviation of this mean reliability is 4%, and the true reliability is sampled from a Normal distribution with these parameters but bounded by 100% on the upside. The *experimental payload* is modeled with a single mixed Weibull-Exponential failure model as per Section 3.4, with a mean base failure rate (computed by summing failure rate estimates for each of the payload subsystems) of  $\lambda = 8 \times 10^{-6}$  and a (high) lognormal uncertainty factor  $\sigma = 1.0$ . Using Section 3.4, this mean failure rate translates into an expected one-year success probability for the payload of 92.8%, although the level of uncertainty for the payload is much higher than for the launch vehicle.

The success probabilities for launch/orbit and payload systems are sampled for each uncertainty trial and then applied to the more-detailed spacecraft bus reliability computations. The latter will change as different spacecraft bus designs are evaluated, but the launch/orbit and payload reliabilities are considered to be *independent* of the optimal spacecraft design and thus are not affected by spacecraft design variations. Admittedly, this is a slight simplification, as the spacecraft/payload interfaces for GP-B do not allow such an easy separation (see Figure 3.1b), but it makes sense given the very simple payload model used.

Finally, there are roles on the spacecraft bus which are "one-shot" operations, such as deploying the solar arrays after the final orbit has been attained. These are represented by a *point* failure probability rather than the time-based model of Section 3.4. For the (internally redundant) solar array release and deployment units required for GP-B, a mean

point success probability of 0.9995 is assessed, and the (Normal) uncertainty standard deviation is 0.001 (the success probability is bounded by a maximum of 1.0).

#### 4.2.3 *Decision-Maker Utility Functions*

The optimal design is driven by the form of the objective function used to model the *utility*, or relative values of outcomes, of the decision maker. Because evolutionary methods (detailed in Section 4.3) can perform an optimal search on any objective function, the designers are free to choose any mathematical model that best describes their preferences. This is a major improvement over conventional formulations such as (4.1), but it is a double-edged sword: the optimal search will tend to exploit any “holes” in the objective function, resulting in solutions that may not seem optimal to the human designers. In general, it is difficult to properly weigh the preferences of both system designers and users into a single objective function on the first try (but it is much more difficult if the form of this function is mathematically constrained) [1-5]. The results of the first few optimization runs therefore fit into an iterative systems engineering framework as they help designers refine their value models to better capture what they want to achieve. It is possible to abuse this process in order to justify a pre-determined conclusion, but that violates the basic precepts of decision theory and results in a decision-maker simply cheating himself or herself.

**4.2.3.1 LMSC Value Model:** The objective function for LMSC is assumed to be the fee, or profit as a percentage of the spacecraft bus cost, it is to receive depending on the outcome of the mission. The following objective functions are based on the LMSC/NASA reward contract for GP-B and thus are a reasonable starting point for expressing LMSC preferences in terms of maximum gain from the mission, although they do not explicitly give the true LMSC fee agreement. For LMSC, profit is divided into three categories with the following random models:

- *award fee*: From 0 to 6% of spacecraft bus cost  $C_s$ , based on progress in meeting development program milestones -- modeled as (unchangeable) *Normal*(4%, 1%).

- *cost fee*: From -6% to 6% of spacecraft bus cost  $C_S$ , based on a linear function of the resulting actual cost of spacecraft bus. The actual cost  $C_A$  includes the amount by which the number of components of a given design solution increases or decreases the spacecraft cost, and a random cost amount of *Gamma*(\$25 million, \$10 million) - \$10 million is added. The cost fee percentage is then determined by:

$$CP_F = \begin{cases} 6\% & \text{if } C_A \leq C_S - \$5 \text{ million} \\ -6\% & \text{if } C_A \geq C_S + \$55 \text{ million} \\ 6 - 0.2 \left( \frac{C_A - C_F + \$5 \text{ million}}{\$1 \text{ million}} \right) (\%) & \text{otherwise} \end{cases} \quad (4.2)$$

- *performance fee*: From -6% to 6% of spacecraft bus cost  $C_S$ , based on results of mission and spacecraft bus on-orbit. A successful mission which obtains valid science data for the one-year length of the primary mission is worth 6% to LMSC. A mission failure during launch or orbit insertion (resulting in no science data) that is attributed to LMSC results in a fee of -6%. If the blame lies elsewhere, the fee is 0%, which also applies to a non-LMSC mission failure during the first six months of science data-taking. The fee for a failure blamed on LMSC during the first six months, or any failure after six months have passed, is bounded by the following Maximum Performance Fee (MPF):

$$MPF = \left[ \frac{(tm-6)^5}{216|tm-6|} + 6 \right] PFF - 6 \quad (4.3)$$

where  $tm$  is the number of months of successful science data-taking, and PFF is an independent, subjective performance evaluation made by Stanford which is assumed to be distributed as (unchangeable)  $Normal(0.8, 0.1) \leq 1.0$ .

NASA regulations bound the overall fee percentage, which is the sum of the fees from the three listed sources, to a minimum of 0% (i.e., no financial loss is possible) and a

maximum of 15%. The estimated base spacecraft bus cost is  $C_s = \$100$  million, which translates into a possible value range of \$0 - 15 million.

This expected-profit model contains few strong incentives for LMSC to maximize on-orbit reliability. Much of Lockheed's profit is determined before the spacecraft is even launched, and there are many factors which the spacecraft redundancy designer cannot directly control. As noted in Section 4.2.2, the relative importance of these factors lessens the marginal value of reliability improvements. Also, the launch vehicle, upper stage, and experimental payload are all intrinsically less reliable than the spacecraft bus. The spacecraft bus designer must nevertheless make the optimal design decisions within this extended context.

**4.2.3.2 Stanford Value Model:** The objective function for Stanford University, which is focused primarily on the amount of high-quality experimental data obtained from the mission, is based solely on an on-orbit performance evaluation similar in intent to that of the LMSC performance fee. It is also normalized to be between \$0 - 15 million, although the resulting LMSC and Stanford values are not directly comparable, since scientific value is not expressed in terms of dollars. Successful science data taking for the length of the one-year primary mission is "worth" \$15 million, mission failure before any data is received is worth nothing, and intermediate results are valued by the following equation:

$$Stanford\ Value = \begin{cases} 0.1\ tm\ (\$15\text{million}) & tm \leq 6\text{months} \\ [0.6 + 0.0667(tm - 6)] (\$15\text{million}) & 7 \leq tm \leq 12\text{months} \end{cases} \quad (4.4)$$

This function increases linearly over the first six top-priority months, then it levels off as further months of mission success are added. It should be noted that additional value would be derived from "co-experiments" which will continue after the first year in orbit, but they are of a lesser order than the relativity mission. Thus, payload and spacecraft bus reliability is only computed out to one year for the purposes of optimization.

Unlike the LMSC value function, the Stanford model does not include extraneous (and uncontrollable) variables. The resulting value is solely a function of how much high-

quality data (i.e., data that meets the science requirements) is obtained. Note that this is admittedly a simplification of reality, since Stanford's value (and LMSC's) is directly connected to cost and schedule objectives during program development which are tied to the complexity of the spacecraft design. A complete decision-theoretic solution would include these concerns along with a model of program development risk and simulation of the development process. This is not done here but is a key area of future work [1-5].

In the Stanford case, the reliability model is assumed to represent failures that make it impossible to obtain good experimental data for one reason or another. Failures which only degrade the amount or quality of data within acceptable limits are not counted as mission failures, although considerable uncertainty exists over the threat posed by specific component failures (see Section 3.5 for a discussion of FMECA's). Since we can compute the overall distribution of failure times based on launch, orbit insertion, payload, and spacecraft bus reliabilities (after uncertain failure parameters are sampled), it is not necessary to run actual orbital simulations of the Stanford value function; the value (4.4) can be computed as a weighted sum of each possible number of months of valid data.

#### 4.2.4 Constraint Modeling

To represent constraints, *penalty functions* are applied which subtract costs for exceeding spacecraft weight, volume, and power budgets from the LMSC and Stanford value functions. The spacecraft bus has been allocated totals of 801.5 kg, 0.983 m<sup>3</sup>, and 223.75 Watts for each of these, respectively, in the LMSC spacecraft proposal [5-5]. Each trial solution has its totals in these categories (as well as unit cost) figured, and ratios of these numbers over the base numbers given above are computed as  $R_w$ ,  $R_v$ , and  $R_p$  respectively. If these ratios are below 1.0 (indicating that the proposed solution is within budget in that category), no penalty cost is applied. If not, penalties are computed based on the following cubic functions whose coefficients are chosen to model the approximate loss of mission value that would result from a given level of budgetary excess:

$$\text{weight penalty} = C_s [0.001 + 0.15 (R_w^3 - 1)] \quad (4.5a)$$

$$\text{volume penalty} = C_s [0.0001 + 0.01 (R_v^3 - 1)] \quad (4.5b)$$

$$\text{power penalty} = C_s [0.001 + 0.09 (R_p^3 - 1)] \quad (4.5c)$$

where  $C_s$  is the base spacecraft cost of \$100 million. Note that these equations rank relative importance in meeting the budget as weight first, then power, then volume. Also, in the weight case (4.5a), additional weight is assumed to be added in the spacecraft structure at the same proportion (to baseline structural weight) that  $R_w$  exceeds 1.0.

The use of penalty functions to represent constraints is a controversial issue in evolutionary systems optimization because such functions cannot easily enforce “hard constraints” which must not under any circumstances be violated. If the spacecraft system budgets truly were absolute black-or-white barriers, this approach would be insufficient. There are two means of enforcing hard constraints in an evolutionary-search format. One is to come up with a problem encoding that simply does not allow a constraint violation. This is possible only if the constraints do nothing more than bound the range of each separate entry of the solution vector. Another approach is to simply check the constraints for each new solution generated and discard those that violate any constraint. Unfortunately, this approach denies the algorithm the ability to search near the constraint borders (as is done in Linear Programming) since many solutions close to the border will be rejected before they can propagate their desirable elements.

Penalty functions are the right choice for most real-world problems because few constraints are truly absolute, and those that are can normally be incorporated into the problem encoding. In the spacecraft design case, the budgets proposed initially normally have a “safety” margin to allow for possible overruns in key subsystems. The margin is considerable in the GP-B case because of the separate set of budgets for the experimental payload, where more budget margin is necessary. The functions in (4.5) account for this by providing a small but noticeable (\$100,000 in the weight and power cases) penalty for any exceedence of the applicable budget and then increase the penalty exponentially as the constraint violation worsens. Once the violation begins to use up most of the slack in the

budget, the penalty cost from (4.5) becomes severe (greater than \$1 million), ensuring that that alternative is not close to optimal.

#### 4.2.5 Final Objective Function Evaluation

After taking these possible penalty costs into account, the final evaluations for the Lockheed and Stanford value functions can be computed:

$$LMSC\ value = award\ fee + cost\ fee + on-orbit\ fee - penalty\ costs \quad (4.6a)$$

$$Stanford\ value = on-orbit\ value + cost\ savings - penalty\ costs \quad (4.6b)$$

Because these evaluations are based on simulation results, many trials must be conducted to produce a result distribution that can be converted into an objective value. In the LMSC case, for each new solution evaluation, 10 mission failure-time simulations are conducted for each of 500 samples of the reliability parameter uncertainties as in Section 3.4. As mentioned before, the Stanford case does not require mission simulations. Instead, the value function (4.6b) is evaluated based on the histogram of mission reliability per month generated by 1000 samples of reliability uncertainty. The result is a considerable speed-up compared to the LMSC evaluation, and the lesser need for simulation reduces the random noise in the values that remain after the simulations are completed.

In order to reduce the sampling noise or uncertainty over time, initial evaluations of design solutions are not “thrown away.” Instead, results for solutions which have been evaluated are stored along with a count ( $N_{old}$ ) of how many separate evaluations have been conducted. When a previously-evaluated solution is re-evaluated, the specified number of simulations per evaluation are conducted, then a weighted value update is conducted as follows:

$$Updated\ evaluation = \frac{N_{old} (old\ evaluation) + (new\ evaluation)}{N_{old} + 1} \quad (4.7)$$

The updated value then becomes the old value, and  $N_{old}$  is increased by one. Since simulated annealing (Section 4.3.1) propagates only one trial solution at a time, this is straightforward. For a genetic algorithm (Section 4.3.2) in which many solutions are maintained at once, a list of previous evaluations needs to be kept.

#### 4.2.6 Value Models for the Risk-Averse Case

If LMSC or Stanford were deemed to be significantly *risk-averse* to the effect of a publicized mission failure, a risk aversion factor [1-3,5] can be added to the risk-neutral value models presented in this section. In the LMSC case, risk aversion could flow from a fear of loss of future spacecraft contracts if GP-B had a publicized failure that was blamed on the spacecraft bus. This would apply to the performance fee only, and it would be offset to some degree by the other fees which are (for the most part) determined before launch. Stanford is concerned with orbital performance only and thus has a clear-cut motivation to be risk-averse, since a failure of GP-B would make it difficult to receive funding for another spacecraft to try the experiment again. However, this preference shows up in the MPF equation (4.3), which places more importance on the first few months of data-taking so that at least some science information is gained.

Risk aversion can be modeled arbitrarily using the preference probability value model of Appendix A, which requires no mathematical assumptions outside of the basic definitions of probability. For large risk-averse companies users whose certain equivalent business (see Section 1.2.2 and Appendix A) for relatively small activities is assumed to be independent of their overall "wealth", the *delta property* can be applied [1-3]. A convenient model for this case is the following utility function, which would be applied to the performance fee [1-5]:

$$u(p) = \frac{1 - \exp(-\gamma p)}{1 - \exp(-\gamma)} \quad (4.8)$$

where  $p$  is the performance fee percentage and  $\gamma$  is a *risk aversion coefficient*. For a strictly risk-neutral decision-maker,  $\gamma = 0$ , which gives  $u = 0$  for any  $p$ . This function is thus inappropriate (and unnecessary) for a risk-neutral decision maker, but in the limit case

as  $\gamma \rightarrow 0$ ,  $u(x) \rightarrow x$ , which makes it logically consistent. Risk-averse users have  $\gamma > 0$ , and risk-seeking users (an unusual case in practice) have  $\gamma < 0$ . Under the delta property,  $\gamma$  must be constant over the range of possible outcomes.

A modified risk coefficient  $r = \exp(\gamma)$  has another useful property for  $\gamma \neq 0$ . For risk-averse users,  $r > 1$ , and  $r$  gives the favorable *risk odds* that such a user would have to be given in a lottery between outcomes  $+X$  (good) and  $-X$  (bad) before being indifferent to a certain equivalent of 0. In other words, while a risk-neutral person would be indifferent between a sure outcome of 0 and a 50-50 chance of winning  $X$  versus losing  $X$  (within the delta property bounds), a risk-averse person would insist on a better than 50-50 chance of winning before accepting the risk. His required chance of winning would then become  $r/(r+1)$ , which always exceeds 0.5 in the risk-averse case.

It is thus possible to derive risk-averse value models from risk-neutral ones directly if the delta property holds. In general, this modification may significantly increase the value of adding reliability to the spacecraft bus, although this effect is not significant for GP-B because of the dominance of payload reliability (the weakest link) on overall mission success. Even very risk-averse spacecraft bus designers can do no better than the payload reliability (which cannot be altered here), thus overall mission reliability improvement is hard to come by no matter what its potential value would be. This should be kept in mind when considering the significance of the optimal results in Section 4.4.

### 4.3 New Design Optimization Approaches

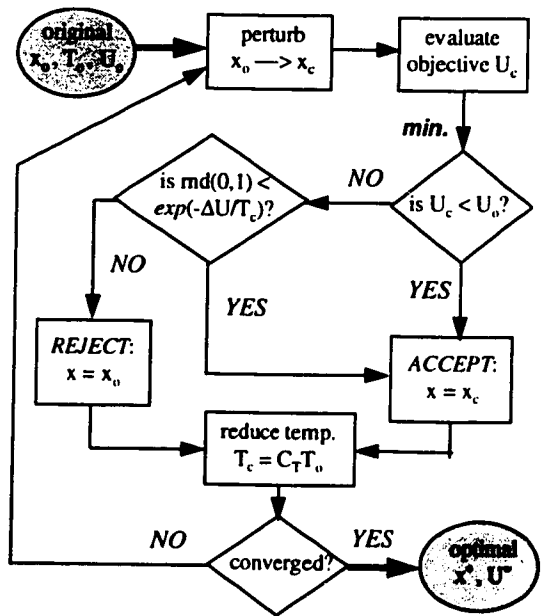
Given an arbitrary user objective function and an uncertain reliability model which cannot be represented analytically, the problem representation in (4.1) no longer fits. This research uses the flexibility of *evolutionary optimization* methods, mentioned in Chapter 1, to overcome these obstacles. Evolutionary approaches do not require a well-behaved, analytic problem definition or value model. Instead, they rely on simulations to generate arbitrary function evaluations. Since objective function *gradients* are not required, we can more easily adopt the more realistic reliability models of Chapter 3.

Variants of two well-known evolutionary methods are developed here for a general reliability design problem. The *simulated annealing* approach uses one trial solution which "evolves" in the search process, while the *genetic algorithm* maintains a population of solutions that evolve according to the concept of natural selection. The means by which a new set of solutions "evolves" from the current one differ considerably between the two methods.

#### 4.3.1 Simulated Annealing (SA) Algorithm

**4.3.1.1 Methodology:** Simulated Annealing, first applied in this thesis in Section 2.4, is designed to carry out a global search of the design space using an analogy to the thermodynamic process of annealing, in which a solid is heated and then slowly cooled to improve its material properties. This cooling process is represented by the temperature parameter ( $T$ ), which is initialized to a value comparable to or above the value of the initial design solution, or initial "energy" ( $U_0$ ). The temperature is slowly decreased as more iterations are conducted, representing the "cooling" of the search procedure and the desired convergence to a single globally-optimal solution. SA is discussed in more detail in [2-10,16,17,3-12].

Figure 4.1 shows a conceptual flow chart of the Simulated Annealing solution procedure. Beginning with a baseline solution vector  $x_0$ , the algorithm sets up outer and inner loops. In the outer loop, temperature is lowered by a constant multiplier ( $C_T = 0.90-0.95$ ), and a convergence check is conducted. The algorithm will terminate and report its current solution and (if different) the best solution ever found if the variation in the objective value, relative to the



**Figure 4.1: Simulated Annealing Flow Chart**

current value, is below a pre-set threshold  $\epsilon$ . The inner loop generates 15  $N_x$  (where  $N_x$  is the number of elements in the solution vector) successive new solutions. In each, a new design  $x_c$  is created by a random perturbation of the current solution vector  $x_o$ . This process is problem-specific, but it basically involves sampling of a new solution element from a distribution whose mean is the number in the current solution.

Once a new solution is generated, the objective function is evaluated for the new solution to give  $U_c$ . If  $U_c$  is an improvement over the current value, the new solution is automatically accepted (i.e., it becomes the new  $x_o$ ). Otherwise, a sample  $p$  is taken of a Uniform(0,1) distribution, and the new solution is accepted (despite being an apparent move in the wrong direction) if  $p < P_{acc}$ , where  $P_{acc}$  is a function of the current temperature (T):

$$(maximize \text{ value } U) \quad P_{acc} = \exp\left(-\frac{U_o - U_c}{T}\right) \quad (4.9)$$

Since  $P_{acc}$  decreases as the temperature is lowered, the algorithm will become less likely to accept solutions with “worse” objective evaluations as the number of outer-loop iterations increases. This allows the algorithm to search more of the design space in the early iterations, hopefully avoiding getting “trapped” in local optima. As the temperature decreases over time, the algorithm will be less likely to take exploratory steps, focusing more on changes that give immediate improvement. Convergence will generally occur by the time that no better solutions can be found and when the temperature has decayed to a number that is small compared to  $U^*$ , the value of the best solution found to that point.

Table 4.1 contains a summary of the key parameter settings for the spacecraft bus optimization problem. The initial temperature  $T_o = \$2$  million is chosen to give an initial probability of accepting a move that causes a decrease of \$1 million in mission value of 0.606

SA Parameter	Value	Notes
Initial temp.	$2 \times 10^6$	$V_{max} = 15 \times 10^6$
Num. temp.	300	# iter. for given T
Temp. mult.	0.90	dec. after 300 iter.

**Table 4.1: SA Algorithm Parameters**

( $= \exp(-0.5)$ ). This high initial probability of a "backward" step decreases markedly as the search proceeds, as the temperature is reduced 10% after each set of 300 new solution trials. These last two parameters of Table 4.1 are chosen based on the studies of SA performance in [2-10,16].

The basic concept behind SA is very simple, and it can quickly be implemented by engineers who have their own system-specific performance analysis codes. Since SA has proven to be a robust means of finding global optima for benchmark problems (such as the Traveling Salesman Problem (TSP) [3-12]) that have eluded more traditional approaches, it has great potential to be rapidly applied to a wide range of design optimization problems.

Various enhancements to the basic SA algorithm have been developed in recent years. SA variants that maintain an "ensemble" of several independent solutions at the same time allow wider searches and can provide a "population" convergence stopping criterion (as the separate SA evolutions converge toward the same solution). It is also possible to adjust the covariance  $S$  of the new solution perturbations by computing the covariance  $S_0$  of the last  $15N_x$  *accepted* new solutions and multiplying by a "growth factor"  $C_s > 1$  to enlarge the search space commensurate with the direction and magnitude of the previous solution vector moves [2-16]. Many other modifications exist [2-10,3-12] and can be applied when the basic algorithm has trouble converging efficiently.

**4.3.1.2 Problem Encoding and SA Perturbations:** The SA algorithm used in this study has a new and unique method of generating a new solution. Trial solutions are specified by a collection of integer "genes" that give the number of units (how many) of each spacecraft component type included in that design. In the spacecraft case, the solution (a vector of integers) is broken down into functional subsystems (as shown in Table 4.3). Each time a new solution is generated, at least one of the changeable subsystems (all but the first two in Table 4.3) is randomly selected for modification. Those not selected retain the same values as in the last solution.

A new method for generating random perturbations of existing SA solutions has been developed in this thesis. It maintains the SA perturbation philosophy of making small

changes in the current design more likely than large ones, and it makes design changes within subsystem partitions rather than randomly altering numbers of unrelated components. For each component in a subsystem to be changed, a pair of staircase functions is computed based on the current number of units ( $nc$ ) and the minimum and maximum number allowed for that component ( $m$  and  $M$  respectively). The minimum number is the number needed for mission success, and the maximum for a given case is the *lesser* of two quantities:  $2 nc$  or the absolute maximum number allowed. The probability function for the number of units in the new solution is chosen to be:

$$\begin{aligned}
 P[new = nx \mid old = nc] &= 2.5 / \{2(M - m)\} && \text{for } nx = nc \\
 &= \frac{1 - 2.5 / \{2(M - m)\}}{\sum_{i=1}^{M - nc + 1} i} nx && \text{for } M \geq nx > nc \\
 &= \frac{1 - 2.5 / \{2(M - m)\}}{\sum_{i=1}^{nc - m} i} nx && \text{for } nc > nx \geq m
 \end{aligned} \tag{4.10}$$

Equation (4.10) creates a "stairstep" distribution that peaks when  $nx = nc$ . Retaining the current number of units is thus quite probable. The more different a new number is, the less likely it is to be selected. Note that there is an equal probability of the new number being either higher or lower than  $nc$ . Because large changes are unlikely in a single trial; new solutions "evolve" from the best ones found thus far.

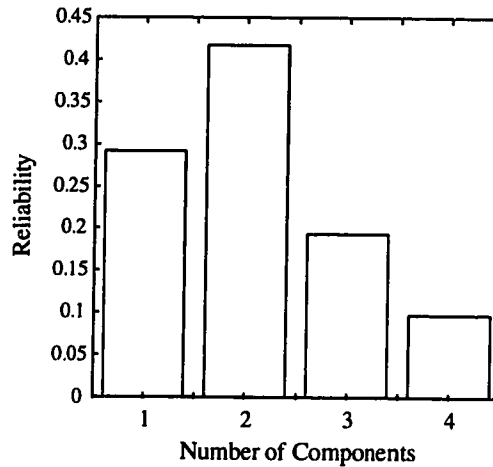
Figure 4.2 shows the stairstep distribution created by (4.10) for the case  $nc = 2$ ,  $m = 1$ , and  $M = 4$ , which is common in the subsystem redundancy allocation carried out in Section 4.4. Note that retaining the current solution component number is the most likely result, and the probability of choosing a different number is evenly divided among  $nx < nc$

and  $n_x > n_c$ ; thus the most likely change is to reduce down to  $n_x = 1$ , the minimum possible number. As  $n_x$  gets farther away from  $n_c$ , its probability decreases accordingly.

When the SA algorithm generates a new solution, it randomly applies the perturbation operator (4.10) to all components in a given *subsystem*. For the GP-B spacecraft bus, seven separate subsystems have been defined:

- |    |                         |                                 |
|----|-------------------------|---------------------------------|
| 1. | structure               | (1 component - unchangeable)    |
| 2. | thermal                 | (1 component - unchangeable)    |
| 3. | power                   | (4 components - 1 unchangeable) |
| 4. | communications          | (5 components)                  |
| 5. | computer/processing     | (5 components)                  |
| 6. | guidance and control    | (6 components)                  |
| 7. | solar array deployables | (5 components)                  |

Unchangeable components refer to things that the SA optimization algorithm is not allowed to change. This includes the design of the structural and thermal subsystems as well as the size (number of cell strings) of the solar panels in the power subsystem. Solar array reliability is a function of the number of strings provided, but adding strings increases the solar array panel area, which affects the solar pressure disturbances on the spacecraft in addition to increasing weight and volume. Thus, a change would require much more complex, integrated performance analysis software which is not practical here, although it is an important long-term goal. Similar issues exist for the helium thrusters (in guidance and control), but the total number of thrusters is allowed to be changed by the optimal search procedure.



**Figure 4.2: Solution Perturbation Distribution**

For the five subsystems that have changeable components, each time SA generates a new solution from the current one, it samples a Uniform(0,1) random variable for each subsystem. Subsystems whose uniform sample is below a threshold level  $P_n = 0.30$  (a 30% chance) will have their component numbers altered by sampling from the staircase distribution (4.10). The rest will not be altered at all for this particular perturbation. At least one subsystem must be altered -- if none of the subsystem samples is less than  $P_n$ , one of the five changeable subsystems is selected randomly for alteration (each has probability 0.2 of being selected). This procedure helps prevent solutions from changing too abruptly from one trial solution to the next, but it also insures a very high probability of at least some change in one of the variable subsystems.

Note that the perturbation approach described here has a couple of key parameters that can be adapted as part of the SA algorithm. First, of course, the subsystem probability threshold  $P_n$  can be varied to increase or decrease the average amount of variation from one solution to the next. Second, the key constant in the staircase distribution (4.10) is the parameter  $N_s = 2.5$  which is in the numerator of the probability calculation for  $n_x = n_c$  (and is by consequence in the other cases as well) and controls the likelihood of retaining the same number of components for a given solution entry. If this number is *increased*, the amount of variation produced by the staircase distribution *decreases*, and vice versa. As mentioned in the last section, it might be desirable to adjust one or both of these parameters to increase the search space (i.e., increase solution variability from step to step) if the current solution covariance  $S_0$  is too small, and the reverse is also true [2-16]. This has not been implemented in the spacecraft optimization procedure simply because SA converges quite readily without adjustments to  $P_n$  and  $N_s$ .

#### 4.3.2 Genetic Algorithms (GA's)

Genetic Algorithms get their name from their attempt to model optimization as a "survival of the fittest" battle among a population of possible solutions to determine the one that best satisfies the objective function. Rather than trying one solution at a time, as in Simulated Annealing, genetic algorithms maintain a population of  $N_p$  solutions and "evolve" them from one generation to the next using operators that are modeled on

Natural Selection and the reproduction of biological organisms. These operators, while not deterministic, tend to favor solutions which have higher objective function evaluations, or *fitnesses*.

Selected higher-fitness solutions can directly *reproduce* into the next generation, but most of the search effectiveness arises from the *crossover* and *mutation* operators. The former is the more important; it mixes the solution-vector entries, or *genes*, of two selected (for their fitness) “parent” solutions to create “children,” some of which will hopefully combine the better attributes of both parents into a superior offspring. The population will tend to converge toward similar-looking solutions that appear to give the best fitness, but the mutation operator exists to prevent premature convergence to a local optimum by randomly altering individual genes within the overall population with a small probability. Populations that appear to reject all mutated variants of the dominant solution can be said to have converged “close” to a global optimum -- there is no guarantee of reaching such an optimum in finite time [2-5].

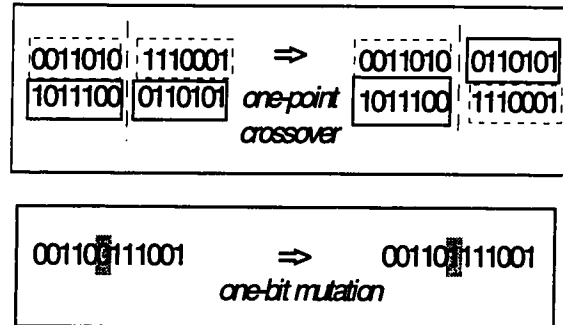
GA's can be said to search the objective function by *hyperplane*. A hyperplane is simply the set of all possible solutions that share a given sub-solution encoding. An example is:

###110###1#0

where the #'s are “wild cards” that can be any allowed entry. Hyperplanes whose common entries have a high fitness will slowly come to predominate in the population as the GA searches for the right combination of wild-card entries to maximize the value of the overall design. A formal explanation of this concept and its consequences is given in the *Schema Theorem* [2-5,9].

In the canonical genetic algorithm (GA) format [2-5], chromosomes, or members of a population of trial solutions, are expressed as binary numbers (0-1), and the standard genetic evolution operators are designed for this type of population. Figure 4.3 shows example cases of crossover and mutation operators for binary encodings. For this system design application, however, the integer solution encoding used for the SA algorithm in

Section 4.3.1.2 is much more natural. Thus, variants of the GA operators for this genetic format have been developed and are discussed here. It should be possible to improve them by incorporating more problem-specific information; this is left for future research.



**Figure 4.3: Canonical GA Operator Example**

As for Simulated Annealing, previous research on using Monte Carlo simulation to evaluate the objective function (or *fitness*) of population members has provided insight into the GA design parameters used here [2-1,6]. These are given in Table 4.2. The revised operators are:

**Reproduction:** *Roulette wheel* selection, in which solutions are randomly chosen with a probability that is proportional to their fitness, is used to choose solutions for the next generation (before crossover). Since the spacecraft bus value function evaluations tend to be similar, the fitnesses are linearly normalized from best to worst by multiplying the difference between the best fitness and a given solution fitness by 10. The best solution is always reproduced into the next generation (*elitism*), and the weighted-average equation (4.7) is used when applicable to update (rather than replace) the fitness of the best solution [2-2].

**Crossover:** Subject to the crossover rate in Table 4.2, after sets of parents are chosen by the reproduction operator, two solutions at a time are “mated” together to produce *one* offspring. The two parents simply average their unit numbers for each component type within a randomly selected (using the SA subsystem selection procedure - see Section 4.3.1.2) crossover window to give the number of

GA parameter	Value	Notes
Population size	25	duplicat. poss.
Crossover rate	0.6	after reprod.
Mutation rate	0.02	use (4.4)
No. simulations	500	per func. eval.
Converge tol.	0.01	

**Table 4.2: GA Parameters**

units in the offspring (randomly rounded up or down if the result includes a decimal:  $n.5$ ). The crossover rate determines the ratio of offspring to reproduced strings in the next generation, as  $N_{Cr}$  ( $= 0.6 N$ ) solutions are crossed over in  $N_{Cr}$  separate combinations to produce  $N_{Cr}$  offspring.

**Mutation:** Each gene (number of units for a given component) is subject to random mutation with the mutation rate probability given in Table 4.2. If a gene is mutated, the current number of units is replaced according to the *SA* probability equation (4.10). This function has a high probability of retaining the same number of units; so the mutation rate is inflated somewhat to compensate.

**Population Convergence:** The convergence test is conducted after fitness evaluation but before the next generation is reproduced. If the average fitness of the population differs from that of the best solution in the population by less than the tolerance given in Table 4.2, the algorithm stops and outputs the final solution population as well as the best single solution obtained thus far.

Genetic algorithms have several advantages over simulated annealing. Most fundamentally, the search of hyperplanes via the Schema Theorem is potentially much more particular than the simpler solution perturbation operators used in *SA*. In other words, GA's have a better chance of making fine distinctions within the problem encoding, making it more probable that they will arrive at the optimal entries for each solution vector element. Also, the evolution of a population of solutions in GA's rather than a single one provides more robustness to the optimal search (*SA* can also be modified to conduct multiple solution propagations simultaneously). Examination of the final population at convergence and comparison to the best solution found shows how much variation exists within the final population and may display a genetic "line" of solutions quite different than the dominant families but which nevertheless have competitive fitnesses. This gives the human designer a basis to do his own search, combining various elements of the final population that he or she thinks might complement each other.

However, GA's have a significant disadvantage for users who are not familiar with their inner workings: they are simply much more complex than SA and require incorporating many special subroutines into the software that computes system performance and value. Many GA source codes can be purchased or obtained through the Internet, but it is not usually possible to just use them as "black box" codes because they are heavily dependent on the solution encoding (as demonstrated above). In marked contrast, the basic SA concept is so simple that any engineer can write his own code of less than 50 lines to execute it, and one can include the solution perturbation algorithm that fits the problem at hand. In addition, SA codes which propagate a single solution are good enough for most applications; thus a great deal of computer time is saved by not having to evolve and evaluate a much larger set of trial solutions.

These decisive advantages of SA when used by design engineers of varying specialties suggest that SA be the preferred optimal search strategy. GA's definitely have their uses, however, especially when the design engineers have significant background in and understanding of them. Generally, GA's should be used only when they can be carried out with populations of non-trivial size (5 at a minimum; preferably at least 10 solutions) in a reasonable computing time and when *at least one* of the following two conditions applies:

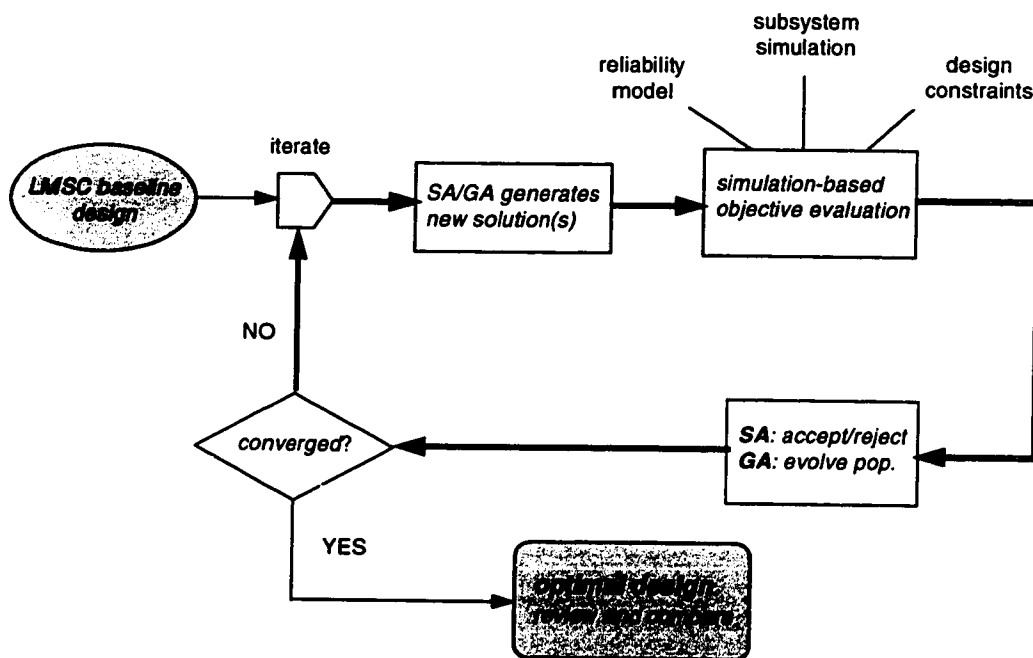
- search *particularity* and/or *robustness* are very important for a particular application (in this case, GA helps by searching more of the design space and by differentiating more between similar-looking solution encodings)
- a *binary* encoding naturally fits the solution space for a given problem (in this case, black-box codes for a "canonical" GA should work with few modifications)

For the spacecraft bus optimization problem, SA search has demonstrated sufficient particularity (ability to distinguish between values of similar solutions) and robustness, especially since, for the majority of component types, the value models are not very sensitive to small changes in redundancy. Attempts to run the GA search algorithm have not been able to come close to convergence in the same time that the SA code does

(i.e., within 18 hours). Faster convergence may be obtained by decreasing the mutation rate, but in any case, the good results from *SA* preclude the need to use GA's for this particular spacecraft bus application. The results in the following section thus come from runs of the *SA* algorithm only. However, Section 8.4 introduces an augmented-GPS network optimization problem where GA's are the preferred solution strategy.

#### 4.3.3 Optimization Process Flow Chart

Figure 4.4 is a flow chart which illustrates the overall simulation-based optimization procedure. All of the elements of the process work together smoothly in a compiled program that includes all of the relevant performance evaluation subroutines shown as well as the code which executes the evolutionary optimal search. Note that the final result is not the last word. Once convergence is obtained, the human designer must compare the result to his expectations and determine if the optimal result was chosen because user preferences were not fully modeled. Revisions to the value models and penalty functions are thus natural outcomes of a single run, and the entire process resembles the iterative design procedure that is familiar to systems engineers.



**Figure 4.4: Spacecraft Bus Optimization Flow Chart**









## 4.4 Spacecraft Bus Optimization Results

### 4.4.1 Baseline Design and Problem Formulation









Figure 3.1 in Chapter 3 shows the component redundancy of the LMSC baseline spacecraft bus design for which the reliability results in Figure 3.2 were computed. The separate lines indicate which components fall into the subsystem classes listed in Section 4.3.1.2. It also shows, at a simplified level, the number of components of each type needed for mission success (usually just one). This number serves as the minimum number ( $m$ ) for that component type in the encoding scheme. The absolute maximum number ( $M$ ) allowed in the encoding varies, and it is a somewhat arbitrary constraint in most cases. Generally, components with 2-for-1 redundancy in the baseline design have  $M = 4$ , and components with  $k$ -out-of- $n$  redundancy where  $k > 1$  and/or  $n > 2$  have  $M = 8$ . For the helium thrusters,  $M = 24$ , which is probably higher than necessary.

The encoding used here allows for single-point failure modes since the minimum number of components allowed ( $m$ ) is always equal to the minimum number needed to work. This is by design; one of the spacecraft design principles being tested here is the one which forbids non-redundant designs as too risky. However, if non-redundancy really were deemed to be unacceptable (regardless of the added weight or expense) by the decision-makers,  $m$  could simply be increased by one for each component class.

Using the reliability results from Section 3.4 along with the non-spacecraft-bus failure models of Section 4.2.2, the baseline design has an LMSC value of \$13.12 million and a Stanford value of \$13.48 million. These are already between 87.5% and 90% of the maximum possible value of a completely successful mission (\$15 million); thus the

Component	Min.	baseln.	optim.
Structure	1	1 (nc)	1
Thermal	1	1 (nc)	1
SA string	92	96(nc)	96
power reg.	1	2	
NiCd battery	1	2	
pwr. control	1	1 (ir)	
omni antenna	1	4	
circ. switch	1	2	2
RF switch	1	2	
trans. switch	1	2	
transponder	1	2	
cmd. proc.	1	2	

(nc) cannot be changed








Component	Min.	baseln.	optim.
telem. proc.	1	2	
remote proc.	1	2	
flight comp.	1	2	2
solid-st. rec.	1	2	
P/Y gyro as.	1	2	
star tracker	1	2	
R/Y gyro as.	1	2	1
thrusters	16	18	
ATC elect.	1	2	2
mass trim	1	2	2
SA release	4	4 (ir)	
SA separ.	1	1 (ir)	

(ir) internally redundant




 *redundancy removed*

 *redundancy added*

**Table 4.3a: LMSC Optimal Design Results**

Component	Min.	baseln.	optim.
Structure	1	1 (nc)	1
Thermal	1	1 (nc)	1
SA string	92	96(nc)	96
power reg.	1	2	
NiCd battery	1	2	
pwr. control	1	1 (ir)	1
omni antenna	1	4	
circ. switch	1	2	
RF switch	1	2	
trans. switch	1	2	
transponder	1	2	2
cmd. proc.	1	2	

(nc) cannot be changed

Component	Min.	baseln.	optim.
telem. proc.	1	2	
remote proc.	1	2	
flight comp.	1	2	2
solid-st. rec.	1	2	1
P/Y gyro as.	1	2	1
star tracker	1	2	2
R/Y gyro as.	1	2	2
thrusters	16	18	
ATC elect.	1	2	1
mass trim	1	2	2
SA release	4	4 (ir)	4
SA separ.	1	1 (ir)	1

(ir) internally redundant

 *redundancy removed*

 *redundancy added*

**Table 4.3b: Stanford Optimal Design Results**

potential improvement from an optimized design will not be dramatic. Optimal design runs using both the LMSC and Stanford objective functions and the SA algorithm have been conducted. Attempts have also been made to determine which input parameter changes show the most sensitivity in the optimal results.

#### 4.4.2 *Design Optimization Results*

Using the SA algorithm of Section 4.3.1, the best spacecraft designs obtained after convergence are given in Tables 4.3a and 4.3b, which contains the minimum numbers of working components, the LMSC baseline design, and the optimal component numbers for the LMSC and Stanford value functions, respectively. The algorithm was always initialized with the baseline solution, so it is to be expected that the objective value would drop as the SA algorithm explores a range of options. Figure 4.5 below has mean result values that show 5.2% improvement over the baseline design for the LMSC case and 3.5% improvement for the Stanford case. Running on a Sparc-10 workstation, convergence was generally obtained within 18 hours, but as noted above, the tolerance could have been increased (stopping the runs earlier) with little change in the results.

Note in Table 4.3 that the optimal designs show some consistent patterns when compared to the baseline solution. While the baseline has the same redundancy for all components, the SA-generated solutions remove redundancy from less-risky or cost-effective components and add redundancy to components that have high failure rates and/or failure rate uncertainties. This is not surprising, and it points out the potential sub-optimality of the traditional method of allocating redundancy, which applies redundancy everywhere because conventional spacecraft risk predictions cannot be trusted to isolate components that have the most impact on overall mission risk (see Section 3.3).

From these results, we see that in both the LMSC and Stanford cases, the optimal result for the NiCd battery is a single non-redundant unit. This not only violates the single-point failure guideline, it removes redundancy from a component that has a relatively high mean failure rate. While it is surprising at first glance, the reason becomes evident when one realizes that a certain amount of weight, power, and volume must be freed up to allow added redundancy for riskier subsystems. Removing one battery frees

up enough room for additional units elsewhere. A non-redundant battery does pose a non-trivial failure risk, but its mean failure rate uncertainty is relatively low since considerable orbital experience exists for it. Since the length of the primary mission is short, the high mean risk and the normal battery wear-out problems may come into play less, allowing the algorithm to take an educated risk in removing one battery to make way for additions in the “electronic box” subsystems (communications, processing, guidance and control), where components typically have higher failure rate uncertainties. Higher-uncertainty components pose special risks because they may have a non-trivial chance of having such a poor reliability (90% or less) that the design of the rest of the spacecraft becomes almost irrelevant. The degree to which these arguments are relevant for GP-B is studied in Section 4.4.6, where a variant SA evolution is run for the two-battery case.

It is also notable that separate optimization runs often give widely different optimal results for several component types, even when the same value model is applied. This pattern, evident for many of the switches and antennas (and the thrusters for reasons of limited cause-and-effect modeling), normally suggests that the objective value is not very sensitive to changes in these design elements. In these cases, spacecraft reliability is not much affected, and the system budgets are not significantly altered by the inclusion or removal of a single unit in one of these classes. Components which demonstrate this lack of sensitivity can thus easily be set to the “intuitive” redundancy level (2-for-1, 5-for-3, etc.) without controversy. Attention will then be directed where it belongs: to more detailed studies of the components, such as the NiCd battery, that give both consistent and surprising results.

#### 4.4.3 *Optimal Design Value Evolutions*

Figures 4.5a and 4.5b show the evolution of the LMSC and Stanford objective values, respectively, for the two runs which gave the results shown in Table 4.3. The dashed line represents the initial evaluation of the LMSC baseline design for each value model. Each evaluation point represents the value of the current SA solution at the end of each constant-temperature iteration. The last points on these plots represent the latest evaluations of the best solutions found. Convergence in Figure 4.5a appears rapid, but the

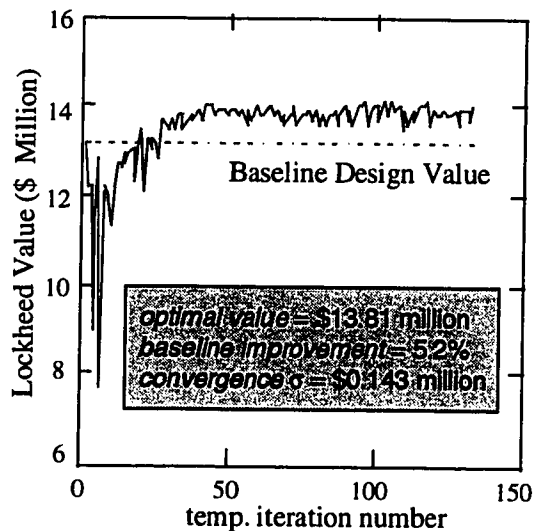


Figure 4.5a: LMSC Optimal Value Evolution

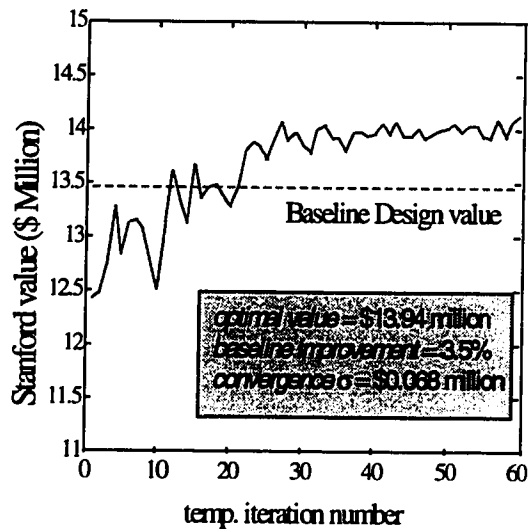


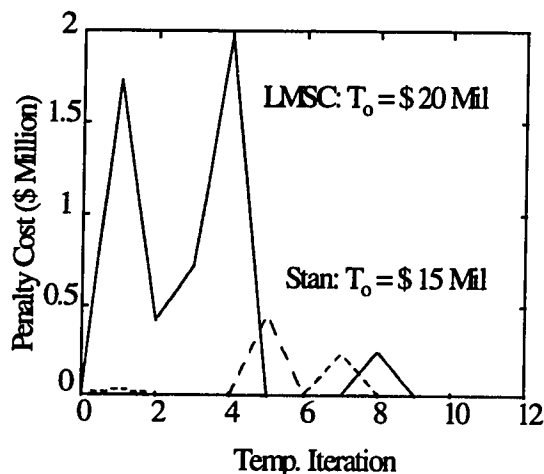
Figure 4.5b: Stanford Optimal Value Evolution

time scale is misleading, since the termination tolerance in the LMSC case is deliberately made tight in order to run more system simulations and to study the steady-state variance of the optimal design values.

One thing that is notable in Figure 4.5a is the decrease in the value of the SA solution during the first five iterations. With an initial "temperature" of \$2 million, the SA search was initially very tolerant of moves in the wrong direction, but this tendency was suppressed as the temperature was lowered, and the wide search of the design space was able to focus in an area that clearly achieved higher values than the starting solution. In the Stanford case, the starting temperature was set to be \$1.5 million because the value model had less randomness (since no award or cost fees were included). This prevented the SA search from accepting as many downward moves and thereby limited the search space explored. However, convergence to an improved solution was obtained more rapidly.

Figure 4.6 shows the evolution of penalty function cost for these two LMSC and Stanford runs. The use of penalty functions works well here, as no "invalid" solutions were accepted after the first 10 iterations in either case. This is evident because the penalty cost, after bouncing around a bit as different non-feasible solutions are tried early in the search process (when the SA temperature is high) falls to zero and stays there for

the duration of the search. If optimal solutions with non-zero penalty costs were to survive until convergence, this would suggest that the penalty function coefficients should be made more severe. However, if the penalty functions are grounded in a valid cost/benefit trade-off, it could also mean that solutions once thought to be infeasible actually give better bottom-line results. The flexibility provided by



**Figure 4.6: Penalty Costs for SA Runs**

“soft” constraint modeling thus has the important advantage of allowing one to check whether the pre-set constraints actually make sense at the system level, rather than enforcing “hard” constraints that are assumed to be inviolate and etched in stone. For example, if the final converged solution included a non-zero penalty cost, that would suggest that the constraint which it modeled may not be as critical as originally thought, since overall mission value was maximized when that constraint was slightly violated. If that constraint really were critical, its penalty function would have to be changed to make small violations of it more expensive.

#### 4.4.4 Optimal Spacecraft and Mission Reliability

Using the Stanford value model, Figures 4.7a and 4.7b show the resulting probability densities of system reliability at  $t = 1$  year (the end of the GP-B primary mission) for the spacecraft bus alone and for the overall mission, respectively. Table 4.4 compares the mean reliability of the optimal solution with the LMSC baseline in both spacecraft and mission categories. The baseline and optimal density curves in Figure 4.7 look very similar. Table 4.4 shows that a small improvement in mean reliability was obtained for the spacecraft bus, which translated into a slightly larger improvement in mean mission reliability. Improvements of a similar scale are also obtained at the worst-

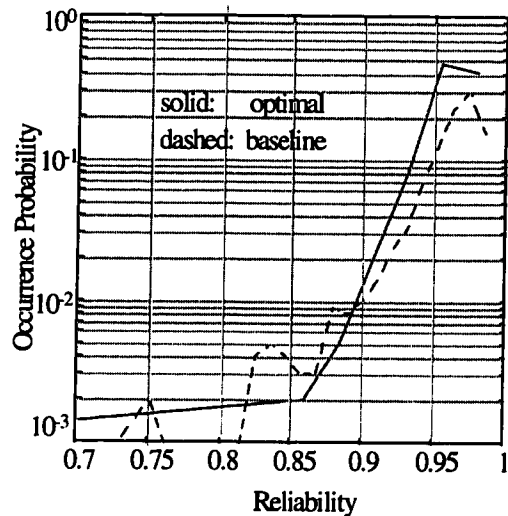


Figure 4.7a: Spacecraft Bus Reliability Density

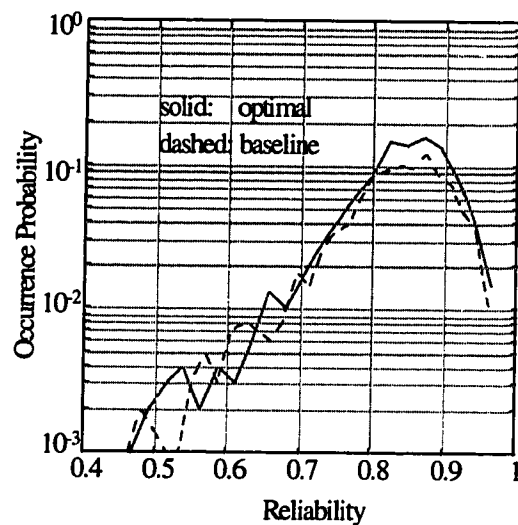


Figure 4.7b: GP-B Mission Reliability Density

case end of the reliability spectrum, meaning that the likelihood that system reliability will turn out to be much worse than the mean prediction has decreased a little.

The reason that more improvement is not forthcoming is simply that limited leverage exists to improve the overall mission reliability by tinkering with the spacecraft bus. After all, spacecraft bus reliability is much higher than that for the launch vehicle and experimental payload, and no control is exercised over these more significant risk sources. Even if enough spacecraft redundancy were added to bring the spacecraft mean reliability to 0.99, the mean overall mission reliability would only improve to about 0.855, which will not provide enough value-model benefit to compensate for the added cost. Furthermore, the constraint penalty functions prevent the optimal search from adding significant redundancy without taking it away somewhere else. In effect, the evolutionary search

	Spacecraft Bus	Overall Mission
<i>Baseline Design Mean Reliability</i>	0.957	0.827
<i>Optimal Solution Mean Reliability</i>	0.960	0.834
Mean Reliability Improvement	0.31 %	0.85 %
Unit Cost Decrease	\$ 3.1 Million	
Cost Savings	9 %	

Table 4.4: Reliability and Cost Improvement for Optimal Design

process has to “learn” that efforts to gain value by improving reliability (i.e., trial solutions which add overall redundancy) are non-optimal for this particular application.

At the same time, however, the search process learns how to achieve much greater improvement over the baseline solution by reducing overall redundancy (and component costs) while maintaining the baseline level of reliability. This is illustrated by the unit cost savings of \$3.1 million (or 9% of the total component cost) over the LMSC baseline solution as indicated in Table 4.4. Obtaining this level of cost reduction while still gaining a marginal system reliability improvement is a significant achievement, and it is obtained by removing restrictions on single-point failure modes and uniform redundancy patterns (e.g., in which every component class has 2-for-1 redundancy). Note that the Stanford value function summarized in (4.6b) includes credit for 10% of this cost savings, or about \$310,000, which accounts for *two-thirds* of the total improvement over the baseline design. The LMSC value function gives greater credit for unit cost savings as part of the “cost fee” (see Section 4.2.3.1), which explains why greater percentage improvement in bottom-line value (5.2% vs. 3.5%) is obtained for the LMSC case in Figure 4.5.

As noted in Section 4.4.2, for certain components, certain elements of the optimal redundancy vector can change significantly from one run of the SA algorithm to the next. It was suggested there that component types which display this trend are amenable to human designers who could adjust the results using their own qualitative design knowledge. Interestingly, attempts to do this for the spacecraft bus application have not yet been able to achieve a higher objective value (for either LMSC or Stanford) that is statistically significant. This supports the earlier hypothesis that the objective function in this case is *insensitive* to components whose optimal unit numbers vary noticeably between optimization runs. It also suggests that the SA optimal search is able to converge to specific component numbers in almost all cases where the objective function would be significantly affected.

#### 4.4.5 Objective Function Evaluation Uncertainty

One other facet that must be addressed is convergence *uncertainty* resulting from two related sources: (1) randomness in the Monte Carlo performance evaluation, and (2)

variation in the optimal solution from one *SA* iteration to the next. Using the Central Limit Theorem (CLT), the simulated function evaluations can be considered to be approximately Normally distributed as the number of simulation trials becomes sufficiently large [3-9]. The effective number of trials in the *SA* search increases over time since previous evaluations of a given solution are factored in using the update equation (4.7). From Figures 4.5a and 4.5b, we see that convergence around a limited range of result values is obtained after about 30 iterations. These figures contain a box which lists the mean optimal values ( $\mu$ ) and standard deviations ( $\sigma$ ) of these values for both cases. The figures make it clear that much less variability exists in the Stanford case. As noted in Section 4.2.5, this is due to the fact that one level of simulation (sampling of failure times for trial missions) is not necessary to evaluate the Stanford objective function. The larger one-sigma variation of \$143,000 in the LMSC case is non-trivial. While it could be reduced by more simulation trials for each function evaluation, it does not seem to be necessary for the *SA* search, since convergence around the mean result is not lost after it is acquired, and the best solution stops changing as the *SA* temperature approaches zero.

As long as objective function variability does not prevent convergence to the "right" solution, no changes to the search procedure are needed. Instead, it is much easier to take the final converged solution and simulate it with many more trials off-line. In the spacecraft bus case, an off-line performance simulation of the LMSC-case optimal result could be conducted with 500 million trials (instead of the 5000 used for each evaluation in the *SA* routine) in a couple of hours or less.





For problems where a more sophisticated response to simulation uncertainty is called for, various approaches exist. A simple one is to monitor the variance statistic in Figure 4.5 in "real time" and to only stop the Monte Carlo evaluations when sufficient certainty is attained. This idea is developed for GA's in [2-1]. The measure of statistical certainty can be based on a measure of risk aversion to the variability of the award result, but this will not normally be necessary. Note that in the *SA* case, the algorithm may continue after solution changes are practically impossible -- the code will simply keep adding simulation trials of the best solution found until the noise reaches an acceptable

level. Another important tool is the intelligent use of *variance-reduction techniques* in the design of the Monte Carlo simulations [3-9].




#### 4.4.6 Variant Solution with Two Batteries

Recall that in Section 4.4.2, the fact that both the LMSC and Stanford objective functions resulted in a non-redundant spacecraft battery (only one was supplied) was discussed. The SA method detailed in Section 4.3.1 makes it easy to do a sensitivity study of the battery redundancy issue. A separate design evolution is run in which the minimum number of possible components  $m$  (see Section 4.3.1.2) is reset from *one* to *two* for the battery only. In other words, although only one battery is needed to work, the designer can deliberately choose to limit the range of possible solutions so that at least one redundant battery is provided.

An SA evolution for this variant case has been done for the Stanford value function., and the resulting optimal solution is shown in Table 4.5. Not surprisingly, since the global optimum for the base case in Section 4.4.2 has only one battery, the optimal result for this more-constrained problem gives two (2) batteries -- the constraint  $m \geq 2$  is

Component	Min.	baseln.	optim.
Structure	1	1 (nc)	1
Thermal	1	1 (nc)	1
SA string	92	96(nc)	96
power reg.	2	2	1
NiCd battery	2	2	2
pwr. control	1	1 (ir)	1
omni antenna	2	4	2
circ. switch	2	2	1
RF switch	2	2	
trans. switch	2	2	
transponder	2	2	
cmd. proc.	2	2	

(nc) cannot be changed

Component	Min.	baseln.	optim.
telem. proc.	2	2	
remote proc.	2	2	2
flight comp.	2	2	1
solid-st. rec.	2	2	
P/Y gyro as.	2	2	1
star tracker	2	2	2
R/Y gyro as.	2	2	1
thrusters	17	18	
ATC elect.	2	2	2
mass trim	2	2	2
SA release	4	4 (ir)	4
SA separ.	1	1 (ir)	1

(ir) internally redundant



redundancy removed



redundancy added

**Table 4.5: Stanford Optimal Result with Redundant Battery**

active here. In adding a battery to the optimal solution shown in Table 4.3b, the optimal search decides to reduce the level of redundancy by one for the RF switch, remote processor, flight computer, and roll/yaw gyro assembly (the latter two become non-redundant as a result). Conversely, one level of redundancy is added to the ATC electronics (which were non-redundant in Table 4.3b), and numbers of components in a few other non-critical classes (such as antennas, and thrusters) go up or down a little as well. This variant solution does not incur any constraint cost (see Section 4.2.4), but it does add 54.5 kg of mass (a 7.4% increase), 12 W of power (a 6.9% increase), and adds 8.3% in unit volume. Thus, it has stayed within the constraint boundaries, but it has still eaten up significant design margin in each of the spacecraft budget categories.

In terms of unit cost, the variant solution has added \$2.8 million, which is an 8.6% increase over the solution in Table 4.3b. This represents a significant increase because it removes almost all of the \$3.1 million in savings that the original optimal solution achieved from the LMSC baseline design (see Table 4.4). This increase does result in a slight but noticeable *mean* reliability improvement. Compared to the results in Table 4.4, mean spacecraft reliability improves 0.83% to 0.968 and mean overall mission reliability improves 0.73% to 0.840. More interesting is the somewhat-unexpected reduction of spacecraft bus reliability *uncertainty* for this two-battery design. Compared to Figure 4.7a, where the  $R = 0.90$  crossover is at a likelihood of  $10^{-2}$ , the variant solution has the same crossover at  $2 \times 10^{-3}$ .

It is thus clear that the addition of the second battery improves spacecraft bus reliability, but the question posed by the Stanford user value function of Section 4.2.3.2 is whether this improvement is worth the \$2.8 million additional unit cost it requires. Recall that even for the LMSC baseline design, spacecraft bus mean reliability and uncertainty are lower than for the other two mission elements, the launch vehicle and the experimental payload. If one looks at the *mission* reliability uncertainty for the two-battery solution, it turns out that the spacecraft bus improvement is almost “washed out” by these two non-controllable categories. The (pessimistic)  $R = 0.60$  crossover in Table 4.7b gives a likelihood of  $4 \times 10^{-3}$ , which is just as good as the same crossover point for the two-battery variant. Therefore, the additional investment in the spacecraft bus does not

translate into significant reliability advantage for the overall GP-B mission. This is borne out by the overall Stanford value of \$13.79 million for the two-battery design, which is \$150,000 (or 1.1%) below the optimal result for the non-redundant battery design as shown in Figure 4.5b.

This process illustrates the relative ease by which a decision maker can tailor an evolutionary search to his or her own qualitative limits and “comfort zone.” If he or she really would not feel comfortable with a non-redundant battery, this implies that a certain degree of user preference is not modeled in the value function used for the search. Updating the value model to incorporate this is possible, but as shown here, it is not difficult to simply preclude a non-redundant battery in the design encoding. The result shown here indicates that this user-defined restriction costs him or her about \$150,000 in terms of the original value function (when compared to the optimal result without this restriction). The decision maker, given this tradeoff, can then make up his or her own mind as to whether a qualitative preference for two batteries is justified.

#### **4.5 Spacecraft Design: Conclusions and Significance**

The results shown here seem to justify the use of global optimization for this spacecraft design problem. Using simulated annealing, improvement can be obtained relative to the LMSC baseline design even though the objective functions are only partially sensitive to changes in component redundancy. Convergence of the global search did not take very long considering the complexity of the reliability analysis and the variance of the Monte Carlo simulations. The smooth pattern of convergence demonstrates that evolutionary global search using simulated annealing is a useful way of conducting reliability optimization based on the new reliability models developed in Chapter 3.

The usefulness of this type of design optimization algorithm is threefold. First, it allows search of the design space to globally optimize an *arbitrary* value function based on an arbitrary system performance model. Second, it demonstrates the flexibility of evolutionary global optimization using simulation evaluations, since it accommodates domain-specific modifications to canonical SA and GA that improve search efficiency.

Lastly, it is a design tool that allows the user to complement the computerized search by making manual variations to the optimal result in an attempt to gain further improvement and to explore the sensitivity of the solution to the value function and search parameters.

One of the results of the optimizations carried out here is controversial: the discovery that the policy of avoiding all single-point failure modes is not optimal for several of the GP-B component classes. This is not really surprising, since the prohibition against designs with single-point failures is meant to bound the amount of uncertainty faced by designers and users rather than to always represent the best course of action. Since predictions of the traditional spacecraft reliability models are taken with a grain of salt, other design principles have been introduced to shield spacecraft customers from some of the resulting performance risk. The key benefit of the reliability models of Chapter 3 is that by providing better estimates of the underlying uncertainty, some of the “safety net” that now exists in spacecraft engineering can be removed, allowing direct cost/benefit optimization to be carried out. The optimization approaches in this chapter take advantage of this because they can handle any models of user preferences and risk aversion. In other words, simulation-based optimal search can determine which users should invest more to reduce and limit spacecraft performance uncertainty and which ones are better off taking more “educated risks”.

In addition to incremental improvements in value models and evolutionary search operators, further development of this probabilistic optimization capability for spacecraft should come from two different areas of research. While this thesis focuses on improved reliability modeling of spacecraft orbital performance, that is only one element of the total uncertainty picture facing a spacecraft development program. Design of the spacecraft itself is but a small part of such a program; the rest lies in managing the pace of technology development, prototype manufacture, testing of both prototypes and finished flight hardware and software, and cost/effective manufacturing and delivery of the product to the launch site. Much more improvement in cost-effectiveness would be possible if engineers had a consolidated model of overall program risk when working on the initial feasibility analyses and contract proposals. Not only would this allow one to optimize the key programmatic decisions early on, it would also provide a complete “uncertainty map”

for the project which can be updated as new information (test results, for example) is obtained, allowing for optimal decision-making every step of the way. The program-development decision tree shown in Figure 1.4 of Chapter 1 serves as a model for making all program decisions within the resulting risk-assessment framework.

At the same time, our current focus on the spacecraft design itself can be extended by integrating the redundancy optimization carried out here with more detailed prediction software that simulates the dynamics of each spacecraft subsystem. To illustrate by example, decisions on optimal redundancy of the guidance-and-control subsystem components could be conducted hand-in-hand with optimization of the controller and estimator feedback gains if the spacecraft performance simulations had an integrated simulator for the resulting controller time histories. In addition to providing a tool for simultaneous optimization of all of the major design variables, more-detailed performance simulations would remove the need to approximate the consequences of component failures on overall performance. The model used here assumes that the minimum-working requirements of Figure 3.1 are absolute, but in reality, functioning at a degraded level may still be possible after “mission failures” occur in a particular component class. The concept of a small spacecraft-design company centered around software and hardware with this unified-simulation capability will be introduced in Chapter 8. It is certainly possible to develop end-to-end software simulations -- the hardest challenge comes from choosing the level of abstraction that provides maximum fidelity while allowing the optimization process flow pictured in Figure 4.4 to converge in a “reasonable” amount of time.

## Chapter 5: GPS System Integrity Monitoring

*Warning systems can be a powerful way to reduce risks. They may also create risks themselves. The best warning systems are not necessarily the most sensitive ones. The sensitivity that gives optimum benefits (i.e., maximum risk reduction) depends on the expected response to signals, the reliability of the warning system, and the values that are involved in the trade-off between the consequences of Type I (missed alert) and Type II (false alert) errors.*

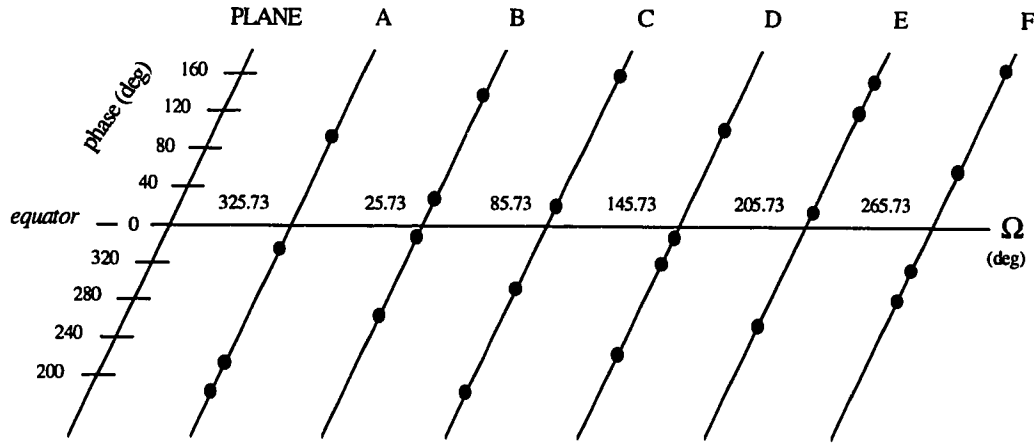
- Prof. M.E. Pate-Cornell, "Warning Systems in Risk Management". pp. 233-34.

### 5.1 Global Positioning System (GPS) Performance Simulation

The *Global Positioning System* (GPS) was introduced in Section 1.6. In order to evaluate the effects of model uncertainty on GPS navigation performance for various classes of users, a simulation of GPS geometries, ranging errors, and the resulting user position errors is needed. The basic elements of GPS satellite simulation are an almanac that describes the nominal ephemerides of all GPS satellites (and other satellites that provide ranging signals, if any), the location of users receiving GPS broadcasts, ranging error models, and code to solve the overdetermined least-squares problem to obtain solutions for 3-D position and user time bias as well as errors in these quantities. This section describes in detail how this is done. This basic simulation format is used in all of the GPS performance studies reported in the next four chapters of this thesis.

#### 5.1.1 GPS Satellite Constellation Almanac

As explained in Section 1.6, the 25 currently-operating GPS satellites are arranged into six orbital planes. Figure 5.1 is a plot of orbital phase vs. right ascension shows the partitioning of 24 "primary" satellite locations into these six planes (additional satellites can serve as active spares). Since this the constellation of orbits repeats itself every (sidereal) day from the perspective of a user fixed on Earth, the simulation of GPS satellite orbits can be structured to sample from a single day of local user time while spanning the complete set of possible geometries. However, once satellite failures are introduced, sampling over many days may be needed for adequate statistical significance.



**Figure 5.1: NAVSTAR Constellation Orbit Geometry**

Satellite orbital simulation is used in all of the GPS performance analyses conducted in this research. For this purpose, a model of nominal circular orbits with the parameter set of Figure 5.1 is constructed, and a matrix  $\mathbf{D}(t)$  of nominal satellite positions in xyz Earth-centered (but non-rotating) coordinates can then be computed as a function of GPS system time  $t$ :

$$\begin{aligned}
 M^i(t) &= \cos(\omega_{sv} t + M_o^i) \\
 \mathbf{D}_x^i(t) &= R_{sv} [\cos(M^i) \cos(\Theta^i) - \sin(M^i) \cos(I) \sin(\Theta^i)] \\
 \mathbf{D}_y^i(t) &= R_{sv} [\cos(M^i) \sin(\Theta^i) + \sin(M^i) \cos(I) \cos(\Theta^i)] \\
 \mathbf{D}_z^i(t) &= R_{sv} [\sin(M^i) \sin(I)]
 \end{aligned} \tag{5.1}$$

where  $i$  is the spacecraft number ( $1 \leq i \leq 24$ ),  $\Theta^i$  is the right ascension of the ascending node for spacecraft  $i$  (depending on which of the six orbital planes it is in),  $I$  is the orbital inclination,  $R_s$  is the radius of the orbit, and  $M_o^i$  is the orbital mean anomaly for each satellite at  $t = 0$  (discussed below). Note that all angles are assumed to be expressed in radians. User positions are also computed as  $\mathbf{u}(t)$  in which the Earth's rotation rate  $\Omega$  (rad/sec) is included (note that a spherical Earth is assumed here):

$$\begin{aligned}
u_x(t) &= R_E \cos(U_{LAT}) \cos(U_{LON} + GRA + \Omega t) \\
u_y(t) &= R_E \cos(U_{LAT}) \sin(U_{LON} + GRA + \Omega t) \\
u_z(t) &= R_E \sin(U_{LAT})
\end{aligned} \tag{5.2}$$

where  $U_{LAT}$  and  $U_{LON}$  are the user latitude and longitude (in radians),  $R_E$  is the Earth's radius, and  $GRA$  is the right ascension of Greenwich, England (at  $0^\circ$  Longitude) on a given date. User line-of-sight vectors to each of the satellites are then computed as explained in Section 5.1.2.

All of the simulations in this research use the Volpe almanac data provided by Karen Van Dyke to serve as a consistent baseline. It provides the mean anomalies for the 24 GPS satellites and the right ascensions of the six orbital planes as of July 1, 1993 (these are slightly different from those shown in Figure 5.1). Of course, these may change over time as the OCS moves spacecraft around to fill “gaps” that may develop in the constellation. Simulations can be run for the latest configuration by simply applying the almanac message from a recent GPS receiver observation, or a recent almanac may be downloaded from the U.S. Coast Guard Navigation page on the Internet (<http://www.navcen.uscg.mil/gps/gps.htm>). However, simulations with other almanacs have not produced results noticeably different from those produced by the Volpe data.

Satellite geometry simulations propagate the orbits forward in time by updating the GPS system time  $t$  by a random amount. Depending on the application, the update parameters will vary, but the basic concept is to update GPS time by sampling from a  $\text{Uniform}(a,b)$  distribution, where  $a$  and  $b$  are the minimum and maximum update times. The default update parameters are  $a = 0$  minutes,  $b = 30$  minutes, which gives a mean update time of 15 minutes. In some cases, a more limited range of [5 min., 20 min.] is preferred, and in simulations that examine navigation service *outage* durations, a short, deterministic update time of 1 minute is used. All updates are cumulative; the system time simply propagates forward from a randomly-sampled starting time. Using constant 1-min. updates, the nominal constellation configuration repeats every 1436 trials (1 sidereal day). The random-update method is designed to avoid this repetition even though the total time

“elapsed” may span many days or weeks. Simulations of satellite service failures also provide an element of non-repeatability (see Section 5.3.2).

As discussed in Chapter 8, many civilian augmentations to the current GPS system are being considered. The provision of GPS-like ranging capability is planned for inclusion on Inmarsat geosynchronous (GEO) satellites which will also broadcast the Wide Area Augmentation System (WAAS) correction message (see Chapters 6-7). Current plans call for the lease of a transponder for this purpose on four Inmarsat GEO satellites stationed over the Equator at AOR-W (54° W Longitude), AOR-E (15.5° W), IOR (63.5° E), and POR (178° E). GEO satellites are easy to include in the satellite orbit simulation: they have  $I = 0$ , satellite radius  $R_{SV} = 5.6 R_E$ , thus:

$$\begin{aligned} \mathbf{D}_x^{i,GEO}(t) &= 6.6 R_E \left[ \cos(SV_{LAT}^i) \cos(SV_{LON}^i + GRA + \Omega t) \right] \\ \mathbf{D}_y^{i,GEO}(t) &= 6.6 R_E \left[ \cos(SV_{LAT}^i) \sin(SV_{LON}^i + GRA + \Omega t) \right] \\ \mathbf{D}_z^{i,GEO}(t) &= 6.6 R_E \left[ \sin(SV_{LAT}^i) \right] = 0 \end{aligned} \quad (5.3)$$

where  $SV_{LAT}^i$  is always zero. Note that their Earth-centered xyz-coordinate positions change with time due to satellite orbital motion.

### 5.1.2 User Locations and Position Calculations

A user with a GPS receiver can determine his location anywhere on Earth if he can receive the broadcasts of at least four GPS satellites and the resulting geometry matrix from (5.7) is non-singular (if only four are visible). Now that the system has reached FOC, this is the case well over 99% of the time, at least in CONUS [6-3]. The C/A signal used by SPS is broadcast by each satellite on the L1 frequency (1.575 GHz). It contains an almanac which gives satellite location and health information as well as ephemeris and ionospheric correction data. The heart of the signal is the satellite time at which each signal is broadcast. The receiver “delay-locks” to the C/A code and can determine, by matching its own C/A code with that received from a known GPS satellite, the receiver can determine a “pseudorange” to that satellite. This range contains an unknown time bias

(the receiver clock is not perfectly synchronized with GPS master time) which affects all pseudorange measurements equally and becomes the fourth location parameter to be solved for (in addition user position in three dimensions). By knowing ranges to four known (from the ephemeris message) locations in space, one's position (and time bias) can be uniquely determined.

In simulations which evaluate user navigation performance, the actual location of the user depends on the application. The location of Stanford University (37.43° N, 122.17° W, 6.94 meters elevation) is the default when only one location is evaluated, as in the work done in this chapter. Stanford has considerably better-than-average GPS satellite visibility, though. Analyses described in the following chapters either use a set list of locations or randomly sample within a defined geographic area.

For a given user location, the satellite visibility for any given constellation geometry sampled using the method of Section 5.1.1 is determined by converting all of the satellite location vectors  $\mathbf{D}^i$  into an East-North-Up coordinate frame fixed at the user's location (and rotating with the Earth). This is done by multiplying by the transition matrix  $\mathbf{T}$  defined by:

$$\mathbf{D}_{enu}^i = \mathbf{T}(\mathbf{D}_{xyz}^i - \mathbf{u}_{xyz}) \quad (5.4)$$

where

$$\mathbf{T} = \begin{bmatrix} -\sin(\lambda) & \cos(\lambda) & 0 \\ -\cos(\lambda)\sin(\phi) & -\sin(\lambda)\sin(\phi) & \cos(\phi) \\ \cos(\lambda)\cos(\phi) & \sin(\lambda)\cos(\phi) & \sin(\phi) \end{bmatrix} \quad (5.5)$$

and

$$\lambda = U_{LON} + GRA + \Omega t \quad \phi = U_{LAT}$$

Once the spacecraft position vectors are expressed in a frame relative to the user, we can determine whether or not that user can see each of the active satellites by computing the user's *elevation angle*  $\epsilon^i$  to each one from the “z”-component (up) of the unit vector to each satellite:

$$\epsilon^i = \arcsin(D_{enu}^i |^u) \quad (5.6)$$

Satellites whose elevation angle is greater than the visibility “mask angle”  $E$  of the user’s receiver antenna are then deemed to be visible. For this research, the user’s mask angle is assumed to be  $7.5^\circ$ , and the mask angle of any DGPS reference station (see Chapters 6-7) is taken to be  $5^\circ$ . Depending on the number of channel’s in the user’s receiver, he may or may not be able to use all of the visible satellites in his position solution. The default assumption here is “all in view”: the user includes all visible (and healthy) satellites. If that is not possible, the user receiver must apply *satellite selection* algorithms to decide which satellites to use. This determination is usually made by finding the set of satellites that gives the best “geometry” in terms of user position-fix accuracy. This concept is explained further below.

Given the matrix  $\mathbf{D}_{\text{enu}}$  of all satellite position vectors with respect to the user, an  $n \times 4$  user geometry matrix  $\mathbf{G}$  is constructed, where  $n$  is the number of usable satellites in view:

$$\mathbf{G} = \begin{bmatrix} -\bar{D}_e^1 & -\bar{D}_n^1 & -\bar{D}_u^1 & 1 \\ -\bar{D}_e^2 & -\bar{D}_n^2 & -\bar{D}_u^2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -\bar{D}_e^n & -\bar{D}_n^n & -\bar{D}_u^n & 1 \end{bmatrix} \quad (5.7)$$

The barred elements of  $\mathbf{D}$  are unit vectors, i.e., the three elements of each applicable row of  $\mathbf{D}_{\text{enu}}$  are normalized by the magnitude of the resulting 3-element vector. The basic measurement equation for user pseudorange then becomes:

$$\mathbf{z} = \mathbf{G}\mathbf{x} + \mathbf{v} \quad (5.8)$$

where  $\mathbf{z}$  is the vector of pseudorange measurements for  $n$  satellites in view,  $\mathbf{x}$  is the  $4 \times 1$  navigation state vector in three position coordinates and the clock bias (in most simulations, the true location is  $[0, 0, 0]$ ), and  $\mathbf{v}$  is a random vector of measurement errors

(noise). For  $n \geq 4$ , the best user position fix vector (relative to truth in this case) is given by:

$$\hat{\mathbf{x}} = \mathbf{G}^* \mathbf{z} = (\mathbf{G}^T \mathbf{G})^{-1} \mathbf{G}^T \mathbf{z} \quad (5.9)$$

where  $\mathbf{G}^*$  is the *pseudoinverse* of  $\mathbf{G}$  for the overdetermined ( $n > 4$ ) case. The key mapping between pseudorange error statistics and those for user position error (assuming ranging errors are i.i.d. and Normally-distributed) is given by the Dilution of Precision (DOP), which is a function of the geometry matrix only:

$$\mathbf{G}_{\text{DOP}} = (\mathbf{G}^T \mathbf{G})^{-1} \quad (5.10)$$

The diagonal terms of the  $4 \times 4$  matrix  $\mathbf{G}_{\text{DOP}}$  give the DOP parameters for horizontal (HDOP), vertical (VDOP), position (PDOP), and overall geometry (GDOP) as follows [6-13]:

$$HDOP = \sqrt{\mathbf{G}_{\text{DOP}}[1,1] + \mathbf{G}_{\text{DOP}}[2,2]} \quad (5.11a)$$

$$VDOP = \sqrt{\mathbf{G}_{\text{DOP}}[3,3]} \quad (5.11b)$$

$$PDOP = \sqrt{\mathbf{G}_{\text{DOP}}[1,1] + \mathbf{G}_{\text{DOP}}[2,2] + \mathbf{G}_{\text{DOP}}[3,3]} \quad (5.11c)$$

$$GDOP = \sqrt{\mathbf{G}_{\text{DOP}}[1,1] + \mathbf{G}_{\text{DOP}}[2,2] + \mathbf{G}_{\text{DOP}}[3,3] + \mathbf{G}_{\text{DOP}}[4,4]} \quad (5.11d)$$

Under the i.i.d. Gaussian assumptions mentioned above, these DOP parameters simply multiply the expected one-sigma pseudorange error  $\sigma_R$  (assumed to be the same for all satellites) to give the expected one sigma position error  $\sigma_P$  for the relevant axis (i.e., VDOP  $\sigma_R = \sigma_P$  in the vertical direction). These DOP parameters primarily measure the effect of the current satellite geometry on user positioning accuracy. VDOP tends to be the worst one, since most satellites are looking “down” on the user, giving the least geometric “leverage” to user position-determination in the vertical axis. As a result,

vertical error is usually the limiting factor for precise aircraft navigation applications, especially since the requirements for this application are the most stringent as aircraft approach the ground. DOP factors are a convenient way to represent the overall “quality” of GPS satellite geometries, even when the assumptions that govern their use do not hold.

### 5.1.3 SPS Ranging and Position Accuracies

The pseudorange errors for the Standard Positioning Service (SPS) are dominated by *Selective Availability* (SA), which is the quickly-changing deliberate error signal applied primarily to the satellite clock (although it could also be applied to the ephemeris message). Table 5.1 lists the SPS error sources and typical one-sigma ranging errors (which assume 10-second averaging of raw pseudoranges) that are used in this research. They date back to the late 1980's and are quite pessimistic for modern receivers, but they are retained here to compare the results obtained here with those published in the original papers on RAIM [7-8,10]. In any case, the dominating SA error term leaves the “bottom-line” one-sigma error (RSS'ed from the individual error sources) more or less unchanged at 32.4 meters [7-11]. Section 5.4.5 presents the solution for the case where up-to-date error models [6-13] are introduced *without* the effects of SA.

### 5.1.4 Performance Evaluation Concepts and Terms

The evaluation of overall user navigation performance for a set of GPS simulations depends on the requirements for the application being studied. In almost all cases, this evaluation can be summarized by four parameters: performance (accuracy), integrity,

Error Source	One-Sigma Error (m)	Noise Type
satellite clock/ephemeris	5	colored
ionosphere and troposphere	10	colored
receiver noise/multipath	15	white/colored
selective availability (SA)	30	colored

**Table 5.1: Errors for Normal SPS Operation**

continuity, and availability, which form the acronym “PICA”. Detailed explanations of each of these terms are given below. Note that they are concept definitions which express the intent of “official” definitions published in (for example) [6-11], but they do not necessarily follow the letter of these sources, as the focus here is on general principles of navigation system assessment.

**5.1.4.1 Performance:** This is measured by 95% user position *accuracy* under so-called “normal conditions”, i.e., when the Signal-in-Space (SIS) and all relevant error sources are within expected tolerances. The Normal pseudorange and position error distributions mentioned in Section 5.1.3 thus apply, and the 95% position error bound in any given dimension can thus be approximated by twice the one-sigma position error. This can be decomposed into ranging error and relevant DOP, but it is more accurate to simply store simulated position errors in a histogram and then compute the 95% error limit from this histogram in post-processing.

Accuracy is the most commonly-cited comparison statistic for different (augmented) GPS system architectures because it is the only one that is easily measurable. Ranging error estimates and DOP calculations as per (5.10-11) allow approximate accuracy predictions from simple covariance analyses, and field tests can obtain enough actual performance data to statistically infer a reasonable mean and 95% error bound. A key problem, however, is that field or flight-test data is very situation-specific. Any given test has a particular prototype mechanization with problems and biases that can be factored out but may take a different shape when a production system is developed. Satellite geometry also varies among field-test results, and a simple translation of the inferred statistics to a different geometry using (5.10-11) is not always representative. Simulations cannot demonstrate basic *feasibility* for a given system, but once that is established by experiment, they can provide a more complete basis for PICA evaluation and comparison over a wide range of operating conditions.

Accuracy for a given system is also highly correlated with the other PICA parameters discussed below, because systems whose accuracy is very good relative to the sizes of errors that pose safety threats or require aborts will be less likely to encounter

such risky situations. PICA assessments should therefore not be seen as four independent evaluations but as a single strongly-correlated whole that is produced by a set of interwoven simulations.

**5.1.4.2 Availability:** Availability is essentially the probability (over time and operating conditions) that the navigation service will be usable at the point at which the planned use is to begin. In an aircraft application, an airplane would have to abort its intended mission if the GPS-based navigation service is known to be unavailable and if it is a “sole-means” (i.e., no separate backup system can fill the role) application.

Loss of availability can result from a number of causes. System failures that prevent the reception of an adequate SIS (or differential correction message if needed) are one obvious source. Even if the system is operating nominally, ranging error behavior and/or poor satellite geometry could provide an expected positioning accuracy which is outside the normal 95% error tolerance. In addition, systems that require integrity monitoring may lose availability if conditions make the use of the monitor impossible or unsafe. The formal definition in [6-11] requires that all accuracy, integrity, and continuity requirements for safe operation be met at a given time for the system to be available.

In simulation, availability can be handled in a couple of ways. It may be determined in post-processing by looking at the resulting distributions of number of satellites in view and satellite geometry (expressed as a DOP) and then computing how many cases would have been unavailable. The PICA user cost model developed in Section 5.3.4 allows the DOP thresholds which determine availability to be optimized in post-processing. Alternatively, availability can be assessed during individual simulation trials by applying predetermined conditions at the availability decision point.

**5.1.4.3 Continuity:** Continuity is similar to availability in that it occurs when the user is warned that current conditions do not allow him or her to proceed with an application that has already begun. The service interruption thus happens unexpectedly during the execution of the mission, *after* it had been determined that the system was available. This can result from a failure during the mission that prevents further position/velocity fixes,

such as the sudden flagging of one or more satellites as unhealthy or a GPS receiver shutdown. In other cases, the problem will be more subtle, requiring the integrity monitor (if one is provided) to detect it and alert the user.

Like unavailability, loss of continuity has an inconvenience cost resulting from the cancellation of the intended operation. In addition, since continuity breaches occur after initiation of the application, a sudden abort may have to be improvised under hazardous conditions. Thus, a continuity breach carries with it a significant threat to safety, albeit a threat much lower than would be the case if the user were not warned of the problem. Because loss of continuity occurs during a relatively brief “exposure time” interval, simulations which measure the probability of loss of continuity must be time-based and have an update time short enough to allow the simulation to register their occurrence. This is not explicitly done in this research, but future plans call for the adaptation of our current failure simulations into a Markov chain with short update times, along the lines of [6-4,15].

A simplified continuity assessment is possible from the perspective of integrity monitoring, however. If we look at a “snapshot” in time, the integrity monitor must make a decision as to whether continued navigation is safe. If it warns the user of a hazardous situation when the true situation is safe enough for continued operation, that constitutes a *false alarm* (FA), which is a loss of continuity and a loss of the availability of the system. Detection of an actual hazard is also a continuity breach but a desirable one. If the monitor should fail to warn of an actual hazard, this is a *missed detection* (MD) which is the least desirable outcome, as explained in the next section. The probabilities of FA and MD events in this abstracted sense can be determined by the simulations described in Section 5.3.

**5.1.4.4 Integrity:** The probability of loss of integrity refers to the probability that the user is not warned of a situation or event that poses an unacceptable risk. It is thus equivalent to the missed detection probability mentioned in the last section if an integrity monitor is present. Depending on the system, it may be possible to switch to redundant systems or isolate the best-guess failure cause without sending a warning. Loss of

integrity would then occur if this decision to continue is faulty. Note that this definition of integrity assumes there is some underlying failure in the system that can be detected or isolated. It is also possible for a dangerous situation to result under “fault-free” conditions, and here it is assumed that no warning is possible. In other words, no specific system element is operating out of its specified range, but a combination of “rare-normal” events nevertheless places the aircraft in a hazardous situation. Integrity threats that cannot be detected must be shown to be “extremely improbable”, meaning a probability of occurrence of no higher than one in a billion. In the fault-free case, this is based on a “tail-extrapolation” of the system accuracy distribution under normal conditions.

For most safety-critical navigation applications, requirements on these four elements are issued by a governing body such as the Federal Aviation Administration (FAA). Performance and availability requirements are set to ensure that the system does not inconvenience its users (or other uninvolved people) by not operating in an efficient, predictable manner or by simply not being reliable. Continuity and integrity requirements, on the other hand, are designed to insure a minimum level of safety. In the case of aircraft precision approach and landing (to be discussed in detail later), safety requirements are based on both the heritage of older systems such as the Instrument Landing System (ILS) and a simple risk allocation to the navigation system of the overall fatal accident risk for a commercial airliner [6-10]. The simulations reported in this research can be used to evaluate PICA performance relative to these requirements, but the focus here is more fundamental. In this research, instead of simply accepting requirements as written, these specifications are used as references in the construction of an objective model of user costs and benefits. This value or preference model follows the philosophy of decision analysis as explained in Section 1.2.2 and Appendix A. It is constructed at several levels of detail; thus it can be used to both optimize system parameters and to help determine what the system-level requirements should be.

## 5.2 GPS RAIM Integrity Monitoring Approaches

This section explains the basics of Receiver Autonomous Integrity Monitoring (RAIM) for stand-alone GPS (SPS) users. Unless GPS navigation is supported by a completely separate source of information or by non-DoD external messages regarding GPS satellite health, a user will have no other means of determining whether or not the GPS signals are safe to use for his or her intended application. The essentials of the traditional approach to “snapshot” (one epoch only) RAIM are introduced, and weaknesses in the assumptions traditionally used to compute decision thresholds are discussed. A more generalized method for finding optimal decision thresholds will be introduced in the Section 5.3.

### 5.2.1 Mathematical Background

The basic concept behind RAIM is the use of additional information to verify position solutions. Since more than the minimum number of four satellites will be visible in almost all cases [6-15], redundant satellite pseudorange information is available, and the position fix may be computed from a best fit to the overdetermined data (5.9). The consistency of the redundant measurements provides a clue as to whether a GPS satellite or some other unit is operating out of specification and whether this error makes the position solution unusable.

Traditional RAIM methods are based on variations of the parity-vector based threshold tests described in [7-8,9,10]. Most of these algorithms have been shown to be functionally identical in [7-2]. Because they are expressed as practical, usable algorithms, the CFAR and CPOD algorithms given in [7-11], which attempt to set thresholds to achieve constant false alarm rate and constant probability of detection, respectively, will be taken as representative of this methodology.

The basic GPS measurement equation is given by (5.8), in which the noise vector  $\mathbf{v}$  is assumed to have a Gaussian distribution with a mean  $\mu = 0$  and a variance of  $\sigma_n = 32.4$  m for each entry (from Table 5.1). Note that for a true position of  $[0, 0, 0]$ , (5.8) can express the *errors* in  $z$  and  $x$  since the equation and assumptions are linear. A position fix

is computed for every epoch (10 seconds in this case, after averaging one-second ranging measurements) where  $n \geq 4$  using (5.9). A least-squares *residual* statistic (a scalar) can then be computed from the magnitude of the projection of the measurement vector into *parity space*, which is defined by the null space of the original geometry matrix  $\mathbf{G}$ . An  $(n - 4) \times 1$  parity vector  $\mathbf{p}$  is computed from:

$$\mathbf{p} = \mathbf{P} \mathbf{z} = \mathbf{P} (\mathbf{v} + \mathbf{b}) \quad (5.12)$$

where the measurement  $\mathbf{z}$  is given by (5.8) and the  $(n-4) \times n$  matrix  $\mathbf{P}$  spans the parity space; i.e., it has  $\text{rank}\{\mathbf{P}\} = n-4$  and satisfies both  $\mathbf{P}\mathbf{P}^T = \mathbf{I}_n$  ( $\mathbf{I}_n$  is an  $n \times n$  identity matrix) and  $\mathbf{P}\mathbf{G} = \mathbf{0}$ , where  $\mathbf{0}$  is a matrix of all zeros. The  $n \times 1$  vector  $\mathbf{b}$  on the right-hand side of (5.12) expresses any *bias errors* that may exist on the measurements in addition to the “ordinary” Gaussian noise contained in  $\mathbf{v}$ . If one or more satellites have bias errors in their measured pseudoranges, the corresponding entries in  $\mathbf{b}$  will contain these (unknown to the user) errors, otherwise the entry will be zero. If  $\mathbf{b} = \mathbf{0}$ , the elements of  $\mathbf{p}$  will be uncorrelated with equal variance  $\sigma_n^2$  from the distribution of  $\mathbf{v}$  [7-10].

In order to check the consistency of each of the pseudorange measurements within the overall position solution, it is desirable to project state (position) and parity space vectors back into measurement space. This results in the  $n \times 1$  *fault vector*  $\mathbf{f}$  defined by:

$$\mathbf{f} = \mathbf{S} \mathbf{z} = \mathbf{P}^T \mathbf{P} \mathbf{z} \quad (5.13)$$

where  $\mathbf{S}$  is  $n \times n$  and has  $\text{rank}\{\mathbf{S}\} = n-4$  and  $\mathbf{S}^2 = \mathbf{S}$  (i.e.,  $\mathbf{S}$  is *idempotent*). The fault vector thus has a mean  $E[\mathbf{f}] = \mathbf{S}\mathbf{b}$  (which is zero under fault-free conditions) and covariance  $\text{Cov}[\mathbf{f}] = \mathbf{S}\sigma_n^2$  if the bias vector components are i.i.d. with a mean of zero (*note*: this is a strong assumption). The matrix  $\mathbf{S}$  can be computed directly by [7-10]:

$$\mathbf{S} = \mathbf{I}_n - \mathbf{G}\mathbf{G}^* \quad (5.14)$$

The RAIM residual decision statistic is then defined to be the 2-norm of the fault vector:

$$D = \|\mathbf{f}\|_2 = \mathbf{z}^T \mathbf{S} \mathbf{z} = \mathbf{z}^T [\mathbf{I}_n - \mathbf{G} \mathbf{G}^*] \mathbf{z} \quad (5.15)$$

This decision variable  $D$  equals the *magnitude* of the parity vector projected into measurement space. In other words, it represents the size of the inconsistency of the position solution from (5.9), which is what the parity vector expresses. Note that it is mathematically equivalent to the “sum of squared residuals” definition used in other papers on RAIM. Given this representation and the assumptions made above,  $D$  represents a sum-of-squares of Normal random variables (possibly plus a constant non-zero bias vector); thus it is assumed to be *chi-square* distributed. These distributions and their relation to false-alarm and missed-detection probabilities are given in the next section.

### 5.2.2 Choosing Thresholds for RAIM Statistics

Normally,  $D$  is computed using (5.15) and compared to a predetermined threshold  $T$ . If  $D > T$ , an integrity *alarm* is issued. Otherwise, normal (no-fault) operation is assumed. As mentioned in Section 5.1.4.2, cases of “bad” GPS satellite geometry are designated as *non-available* RAIM cases, i.e. no failure determination can be issued. A separate geometry statistic (DOP or a variant of DOP) and threshold are also chosen to make this determination.

Generally, traditional RAIM algorithms choose  $T$  either by Monte Carlo sampling [7-8] or by the *chi-square* probability distribution [7-8,10,11] to give desired *false alarm* (FA) and *missed detection* (MD) probabilities:

$$P_{FA} = \Pr[D > T | \delta x < RPE] \quad (5.16)$$

$$P_{MD} = \Pr[D < T | \delta x > RPE] \quad (5.17)$$

where  $\delta x$  is the true (unknown) position error and RPE is the allowed position error limit. Various sources of maximum allowed FA and MD probabilities are issued based on the application (see [6-11,6-18,8-13]). Non-availability thresholds are chosen based on

geometry dilution of precision parameters (such as HDOP and PDOP) which relate the noise standard deviation  $\sigma_n$  to the position error standard deviation. These limits may correspond to a specification on  $P_{MD}$  when the threshold  $T$  is chosen to meet a false alarm requirement. More complicated availability criteria are described in [7-2], such as those which utilize  $\delta H_{max}$  as described below.

Because the residual statistic  $D$  is a quadratic function of  $\mathbf{z}$ , which is itself a linear function of the normal random vector  $\mathbf{v}$  in (5.8), the false alarm probability (which assumes fault-free operation) is given by the chi-square distribution:

$$P_{FA} = Q\left(T^2/\sigma_n^2 \mid n-4\right). \quad (5.18)$$

where the operator  $Q$  represents one minus the incomplete gamma function  $P$  [3-12]. In traditional RAIM, only *bias* failures on a single satellite are considered (i.e.,  $\mathbf{b}$  can have no more than one non-zero entry). The missed detection probability is then bounded by the *non-central* chi-square distribution:

$$P_{MD} \leq P\left(T^2/\sigma_n^2 \mid n-4, RPE/[\sigma_n \delta H_{max}]\right). \quad (5.19)$$

in which  $\delta H_{max}$  is a geometry parameter defined to be the worst (largest) DOP parameter over the  $n$  sets of visible satellites each with one satellite dropped out [7-11]. This worst-case “subset DOP” measures the effect if the most critical (from a geometry point of view) satellite is the one which fails.

The CFAR algorithm in [7-11] sets a decision threshold  $T_n$  for each number of satellites ( $n$ ) in view from (5.18) to meet a (constant) requirement on  $P_{FA}$  (note that  $\sigma_n$  is considered to be constant). These thresholds would then be checked using (5.19) to determine whether the requirement on  $P_{MD}$  is also met. The more complex CPOD algorithm in [7-11] uses (5.19) to set a threshold as a function of  $n$  and  $\delta H_{max}$  to achieve a required  $P_{MD}$ . Note that this latter approach adjusts the threshold as a function of geometry; thus it has an implicit availability screen. If it is impossible to meet the PFA

requirement for a given  $n$  and/or  $\delta H_{\max}$ , no valid threshold can be set, and that geometry is declared unavailable.

### 5.2.3 Weaknesses of Traditional RAIM

In theory, these conventional RAIM approaches allow one to choose  $T$  to meet specifications on either  $P_{FA}$  or  $P_{MD}$  while checking to ensure that the requirement on the other is not exceeded. However, there are many assumptions embedded in the above equations. First, note that  $P_{FA}$  in (5.18) implicitly assumes that any bias fault (i.e., any case where  $\mathbf{b} \neq \mathbf{0}$ ) will lead to a position error exceeding the RPE. In practice, a specific “worst case” bias size (normally a bias on the worst-case satellite that leads to just exceeding RPE) is assumed during threshold selection. Even so, for systems with high accuracy margin relative to the protection limit (i.e.,  $RPE \gg \sigma_n$ ), a focus on worst-case bias will tend to “tighten up” the thresholds such that many biases are detected which do not cause the position error to exceed RPE. This leads to a higher false-alarm rate than is indicated by (5.18).

The assumption of no multiple-failure cases is also directly built in to the standard RAIM algorithm. This seems reasonable because, since the elements of  $\mathbf{b}$  (and the underlying failures on different satellites) are assumed to be independent, the likelihood of two or more separate failures occurring at the same time is remote. This reasoning is questionable on at least two counts. First, failures that cause position error to exceed RPE should by definition be very rare events, much rarer than failures of spacecraft components (where this assumption is often made, see Section 3.2). In this case, neglecting multiple failures may be frequent enough to affect user integrity on the one-in-a-million or rarer level. Second, as just noted, high-accuracy-margin systems will not often exceed RPE due to a single spacecraft bias. Errors substantial enough to threaten the protection limit may instead come from an unusual underlying failure that manifests itself in more than one satellite pseudorange. One example would be an OCS (or differential reference station) error that (before being detected) mangles the ephemeris corrections for more than one satellite in a non-evident way. In short, the independence assumption is dangerous for

SPS and should be investigated more carefully. This is even more true of DGPS applications because atmospheric errors will be correlated across multiple satellite pseudoranges. This will be discussed further in Chapter 6.

The common thread to the above flaws is that conventional RAIM assumes no prior knowledge of satellite, receiver, and ground station failure modes. Instead, failure mode and effect models are represented in a very simplified way, and failure uncertainty is effectively paid for by becoming very risk-averse and setting parameters for supposed worst-case events. In addition to possibly being inefficient in cost/benefit terms, the fact that only single failures are considered can bias the “worst-case” analysis in the wrong direction; i.e., the worst-case may actually involve multiple failures. The probability models introduced in the next section address this point.

Finally, the true utility of the RAIM algorithm should be a function of more than just  $P_{FA}$  and worst-case  $P_{MD}$  as defined here. The cost model to be introduced in the next section addresses all of the possible RAIM outcomes as functions of the resulting fatal accident risk, thus providing a more general context within which to select the best possible decision thresholds along the lines of Chapter 1. Using this approach and extensive simulation of a GPS uncertainty model, the approximations on which conventional RAIM is based can be carefully examined.

### **5.3 A New RAIM Approach Using Cost-Based Optimization**

This section presents the results of a RAIM study in which the traditional assumptions are set aside. These results are based on a simulation of the GPS constellation geometry along with distributions which model user uncertainty regarding the probability of failures in the satellites and ground equipment. The outputs of the simulation are observed probabilities of errors given knowledge of the current geometry. Thus, given a cost function which models the negative utility of RAIM decision errors for a given application, we can not only check the theoretical error predictions assumed by current RAIM methods; we can also directly choose an optimal threshold for each geometry case.

### 5.3.1 GPS System Faults: Prior Probability Model (PPM)

The assumptions made by traditional RAIM not only contain simplifications; they do not consider any *prior probability information* that might influence  $P_{FA}$  and  $P_{MD}$ . They essentially consider that there is one possible spacecraft failure mode of unknown but small probability. While it is true that only limited data on navigation satellite and receiver integrity has been made public, it is sufficient to at least attempt to express a *Prior Probability Model* (PPM) and examine its usefulness in computing optimal RAIM thresholds. This PPM would express our uncertainty regarding whether or not system failures are present *before* any measurements are conducted. Note that this PPM is designed to satisfy the requirements for decision making under uncertainty as promulgated in Section 1.2; thus decisions derived from it will have the property of being optimal under the chosen objective function and “rational thought” axioms from Appendix A.

In an attempt to predict the likelihood of GPS position fix availability for various classes of users, two studies have been conducted from which our PPM can be drawn. The more general study is [6-4], which is based on failure models drawn from previous space-based and inertial navigation systems. A spacecraft failure/renewal model was assumed, and ten separate parameter sets were selected as representative of the uncertainty of the GPS constellation. The satellite model in [6-15] uses the “most likely” set of failure parameters from [6-4] and updates them with later constellation information and the possibility of geosynchronous augmentation satellites. The probability distribution for numbers of healthy satellites presented in [6-15] is now used as a standard for simulations of satellite availability.

Using the model in [6-4], we divide failures into “hard” and “soft” cases, “hard” meaning complete spacecraft shutdowns and “soft” meaning spacecraft operating with GPS signal errors. These “hard” and “soft” classifications more or less match the “long-term” and “short-term” distinctions (respectively) used in [6-4] and use the corresponding MTBF and MTTR values. Rather than using ten discrete failure parameter sets, our PPM uses continuous  $\text{Gamma}(\alpha, \beta)$  and  $\text{Normal}(\mu, \sigma)$  distributions to model the uncertainty in the failure parameters based on the data in [6-4]. For spacecraft failures, the key parameters in a simple renewal model are the mean-time-between-failures (MTBF) and the

mean-time-to-repair (MTTR). These distributions and the relevant parameter uncertainty models are shown in Table 5.2.

In the case of a hard failure, the satellite either stops transmitting or has been flagged by the OCS as unhealthy. In the soft failure case, the satellite continues broadcasting an erroneous message (i.e., it does not meet the SPS specifications [6-3]) and does not initially warn the user of that fact. Given a soft failure, a distribution is given of the size of the resulting pseudorange error (it is a “bias” because we are sampling a snapshot in time). It is assumed that the OCS is more likely to eventually “flag” a soft failure (thereby warning the user) if the resulting error bias is large (see Section 5.2.3).

In addition, models for GPS receiver soft failures are given. These represent errors in the receiver’s internal processing of one or more healthy received signals. If a such a failure exists, it is assumed to be on a single channel with 50% probability, inducing an apparent bias on the relevant spacecraft. Otherwise, it is assumed to affect all channels, thereby increasing the effective receiver noise by a randomly-generated multiplicative factor (see Table 5.2). It is still assumed that most receiver faults will be hard failures which make GPS position fixes impossible. From these models of our failure parameter uncertainty, we can generate probabilities of system failures for each trial and then apply them to the GPS constellation.

Parameter	Distribution	mean ( $\mu$ )	standard dev. ( $\sigma$ )
SV hard MTBF	Gamma	100 mo.	12 mo.
SV hard MTTR	Gamma	1.5 mo.	0.6 mo.
SV soft MTBF	Gamma	16 mo.	6 mo.
SV soft MTTR	Normal	28 hr.	8 hr.
SV soft PR bias	Gamma	300 m	240 m
SV soft failure flag	Uniform	400 m	230.9 m
RCVR failure prob.	Normal	0.0015	0.0025
RCVR channel bias	Gamma	30 m	24 m
RCVR noise multiple	Normal	8.0	2.5

*Note:* Normal distribution outcomes  $< 0$  are taken to be 0.

**Table 5.2: Prior Probability Model Parameters**

Modeling the effects of system failures is very difficult because of the many possible failure modes that can exist. There is little information to go by; thus the Gamma distributions in Table 5.2 which model the likelihood of bias magnitude have a large variance. This represents considerable prior uncertainty. No detailed attempt has been made to break down the causes of failures because there are so many possibilities. The models in Table 5.2 simply try to model our uncertainty regarding possible bias outputs as required by the decision-analysis paradigm of Chapter 1. Note that these failures are taken to be above and beyond the “normal-case” uncertainties modeled as noise in  $\sigma_n^2$  from Table 5.1.

### 5.3.2 *Simulation of GPS System under PPM*

The GPS performance consequences of the prior probability model outlined above cannot be calculated analytically; Monte Carlo simulation is the best way to perform the analysis without having to make severe limiting assumptions on the PPM. For this study, a simulation of the GPS primary 21+3 satellite constellation with Volpe almanac parameters (as described in Section 5.1.1), the PPM, and the results of RAIM-assisted position fixes was written in C. Each simulation run executes  $N_{\max}$  position fix trials. From one trial to the next, a clock is incremented by a random time interval that is uniformly distributed between 0 and 30 minutes. The GPS constellation position is then updated accordingly. Beyond this step, however, no continuity between trials is assumed; i.e. no correlation with time is included in the probability models. For each trial, new values of the PPM failure parameters are randomly generated from the distributions given in Table 5.2. Given the MTBF and MTTR for both hard and soft satellite failures for a given trial, the probability of being in a failure state at that instant is given by a simple application of renewal theory, where separate results are derived for hard and soft cases [6-4]:

$$P_F = \frac{MTTR}{MTBF + MTTR} \quad (5.20)$$

Each of the 24 satellites is then checked for failure against the resulting  $P_F$ 's. Those that suffer hard failures are simply removed from possible visibility for that trial (only). Those which suffer soft failures receive a randomly generated bias which is assumed to be static for the purposes of a single trial. Dynamic effects such as ramp errors are not specifically simulated, but the sampling from the bias distribution in Table 5.2 implicitly assumes some combination of possible error dynamics. This random bias is then compared to a uniform random number to determine if the bias has been "flagged" by OCS ground control. The higher the bias, the more likely it is to be flagged. Flagged soft failure effects are the same as those for hard failures; i.e. these satellites are not used if they are visible. Note that this means of simulating OCS flagging of short-term failures fits with (but is not limited to) a ramp-error time history, since a steadily-growing bias error will be more likely to be flagged as more "time" passes. The simulation simply picks out one moment in time and then simulates the error states in the context of how much time might have passed since the underlying failure occurred.

Once the status of all satellites is determined, the GPS geometry is resolved to determine the number of satellites visible to a receiver positioned at San Francisco (this position does not change) with a  $7.5^\circ$  mask angle. Using the PPM, it is possible that one or more of the apparently usable satellites has an unflagged bias error. The probability of receiver error is then generated, and the receiver error state (either single-channel error or an increase in the overall receiver noise variance) is sampled for this trial.

Next, white noise with variance  $\sigma_n^2$  is added to the output pseudorange of each satellite using the variances given in Table 5.1 (which assume a 10-second averaging time). GPS position is then computed and compared to the known user position. In addition to the position fix error magnitude  $\delta x$  from (5.9), the least-squares residual magnitude  $D = \delta r$  is computed as the RAIM decision test statistic from (5.15).

These results are stored in histogram arrays that contain the number of occurrences of the discretized index values in question. The application of SPS navigation that we will focus on in this chapter (see Section 5.3.3) is most concerned with horizontal (East-North) position error; so the geometry criterion uses horizontal DOP (HDOP). One bin equals 10 meters in  $\delta r$  and  $\delta x$  and 0.1 in HDOP. Each trial is recorded according to the number of

usable (non-flagged) satellites in view (called  $n_{\text{view}}$ ) and the HDOP of the visible spacecraft geometry. In addition to recording  $\delta r$  vs.  $\delta x$  within the prevailing  $n_{\text{view}}$  and HDOP, marginal distributions of  $\delta r$  vs.  $\delta x$  along with satellite failure state, pseudorange error, receiver error, and others are stored for later analysis.

### 5.3.3 *Non-Precision Approach using SPS*

The application of the GPS Standard Positioning Service considered in this chapter is known as non-precision approach. This mission involves navigating an aircraft in the horizontal (E,N) plane down to several hundred feet of altitude. Altitude navigation is assumed to be provided separately, so the limits on accuracy are not as burdensome. In fact, the 100-meter 95% horizontal error characteristic of SPS should be sufficient.

Overall PICA requirements for non-precision approach were published by the RTCA Special Committee 159 (SC-159) in [6-18]. In [7-11], these requirements are converted into  $P_{\text{FA}} \leq 1.4 \times 10^{-5}$  and  $P_{\text{MD}} \leq 10^{-4}$  at every 10-second RAIM decision epoch. Also, a required protection error of 550 meters is cited, which is 17 times the base pseudorange standard deviation  $\sigma_n$ . Availability requirements are not specifically called out. In [7-11], the assumption is that RAIM is unavailable for geometries in which a threshold that meets both of the above requirements cannot be found. When all 24 GPS satellites are healthy, it calculates RAIM availability for non-precision approach as 98.9%. Heretofore, specifications of this sort have been the only guidelines for RAIM threshold calculation and performance analysis.

### 5.3.4 *Risk-Based RAIM Cost Model*

Instead of just taking the published requirements as a given, it is important to ask where these specifications come from and what is the underlying philosophy. Clearly, requirements should be based on the “cost” to the user of a false alarm, missed detection, or non-availability result. These costs may not be uniform for all users. Furthermore, as pointed out in [7-4], absolute specifications cannot be demonstrated to 100% confidence in any case. Since uncertainty is unavoidable, it may be better to use a cost-based method that more easily models uncertainty and risk aversion.

A basis for both the RTCA specifications and a cost-based model can be found in the *RNP Tunnel Concept* for precision aircraft approaches and landings [6-10]. It describes a means of setting position error specifications along an approach “tunnel”, and it estimates the consequences of straying outside the tunnel based on the historical record of commercial aviation accidents. For a precision approach, integrity decisions must be made close to the ground. A false alarm (causing an aborted landing) has a small but significant probability of leading to a fatal accident, while the more serious missed detection has a much higher chance of leading to a fatal accident. Since non-availability decisions are made before the final approach begins (based on satellite geometry), there is minimal risk involved, but there is still a small inconvenience cost that must be weighted against the fatal accident risk.

The non-precision approach case studied here does not face the same exacting requirements, so the relative costs are harder to specify. The RPE in this case is much looser, however; violating it still poses considerable risk if not detected. Any detected violation (known as a “bad” position result) requires an alternative approach method, so there is an inconvenience cost plus some minimal risk. A false alarm should have the same cost as a “bad” position result except that the availability of the GPS system is wasted. Thus, the false alarm cost is equal to the “bad” cost plus the small non-availability (NA) inconvenience cost. Although many different cases have been tested, a generalized set for SPS which is used for the results shown here is given in Table 5.3. Note the variable cost of the MD event means that in addition to the base cost of any MD, an additional cost is paid for each bin (10 m) beyond the RPE limit that the missed position error is located.

RAIM Result	Base Cost	Variable Cost
good position	0	0
detected bad pos. ( $C_{\text{bad}}$ )	1	0
missed detect. ( $C_{\text{MD}}$ )	200	5 (per 10 m)
false alarm ( $C_{\text{FA}}$ )	1.01	0
non-available ( $C_{\text{NA}}$ )	0.025	0

**Table 5.3: SPS RAIM Cost Parameters**

The RNP identifies relative fatal accident risks most clearly for Category III precision landing, where the aircraft is guided all the way to the touchdown point. It estimates the probability of a fatal accident given that a missed detection occurs (i.e., the aircraft continues the approach while not being warned of a hazardous navigation error) as 10-25% for non-Category III approaches (100% is used for Category III). The risk of a fatal accident during a missed approach caused by a RAIM-derived abort (a *continuity* breach - either a detected bad position or a false alarm) is estimated to be 0.001. Most likely, the same risks would not apply to an abort or a violation of the RPE during non-precision approach, but there is still substantial risk to the affected aircraft and to aircraft in its vicinity. The cost model in Table 5.3 thus is assumed to be approximately the same for all aircraft approach applications, as the *relative* risks of continuity and integrity breaches are thought to be similar. What varies, then, is an (uncertain) constant parameter  $R_C$  which is defined to be the ratio of actual accident risk to RAIM cost from Table 5.3. For Category III,  $R_C \cong 0.001$ , meaning that an actual fatal accident has an implied RAIM cost of  $1 / R_C \cong 1000$ . For non-precision approach, on the other hand, the fatal-accident risk is taken to be 20 times lower; thus  $R_C \cong 5 \times 10^{-5}$ . This translation of RAIM cost to bottom-line risk establishes a means of tracing navigation and RAIM algorithm design to top-level system goals and constraints; thus it can serve to place existing requirements within a larger context.

The NA cost for non-precision approach is perhaps the most important to specify. Unlike the other costs, which trace back to accident risk, this cost derives from the economic inconvenience of not having the service available at a specific time it is needed. One way to derive this cost into dollar terms is to estimate the average additional flying time required in a non-available case and multiply that by the direct operating cost and fuel cost for an aircraft over that time. From a recent ICAO study [6-8], an average cost per hour for commercial aircraft is about \$4800. Assuming a half-hour average delay, the per-occurrence NA cost becomes \$2400.

To translate this economic cost into the risk domain of the RAIM cost model, we refer back to Chapter 1, where the equivalence of fatality risk to dollars was justified for sufficiently small risks (i.e., less than 0.0001). A conservative fatality-equivalent cost of

\$10 million per life was chosen there. Based on the weighted aircraft usage parameters in [6-8], a fatal accident is assumed to result in a (conservative) average of 200 deaths (which may come from more than one plane if a mid-air collision results); thus the cost per fatal accident is about \$2 billion. For non-precision approach, where  $R_C$  is conservatively estimated to be  $5 \times 10^{-5}$ , the comparative cost of a single non-availability event would be:

$$C_{NA}^{np\ app} = \frac{\text{NA cost (\$)}}{R_C [\text{fatal acc. cost (\$)}]} = \frac{\$2400}{(5 \times 10^{-5})(\$2 \times 10^9)} = 0.024$$

Since this estimate is based on several assumptions regarding the parameters in (5.21) below, a range of non-availability costs from 0.01 to 0.025 has been tested in the RAIM optimization described in Section 5.4.

### 5.3.5 Cost-Based Threshold Optimization

Since each combination of  $n_{\text{view}}$  and HDOP for non-precision approach represents an independent decision case, an optimal residual threshold can be chosen separately for each one. Recall that the simulation described above stores  $\delta r$  as a function of  $\delta x$  for each discretized geometry case. Once all trials are completed, threshold optimization is conducted simply by computing the cost of all possible discrete thresholds  $T = \delta r_{\text{limit}}$  using the following equation:

$$J_{\text{RAIM}}(T) = P_{\text{FA}} C_{\text{FA}} + P_{\text{MD}}(\delta_e) C_{\text{MD}}(\delta_e) + P_{\text{bad}} C_{\text{bad}} \quad (5.21)$$

where  $\delta_e$  is the amount by which RPE is exceeded. The cost parameters  $C_{\text{bad}}$ ,  $C_{\text{MD}}$ , and  $C_{\text{FA}}$  are defined in Table 5.3, and cost numbers are given there for each of these categories. The threshold that gives the lowest  $J_{\text{RAIM}}(T)$  is the optimal choice.

At this point, the expected cost of the optimal threshold is compared to the GPS non-availability cost. If the optimal threshold gives a lower cost, then GPS is available for that case and the optimal RAIM threshold is set. Otherwise, that case of  $n_{\text{view}}$  and

HDOP is designated as non-available for integrity, and GPS is assumed to be unusable. Note that the NA cost is thus a critical number, as it measures the user's *risk aversion* regarding the reliability of RAIM decisions. Higher NA costs, representing greater degrees of inconvenience, result in higher availability probabilities, as the optimization algorithm is willing to tolerate more risk in choosing the best possible threshold. Finally, the optimal thresholds (or zero if non-available) are output in a lookup table of  $n_{\text{view}}$  vs. HDOP which can be directly fed into aircraft avionics. Overall results for all of the cases, weighted by the likelihood of each, are also computed to measure the general utility of this RAIM methodology as well as the marginal probabilities of FA, MD, and NA outcomes.

Figure 5.2 illustrates the selection of the optimal threshold for a given geometry case. The 2-D plots show the simulated and stored  $\delta x$  vs.  $\delta r$  for each geometry [ $n_{\text{view}}$ , HDOP], where each "x" represents a  $[\delta r, \delta x]$  histogram bin. Taking the RPE (the boldface horizontal line  $\delta x = \text{RPE}$ ) as fixed, a RAIM cost  $J(T)$  is computed for a given  $T$  (the boldface vertical line) using (5.21) as shown in the upper left quadrant of the figure.

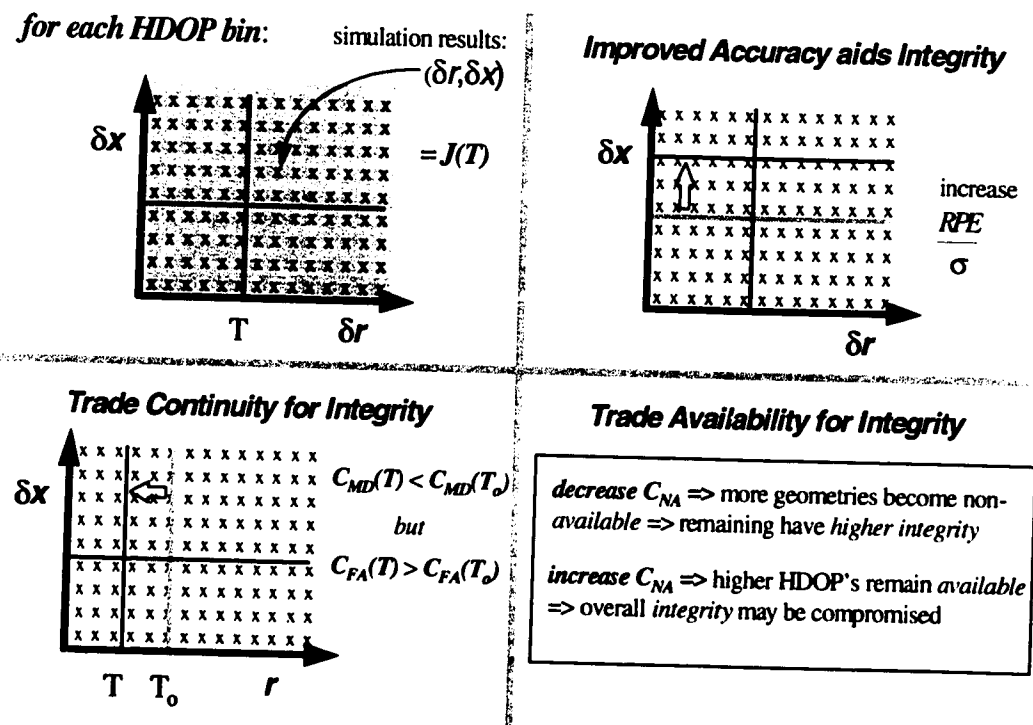


Figure 5.2: Optimal Threshold Selection Process

Even if RPE is fixed, improving the system accuracy (i.e., reducing  $\sigma_n$ ) effectively improves  $J(T)$  for all  $T$  by shifting the RPE line (normalized by  $\sigma_n$ ) upward as shown in the upper right quadrant. In the lower left quadrant, the trial threshold is reduced, moving the vertical line leftward. This shrinks the missed detection region ( $\delta x > \text{RPE}$  but  $\delta r < T$ , which is the upper left corner of the 2-D plot) but widens the false alarm region (the lower right corner of the plot where  $\delta x < \text{RPE}$  but  $\delta r > T$ ). The RAIM cost model in Table 5.3 effectively determines the optimal tradeoff between these two undesirable outcomes for a given  $[\delta r, \delta x]$  simulation result. Thus, the bold vertical line is moved from the far left to the far right of the residual statistic domain until the location that gives the lowest RAIM cost  $J(T^*)$  is found. That  $T^*$  is the best possible choice for the situation in question. Also note that the lower right corner expressed the availability vs. risk tradeoff discussed above.

#### 5.4 Simulation and Optimal Threshold Results

All the results shown in this section use the parameters given in Tables 5.2 and 5.3 and a sample size of  $N_{\max} = 10$  million trials. Simulations have been run with variations of the Prior Probability Model and cost parameters; thus we can draw some conclusions regarding solution sensitivity to parameter changes while we study the nominal results.

##### 5.4.1 Spacecraft and Receiver Reliability

The simulation outputs contain important marginal output probabilities that deserve study so as to illustrate the effects of the prior probability model with regard to spacecraft and receiver health. Table 5.4 shows the observed probability of these failure categories. Note that although flagged soft failures have the same effect as hard failures, they are probabilistically separate events in the simulation results from Section 5.2.3.

Observed Event	Probability
no SV failure	0.98237
SV hard failure	0.01492
SV soft failure-unflagged	0.00173
SV soft failure-flagged	0.00098
RCVR failure (separate case)	0.00191

**Table 5.4: Observed Failure Probabilities**

These results contain few surprises. Most spacecraft failures are "hard"; they only degrade GPS geometry. The 0.014 probability of hard failure (or "unhealthy" status) for a given satellite agrees with the numbers implied by the results in [Phlong]. The issue of flagged soft failures could use further study, as the PPM distributions from Table 5.2 result in a 64% chance of a given soft

failure being unflagged. With the information currently available, it is hard to say how accurate this result is. Although receiver failures are not too uncommon, their effects from the PPM are usually not very significant (see Table 5.2).

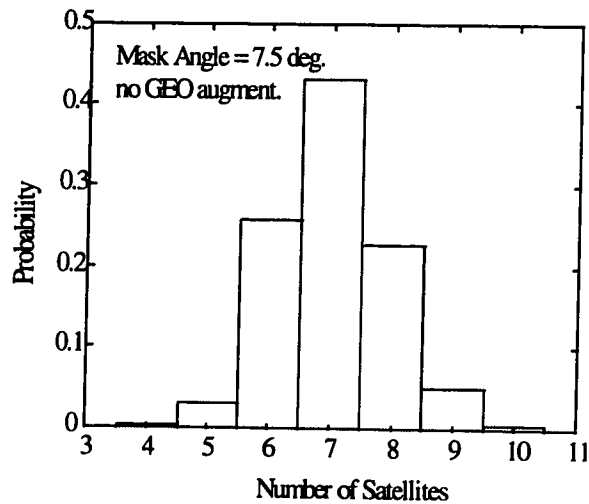


Figure 5.3: Distribution of Satellites in View

#### 5.4.2 GPS Geometry Results

Figure 5.3 gives a bar-chart histogram of the number of healthy satellites visible to the San Francisco user over the set of simulations. Using only the basic 24-satellite NAVSTAR constellation and a 7.5° mask angle, the user can see six or more satellites 97% of the time. The six-to-eight satellite range occupies 91% of the cases, and nine or more satellites are in view about 6% of the time. This level of satellite availability gives a reasonable basis for RAIM, but note that the 0.5% of the cases where four or fewer satellites are visible places an upper

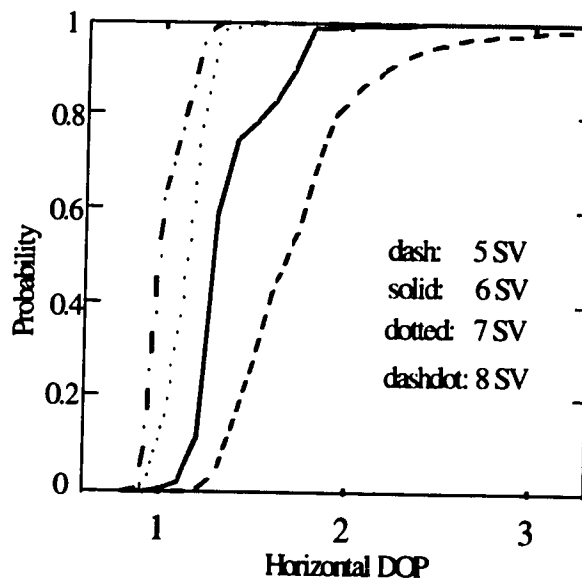


Figure 5.4: HDOP Cumulative Probability

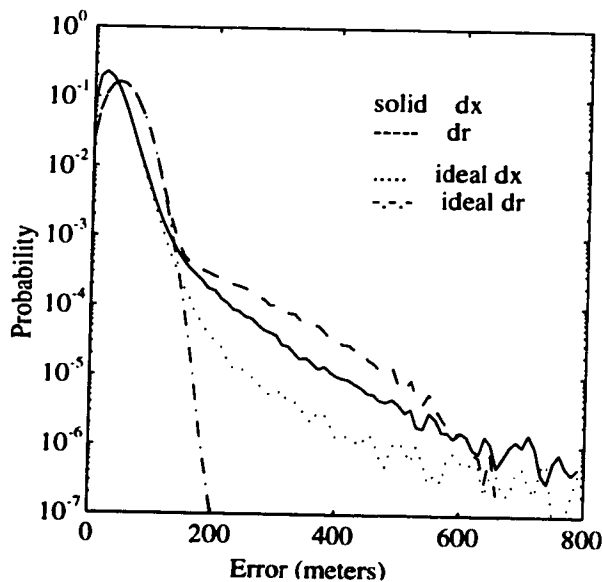
bound of 99.5% on RAIM availability, since at least five satellites are needed to provide redundancy in the position solution.

Figure 5.4 provides separate curves of cumulative probability for HDOP for 5, 6, 7, and 8 satellites in view. The probability on the y-axis is the likelihood that the actual HDOP is *lower* than that shown on the x-axis. Thus, for five satellites in view, HDOP above 2.5 occurs about 5% of the time, whereas for six or more satellites in view, it does not happen at all (at least within the 10 million samples done here). This suggests that many 5-satellite cases will end up being unavailable for RAIM due to high HDOP, while almost all cases with 6 or more satellites in view will be feasible. Note that HDOP limits for RAIM availability will vary with numbers of satellites in view.

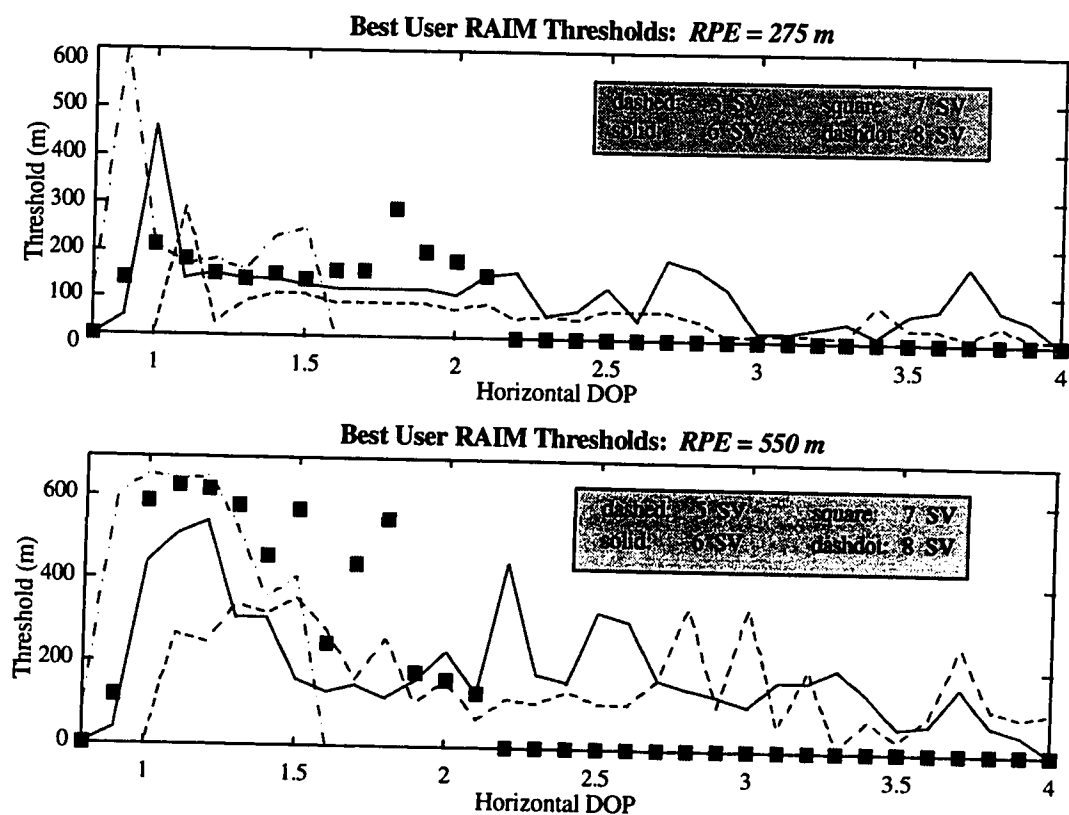
#### 5.4.3 Optimal RAIM Results for SPS Users

Figure 5.5 shows the marginal probabilities of  $\delta r$  and  $\delta x$  over all geometry cases. The dotted and dashdot "theoretical" probability curves are generated from an "ideal" simulation (with no SV soft failures) and are included for comparison. Agreement is reasonably good for the low error, high probability cases, but above 150 meters, the ideal plots dive well below what is observed using the PPM, as the "soft" ranging error models significantly extend the tail of the position error distribution. Disagreement between the ideal and expected residual statistic curves is even greater. The small oscillations in the lower right-hand corner are due to limited sample size for the lowest probabilities.

Figure 5.6 shows two plots of the best RAIM thresholds found as functions of HDOP, where the four different curves break the best



**Figure 5.5: Ranging and Position Error Densities**



**Figure 5.6: Best RAIM Threshold Results**

satellites in view. These thresholds are simply the magnitude of the residual statistic from (5.15) above which a failure warning is issued. The two plots shown are for cases which differ only in the assumed RPE. The top graph is for an RPE that was halved from the 550-meter one specified for non-precision approach and shown in the bottom plot. In both cases, the best thresholds tend to increase with the number of satellites in view and fall with HDOP. This is the expected result because, as the GPS geometry worsens, the likelihood of severe position errors increases, producing a greater threat to user integrity. In these cases, the allowable threshold on the RAIM statistic will be lower than cases where the GPS geometry is better and thus the “prior” probability of extreme position errors is much lower.

Normally, we would expect the change in the optimal threshold to vary continuously with HDOP, but this is clearly not the case in Figure 5.6, as there is a lot of “noisy” wandering of the best threshold within the clearly-visible larger trend. This is

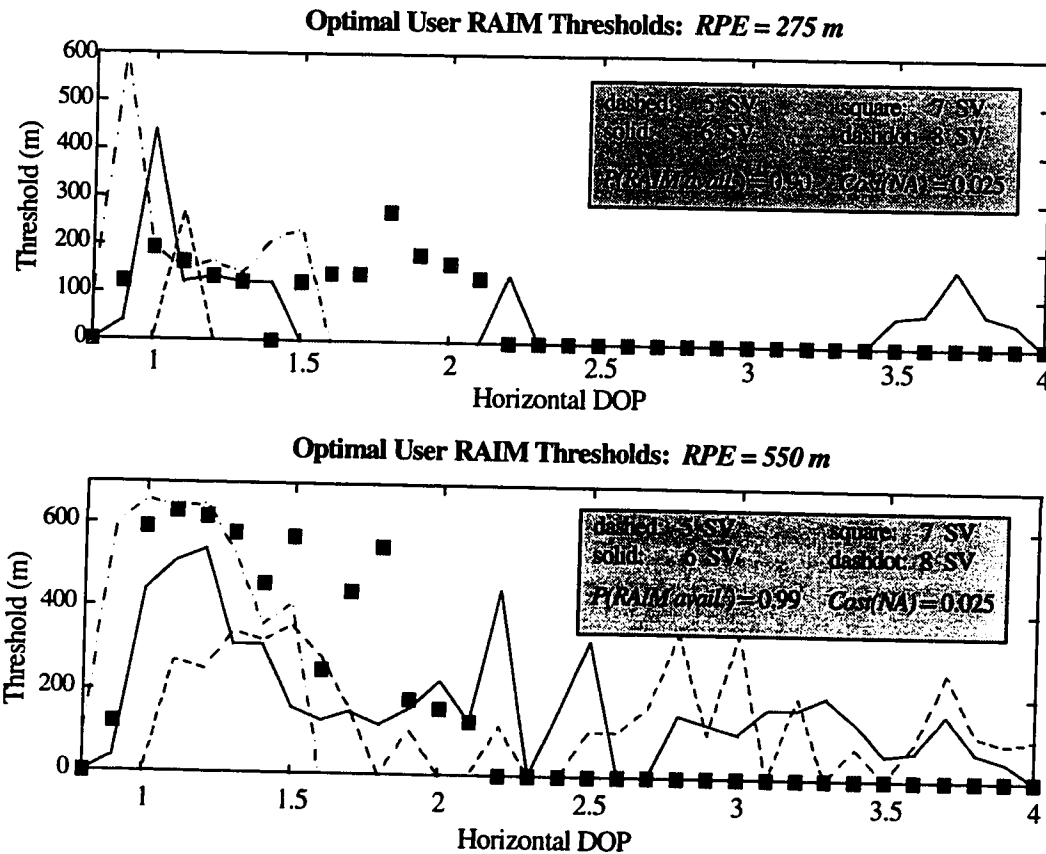


Figure 5.7: Optimal RAIM Threshold Results

especially visible for the 6 and 7 satellite-in-view cases. By discretizing the GPS geometry space by both number of visible satellites and HDOP (divided into bins of width 0.1), relatively few failure events (position errors exceeding RPE) occur in each case; thus the results will be inconsistent even for similar geometries. The more events recorded in each separate geometry case, the more significant the best threshold result. This is visible in the upper plot for the 275-meter RPE case, which is considerably less jumpy because there are many more position errors that exceed this lower protection limit.

Figure 5.7 shows the *optimal* thresholds as a function of GPS geometry for both RPE cases. The only difference between this figure and Figure 5.6 is that the availability determination has been made for each geometry, and cases where the cost of the best threshold is higher than the non-availability cost are declared *unavailable* and shown with an optimal threshold of zero. From looking at these plots, one can determine HDOP limits

Number SV in View	HDOP limit (% available) <i>RPE = 275 m</i>	HDOP limit (% available) <i>RPE = 550 m</i>
5	1.3 (17%)	2.0 (78%)
6	1.5 (84%)	2.6 (99.5%)
7	2.2* (100%)	2.2* (100%)
8	1.6* (100%)	1.6* (100%)

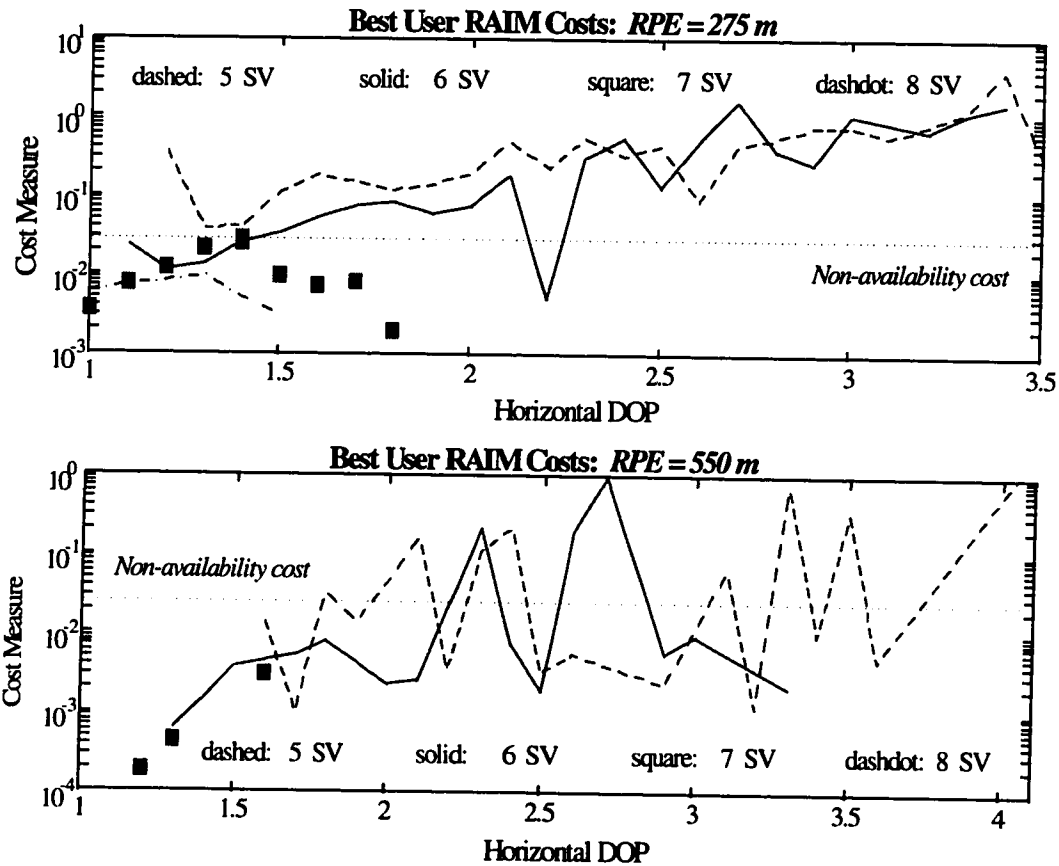
\*: represents maximum HDOP from any simulation trial

**Table 5.5: HDOP Availability Thresholds**

above which RAIM is unavailable for each  $n_{\text{view}}$ . The input non-availability costs ( $C_{\text{NA}}$ ) and output availabilities are indicated in the legend.

Table 5.5 has the HDOP availability thresholds and resulting availabilities as a function of number of satellites in view for both RPE cases. Note that for 7-8 satellites in view, all HDOP's are available; thus the availability limit is simply the largest HDOP result for that case. The limits for five or six satellites in view do eliminate some geometries, but their trend in Table 5.5 makes sense; i.e., more spacecraft in view give a higher HDOP limit even as they provide a lower HDOP probability distribution. As a result, availability improves markedly when at least six satellites are in view. However, the noise in the RAIM cost evaluations causes jagged availability results as well. For example, it is not clear where the right limit is for six satellites in view in the bottom plot, as it shows HDOP = 2.3 non-available, then HDOP = 2.4-2.7 are OK, then HDOP > 2.8 all seem available. The listed cutoff of 2.6 in Table 5.5 is an estimate, as the HDOP bins above that are considered unreliable because they contain few samples and thus have limited statistical significance.

Figure 5.8 illustrates the source of the randomness in the optimal threshold results by plotting the RAIM cost of the best threshold found for each geometry. The y-axis shows the RAIM cost in logarithmic scale, and the non-availability cost  $C_{\text{NA}} = 0.025$  is shown as a horizontal dotted line to indicate which geometries beat the availability cutoff. The upper plot for RPE = 275 m is relatively well-behaved, as costs for 5 and 6 satellites in view increase with only a few jumps as HDOP is increased. The curves for 7 and 8 satellites in view have costs which drop off as HDOP increases simply because few



**Figure 5.8: Best RAIM Cost Results**

samples exist for the higher HDOP's in these cases. The lower plot for  $RPE = 550$  m shows a great deal of local variation, as the best cost jumps above and below the non-availability cost at an excessive rate, forcing us to estimate the point at which the "true" RAIM cost curve crosses the non-availability cost line.

The best way to make use of the noisy simulation-based cost evaluations is to fit a mean curve to the raw result from Figure 5.8. In most of these plots, the underlying trend looks linear; so a linear fit to the results for each of the satellite-in-view cases gives a reasonable HDOP availability threshold for each one (see Table 5.5). However, substantial uncertainty about the validity of this approximation remains. For actual use, many more simulation trials should be conducted to gain further statistical significance.

#### 5.4.4 RAIM Optimal Result Summary

Table 5.6 gives the bottom-line RAIM results for this minimal-cost approach in terms of overall SPS user RAIM cost and the RAIM event parameters important to PICA evaluation (see Section 5.1.4). Looking at the PICA event probabilities, we see that they vary considerably from the results published for traditional RAIM [7-8,10] and the implied SPS RAIM requirements from Section 5.3.3 [6-18,7-11]. For example, the false alarm probability for the RPE = 550 m case is about twice that specified in the implied RTCA requirement. It is an order of magnitude worse for the RPE = 275 m case simply because it is much harder to reliably detect a position error reduced by 50% when the noise floor ( $\sigma_n$ ) is not improved. We see significant differences for the 275-meter case precisely *because* only RPE was changed. Bad position fixes are more probable, so the thresholds must "tighten up" to reduce  $P_{MD}$  at the cost of increasing  $P_{FA}$ .

The missed-detection results are particularly interesting. For the 550-meter case,  $P_{MD}$  is very high (28%), much higher than is thought to be tolerable by traditional RAIM. The reason is twofold. First, the RAIM cost model has a different weighting of missed detections to false alarms than does the implied requirement. Since the cost model is directly tied to user accident risk, it should be a better guide according to the principles of Section 1.2. Second, the missed detection probability derived from (5.19) is based on an arbitrary bias that should lead to a position error exceeding RPE, but the simulations of the PPM suggest that reality is much more complex.

Recall that the simulation approach in Section 5.3.2 samples failures from the PPM in Table 5.2 based on first sampling uncertain failure model parameters. While two or more serious failures at the same time are quite uncommon, they become more likely in trials where the sampled failure parameters turn out to be dangerously bad. The resulting

Simulation Output	RPE = 550 m	RPE = 275 m
Overall User RAIM Cost	0.00100	0.01216
<i>False Alarm</i> prob.	0.00040	0.00574
<i>Bad Position</i> prob.	$1.3 \times 10^{-5}$	0.000325
<i>MD   bad position</i> prob.	0.282	0.039
<i>Availability</i> prob.	0.990	0.904

**Table 5.6: RAIM Simulation Output Summary**

high failure rates are then applied to each visible satellite in turn, introducing significant correlation between their failure states. This indirect correlation invalidates the i.i.d. assumption of traditional RAIM (Section 5.2.1). Note that this same correlation effect was introduced for spacecraft reliability evaluation in Section 3.4.2.

As mentioned in Section 5.2.3, most single-channel bias errors do not result in a position error exceeding RPE. Single large-bias failures that do lead to errors exceeding RPE can be detected, but many threatening cases are multiple failures that are often less evident in the RAIM parity space. In other words, the missed detection probability derived from simulation results is not directly comparable to the traditional result of (5.19), since the latter does not represent a “true” missed detection in the sense of (5.17) for high  $RPE/\sigma_n$  ratios. Reducing the RPE by 50% gives a much lower missed detection probability (4%) because easier-to-detect single-bias failures become more threatening, but the price paid is a decline in availability (down to 90%).

The RAIM availability probability is quite good for both cases. It is much higher for  $RPE = 550$  meters because fewer position errors introduce unacceptable risk; thus it is safer to rely on RAIM over a wider range of GPS geometries. This fact is reflected in the most important number in the table: the overall RAIM user cost, which is an order of magnitude lower for the 550-meter case because far fewer events threaten user integrity. Note that the 275-meter results can be seen as a “controlled” case in which simulation-based optimization is easier and the results are more predictable. The 550-meter case corresponds more to the actual risk inherent in non-precision approach. For this application, the conversion factor  $R_C \equiv 5 \times 10^{-5}$  gives an estimate for the *fatal accident risk* per mission event to be (overall RAIM cost)  $R_C \equiv 5 \times 10^{-8}$ . This is the bottom-line acceptability factor as developed from the RNP; thus it demonstrates the relative acceptability of the RAIM approach without relying too heavily on arbitrary requirements which, as demonstrated by Table 5.6, will be difficult to meet given a realistic PPM and cost model.

#### 5.4.5 Results for Improved SPS Ranging Accuracy

The results in Table 5.6 hold for the original nominal SPS error assumptions made in Section 5.1.3. As noted there, they are now quite pessimistic because GPS user equipment has improved substantially [6-13]. Furthermore, studies of SPS user accuracy have concluded that Selective Availability (SA) over time does not degrade user accuracy as much as is suggested by the SPS specification [6-3]. Because SA is of limited value in denying the use of GPS to U.S. adversaries and because its presence retards the growth of civilian GPS applications, considerable debate has arisen over whether it should be terminated as a matter of U.S. government policy [6-12]. While the improvements in SPS user equipment are of limited consequence to integrity if SA remains, the possible removal of SA would significantly improve SPS accuracy and therefore improve user integrity significantly for the aircraft enroute and non-precision approach applications.

Table 5.7 shows the revised  $1\sigma$  SPS ranging errors for improved user equipment along with the amount of improvement obtained over the numbers in Table 5.1. Retaining the 10-second averaging of receiver noise, the resulting  $1\sigma$  rms ranging error would be 6.5 meters -- a dramatic 80% reduction from the SPS error model in Section 5.1.3. This error model variant can be applied to the RAIM threshold-optimization methodology to determine the risk reduction that would result (in terms of the RAIM cost function). As before, cases were run for both RPE = 550 m and 275 m.

The results for the improved noise model are a little surprising: *the only cases where horizontal position errors exceeded 275 m had only four satellites in view with HDOP > 15*. Position errors in these cases which exceeded 275 m occurred with

Error Source	One-Sigma Error (m)	Error Reduction
satellite clock/ephemeris	4	20%
ionosphere and troposphere	5	50%
receiver noise/multipath	4	73%
selective availability (SA)	N/A	<i>removed</i>

**Table 5.7: Revised Errors for Normal SPS Operation**

probability  $3.6 \times 10^{-7}$  (errors exceeding 550 m occurred with probability  $1.9 \times 10^{-7}$ ). By definition, these geometries would be screened out by the availability check. If all geometries with 5 or more satellites in view are declared available (the optimal solution in this case), the availability probability becomes 0.9997, which should meet any proposed requirement (and is much better than the RAIM-limited result in Table 5.6).

This result, based on the SPS PPM in Section 5.3.1, indicates that the probability of dangerous position error for 5 or more satellites in view is very likely below  $5 \times 10^{-8}$  (since none were found in 20 million trials). This is a major improvement over the result in Table 5.6. According to the optimal result, RAIM would be unnecessary, as all geometries that allow RAIM are safe for any reasonable failure. Not requiring RAIM has a further availability reward, as including four-satellite geometries where  $\text{HDOP} \leq 5$  (a conservative bound) would improve availability to a remarkable 0.999995. However, other considerations suggest that RAIM be maintained along with the consequent 5-satellite availability requirement. One is that the PPM is limited to simulating "snapshot" errors at a given point in time. According to the PPM, the probability of a *sudden* pseudorange error of many kilometers that would be needed to threaten user safety is exceedingly remote. However, it could result from an incipient error that grows to dangerous levels over time and is not flagged by the DoD Operational Control Segment before it becomes hazardous. Although this possibility is also remote, RAIM with the improved error model provides a nearly *ironclad guarantee* of detecting and single-channel and almost all dual-channel errors because the margin between nominal and dangerous error is so large (this is also the case with the Integrity Beacon Landing System developed at Stanford for Category III landings, see [7-9,15]). This protection could be obtained without significantly affecting availability and continuity because the detection

Simulation Output	RPE = 550 m	RPE = 275 m
Overall User RAIM Cost	0.00100	0.01216
<i>False Alarm</i> prob.	0.00040	0.00574
<i>Bad Position</i> prob.	$1.3 \times 10^{-5}$	0.000325
<i>MD   bad position</i> prob.	0.282	0.039
<i>Availability</i> prob.	0.990	0.904

**Table 5.8: RAIM Output Summary for no-SA Case**

thresholds could be set very loose. Further study of RAIM effectiveness for very large SPS pseudorange errors (based on a time-dependent PPM that simulates errors growing in time) is a key topic for further research.

This result, which demonstrates that the key to improved integrity for SPS users is simply to turn off SA, is important in the context of the current political debate on the future of SA, which is pursued in great detail in [6-1,12]. As of April 1996, a presidential directive has indicated that in accordance with the increased priority being put on the use of GPS for civilian applications, SA will be turned off within a decade. But within this time frame, the use of SPS for enroute and non-precision approach will expand dramatically, and the increased hazard due to the presence of SA (as indicated in Table 5.6) amounts to a mean of one serious accident in every 20 million approaches. With SA off, this risk drops by at least of factor of 100. The safety gain for unaugmented users of SPS amounts to a benefit in economic terms (based on the cost model in Section 5.3.4) of at least \$100 per flight. Projected over the millions of SPS approaches to be conducted over the next decade, this benefit of removing SA by itself runs into the hundreds of millions (if not billions) of dollars.

## **5.5 Comparisons with Traditional RAIM**

It is difficult to compare directly the performance of this PPM-based cost optimization method with that of traditional RAIM methods in [7-10,11] because their thresholds were set to meet arbitrary specifications. In order for a fair comparison to be made, the same user preference model must be used to set thresholds for both methods. This is very difficult for the case of traditional RAIM, as it is not set up to optimize any realistic cost model but instead focuses on black-and-white PICA requirements and evaluations.

Requirements such as those for SPS enroute and non-precision approach issued by RTCA SC-159 [6-18] are normally motivated by an implicit “target level of safety,” but they often do not attempt to model prior performance uncertainty when analytical studies suggest that an acceptable level of integrity can be provided by RAIM alone. Traditional

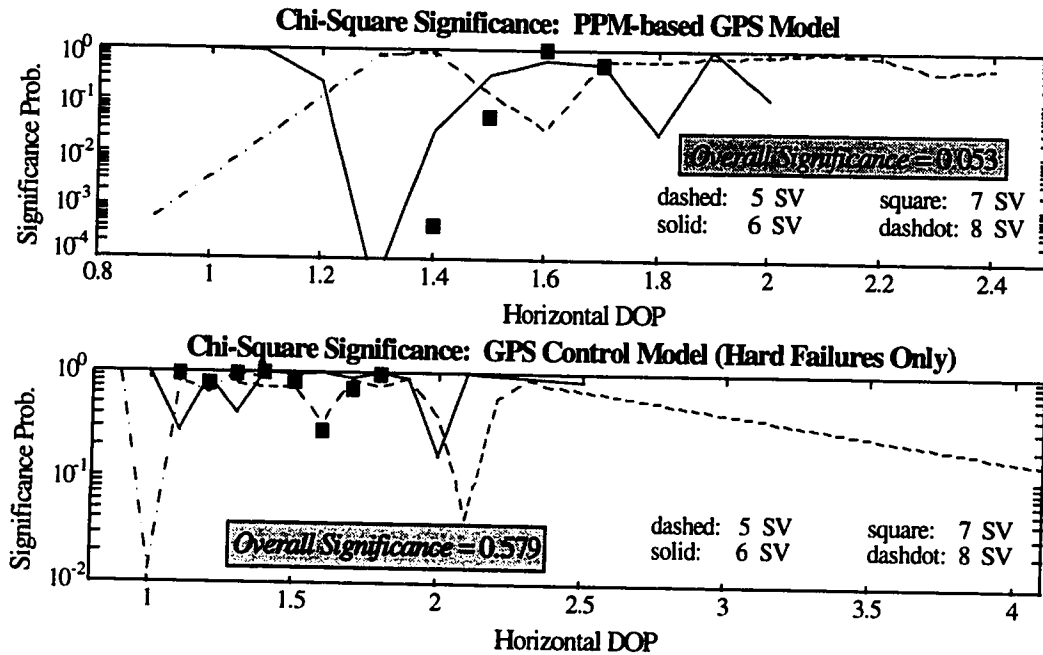
RAIM is shown in [7-8,10] to be capable of missed detection probabilities of 0.001 or below with reasonable availability. Once traditional RAIM assumptions are accepted, these results may influence the committees who write the requirements. This is despite the fact the assumptions outlined in Section 5.2 are questionable in practice. For example, it is unreasonable to assume that a position error just under the required limit is perfectly safe while one just over it is a dire safety threat. A user cost model along the lines of Section 5.3.4 not only represents risk more reasonably; it is also completely independent of what can or cannot be done using a specific technique.

Lacking a fair bottom-line performance comparison, another approach is to determine the accuracy of the chi-square model (5.18, 5.19) used in traditional RAIM in the context of the PPM used in our simulations. The RAIM method outlined here chooses optimal thresholds based on the simulation outputs instead of relying on (5.18, 5.19). It is also possible to compute the observed  $P_{FA}$  as a function of the detection threshold for each  $n_{view}$  and HDOP. Using (5.18), we can also compute the expected  $P_{FA}$  given any chosen threshold  $T$  and  $n_{view}$  (it is independent of HDOP) and then use a *chi-square statistical test* to determine the significance ( $\Sigma$ ) of the chi-square assumption; that is, the likelihood that the simulation output  $P_{FA}$  was produced by the theoretical distribution (5.18).

For a discrete dependent variable  $N$ , the chi-square statistic (distinct from the chi-square residual distribution of Section 5.2.2) is given by [3-12]:

$$\chi^2 = \sum_i \frac{(N_i - n_i)^2}{n_i} \quad (5.22)$$

where  $N_i$  is the number of events observed in bin  $i$  and  $n_i$  is the theoretical number of events from (5.18). The definition of “event” here is a false alarm, so in fact the inputs to (5.22) are the actual and predicted *numbers* of false alarms rather than the relevant probabilities. Note that  $N_i$  is an integer while  $n_i$  (derived from (5.18)) can be a fraction. The bin counter  $i$  represents all 80 possible threshold bins in  $\delta r$ ; thus an actual and predicted number of false alarms is computed for each threshold size, and a single chi-square statistic is then computed for each geometry. The *significance* of that statistic, or



**Figure 5.9: Chi-Square Comparison Results**

the likelihood that the simulated data is described by the theoretical distribution, is computed from the incomplete gamma function  $Q(\chi^2 | v)$ , where  $v$  is the number of degrees of freedom, which in this case equals 80, the number of threshold bins.

For comparison, a "control" simulation was run in which all *soft* failure models were removed (hard failures were retained since they simply reduce the number of available satellites). In this case, (5.18) should hold almost exactly, since the assumptions which underlie it (see Section 5.2.2) are all satisfied. In practice, however, bin discretization error and limited sample size prevent 90-100% statistical confidence from being achieved.

Figure 5.9 shows the levels of significance obtained for each geometry case, where the upper plot uses the PPM simulation results and the lower one uses the control simulation. Geometry is on the x-axis, while significance probability is plotted on log scale on the y-axis. Note that the significance scale on the upper plot is much lower than that of the control case, as the visible results for moderate HDOP give significances on the order of  $10^{-2}$ , whereas the control case significances are much greater. The random pattern of these results is important, as it suggests that noise in the underlying simulation will make

high significance unlikely even for the control case. It can also be observed from the test data itself that, as expected, agreement is better for lower thresholds (where many false alarms will result) than for the thresholds likely to be selected as optimal.

A summation of these results weighted by the number of position fixes in each bin is shown as *Overall Significance* on both plots in Figure 5.9. For the control case, the result is 0.579, meaning that it is quite probable that the false alarm results do come from the chi-square residual distribution of (5.18). For the PPM simulations, however, the result of 0.053 means that we can *reject* the hypothesis that the PPM simulation outputs are chi-square with 94.7% confidence. We must conclude that using what we believe to be a reasonable PPM, the chi-square distributions used in RAIM are likely to give somewhat incorrect results.

Just how incorrect the assumptions behind traditional RAIM are is still unclear, as it will vary with the application and amount of underlying system performance uncertainty. It should be noted that if (5.18) were completely inaccurate, the resulting significance would be zero. Thus, it is likely that traditional RAIM is not too far off the mark for most applications. Since it produces thresholds without a lengthy simulation-based optimization process, it can be used to set preliminary thresholds that can be substantially improved by the methods and cost functions derived in Section 5.3. A fair bottom-line cost comparison is needed before we can be more specific about the relative penalty incurred by traditional RAIM methods.

## 5.6 Bayesian Updating and GIC Potential

While the method used to compute the optimal thresholds shown in Figure 5.7 can be used as part of a "snapshot" RAIM algorithm (which uses only the current sample of pseudorange data), more benefit may eventually be gained by applying the PPM and the cost model to a multi-step updating algorithm. Fault Detection and Isolation (FDI) using multi-step filters is an established field of research (see [7-5,14]). Its use can make the integrity warning algorithm much less sensitive to aberrations in a single epoch, either by

simple averaging of pseudorange measurements or by using single-epoch measurements to update a “prior distribution: of failure uncertainty.

A Kalman filter algorithm for this role is derived in [7-7]. It relies on "censoring" out the estimated bias from a single source. This idea utilizes past sensor information and requires fewer assumptions than does traditional RAIM, but failures are still assumed to come one at a time, and random effects are presumed to be perfectly white and Gaussian once the bias estimate is removed. Another approach is implied by current plans for a GPS Integrity Channel (GIC), which uses the FAA Wide Area Augmentation System (WAAS -- see Chapters 6-8) to do an integrity check for each satellite visible to the WAAS network in addition to broadcasting precise pseudorange corrections [8-8]. WAAS observes past pseudorange data; so it indirectly conducts multi-step updating.

We can avoid the white noise limitation built into Kalman filters by conducting *Bayesian* probability updates at each time step. This type of algorithm is presented in [7-1] using a hybrid version of Bayes' rule:

$$h(\theta_i | \mathbf{e}_t) = \frac{f(\mathbf{e}_t | \theta_i) h(\theta_i)}{\sum_i f(\mathbf{e}_t | \theta_i) h(\theta_i)} \quad (5.23)$$

where  $\mathbf{e}_t$  is the residual vector,  $\theta_i$  is a fault hypothesis,  $h(\theta_i | \mathbf{e}_t)$  is the posterior probability of  $\theta_i$  given  $\mathbf{e}_t$ ,  $f(\mathbf{e}_t | \theta_i)$  is the likelihood of the observed residual vector given an assumed fault state, and  $h(\theta_i)$  is the *prior probability distribution* given by:

$$h(\theta_i) = \begin{cases} 1 - m\alpha & \text{for } i = 0 \\ \alpha & \text{for } 1 \leq i \leq m \end{cases} \quad (5.24)$$

Here,  $\alpha$  is the assumed prior probability of a single spacecraft soft failure and  $m$  is the number of visible satellites. This model thus has only  $m + 1$  independent failure modes (either no failure or one of  $m$  single-satellite bias failures). In [7-1],  $f(\mathbf{e}_t | \theta_i)$  is computed

based on the least-squares fit model in Section 5.2.2 after applying a correction for the most likely bias estimate.

This formulation can be modified to incorporate a PPM and more general fault assumptions. In [7-1],  $\alpha$  is not set from prior information; it is chosen to give a desired  $P_{FA}$  just as in traditional RAIM. Introducing the PPM from Section 5.3.1 complicates the problem, but it can be simplified by adding just one additional fault mode to (5.24): a "grab bag" of miscellaneous receiver and multiple satellite faults. The overall prior likelihood of these miscellaneous cases can be estimated from our PPM simulations, and  $f(\mathbf{e}_t|\theta_j)$  could be obtained by storing residuals conditioned on failure causes  $\theta_j$ . If the simulation does not provide enough samples of the miscellaneous  $\theta_{m+1}$  failure hypothesis, we can arbitrarily represent our uncertainty here by:

$$f(\mathbf{e}_t|\theta_{m+1}) = 2 \|\mathbf{e}_t\| / \|\mathbf{e}_{\max}\|^2; \quad \|\mathbf{e}_{\max}\| > 2. \quad (5.25)$$

This is simply a ramp-shaped probability density which suggests that the likelihood of a residual vector  $\mathbf{e}_t$  given the occurrence of this "unexplained" fault class increases linearly as the magnitude of  $\mathbf{e}_t$  increases. The higher the expected "maximum" residual  $\mathbf{e}_{\max}$  is chosen, the more uncertainty exists (because the probability band is spread over more possible results). Of course, an attempt to *isolate* a single satellite failure can give more information about this likelihood (see [7-10]). Finally, a "loss function" decision cost model (which does not need to set thresholds *per se*) is proposed in [7-1] and could be modified to handle risk-based cost models using the concepts in Section 5.3.4.

An interesting system design problem results from the introduction of GIC: how should system-wide algorithms and parameters be designed to optimally utilize all available information and transform it for use by independent receivers with simple RAIM test capability? WAAS GIC should be able to carry out the complex filtering or probability updating algorithms outlined here and then transform its posterior results to test thresholds which users can vary based on their distance from the station, their own observable GPS geometry, and their risk aversion. The optimization of the overall shared RAIM algorithm

is a system design problem suitable for global search optimization algorithms such as Simulated Annealing and Genetic Algorithms (described in Section 4.3). Top-down optimization of this type avoids over-reliance on simplifying assumptions, as will be demonstrated in Chapter 8.

## 5.7 Summary and Future Developments

In this chapter, a prior probability model of the GPS system was constructed and applied to RAIM algorithm threshold optimization. Monte Carlo simulations of the PPM and the GPS non-precision approach application produced output probability distributions from which optimal decision thresholds were computed to minimize a risk-based cost model. The outputs were also used to show that the chi-square assumption of traditional RAIM is questionable in real GPS operating conditions. The PPM-based thresholds represent a more general solution to the RAIM problem while using the snapshot residuals algorithm of traditional RAIM. With more extensive design-stage simulation and cost modeling effort, it could be turned into an independent high-fidelity RAIM algorithm.

One key limitation is the discretization of GPS geometries by *both* HDOP and the number of satellites in view. Since this doubly partitions the set of observed position fix errors and residual statistics, there are fewer samples in each bin from which to search for the best threshold. The noisy RAIM cost results in Figure 5.7 demonstrate this problem clearly. As mentioned there, we can get reasonable thresholds by simply fitting a line to the jumpy curve of raw best thresholds, but a more general solution is simply to normalize the residual test statistic by the number of degrees of freedom (number of satellites in view) as motivated by the chi-square distribution assumption. This means that the position-fix samples will only be discretized by DOP, increasing the number of samples in each bin by a factor of 4-5 or so given the same number of overall trials. This is done in the RAIM optimization for WAAS as reported in Chapters 6 and 8. Also, the use of DOP by itself is not the only geometry parameter that can be used. Using “worst-subset DOP” ( $\delta H_{\max}$  from Section 5.2.2) or the mathematically similar maximum failure slope might

work better even though it implicitly assumes only single-channel failure modes. This hypothesis will be tested for the European WAAS application in Section 8.3.5.

Finally, a key issue is to what degree simulation-based and risk-based RAIM threshold optimization should supplant the more traditional approaches. The traditional RAIM approach is used by almost all GPS system designers, and while it may be approximately accurate, much greater robustness to the underlying uncertainty can be obtained from more-detailed probability models, cost functions, and system simulations. This suggests that conventional RAIM can be used during the initial stages of GPS system design to derive a simple set of approximate thresholds. As development proceeds, simulation-based evaluation should provide two things: (1) a means of revising the original thresholds to take the more-complex risk models into account, and (2) a PICA evaluation for safety and certifiability analysis that is a major improvement over the numbers provided by traditional RAIM. Thus, some sort of detailed simulation methodology will be needed to demonstrate PICA feasibility with the optimal integrity parameters. Reliance on the single-failure chi-square models may be fine for development of the basic integrity algorithm, but using those results as a “proof” of PICA acceptability is a trap that we, as engineers working in a new field with many unknown and unmeasurable failure modes, must be careful to avoid.

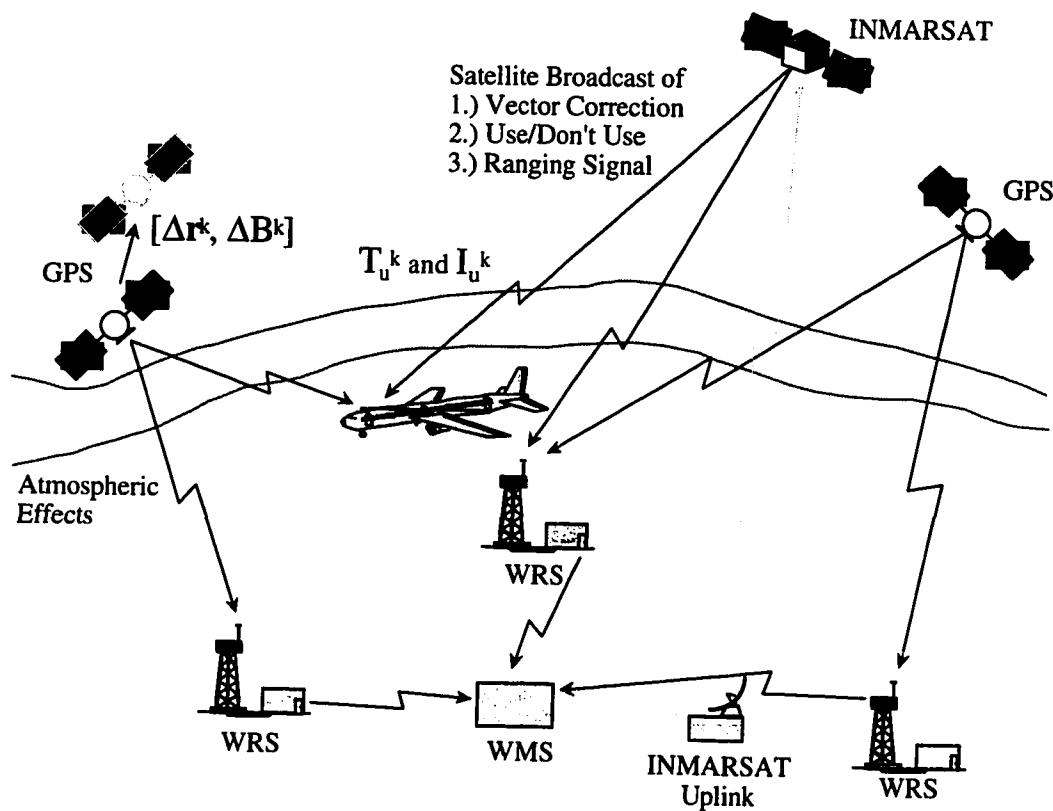
## Chapter 6: Wide Area Augmentation System Ionosphere Integrity

*Coupled with satellite communications, GPS will contribute to increased safety and efficiency of international civil aviation by supporting real-time surveillance of aircraft and reducing the separation requirements - and increasing the number of flights possible - on busy transoceanic routes that represent the most favorable routes between origins and destinations.*

- Robert Loh, "Seamless Aviation: FAA's Wide Area Augmentation System". GPS World, April 1995, pp. 20-21.

### 6.1 Introduction to the Wide Area Augmentation System (WAAS)

In recent years, several approaches have been proposed which utilize additional ground and space-based systems to *augment* the performance attainable from "stand-alone" GPS. Differential GPS (DGPS) is an augmentation approach that offers substantial user accuracy improvement. The basic concept of DGPS is the use of one or more *reference stations* at known, pre-surveyed locations. These stations have accurate, well-calibrated GPS receivers which determine their position and compare to their known "true" locations. Several algorithms exist for calculating scalar ("lumped" pseudorange) or vector (error-specific) corrections for each GPS satellite in view of the reference station. A key distinction exists between "local-area" and "wide-area" DGPS. In the former, also known as LADGPS or LAAS, a single reference station provides corrections for users in its vicinity (i.e., within about 100 km). The latter, known as WADGPS or WAAS (for the FAA's "Wide Area Augmentation System"), is composed of a network of widely spread reference stations (or WRS's) which transmit their observations to a master station. The master station, or WMS, fits the combined observations into corrections that encompass the set of GPS satellites visible to users over a large geographic area. These corrections are ideally transmitted over a hemispheric area by geosynchronous communication satellites, which broadcast their signal on the GPS L1 frequency and also provide additional GPS ranging signals. A pictorial outline of the FAA WAAS system is shown in Figure 6.1. Commercial WADGPS systems may substitute ground-based communications schemes, such as FM-subcarrier broadcast [6-14].



**Figure 6.1: WAAS Network Layout Overview**

Because wide-area DGPS ties the observations of its reference stations together into a uniform set of corrections, it promises to provide navigation accuracy sufficient for FAA Category I precision approaches to a large region of users at a much lower price than would be required if the same service were to be provided by many independent local-area DGPS stations. However, key uncertainties regarding WAAS performance are raised by the fact that, unlike denser LAAS networks, users must rely on observations from points hundreds of kilometers away. While it is expected that spatial decorrelations are well-behaved over these distances, insufficient data has been gathered to verify this to the necessary statistical certainty.

At Stanford, a prototype WAAS network has been built with three wide-area reference stations (WRS's) in Arcata, CA., San Diego, CA., and Elko, NV. Each of these stations has a dual-frequency GPS receiver and a PC. The wide-area master station (WMS) computer, which receives all of the WRS observations and computes the WAAS

corrections, is located at Stanford. The Stanford site also has a separate dual-frequency receiver which serves as a static user for purposes of WAAS performance evaluation but does not contribute to the WAAS message solution. This network has demonstrated sub-3 meter 95% vertical accuracy during flight trials at Palo Alto Airport (near Stanford), Livermore, and Truckee airports [8-18].

The FAA is developing the WADGPS concept further by introducing a nationwide network of reference stations to cover all of the Continental U.S. (CONUS). This operational system will be preceded by a so-called "National Satellite Testbed" (NSTB) of as many as 24 reference stations, possibly including three each in Canada and Mexico. The NSTB network is separate from the operational system and will serve to verify the WAAS concept in full-scale operation as well as help determine the accuracy and integrity to be expected of WAAS when the operational system comes on-line [8-11]. In addition to the SPS requirements mentioned in Section 5.2, RTCA, a Federal Advisory Organization comprised of industry, government, and academia, has promulgated Minimum Operational Performance Standards (MOPS) for WAAS user equipment [8-13]. The FAA has also released the WAAS Specification [8-16] which attempts to provide a set of requirements which the networks under development will be expected to meet.

This chapter applies the RAIM concepts introduced in the preceding chapter to WAAS. A WAAS integrity study is conducted which focuses on rare-event ionospheric spatial decorrelations. The extent and likelihood of these are unknown at present; thus an uncertain probability model is fitted to limited experimental data and then extrapolated by Monte Carlo simulations. In other words, the PPM for the SPS user is replaced by a model of the uncertain magnitude of ionospheric correction errors based on the best available information. The process of simulating WAAS-aided user position fixes along with these error modes is similar to that of the SPS case. The key application considered is aircraft Category I precision approach, which would rely on GPS/WAAS to provide a 3-D (not just horizontal) position fix accurate to within a few meters (under normal conditions) down to a minimum decision height 200 feet above the ground. This puts great stress on vertical position accuracy, which is the weakest dimension for GPS because of the generally poorer geometry (VDOP) in that dimension. Therefore, all

WAAS accuracy and integrity simulations carried out here focus on vertical position error only.

A deeper understanding of WAAS operations and performance will be provided in subsequent chapters. Chapter 8 will introduce a system-wide evaluation procedure for WAAS user accuracy performance under “normal conditions,” i.e., the 90-95% of the time that all elements of the system are within normal operating parameters. Under these conditions, massive simulation of user position fixes is not necessary, as system errors can be propagated from one element to another using stochastic linear system theory since all of the basic assumptions are met. The integrity approach developed in Chapters 5-6 is then applied to the results of the normal-case analysis to simulate how RAIM handles a variety of possible failures modeled in prior failure probability distributions. After combining the accuracy and integrity studies into a unified PICA evaluation, Chapter 8 proposes a system value model for the candidate WAAS architectures being considered. Finally, using this model as the objective function for the WAAS-based capability being provided, a genetic algorithm formulation is introduced which can evolve an optimal WAAS network architecture.

## **6.2 Rare-Event Ionosphere Decorrelation Models**

### **6.2.1 Introduction**

Recent experimental data suggests that ionospheric spatial decorrelation is the most serious systematic threat to WAAS accuracy and integrity. Whereas other WAAS error sources, such as ephemeris and tropospheric error, appear to have reasonably well-bounded tails, ionospheric decorrelation has the potential to become very large (over 5-meter differences) when pierce-point separations exceed 1000-1500 km. A probability model based on the latest available data has been constructed to model our uncertainty regarding the magnitude of “worst-case” ionospheric errors as well as the error magnitudes to be expected under more normal conditions. This model forms the basis for a series of WAAS performance simulations in which accuracy and integrity for users randomly located in the Continental U.S. is measured. Our goal at this stage of WAAS

program development is to use computer simulation to project WAAS performance given what is now known. Due to the limited error data and uncertain error models we have now, it is very difficult to verify that WAAS meets the continuity and integrity performance requirements for Category I aircraft precision landing [8-16], but our results under these limitations suggest that it will be feasible to certify this once data from WAAS full-system tests is available in a couple of years.

#### 6.2.2 Breakdown of WAAS Error Sources

WAAS employs GPS corrections computed by a network of reference stations to remove most of the satellite-based errors that exist without differential corrections. Small errors due to spacecraft clock and ephemeris remain, however, and depending on the *latency* (time to reception) and *age* (time to last usage) of DGPS corrections, Selective Availability (SA) will contribute a ranging error estimated by the simple kinematic relation:

$$SA \text{ error (m)} \equiv \frac{1}{2} a t^2 ; \quad a \equiv 0.004 \text{ m/sec} \quad (6.1)$$

where  $t$  represents latency plus age, and the "acceleration factor"  $a$  has been estimated by studies of SA over time [6-14]. This parabolic approximation should be valid for  $t \leq 40$  seconds. Using  $t = 12$  seconds as a conservative number for WAAS, this *time-decorrelation* error is very small (see Table 6.1 below).

Errors that are *spatially decorrelated* depend on the geographic separation between the user and the site for which the corrections were computed. Local conditions within the ionosphere (about 350 km above the Earth) and the troposphere (the band of the atmosphere from the surface to about 7 miles up) cause variable delays in the GPS ranging signals which distort the true range to the user. These are described in detail in Section 6.3.3.

Other user-specific errors are due to conditions at or near the user itself, such as multipath (due to reflected ranging signals) and receiver noise. The user should be aware of these local error conditions in order to make good integrity decisions; this is one reason why RAIM is traditionally handled by the user. A summary of the normal-

condition user error model (excluding spatial decorrelation errors) is given in Table 6.1 [8-5,7]. The ephemeris and clock error terms are after the WAAS correction is applied; thus they are much better than the uncorrected SPS numbers in Table 5.1 of the previous chapter. The rms of the variances in Table 6.1 is about one meter. In our WAAS simulations, the effects of these errors are modeled by Gaussian noise with a standard deviation of 1 m.

However, for WAAS, the quality of the DGPS corrections is another key integrity factor. The network of remote monitor stations each has backup hardware that computes the corrections independently and looks for any significant discrepancies which might indicate a WAAS hardware or software fault [8-12]. Flagged faults may lead to non-availability of one or more reference stations; WAAS applications would not be possible during that time for users located in areas whose corrections without those stations do not meet integrity requirements at the monitor station. Unflagged faults could lead to integrity breaches and are thus very serious, as they would affect many users without their knowledge. As a result, WAAS integrity is a dual responsibility of the network and of each individual user.

Error Source	Standard Deviation	Notes
receiver noise	0.5 m	user-dependent
multipath	0.5 m	user-dependent
SV ephemeris	0.4 m	after WAAS corr.
SV clock (with SA)	0.3 m	12 sec. latency + age
reference survey errors	0.5 m	misc.

**Table 6.1: WAAS Error Source Summary**

### 6.2.3 Spatial Decorrelation Error Models

The ionosphere layer of the upper atmosphere is awash in charged particles that can affect the delay experienced by electromagnetic signals sent through it. Ionospheric conditions in general are dependent on seasonal and daily variations as well as solar effects, which include an 11-year cycle of waxing and waning solar emissions of charged particles (the "solar wind") and the occasional solar flare or other temporary event that creates higher charge levels in the ionosphere. At a smaller scale, charged-particle

conditions in the ionosphere will vary from site to site on the globe. GPS ranging signals received by a user are assumed to penetrate the ionosphere at a variable altitude that averages about 350 km (this penetration point is called a *pierce point*, or PP) and suffer a variable propagation delay in doing so. Note that signals from satellites at lower elevation angles will pass through this layer at a more oblique angle, suffering relatively more time delay. This delay appears as an error to the measured pseudorange for that satellite.

The unaided GPS user relies on the *Klobuchar Model* to adjust for ionospheric conditions based on almanac data contained in the GPS ranging message [8-6]. This model is accurate to 5-10 meters one-sigma, so the remaining error is significant. Differential stations can greatly improve this accuracy by using high-quality *dual-frequency* GPS receivers to measure the local ionospheric delay to around 0.5 meters one-sigma (the remainder is primarily *Tgd error*, which is the error between the L1 and L2-frequency satellite timebases). Local-area DGPS systems without dual-frequency reference receivers assume that the user is close enough to the reference site (that broadcasts DGPS corrections) that the variance in the local ionosphere between the user's satellite pierce points and those of the reference station are manageably small [6-14]. For WAAS, however, the relatively low density of monitor stations across the U.S. means that these distances, or *baselines*, are much larger [7-12]. Thus, algorithms to model ionospheric delays across a wide area are critical to the accuracy of WAAS corrections.

Research at MITRE has developed the "grid algorithm" to handle ionospheric corrections for WAAS [8-6]. Figure 6.3 in Section 6.2.4 illustrates this process. Basically, the remote monitor stations report their measured ionospheric delays for their satellite pierce points to the master station, which computes vertical (i.e. from a hypothetical satellite directly overhead) ionospheric corrections for each point in a lattice that is superimposed over the U.S. Grid points could be separated by either 5 or 10 degrees of latitude and longitude, or the grid may use higher gridpoint densities at lower latitudes, where grid points separated by a constant longitude are further apart. Lower gridpoint separations clearly are less susceptible to spatial decorrelation, but they become difficult from a communications standpoint due to the greater number of estimates that

must be uplinked to the user, who receives the corrections for each grid point as part of the WAAS message.

For each of his or her satellite pierce points, the user determines the four grid points that surround it, the distance  $d_i$  ( $i = 1, \dots, 4$ ) between the pierce point and each grid point  $i$ , and then he or she interpolates the vertical correction at that pierce point using the following equation [8-6]:

$$\text{Iono. Vertical Correction} = \frac{\sum_{i=1}^4 \mathbf{I}_{est}^i \frac{1/d_i}{\sum_{k=1}^4 1/d_k}}{\quad} \quad (6.2)$$

where  $\mathbf{I}_{est}^i$  contains the WAAS gridpoint estimates. This vertical correction must be adjusted for the obliquity of that satellite, which can be calculated from its local elevation angle  $\theta$  and pierce-point elevation  $\theta'$  using the obliquity factor ( $OF$ ):

$$OF = 1 / \sin(\theta') \approx 1 + 2 \left[ \frac{96 - \theta(\text{deg.})}{90} \right]^3 \quad (6.3)$$

Note that this is an approximation based on a flat Earth and is only valid for  $0 \leq \theta \leq 90$  degrees. Multiplying this factor by the interpolated vertical delay gives the final pseudorange correction for that satellite.

**6.2.3.1 Ionosphere Sampling Model:** Various researchers have attempted to measure the accuracy of ionospheric corrections of this type using two to four reference stations. In [3], daytime differences between slant ionospheric error predictions at reference stations from 300 to 1800 km apart were measured in 1992-93, when ionospheric conditions were normal apart from a few "stormy" days. In [4,13], grid algorithm corrections were computed based on a network of three reference stations in the Mid-Atlantic area, and the interpolated slant error prediction was compared to the error measured at a fourth station at the assumed user site. This data was taken from November 1992 - July 1993, and again short periods of peak activity were observed.

Using the data in [8-5,6,10], a model of ionospheric delay likelihoods suitable for WAAS performance simulation has been developed. This model gives a distribution for the relative *slant* ionosphere error between two locations as a function of the following factors:

(a) *Solar Cycle*: Most of the experimental data obtained to date is from the early 1990's, which are between the high and low peaks of the current 11-year solar cycle. Delays in peak years would be approximately *double* those measured in these experiments [8-10]. Similarly, the bottom of the cycle would give delays about half those observed. This is modeled by a correction factor for solar cycles,  $CF$ , which ranges from 0.5 to 2.0:

$$\begin{aligned} CF &= 1 + \sin(2\pi t_y / 11) && \text{when } \sin(\bullet) \geq 0 \\ CF &= 1 - 0.5 \sin(2\pi t_y / 11) && \text{when } \sin(\bullet) < 0 \end{aligned} \quad (6.4)$$

In our WAAS simulations, the current date within the solar cycle,  $t_y$ , is sampled in the outermost loop by a uniform distribution over an 11-year time span. Note that in the covariance projection method of Chapter 6, this distribution is not sampled. Instead, a reasonable worst case value  $CF = 2$  is used to model “normal conditions” conservatively.

(b) *Seasonal*: Slant error results are provided in [8-10] over three seasonal conditions: winter, summer, and equinox. In addition, the presence of ionospheric storms on a handful of days in 1992-93 is a fourth possible condition. Data for various station separations was collated by seasonal condition to calculate base ionospheric standard deviations for a baseline of 348 km. These two-sigma results and the seasonal probabilities are given in Table 6.2.

Condition	$2\sigma$ Deviation	Probability
summer	0.68 m	0.45
winter	1.05 m	0.45
near equinox	1.18 m	0.09
ionosphere storm	1.85 m	0.01

**Table 6.2: Ionosphere Deviations by Season**

Local conditions also vary according to the time of day, with the peak deviation occurring at about 2 PM local time, and errors are generally much lower at night [8-10]. This effect is not modeled in any of the WAAS simulations in this thesis, but it could be easily added if a distribution of local user time is incorporated. This would be desirable for a simulation that is used to project the risk inherent in conditions at a specific local time of day. For this work, the conditions prevailing near the 2 PM peak delay time are assumed to always be in effect.

(c) *Baseline*: Data was taken in [8-10] at various station separations, allowing us to fit a line of ionospheric standard deviation to same-season data over varying-length baselines and then normalizing by a standard 348 km separation. The following linear fit was obtained, where  $RM$  is the differential slant error in meters:

$$RM = 0.416 + [0.542 (\pm 0.06)] \frac{d}{348} \quad (6.5)$$

where  $d$  is the separation distance (in km) and the  $(\pm)$  represents the one-sigma deviation in the slope of the linear fit. Note that the constant factor of 0.416 in this equation, which would exist at a separation of zero, partially represents the effects of L1-L2 interfrequency biases in reference receivers as well as simple noise decorrelation between receivers.

(d) *Tail Distributions*: Data provided in [8-10] represents 90-98% values of slant ionospheric errors. It is clear from this data that the 98%/90% error ratio is greater than 1.416 as predicted by a Normal distribution. The amount of excess varies between individual data points, but it can be approximated by a multiplicative factor  $TM$  ("tail modifier") using the standard Normal(0,1) distribution as follows:

$$1.645 \leq |z_{samp}| \leq 2.33 \Rightarrow TM = 1.13 \pm 0.11 \quad (6.6a)$$

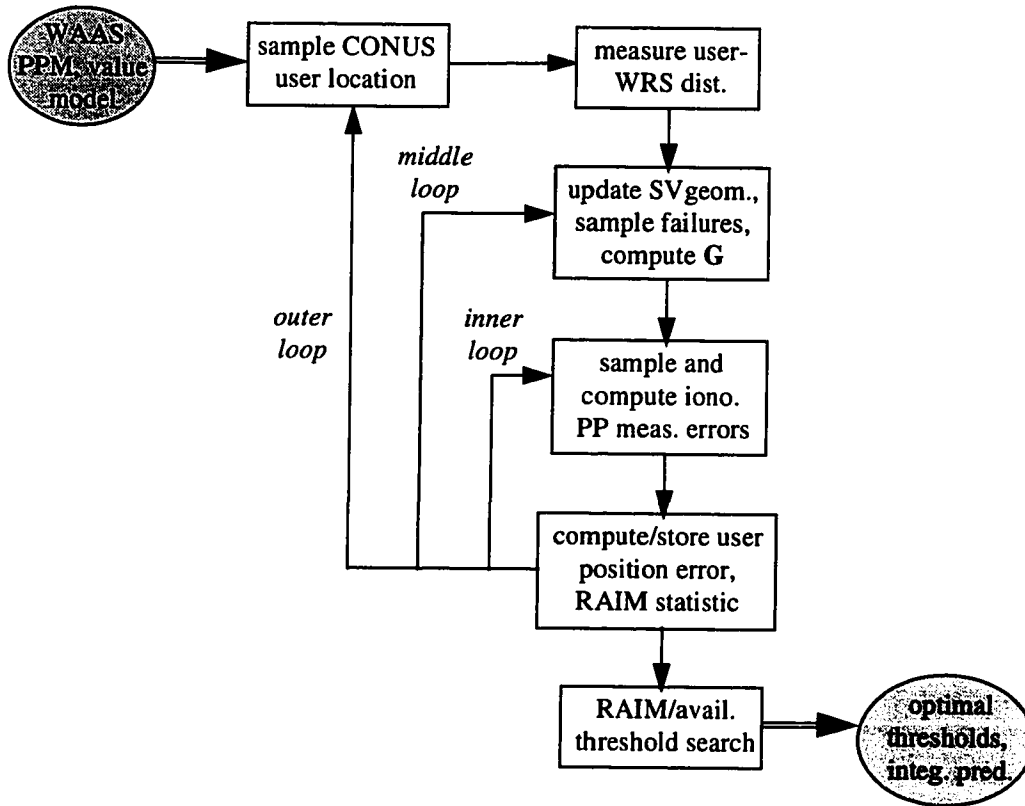
$$|z_{samp}| \geq 2.33 \Rightarrow TM = 1.40 \pm 0.25 \quad (6.6b)$$

$$else \quad TM = 1.0$$

No breakdowns are given for less than 90% confidence; thus it will be assumed that the distribution of slant delay is Normally distributed between the 5th and 95th percentiles. The delay error plots in [8-6,10] support this conclusion, although significant uncertainty remains. Between the 1st and 99th percentiles, we continue to use the Normal distribution framework but expand the base variance by multiplying by the (uncertain) *TM* factor given in equation (6.6a). In the worst 2% of cases, we multiply by the larger factor given in equation (6.6b). This is not the most convenient analytical model for ionospheric tail distributions, but it is well suited for Monte Carlo simulations using standard Normal random numbers. Computer simulations of this type allow us maximum flexibility in expressing our model's uncertainty.

**6.2.3.2 WAAS Ionospheric Sampling Procedure:** Monte Carlo sampling of this WAAS ionosphere model is conducted in the C programming language. Figure 6.2 gives a flow chart of the processes and iteration loops executed by this simulation.

The outermost of three loops randomly places an airborne WAAS user within the Continental U.S. (CONUS) by uniformly sampling his latitude and longitude (note that each 2-D point has an approximately equal chance of being chosen) between [25°, 49° N] and [66°, 125° W] respectively. Sampled points that happen to lie outside the borders of CONUS are discarded and re-sampled. The resulting position is compared to the locations of 20 FAA Air Route Traffic Control Center (ARTCC) sites proposed as WRS locations for the operational WAAS network to measure the relevant separation distances [8-11]. In the second loop, for each user position, approximately one day's worth of GPS satellite geometries is simulated in approximately 15-minute intervals using the sampled time update explained in Section 5.1.1. The Volpe data for the Primary-21 constellation is used, along with the four geosynchronous satellites listed as proposed WAAS augmentations in Section 5.1.1. Two of these extra GPS ranging sources (AOR-W and POR) are normally visible within CONUS. Hard failures (only), which render satellites temporarily unusable, are also simulated for the 24 GPS satellites based on the PPM probabilities and procedure given in Section 5.3.



**Figure 6.2: WAAS Integrity Simulation Flow Chart**

For each GPS geometry, the observation matrix  $G$  and the vertical dilution-of-precision (VDOP) are computed along with elevation angles for each satellite in view of the user using (5.7) and (5.11b). In the innermost loop, 100-200 individual samples of slant ionospheric error are generated for each satellite in view. Not only does this allow us to tabulate the overall distribution of ionospheric errors, but it allows separate position error computations for each inner-loop trial, even though the satellite geometry has not changed.

For each inner-loop iteration, the current state of the 11-year solar cycle is sampled from a uniform distribution, and  $CF$  is computed from equation (6.4). The seasonal condition is sampled using the distribution in Table 6.2 to obtain the base sigma,  $\sigma_b$ . Given a baseline distance, we compute the separation modifier  $RM$  from equation (6.5), sampling the uncertain slope parameter when necessary. The ionospheric error variable  $z$  is sampled from a standard Normal(0,1) distribution, and outcomes outside the 5-95%

"normal" range result in the tail modifier  $TM$  from equation (6.6). Finally, since the base data in Table 6.2 is slant error (not vertical), it already includes an implicit obliquity factor that must (imprecisely) be factored out. From the data in [8-10], a "mean obliquity factor"  $MO = 1.765$  was estimated. This is a necessary approximation which represents the normalized obliquity based on a set of GPS orbit simulations which give satellite elevation distributions. The final slant ionospheric error  $SE$  is computed as:

$$SE = RM\ CF\ \frac{OF}{1.765}\ TM\ \sigma_b\ z_{samp} \quad (6.7)$$

Note that  $SE$  represents an error, reported by the nearest "monitor", relative to the "true" ionosphere error of zero that would be obtained with zero spatial decorrelation.

**6.2.3.3 Troposphere Sampling Model:** Local atmospheric conditions in the troposphere also create transmission delays. Simple models exist for the troposphere, and the WAAS user must rely on them to help correct for this delay [6-14]. Because the troposphere is closer to the Earth than the ionosphere, the delay error is even more sensitive to satellite elevation than for the ionosphere. Data relating 95% tropospheric errors observed by a WAAS user as a function of satellite elevation is given in Figure B-1 of [8-5]. Using two cubic polynomial fits to this data, we can compute the standard deviation for any satellite elevation angle  $\theta$  (in radians):

$$\sigma_{tr} = -61.23\theta^3 + 55.9\theta^2 - 19.7\theta + 3.045 \text{ (low el.)} \quad (6.8a)$$

$$\sigma_{tr} = -0.495\theta^3 + 1.73\theta^2 - 1.99\theta + 0.98 \text{ (high el.)} \quad (6.8b)$$

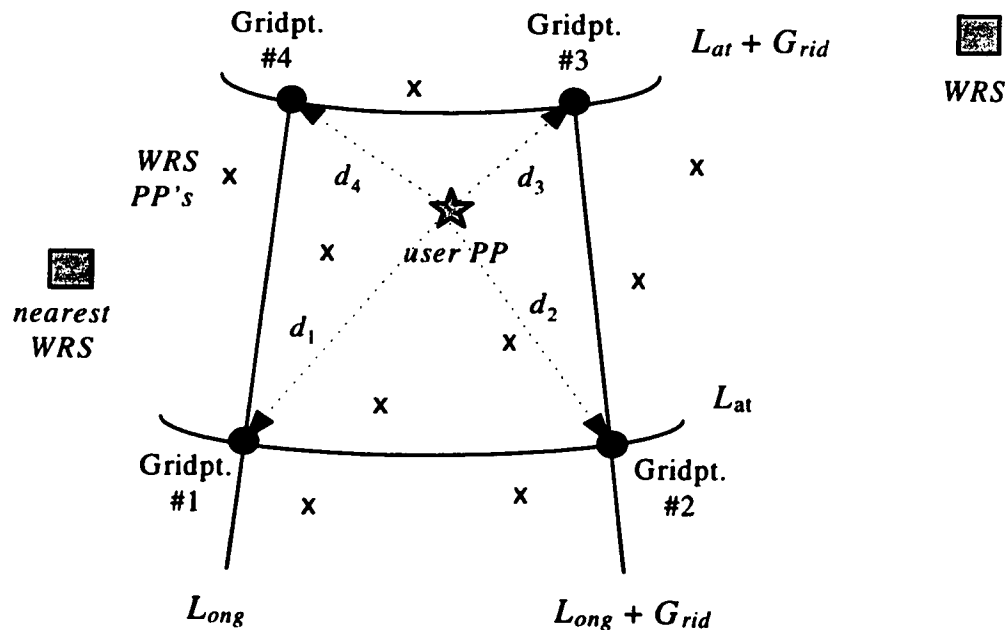
where the boundary for low/high elevation is  $18.75^\circ$ . Tropospheric errors are generally a lesser problem than are ionospheric errors, and their error distribution *saturates* in [8-5] rather than indicating tail inflation. Thus, Gaussian sampling of the standard deviations computed in (6.8) is actually conservative. In the simulation, tropospheric errors are

sampled from  $\text{Normal}(0, \sigma_{tr})$  distributions for each satellite pierce point in view. This error is added to the sampled slant ionospheric error from the previous section.

#### 6.2.4 Simulation of DGPS Corrections

The spatial decorrelation sampling algorithms can be packaged in various ways to model differing DGPS correction algorithms. The simplest application is local-area, or LADGPS, corrections which are broadcast from the nearest monitor site that can see the satellite in question. The user is thus dependent on a single monitor. Using only the ARTCC locations proposed for WAAS, long baselines of hundreds of kilometers result. For each SV in view of the user, an *independent* slant ionospheric error is generated using (6.7) based on the distance to the nearest qualified monitor. Tropospheric errors are also sampled, and a sample of the other error sources from Table 6.1 is added. Using the standard GPS observation equations from Section 5.1.2, the 3-D and vertical position errors can be computed and stored for each trial.

Simulating the grid algorithm is much more complex. Without a complete simulation of all WAAS ground operations, an exact model is not possible. The grid



**Figure 6.3: User Pierce Point and WMS Grid Geometry**

model is designed to represent the essence of interpolation in correcting for spatial decorrelations given the layout of monitor stations on which the gridpoint estimates are based. The grid is composed of 5 or 10-degree cells that cover the geographic area from 10-70° N latitude and 50-160° degrees W longitude. The pierce point of each satellite in view of the user is within one of these grid cells. An illustration of this pierce point separation geometry is shown in Figure 6.3.

To sample an ionospheric delay error at a given user PP, an ideal normalized slant ionosphere error is derived for each grid point by sampling, using (6.7), given the distance from that grid point to the user. Equation (6.2) interpolates for the effects of already-sampled grid points (this is a necessary abstraction since no multiple-baseline dependent ionosphere data exists). Another error sampled from (6.7) is added to each grid point based on the distance from that grid point to the nearest monitor station. Finally, the user's slant ionospheric error is computed by a final interpolation from the four gridpoints using equation (6.2) and applying the obliquity correction ( $OF / MO$ ). Troposphere and other errors are handled as before, as is the GPS position fix calculation and storage of results. The result is a representation of temporal correlation among the separated user pierce points; they will all be affected by similar conditions at any given time.

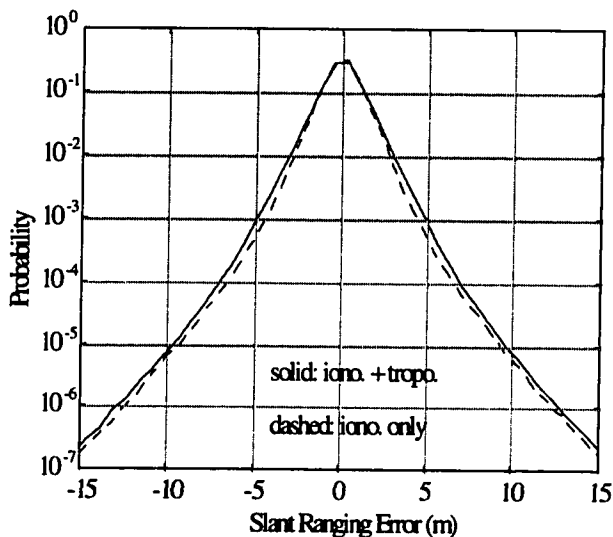
### **6.3 Ionosphere Decorrelation: Simulation Results**

#### **6.3.1 Spatial Decorrelation Error Results**

Although the spatial decorrelation models presented in Section 6.2.3 are inexact and contain substantial parameter uncertainty, they represent the best knowledge available to us with the current set of experimental data. The plots that follow show the results of simulations using these models, allowing us to compare and contrast differing DGPS ionospheric correction methods. All the results in this chapter come from simulations of 2000-5000 user position samples, then 100 GPS geometry samples (over approximately 24 hours) for each user position, then 100-200 spatial decorrelation samples for each spacecraft in view. In other words, 100-200 separate error samples were taken for each user-GPS geometry combination, and each error sample led to a position fix error

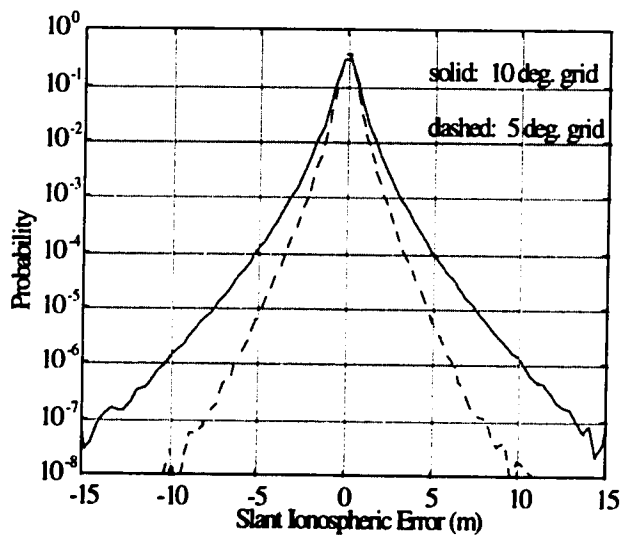
computation. Overall, 20 to 80 million error samples were conducted, and the entire process took 12-18 hours of CPU time on a Sparc-20.

Figure 6.4 is a semi-log plot of the density function of slant ionospheric delay for LADGPS, in which only the nearest ARTCC reference monitor is used for ionospheric corrections. The outer solid line on this "volcano-shaped" plot represents samples of summed ionospheric and tropospheric errors for a given spacecraft. The dashed line inside that represents samples of ionospheric errors only. Clearly, the ionospheric errors have much larger magnitudes. Also note that a Gaussian 1-meter-sigma error distribution has tails which fall off much more rapidly than does the distribution of Figure 6.4. This illustrates the extent to which the tail distributions of ionospheric errors have been magnified by our model.



**Figure 6.4: LADGPS Spatial Decorrelation Errors**

Figure 6.5 is a similar semi-log plot for the WAAS case in which the MITRE grid algorithm is simulated with 10° (solid line) and 5° (dashed line) cells. The 10° case result is generally 2-10 times better than the LADGPS result in Figure 6.4, as expected, although the tail-distribution expansion pattern is not much improved. Substantial further improvement in both error



**Figure 6.5: WAAS Ionosphere Decorrelation Error**

magnitude and rare-event spread is gained by using a  $5^\circ$  grid. However, a  $5^\circ$  grid multiplies the number of needed ionosphere correction points by 4; thus it may overstress the current 250-bps WAAS communications signal format [8-8].

### 6.3.2 Position Error Results

The same simulations that generated the above spatial-decorrelation distributions were used to compute 3-D and vertical DGPS position errors. As mentioned before, vertical position errors dominate this picture and are the most troubling errors for precision aircraft landings. Figure 6.6 shows the observed cumulative distribution of user VDOP from the Inmarsat-augmented and unaugmented GPS geometries sampled. For the basic 24-satellite GPS constellation, VDOP is below 3.2 about 99% of the time. With the extra geosynchronous ranging sources,  $\text{VDOP} \leq 3.2$  occurs 99.87% of the time, and 6 or more satellites are visible 99.9% of the time (using a  $7.5^\circ$  user mask angle). These results suggest that very high system *availability* for precision landings, approaching the desired figure of 99.9% set by RTCA, may be possible [6-15,8-7]. In our simulations, a preliminary availability limit of  $\text{VDOP} = 3.2$  was set; i.e. all GPS

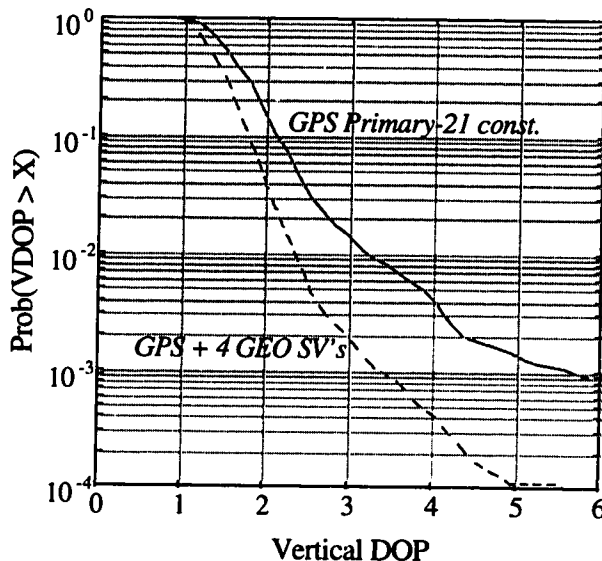


Figure 6.6: VDOP Cumulative Distributions

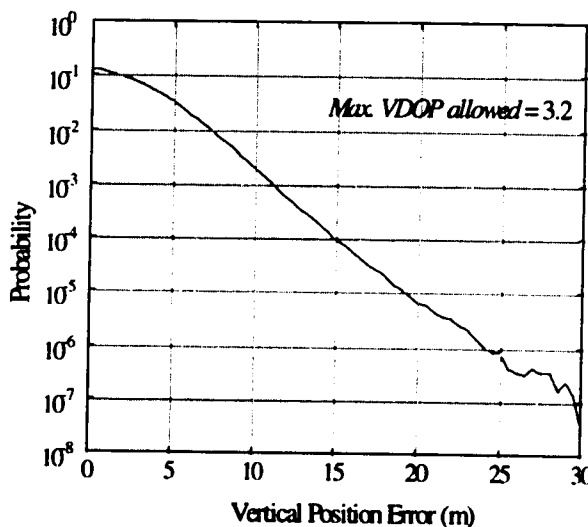
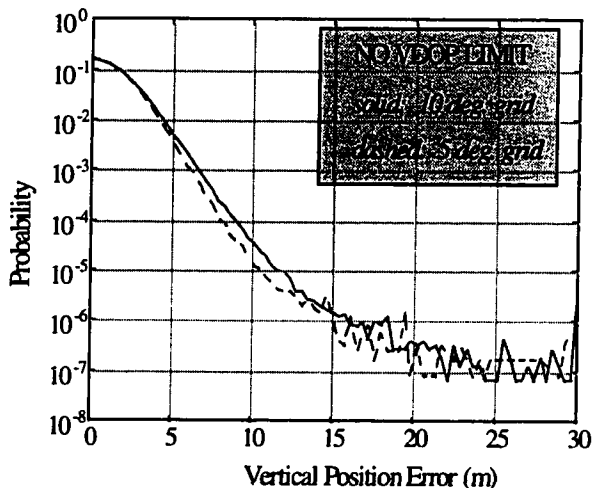


Figure 6.7: LADGPS Position Error Distribution

geometries with higher VDOP would be counted as *non-available* for the precision landing application. The application of RAIM in the following section will be used to confirm this VDOP availability threshold.

Figures 6.7 and 6.8 plot the *density* of vertical position errors for the three cases discussed above: LADGPS (in Fig. 6.7), 10°-grid WAAS (Fig. 6.8, solid), and 5°-grid



**Figure 6.8: WAAS Position Error Distributions**

WAAS (Fig. 6.8, dashed). The LADGPS plot in Figure 6.5 uses the  $VDOP = 3.2$  limit, which helps limit the position-error effects of rare-event ionospheric errors. However, the two WAAS curves shown in Figure 6.8, which are generally superior to those for LADGPS for errors under 25 m, *do not* use this limit for this illustrative simulation run (Figure 6.11 will show the result with the VDOP limit). Without that availability limit, the probabilities of vertical errors stop decreasing past 20 m. From this, it seems that this VDOP limit is the key to limiting the propagations of rare-event spatial decorrelations into unacceptable position errors. Substantial improvements are gained with availability and other RAIM checks, and these will be demonstrated in the following section on RAIM optimization.

Despite being based on a highly uncertain ionosphere error model, the overall results obtained for WAAS from these stations are about 30-35% better than those obtained experimentally by MITRE and Stanford from their 3-station systems [8-5,18], demonstrating the potential of this 20-station monitor configuration. Furthermore, WAAS is not limited to the configuration under study, as further improvements could be obtained from denser monitor networks and/or more geosynchronous satellites. The effects of hardware augmentations will be discussed in the following two chapters.

## 6.4 WAAS Integrity Methods and Results

### 6.4.1 Residuals Threshold Test Algorithm

Given the basic WAAS configuration represented by these simulations, it is possible to apply various integrity algorithms to the simulations in an attempt to further improve the system's accuracy and to warn users of conditions where accuracy risks are unacceptable. This includes availability problems (produced by poor DOP, which is known *a priori*) and continuity and integrity concerns, which result from detected and undetected system faults that occur during a precision approach.

As noted in Section 5.2.2, the standard RAIM algorithm that has been applied to both uncorrected GPS and DGPS consists of a DOP availability check and a *residuals threshold test*. For WAAS, in order to reduce the number of separate geometry bins, the residuals statistic  $D_n$  is normalized by the number of spacecraft in view. The calculation of this statistic, revised from equation (5.15), is done as follows:

$$D_n^2 = \mathbf{z}^T [\mathbf{I}_n - \mathbf{G}\mathbf{G}^*] \mathbf{z} / (n-4) \quad (6.9)$$

As before, the user compares  $D_n$  at each position fix step to a preset threshold  $T$  for each VDOP bin, and an *integrity alarm* is issued if  $D_n > T$ . Analytical methods for calculating desirable thresholds described in Section 5.2.2 can be applied here, but this research uses the more general user cost optimization developed in Section 5.2.3. In this approach, a full set of WAAS simulations is conducted, and each resulting set  $(x, D_n)$  is stored as a function of VDOP (only). Afterward, a simple optimal threshold search is conducted for each VDOP cell to find the threshold that minimizes the *expected user cost*.

### 6.4.2 WAAS User Cost Model

For a precision approach, integrity decisions must be made closer to the ground. The maximum allowable vertical position error for Cat. I precision approach is set at RPE = 15 m, which is still quite conservative. The user costs listed in Table 6.3 below for *false alarms* ( $D > T$  but error  $\leq$  RPE), *detected errors* ( $D > T$  and error  $>$  RPE), and *missed*

RAIM Result	Base Cost	Variable Cost
good position	0	0
detected error	1	0
missed detection	300	10
false alarm	1.025	0
RAIM non-available	0.025	0

**Table 6.3: WAAS User RAIM Cost Parameters**

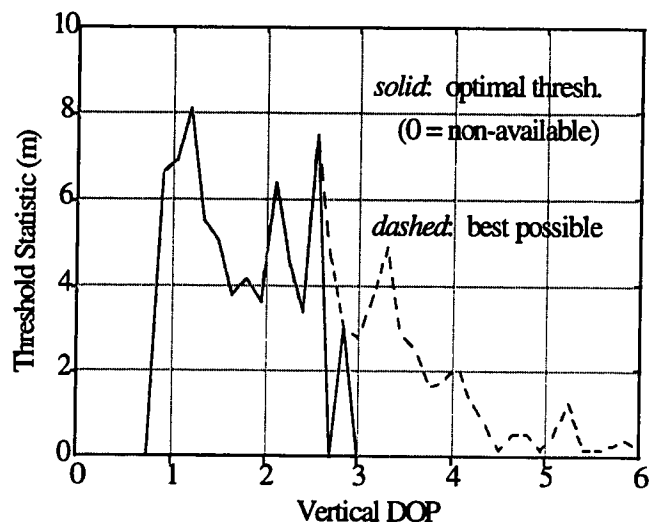
*detections* ( $D \leq T$  but error  $> RPE$ ) are derived from the allowed risks of an aircraft accident as developed in Section 5.3.4. For Category I precision approach, we assume that a fatal accident, given a position error exceeding RPE, has about the same likelihood as for the non-precision approach case; thus the risk multiplier  $R_C = 5 \times 10^{-5}$ . The non-availability cost remains more or less the same because the inconvenience of not being able to use the system is also assumed to be similar. Therefore, only the missed detection cost has changed from the SPS RAIM costs listed in Table 5.3.

Note that the variable cost of a missed detection means that in addition to the base cost of any MD, an additional cost is paid for each "bin" (of width 0.5 m) beyond the RPE limit that the missed position error is located. Optimal thresholds for each VDOP bin are determined using the cost optimization procedure explained in Section 5.3.5. This cost minimization will determine what VDOP should be chosen as the availability limit.

#### 6.4.3 WAAS User Integrity Results

Running this threshold search on the WAAS simulation results in Section 6.3 gives the set of optimal thresholds shown in Figure 6.9. The discontinuities in the optimal thresholds for adjacent VDOP cells indicate that the statistical significance of the WAAS Monte Carlo simulation is still limited. We expect the best thresholds to uniformly decrease for increasing VDOP, since higher VDOP gives a greater probability of position errors exceeding RPE and thus should lead to more integrity alerts. This result would be realized from an infinite number of simulation trials. Instead of this, simply fitting a line to the optimal thresholds in Figure 6.9 (as suggested in Section 5.4.3) should be adequate, and the resulting thresholds can then be tested on a longer set of simulation trials. Note

that the threshold curves shown here are somewhat smoother than in the SPS case (Figures 5.6 and 5.7), indicating that the normalization by number of satellites in view gives a better result. In any case, VDOP greater than 3.0 results in non-availability for this approach, which is close to the limit of 3.2 used in our other algorithms.



**Figure 6.9: WAAS RAIM Optimal Thresholds**

Table 6.4 shows the overall results for this RAIM method. For a VDOP limit of 3.0, availability is still very high (although not quite up to the desired 99.9%). VDOP serves as a reasonable availability criterion, although some improvement is obtained for the European WAAS case using maximum  $V_{\text{slope}}$  as described in Section 8.3.5. False alarms are relatively infrequent, but they occur about 50 times more often than the MOPS number of  $10^{-5}$ . One reason for this is that the calculation done here assumes that any position error does not exceed RPE but is flagged by RAIM is a false alarm, but traditional RAIM tends to assume linearly-growing failures, so that what are “false alarms” in the simulations would be treated as “early detections” of a soon-to-violate-RPE failure. Thus, to the degree that real failures behave predictably, the MOPS requirement may make sense, but without such an assumption, it appears difficult to meet this arbitrary number under current WAAS performance uncertainty.

Pr(WAAS available)	0.9958
Pr(false alarm)	0.0005
Pr(vertical error > 15 m)	$1.0 \times 10^{-6}$
Pr(missed detection   error > 15 m)	0.2691
Overall User Cost	0.00067

**Table 6.4: WAAS RAIM Results Summary**

The overall user cost for the optimized RAIM parameters is reasonably low, and we see that the probability of an undetected penetration of the outer precision-landing tunnel is simply  $\Pr(\text{vert. error} > 15 \text{ m}) \Pr(\text{MD} \mid \text{vertical error} > 15 \text{ m}) = 2.7 \times 10^{-7}$ . This result is very close to the desired probability of  $1 \times 10^{-7}$  as specified in the RNP tunnel concept [6-10], although it does not necessarily include the effects of all possible ranging errors. With  $R_C = 5 \times 10^{-5}$ , the corresponding fatal accident risk per approach (from the ionospheric integrity threat alone) is  $3.35 \times 10^{-8}$ . This is still about 30 times higher than what would be desirable for a fully certified system, which points out the need to gather more complete data from which to reduce the underlying rare-event uncertainty that drives the final risk estimate. Also, the use of  $\text{RPE} = 15$  meters seems overly conservative for approaches that are not supposed to get closer than 200 feet to the ground. The risk multiplier used here most likely suits a considerably higher error limit.

## 6.5 WAAS Isolation/Correction Algorithms

Another RAIM approach that may be especially suitable for WAAS is based on the very probability models we have built to describe spatial decorrelation errors. Since an airborne WAAS user could possess a similar model, he or she could use it to determine the "believability" of the ionospheric grid corrections received from WAAS. Knowing the positions of visible satellite pierce points within grid cells, the user could potentially flag a satellite whose ionospheric correction looks highly uncertain. Satellites whose corrections have a sufficient uncertainty could be *isolated*, or dropped, from the observation matrix used to compute a position fix, or they could be de-emphasized as part of a weighted least squares position computation. A preliminary version of this idea has been tested and is outlined in this section.

Recall that the user will interpolate an ionospheric correction (call it  $I_j$ ) for each satellite pierce point from the broadcast grid corrections using equation (6.2) and then multiplying by the obliquity factor  $OF$  from (6.3). From this, an "uncertainty variance"  $\sigma_{uv}^2$  is computed from the four grid point estimates  $I_{jk}$  as follows:

$$\sigma_{uv,j}^2 = \sum_{k=1}^4 OF_k \frac{(I_j^2 - I_{jk}^2)}{RM_k^2} \frac{d_1 + d_2 + d_3 + d_4}{d_k} + \sigma_{ir}^2 \quad (6.10)$$

where SF, RM, and  $\sigma_{ir}$  are defined in equations (6.3, 6.5, 6.8). Note that the interpolation function (6.2) has been used to combine the likelihoods of the estimates between the PP and each grid point. A limiting threshold could be set on this statistic, or one could project this spatial decorrelation variance from the pseudorange into the vertical position error domain by using:

$$\sigma_{zs,j} = \mathbf{G}_{3j}^* \sigma_{uv,j} = \left[ (\mathbf{G}^T \mathbf{G})^{-1} \mathbf{G}^T \right]_{3j} \sigma_{uv,j} \quad (6.11)$$

where the  $[3,j]$  entry of matrix  $\mathbf{G}^*$  is the scalar projection of ranging error into vertical position error. Once we have this statistic for each satellite in view, we perform the following process of checks:

**Spatial Decorrelation Isolation Algorithm (SDIA)**

- (1) RAIM *non-available* if VDOP > 3.2
- (2) for  $i = 1, \dots, n$ , compute  $\sigma_{zs,i}$  from (6.11) and compare to threshold  $T$
- (3) if  $\sigma_{zs,i} > T$  for any  $i$  and  $n > 6$ , remove row  $i$  from observation matrix  $\mathbf{G}$  to get  $\mathbf{G}_r$
- (4) use  $\mathbf{G}_r$  to compute position fix *unless*:
  - a) VDOP for  $\mathbf{G}_r > 3.2$ , or
  - b)  $\sigma_{zs,i} > 0.75 T$  for any other  $i = 1, \dots, n$
- (5) if a) or b) holds, situation is *non-available*

Unlike the residuals RAIM algorithm, it is difficult to choose an optimal threshold  $T$  from cost-based search because  $T$  is internal to the position-fix computation and must be set before WAAS simulations are run. The results shown in Figures 6.10 and 6.11 below are for  $T = 4.5$  m, which is 30% of the vertical RPE of 15 meters.

Figure 6.10 shows the ionospheric slant delay distributions for a 10-degree WAAS grid. The solid line includes all samples, while the dashed line excludes slant ionospheric errors for satellites that are isolated by the SDIA algorithm. Note that the only visible difference is at the rare-event tails of the distribution (beyond errors of  $\pm 8$  m), where the isolated case probabilities decrease by a factor of 2-10 from the non-isolation distribution.

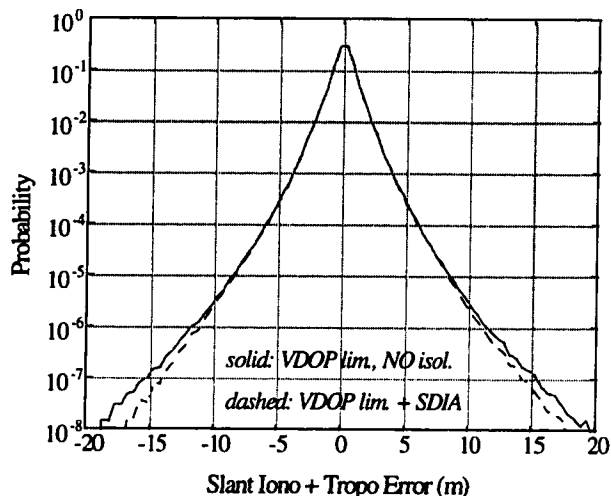


Figure 6.10: WAAS Spatial Decorrelation with Isolation

Figure 6.11 shows the resulting vertical position error distribution densities. In this case, performance improvement due to SDIA is difficult to see even for the lowest probabilities. There is some improvement, however, and the trend of decreasing risk as error increases continues longer for the SDIA case than for the no-isolation case.

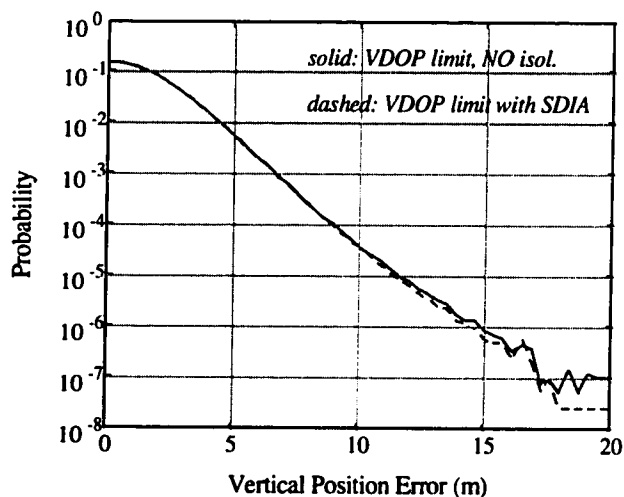


Figure 6.11: WAAS Position Errors with Isolation

Table 6.5 summarizes the results of WAAS simulations with SDIA. It appears from these results that the improvements gained by using SDIA are noticeable but not decisive. Both excessive position error probability and overall user cost decrease by about 40-50%, and note that SDIA does not hurt availability much at all despite isolating out uncertain-looking satellites in 0.5% of all cases. This suggests that further reducing the isolation threshold (searching for the minimal user cost value) would give better results.

<b>Attribute</b>	<b>With SDIA</b>	<b>Without SDIA</b>
Pr(SV isolation)	0.0051	0 (N/A)
Pr(available)	0.9975	0.9983
Pr(position error > RPE)	$1.3 \times 10^{-6}$	$2.5 \times 10^{-6}$
Overall User Cost	0.00071	0.00129

**Table 6.5: SDIA Results Summary**

In comparing these results to the optimal-RAIM results in Table 6.4, we note that the probability of exceeding RPE using SDIA is 30% larger than when using optimal WAAS due to the acceptance of more geometries as available. Also, when using SDIA, no additional residuals-check exists to catch at least 75% of these events, as occurs for optimal RAIM. Thus, the fatal accident risk for SDIA is five times greater than for optimal RAIM. On the other hand, no false alarms can occur. Overall, SDIA gives a user cost only 6% higher than optimal RAIM, while the control case cost is almost twice as much. This suggests that the current user cost function has a relatively flat surface near the optimum, given the design variables we can control. In other words, the nature and level of uncertainty in the rare-event ionosphere model for WAAS (along with the fixed WRS network) dominates the result, leaving little room for RAIM algorithms and parameter adjustments to make a substantial difference. Reducing this uncertainty by conducting extensive tests of the WAAS concept with prototype networks should lead to a substantial improvement. Parameter optimization will then provide more improvement.

SDIA is but one concept of how satellite isolation could be attempted. It is really only a starting point for the development of more sophisticated algorithms based on the idea of checking observed errors against their prior-probability likelihoods. The WAAS master station will have the role of checking the consistency of the current WAAS corrections and possibly broadcasting an estimate of the variance in the ranging correction for each satellite [8-12]. This should allow the WMS to warn users in areas whose ionosphere delay error (as measured by individual WRS's or independent monitor stations) disagrees from what would be interpolated from the WMS grid. If the WMS estimate of the gridpoint ionospheric vertical error (GIVE) is broadcast, users could include it in their own estimate of the error in their interpolated pierce point corrections in (6.11).

Error models and distributions can also be combined into the user's uncertainty estimates. In addition to updating a prior model for real-time RAIM decisions, the user can use such information in a weighted-least-squares position solution which, instead of choosing whether or not to use visible satellites, simply adjusts a Gaussian model of ranging error to give different weights to each satellite measurement. It follows that uncertainty models form an important basis for ground-user cooperation in future WAAS RAIM procedures, since the full picture of system performance can be incorporated into position calculations and integrity assurance.

## **6.6 Rare-Event Ionosphere: Conclusions**

This chapter began with the goal of extrapolating experimental measurements of 95%-bounded ionospheric errors to model uncertain rare-event probabilities for DGPS ionospheric decorrelation errors. The method of estimating the uncertain degree to which tail-probability errors are worse than predicted by a Normal distribution has allowed reasonably efficient Monte Carlo computer sampling of rare-event ionospheric errors, and combining these with samples of other ranging errors has resulted in simulations of vertical position error and other key performance measures for the proposed WAAS network. These simulations take into account the uncertainty of these extrapolations of experimental data; thus they provide the best possible model of WAAS performance as defined by the decision criteria in Chapter 1.

Our results show that this provisional 20-station WAAS network and geosynchronous satellite-augmented GPS constellation provide substantially better correction of spatial decorrelation errors for users scattered throughout the Continental U.S. than do local-area corrections. Along with the improved availability of the augmented constellation, user position fix error distributions are much improved, making possible the use of WAAS for Category I aircraft precision landing. The combination of ground integrity information and user RAIM further reduces integrity and continuity errors that could lead to penetrations of the outer tunnel proposed by the RNP [6-10].

Given the prevailing uncertainty in our error model, a definitive word regarding WAAS integrity, continuity, and availability performance is not yet possible. Spatial decorrelation experiments using more reference stations will provide better and more useful experimental bases for our predictions in the future. As of mid-1996, results from larger WADGPS networks, such as the FAA NSTB and the DCI EAGLE network, are becoming available. Data from these networks should help fill in the gaps in our current knowledge. Simulations and flight tests which consider other error modes (such as communications, software, and other non-physical integrity threats) also need to be conducted. The simulation model developed in this research should be expanded to consider other error sources that are unavoidably built into the WAAS architecture. In addition, a better picture of the relationship between RPE and fatal accident probability for Category I approaches needs to be developed to make RAIM optimization more realistic.

The most important question remaining in the practical design of an operational WAAS is the marginal utility of (1) improved ground and/or user RAIM software, (2) additional remote monitors, and (3) additional satellites to augment ranging and/or communication. A combination of these should be sufficient to meet all requirements for Category I aircraft precision approach. Improved RAIM algorithms are being formulated along with new ideas for ground/user RAIM cooperation. This capability may allow the currently proposed 20-station WAAS network to meet the precision landing requirement for integrity without further hardware augmentation.

The development of a flexible tool for analysis of the overall performance of WAAS system architectures cannot depend solely on the limited integrity-based studies shown here. Performance as defined by PICA includes so-called "normal" situation performance, in which the system is operating under normal conditions with no system failures. Under these conditions, it is possible to evaluate user performance over a wide area without extensive Monte Carlo simulation. The development of such a method is described in the next chapter, and it will provide a normal-case model from which to carry out RAIM optimization using the methods developed here.

## Chapter 7: Wide Area Augmentation System Coverage Prediction

*If on the other hand he went to pay his respects to The Door and it wasn't there . . . what then? The answer, of course, was very simple. He had a whole board of circuits for dealing with exactly this problem, in fact this was the very heart of his function. He would continue to believe in it whatever the facts turned out to be, what else was the meaning of Belief? The Door would still be there, even if the Door was not.*

-- Douglas Adams, spoken by Dirk Gently, *Dirk Gently: Holistic Detective Agency*

### 7.1 Introduction

As mentioned in the last chapter, studies of corrected pseudorange accuracy and satellite availability for WAAS prototypes have demonstrated the potential to achieve vertical position accuracies of 2-3 meters at specific user sites [8-7,18]. However, it remains unclear how results achieved at specified user locations can be extended to predict user accuracies across a wide geographic area, which is the purpose of WAAS. Therefore, while it is apparent that WAAS has the potential to provide Category I accuracy for aircraft approach and that baselines of hundreds of kilometers are possible, it is not clear just how many wide-area reference stations (WRS's) are needed to meet accuracy requirements over the entire geographic spread of users.

The experimental results obtained to date suggest designing networks of WRS's that are both numerous enough and in close-enough proximity to ensure that users are "close enough" to the nearest WRS with near-certainty. WAAS networks are currently laid out with the help of only these preliminary guidelines. This chapter develops a method for expanding on our current understanding of WAAS performance capability by projecting linear least-squares error *covariance* matrices from WRS's through the master station (WMS) to users. Although this approach approximates the detailed workings of the WRS's and WMS, it succeeds in modeling the underlying error uncertainties; thus the results are indicative of the performance that can be achieved by a canonical WAAS architecture. It allows user accuracy predictions over a wide geographic area for any proposed network of WRS's and geosynchronous satellites (which communicate WAAS corrections and serve as redundant ranging sources).

Section 7.2 describes this new “coverage prediction” methodology in detail, including the assumed ranging error variances and least-squares covariance equations. Section 7.3 gives accuracy predictions for the Stanford network of three WRS’s in California and Nevada, along with variations which examine the effects of adding a WRS in Hawaii. Results for a much larger network, the proposed FAA WAAS testbed (NSTB), are then shown in Section 7.4, along with studies of variations of the baseline NSTB network. User coverage predictions (under normal conditions) can also be used as a basis for failure-event extrapolation in Monte Carlo simulations. Section 7.5 merges these predictions with the simulation-based optimal RAIM approach of the previous two chapters to produce system-wide integrity predictions for a proposed WAAS in Europe.

Overall, the results demonstrate the utility of WAAS accuracy predictions over large areas, and they point out possible weaknesses in the coverage provided by the WAAS networks mentioned above. This allows sensitivity studies to be conducted that promise to greatly aid the process of designing future wide-area systems. The tools developed here will be used in the following chapter to evaluate the system-level utility of proposed WAAS architectures and to make possible the optimal design of WAAS network augmentations.

## **7.2 Covariance Analysis of Normal WAAS Performance**

### **7.2.1 Overview of WAAS Simulation Approach**

The coverage prediction approach used here is based on the solution of least-squares covariance equations for given GPS and WRS geometries. Accuracy predictions for large geographic areas are generated by a simulation which samples a large number  $N_i$  (from 1,440 to 10,000) of separate satellite geometries using the GPS orbit model introduced in Section 5.1. The orbit model includes three of the proposed Inmarsat geosynchronous satellites located over the Equator at longitudes 178° E (POR), 18° W (AOR-E), and 55° W (AOR-W) respectively, all of which can be seen by at least some users in the Continental U.S. Spacecraft are assumed to always be failure-free, and geometries are sampled using both random Uniform and constant-interval time updates as

explained in Section 5.1.1. The former approach (with 10-minute mean updates) provides a non-repetitive collection of geometries. In the latter case, one day (1436 trials) of satellite orbits is cycled through, and it is possible to keep track of the *maximum outage duration* at each user site, which is the longest period in which the predicted vertical position accuracy does not meet a specified vertical position accuracy requirement. The ILS requirement of 4.1 m (95%) for Category I precision approach is considered to be ambitious for WAAS by many, but a more flexible requirement of 7.6 m has been selected for the WAAS Category I requirement based on the RNP Tunnel methodology [6-10].

For each satellite geometry, the matrix of direction cosines to each visible satellite  $G_w^i$  is computed for each WRS location  $i$  (using a  $5^\circ$  mask angle). At this stage, the ranging observation errors for each satellite visible at each WRS are computed from the RMTSA model in Section 7.2.3, and the large WRS ionosphere covariance matrix  $P^P$  can be computed element-by-element. The program then cycles through a pre-specified grid of user locations separated by  $1-3^\circ$  in latitude and longitude which comprises the coverage area or “service volume” to be examined. For each user location in this grid, the geometry matrix  $G_u$  is computed, and two separate processes of covariance propagation are carried out in parallel. The first is the *clock/ephemeris* error for satellites in view of the user (using a  $7.5^\circ$  mask angle) based on the WRS’s that can see that satellite and can thus provide clock/ephemeris corrections. The second is *ionospheric* spatial decorrelation projected from the pierce points observed by each WRS to the WMS, which fits a set of predictions to a grid, and finally to each user.

Covariance projections from ionosphere and RMTSA error sources are brought together into a single pseudorange error covariance matrix  $P_v^*$  for each user. Using the matrix  $P^{SV}$  of clock/ephemeris errors for each satellite, the weighted least-squares position error covariance  $\hat{P}_x$  is computed, and the vertical position error variance (assumed to be Gaussian) is given by the [3,3] ( $3^{rd}$  row,  $3^{rd}$  col.) entry of this final matrix. The vertical error result for each geometry is stored in a histogram for that user, as is the Vertical DOP for the satellite geometry visible to that user. “Availability” in this case is defined as the percentage of geometries for which a given user’s vertical one-sigma error (given by

$\sqrt{\hat{\mathbf{P}}_x[3,3]}$  ) is within the ILS one-sigma requirement of 2.05 meters (or one can use the 3.8-m WAAS requirement from the RNP). Geometries for which this requirement is exceeded are deemed “non-available”, and if this state persists over more than a few minutes, a measurable outage period for Category I approaches results.

Figure 7.1 gives a conceptual flow chart for this covariance propagation method. Sections 7.2.3 and 7.2.4 describe the algorithms for computing clock/ephemeris and ionosphere covariances, and Section 7.2.2 gives the receiver-specific RMTSA ranging error model used. The relevant equation numbers used at each step in the procedure are referenced in the figure.

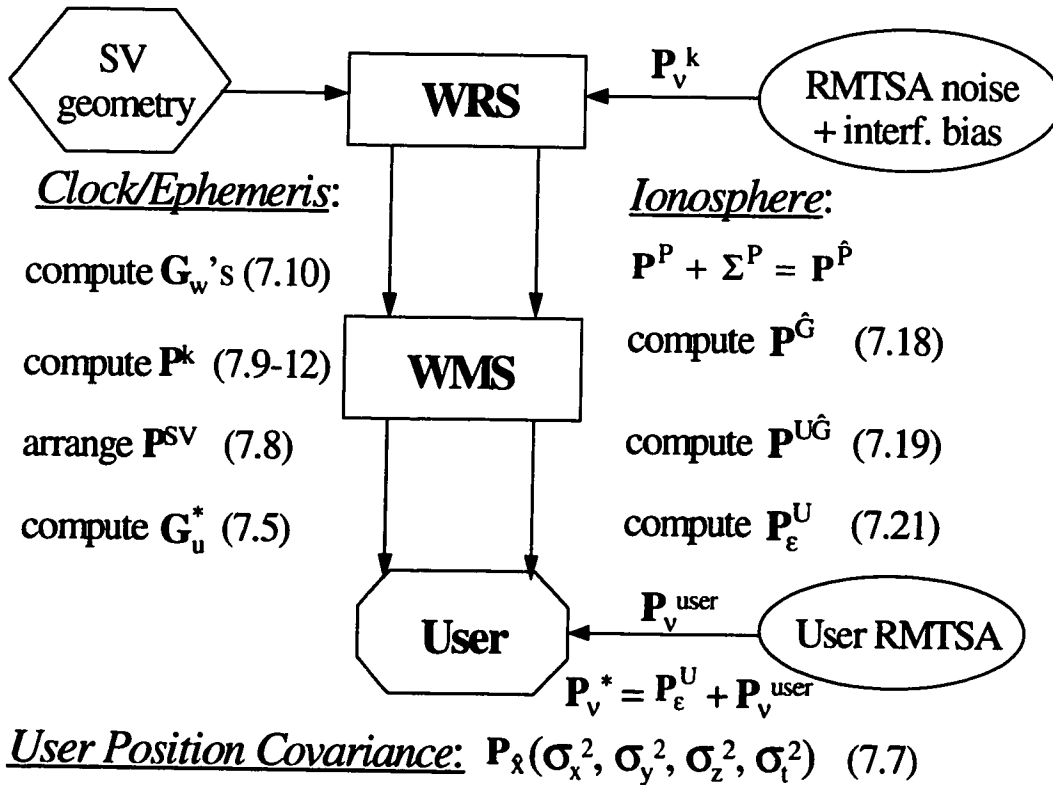


Figure 7.1: WAAS Covariance Overview

### 7.2.2 RMTSA Ranging Error Model

As described in Section 6.1, WAAS employs GPS corrections computed by a network of reference stations to remove most of the satellite-based errors that exist

without differential corrections. This process is modeled by the propagation of clock/ephemeris and ionosphere covariances described here. However, it is corrupted by receiver and location-specific errors at each WRS and at the user. These errors are collectively labeled RMTSA, representing (R)eceiver noise, (M)ultipath, (T)roposphere, and remaining (S)elective (A)vailability errors. In this analysis, these errors are assumed to be independent; thus their combined variance can be obtained by taking the sum of the individual noise variances. The assumption that the error components are Gaussian is not strictly required here, but that assumption is also normally made [7-13].

Noise Source	WRS Error (m)	User Error (m)
receiver noise	0.33	0.50
SA latency	not applicable	0.20
multipath	$0.20 / \tan(\epsilon)$	$0.30 / \tan(\epsilon)$
troposphere	$0.07 / \sin(\epsilon)$	$0.20 / \sin(\epsilon)$

**Table 7.1: One-Sigma RMTSA Errors**

Table 7.1 gives the individual error standard deviations for each individual RMTSA error source. These models are more detailed than the simple user model numbers in Table 6.1, and they incorporate estimates of the accuracies obtained from the Stanford WAAS network. In general, WRS errors are assumed to be of smaller magnitude compared to a generic single-frequency user receiver. In the tropospheric case, each WRS is assumed to have its own weather station, whereas users will have to apply a tropospheric correction model. User SA latency error assumes a fast-correction average age of 10 seconds [8-5,7]. Note that both multipath and troposphere errors are functions of the elevation angle  $\epsilon$  from the user to the satellite in question. This represents the additional “slant” delay due to atmospheric effects and the greater GPS signal reflection magnitude for lower satellite elevations. RMTSA error terms are introduced into the covariance equations as diagonal  $n \times n$  matrices (where  $n$  satellites are in view) in which each diagonal element contains the combined RMTSA variance for that satellite.

With the exception of receiver noise, the error sizes in Table 7.1 are not meant to be especially conservative. Instead, they represent a reasonable estimation of the errors that will be experienced by WAAS-certifiable equipment. More conservative noise distributions have been examined in other studies to show the position accuracy degradation that might result from “looser” hardware standards [7-13].

Although the RMTSA distributions assume “snapshot” (one point in time) WAAS corrections and position solutions, some representation of the *carrier smoothing* conducted by each reference station (but not necessarily by a user) is desirable. The Stanford WAAS network implements carrier smoothing to reduce WRS observation errors. A Hatch/Eshenbach filter is used to average code pseudorange observations with much more precise carrier information (which has only 1-2 mm of noise) [6-7]. When a WRS first sees a given GPS satellite, the averaging process begins, leading to a reduction in the magnitude of receiver and multipath noise as a function of the time that satellite has been observed (without a cycle slip). Receiver noise has a short correlation time, but multipath takes much longer to average out. We now use an abstract exponential-decay model which gives a combined noise reduction factor *NRF* defined as follows:

$$NRF = \exp\left(-\frac{t_{obs}}{\tau_{cs}}\right) \quad (7.1)$$

where the decorrelation time constant for receiver noise and multipath,  $\tau_{cs}$ , is conservatively estimated to be 60 minutes. This single-pole estimate is based on overall WRS noise reduction observed from the Stanford WAAS. Identification of this noise with a two-pole model (one decorrelation time for receiver noise and another longer one for multipath) is now being considered to give a more detailed representation of carrier smoothing, but (7.1) suffices for the level of detail modeled in the covariance propagation algorithm. In the simulation code, the cumulative time  $t_{obs}$  is tallied as the satellite geometry is updated. The receiver and (elevation-dependent) multipath standard deviations (from the RMTSA) are then reduced by multiplying by *NRF* computed from (7.1).

### 7.2.3 Clock/Ephemeris Covariance Prediction

**7.2.3.1 Covariance Prediction Equations:** The covariance model developed here separates clock/ephemeris errors from all other ranging error sources. In practice, clock and ephemeris corrections can be handled separately by the master station, which uses a single difference to separate fast-changing clock error from slowly-varying satellite position errors in order to keep up with the Selective Availability error within the satellite clock [8-7]. However, from a snapshot perspective, it is more convenient to combine them into a single problem within the system covariance model and to treat SA error as the error remaining after WAAS fast corrections are applied. As a result, both residual ionosphere errors (Section 7.2.4) and RMTSA errors (Section 7.2.2) appear as external error inputs ( $v_u$ ) to the clock/ephemeris prediction equations (see (7.7)). In this formulation, user pseudorange error  $\Delta\rho$  is expressed as a combination of user ephemeris error ( $\Delta\mathbf{r}$ ), user clock error ( $\Delta\mathbf{b}$ ), and errors in the WAAS correction ( $\Delta\mathbf{x}^{sv}$ ) as follows:

$$\Delta\rho = \mathbf{G}_u \begin{bmatrix} \Delta\mathbf{r}_u \\ \Delta\mathbf{b}_u \end{bmatrix} - \tilde{\mathbf{G}}_u \Delta\mathbf{x}^{sv} + v_u \quad (7.2)$$

where  $\mathbf{G}_u$  is the user direction-cosine satellite geometry matrix (size  $K \times 4$ , where  $K$  is the number of satellites in view) and  $v$  is a term representing a Gaussian white-noise model of the other error sources. The WAAS clock/ephemeris correction error vector  $\Delta\mathbf{x}^{sv}$  of size  $2K \times 1$  has the following elements:

$$\Delta\mathbf{x}^{sv} = \begin{bmatrix} \Delta\mathbf{r}^{k_1} \\ \Delta\mathbf{B}^{k_1} \\ \vdots \\ \Delta\mathbf{r}^K \\ \Delta\mathbf{B}^K \end{bmatrix} \quad (7.3)$$

where  $\Delta\mathbf{r}^k$  represents the (unknown) error in the 3-dimensional ( $x,y,z$ ) WAAS ephemeris correction for satellite  $k$ , and  $\Delta\mathbf{B}^k$  represents the error in the (scalar) clock correction for

that satellite. The augmented matrix  $\tilde{\mathbf{G}}_u$  of size  $K \times 4K$  is constructed by placing each row of  $\mathbf{G}_u$  along a diagonal pattern as follows:

$$\tilde{\mathbf{G}}_u = \begin{bmatrix} -1_u^{k_1} & 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & -1_u^{k_2} & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -1_u^K & 1 \end{bmatrix} \quad (7.4)$$

Since (7.2) conceptually includes all GPS error sources, we can proceed to find position error covariance from it. Covariance propagation automatically incorporates a form of measurement *weighting* based on measurement error and prior-estimate covariance matrices. Since weighting the position fix calculation by numbers which represent the relative quality of each input measurement will by definition improve user accuracy, a weighted-least-squares user computation is assumed in this chapter and the next. This modifies the unweighted solution (5.9) in a straightforward way. The “weighted” pseudoinverse definition for an *overdetermined* system (more satellites in view than unknowns) is [8-17]:

$$\mathbf{G}_u^* = (\mathbf{G}_u^T \mathbf{W}^{-1} \mathbf{G}_u)^{-1} \mathbf{G}_u^T \mathbf{W}^{-1} \quad (7.5)$$

The user position fix error is then calculated as in (5.9):

$$\hat{\mathbf{x}}_u = \mathbf{G}_u^* \Delta \rho = (\mathbf{G}_u^T \mathbf{W}^{-1} \mathbf{G}_u)^{-1} \mathbf{G}_u^T \mathbf{W}^{-1} \Delta \rho \quad (7.6)$$

The weighting matrix  $\mathbf{W}$  is given by the user pseudorange error covariance matrix  $\mathbf{P}_v^*$  to be defined next. Using the definition of covariance, the final user position error covariance  $\mathbf{P}_x$  is given by:

$$\mathbf{P}_{\hat{\mathbf{x}}} = \mathbf{G}_u^* \tilde{\mathbf{G}}_u \mathbf{P}^{SV} \tilde{\mathbf{G}}_u^T (\mathbf{G}_u^*)^T + \mathbf{G}_u^* \mathbf{P}_v^* (\mathbf{G}_u^*)^T \quad (7.7)$$

where  $\mathbf{P}_v^*$  is the  $K \times K$  diagonal matrix of user noise variances for each satellite due to ionosphere and RMTSA errors. It is calculated from the combined results of the ionosphere covariance model (Section 7.2.4) and the RMTSA model (Section 7.2.2). The use of this matrix as  $\mathbf{W}$  in (7.6) above should give a minimum-variance estimate of the true user position, but in this model, we assume that  $\mathbf{W}$  cannot be known perfectly by the user. Instead, each entry of  $\mathbf{W}$  is calculated by sampling from a Normal distribution whose mean is the corresponding entry of  $\mathbf{P}_v^*$  and whose standard deviation is a fixed ratio of that mean (usually 25%). We have found that the WAAS accuracy results are not very sensitive to this “uncertainty ratio” if it is below 75%.

$\mathbf{P}^{SV}$  in (7.7) is a user-specific block diagonal matrix of  $K$  separate  $4 \times 4$  covariance matrices constructed as follows:

$$\mathbf{P}^{SV} = \begin{bmatrix} \mathbf{P}^{k_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathbf{P}^K \end{bmatrix} \quad (7.8)$$

These separate covariance matrices  $\mathbf{P}^{k_1}, \dots, \mathbf{P}^K$  each represent the post-correction covariance in  $x$ ,  $y$ , and  $z$  components of spacecraft ephemeris, as well as the clock error in  $t$ . If no WAAS corrections are available (because this satellite is not visible to any WRS), this matrix has diagonal terms representing uncorrected errors in GPS ephemeris messages (the first three entries) and uncorrected SA error (the fourth entry) as follows [8-17]:

$$\mathbf{P}_{\text{no}}^k = \begin{bmatrix} \frac{|\Delta r_k|^2}{3} & 0 & 0 & 0 \\ 0 & \frac{|\Delta r_k|^2}{3} & 0 & 0 \\ 0 & 0 & \frac{|\Delta r_k|^2}{3} & 0 \\ 0 & 0 & 0 & \frac{|\Delta B_k|^2}{3} \end{bmatrix} = \begin{bmatrix} 9 & 0 & 0 & 0 \\ 0 & 9 & 0 & 0 \\ 0 & 0 & 9 & 0 \\ 0 & 0 & 0 & 100 \end{bmatrix} (m^2) \quad (7.9)$$

If WAAS corrections are available from at least one WRS (assume  $M > 0$  WRS's can see a given satellite), we form the matrix  $\mathbf{G}_w$  as follows for each satellite  $k$ , where  $k = 1, 2, \dots, K$ ;  $m = 1, 2, \dots, M$ :

$$\mathbf{G}_w = \begin{bmatrix} \mathbf{1}_{m_1}^k & -1 \\ \mathbf{1}_{m_2}^k & -1 \\ \vdots & \vdots \\ \mathbf{1}_M^k & -1 \end{bmatrix} \quad (7.10)$$

where  $\mathbf{1}_m^k$  is the direction-cosine vector from WRS  $m$  to satellite  $k$ . Note that the clock/ephemeris error can be estimated directly by weighted least squares. However, a direct solution (such as in equation (7.6)) tends to be too sensitive to measurement errors for underdetermined ( $M < 4$ ) and exactly-determined ( $M = 4$ ) cases. Instead, we include a prior state covariance matrix  $\Lambda$  and solve the problem as a Kalman-filtered measurement update to the prior covariance. The inclusion of prior information helps ensure that a wildly unlikely solution is not force-fit by “naive” least squares [8-17].

Although normally the matrix  $\Lambda$  would be the same as the prior (no corrections) covariance in (7.9), we have the freedom to vary the diagonal-entry “weights” of the matrix. We do this by multiplying the  $x$ ,  $y$ , and  $z$  variances by a factor of 10, which effectively increases the weight we assign to the WRS measurements. Further, we make the prior clock variance very large since it will in practice be estimated separately from

ephemeris, and we wish to avoid having the prior clock variance significantly affect the ephemeris estimates. Our final choice for the state prior covariance is:

$$\Lambda = \begin{bmatrix} 90 & 0 & 0 & 0 \\ 0 & 90 & 0 & 0 \\ 0 & 0 & 90 & 0 \\ 0 & 0 & 0 & 1 \times 10^6 \end{bmatrix} (m^2) \quad (7.11)$$

The measurement-update equation for the WAAS-corrected clock/ephemeris covariance  $\mathbf{P}_w^k$  is given by:

$$\mathbf{P}_w^k = \Lambda - \Lambda \mathbf{G}_w^T (\mathbf{G}_w \Lambda \mathbf{G}_w^T + \mathbf{W}^w)^{-1} \mathbf{G}_w \Lambda \quad (7.12)$$

where the measurement-error weighting matrix  $\mathbf{W}^w$  is equal to the  $M \times M$  matrix of RMTSA error variances for each WRS that can see satellite  $k$  [8-17].

**7.2.3.2 Computer Implementation Procedure:** In our computer code, for each updated GPS satellite geometry, the satellite visibility for each WRS is computed and formatted as shown by the matrix  $\mathbf{G}_w$  in (7.10). At this point, the algorithm loops through the grid of user locations that are affected by the WAAS corrections. For each user,  $\tilde{\mathbf{G}}_u$  and  $\mathbf{G}_u^*$  are computed from  $\mathbf{G}_u$  using (7.4) and (7.5).  $\mathbf{P}^k$  for each satellite in view of this user is then computed using (7.9) or (7.12) as required; thus giving the block-diagonal matrix  $\mathbf{P}^{sv}$  (7.8).

At the same time, the parallel computation of the user pseudorange error covariance matrix  $\Sigma_u$  for ionospheric errors is computed using the algorithm in Section 7.2.4. The effects of RMTSA user ranging errors are computed (see Section 7.2.2), arranged into a  $K \times K$  diagonal matrix, and summed with  $\Sigma_u$  to get the final user error covariance  $\mathbf{P}_v^*$ . Finally, the user position covariance  $\mathbf{P}_x$  for this geometry is computed using (7.7). This process is repeated for each user location. We store the resulting

vertical position error variance  $\sigma_z^2$ , given by the third diagonal element of  $\mathbf{P}_{\hat{x}}$ , for each user in a histogram unique to that user location for post-sampling analysis. We also compare the vertical error variance at a given user for each sample geometry to the ILS and RNP tunnel accuracy requirements to compute the overall probability that the resulting vertical position error is within the specified accuracy.

#### 7.2.4 Ionospheric Grid Error Propagation Model

The ionospheric grid covariance propagation model, run in parallel with the clock/ephemeris model detailed above, directly propagates ionospheric uncertainty from the WRS's to the WMS, which projects the combined corrections onto an artificial "grid" of ionospheric pierce points separated by 5, 10, or 15 degrees of latitude and longitude (depending on the size of the coverage area). This covariance is then propagated to each user depending on the location of his pierce points within the grid structure. Note that this grid concept is similar to that originally proposed by MITRE [8-6], but it uses weighted least-squares fits to interpolate corrections within the grid rather than simple linear inverse-weighted interpolation, in which users compute a weighted average of the corrections of the four surrounding grid points based on the distance to each of them.

It is basically assumed by this model that the *vertical* ionospheric delay (i.e. not affected by satellite elevation) observed by a GPS receiver through a given pierce point, denoted as  $\mathbf{I}^{\hat{p}}$ , can be expressed as the sum of two independent random variables: the *true delay* (denoted as  $\mathbf{I}_p$ ) and the *observation error* (denoted as  $\Sigma_p(\epsilon)$ ). This basic relation holds for the WRS's, the WMS, and the user (with the appropriate subscripts). Note that  $\Sigma_p(\epsilon)$  is elevation-dependent because it represents the RMTSA error terms discussed in Section 7.2.2.

The propagation of ionosphere error covariance is executed in several distinct steps as follows:

**7.2.4.1 WRS Covariance:** The vector  $\mathbf{I}^{\hat{P}}$  contains the measured (via dual-frequency receivers at each WRS) vertical ionospheric delay at each pierce point observed by all WRS's:

$$\mathbf{I}^{\hat{P}} = \begin{bmatrix} \mathbf{I}_{m=1}^{\hat{P}} \\ \mathbf{I}_{m=2}^{\hat{P}} \\ \vdots \\ \mathbf{I}_{m=M}^{\hat{P}} \end{bmatrix} \quad (7.13)$$

where  $\mathbf{I}_m^{\hat{P}}$  represents the observed delays for each satellite visible to WRS  $m$ ,  $m = 1, \dots, M_{\text{rms}}$  (the total number of WRS's). The vector  $\boldsymbol{\sigma}^P$  is a sample of the RMTSA noise distribution. Its covariance, denoted as  $\boldsymbol{\Sigma}^P$ , is a diagonal matrix whose variances are computed by summing the RMTSA numbers in Section 7.2.2 with a constant *interfrequency bias*. This bias represents miscalibrations between the L1 and L2 broadcast frequencies on each GPS satellite as well as between WAAS reference receivers. Based on current bias estimates for prototype networks, we add the square of a conservative bias deviation estimate of  $\sigma_{\text{bias}} = 0.75$  m to each diagonal entry of  $\boldsymbol{\Sigma}^P$ . Operational WAAS systems should be able to reduce this by as much as 50% [8-2].

The true ionospheric delay covariance, denoted as  $\mathbf{P}^P$ , is a function of the geographic separation between the WRS pierce points and the assumed “base” delay variation at a given point over time. The base delay standard deviation  $\sigma_b$  is taken to be 2.8 meters, which represents the long-term variation (after correction by the Klobuchar model [6-14]) in vertical ionospheric delay at a given mid-latitude location. This estimate is conservative; it represents a near-peak activity time in the 11-year solar cycle. Based on curve fits to the absolute and relative ionospheric delay data used in Chapter 6 [8-6,10], the following covariance formulas are used for each entry in  $\mathbf{P}^P$  (which will consequently have no zeros in it):

$$\sigma(d) = \frac{I_{\text{base}} I_{\text{mult}}}{OF_{\text{mean}}} \frac{R_{\text{slope}}}{R_{\text{base}}} d \quad (7.14)$$

where:

$d$  = distance between reference and observed PP's (km)

$R_{\text{base}}$  = base separation in Klobuchar data = 348 km

$R_{\text{slope}}$  = linear slope with distance in data fit = 0.542

$I_{\text{base}}$  = base ionosphere decorrelation for ( $d = 0$ ) = 0.417 m

$I_{\text{mult}}$  = ionosphere decorrelation multiplier = 2.0

$OF_{\text{mean}}$  = mean obliquity factor in data = 1.763.

If  $d > D$ , where  $D = 1200$  km, the effective  $\sigma(d)$  is modified by an exponential curve such that  $\sigma(d) \leq \sigma_b$ , where  $\sigma_b^2$  is the “base” ionospheric variance over time for a given pierce point. This normally varies with time and geomagnetic position, but we use the conservative value  $\sigma_b = I_{\text{mult}} \sigma_{\text{norm}} = 2.8$  m. The resulting equation is:

$$\sigma(d > D) = \sigma(D) + \sigma_{\text{marg}} \left[ 1 - \exp \left( \frac{-[\sigma(d)_{\text{fit}} - \sigma(D)]}{\sigma_{\text{marg}}} \right) \right] \quad (7.15)$$

where  $\sigma_{\text{marg}} = \sigma_b - \sigma(D)$ , and  $\sigma(d)_{\text{fit}}$  is the variance given by (7.14) for  $d > D$ . Figure 7.2 shows a plot of  $\sigma(d)$  for  $0 \leq d \leq 3000$  km. This variance from one PP to another is converted to a covariance entry  $\bar{\sigma}_{i,j}^2$  in matrix  $\mathbf{P}^P$  using the Gaussian bivariate relation, which assumes probabilistic independence between the spatial decorrelation and the long-term point variation expressed by  $\sigma_b$ :

$$\bar{\sigma}_{i,j}^2 = \sigma_b^2 \sqrt{1 - (\sigma_{i,j}(d)/\sigma_b)^2} \quad (7.16)$$

Summing the covariances  $\mathbf{P}^P$  and  $\Sigma^P$  gives  $\mathbf{P}^{\hat{P}}$ , the final WRS ionospheric measurement covariance matrix.

Ionospheric decorrelation results observed from (7.16) indicate that the bivariate gaussian assumption provides very high correlation of ionospheric delay over the baselines typical of WAAS networks. As a result, this assumption has been relaxed by simply increasing the

exponent of the term which multiplies  $\sigma_b^2$  on the right-hand-side of (7.16) from 0.5 to 1.0. Although this is an arbitrary change, the spatial decorrelations resulting from it look more reasonable (and more conservative). For example, the amount of correlation for an 800-km pierce-point separation decreases from 97.7% to 95.5% of the base (no separation) correlation once this change is made.

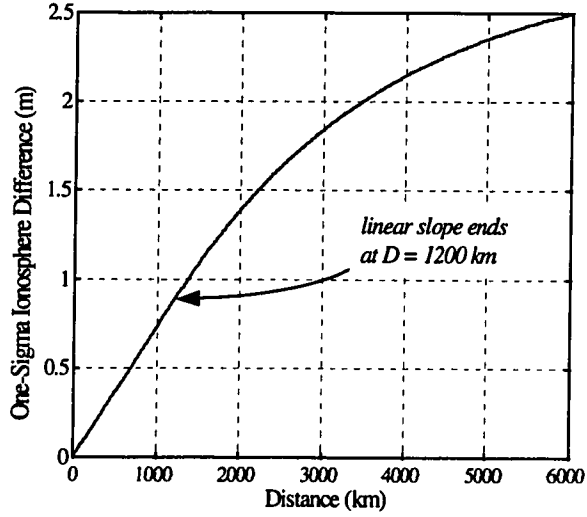


Figure 7.2: Ionosphere Error Spatial Decorrelation

**7.2.4.2 WMS Covariance:** The Wide Area Master Station (WMS) is where the pierce-point measurements of the individual WRS's are collected to form  $\mathbf{P}^{\hat{P}}$ . The WMS then relates the pierce point delay measurements  $\mathbf{I}^{\hat{P}}$  to those that would exist at the WMS grid points by solving for the optimal choice of mapping matrix  $\mathbf{A}$  using least-squares:

$$\mathbf{A}^T = \mathbf{P}^{GP} [\mathbf{P}^{\hat{P}}]^{-1} \quad (7.17)$$

where  $\mathbf{P}^{GP} = \mathbf{P}^{G\hat{P}}$  is the "true" covariance (computed using (7.14-16) between the fixed WMS ionosphere grid points (G) and the set of WRS pierce points (P). From the definition of covariance, we can simplify the application of the mapping matrix to get the final grid point covariance:

$$\mathbf{P}^{\hat{G}} = \mathbf{A}^T \left[ \mathbf{P}^{G\hat{P}} \right]^T = \mathbf{P}^{G\hat{P}} \left( \mathbf{P}^{\hat{P}} \right)^{-1} \left[ \mathbf{P}^{G\hat{P}} \right]^T \quad (7.18)$$

where  $\mathbf{P}^{\hat{G}}$  is the resulting covariance of the WMS ionosphere grid points. Note that WRS ionosphere measurement errors are represented in  $\mathbf{P}^{\hat{P}}$  and get propagated into  $\mathbf{P}^{\hat{G}}$  through this calculation.

**7.2.4.3 User Covariance:** WAAS users (located on a grid of predetermined points covering a wide geographic area) use the transmitted WMS estimates of vertical ionosphere delay at each ionosphere grid point  $\mathbf{I}^{\hat{G}}$  to project ionosphere errors at their pierce points to the satellites they can see. As with WMS propagation, we use least-squares optimal mapping. From the mapping matrix  $\mathbf{A}$  in the previous section, we can find the user-to-grid covariance  $\mathbf{P}^{UG}$ :

$$\mathbf{P}^{UG} = \mathbf{P}^{UP} \mathbf{A} = \mathbf{P}^{UP} \left( \mathbf{P}^{\hat{P}} \right)^{-1} \left[ \mathbf{P}^{G\hat{P}} \right]^T \quad (7.19)$$

where  $\mathbf{P}^{UP}$  is the user-to-WRS pierce point covariance, computed (as with  $\mathbf{P}^{G\hat{P}}$ ) using the spatial decorrelation equations (7.14-16). The definition of covariance and a second mapping matrix  $\mathbf{B}$  allows us to project the error in the WMS grid to the user's pierce points as follows:

$$\begin{aligned} \mathbf{B}^T &= \mathbf{P}^{UG} \left[ \mathbf{P}^{\hat{G}} \right]^{-1} \\ \mathbf{P}^{\hat{U}} &= \mathbf{B}^T \mathbf{P}^{\hat{G}} \mathbf{B} = \mathbf{P}^{UG} \left[ \mathbf{P}^{\hat{G}} \right]^{-1} \left( \mathbf{P}^{UG} \right)^T \end{aligned} \quad (7.20)$$

where  $\mathbf{P}^{\hat{U}}$  is the user ionosphere error covariance due to the WAAS fitting algorithm. By our previous assumptions, the basic equation for user ionosphere vertical error covariance  $\mathbf{P}_\varepsilon^U$  is:

$$\mathbf{P}_\varepsilon^U = \mathbf{P}^U - \mathbf{P}^{\hat{U}} = \mathbf{P}^U - \mathbf{P}^{UG} \left[ \mathbf{P}^{\hat{G}} \right]^{-1} \left( \mathbf{P}^{UG} \right)^T \quad (7.21)$$

where  $\mathbf{P}^U$  is the overall covariance, which can be computed from the underlying decorrelation between the user's pierce points using (7.14). The last step is to multiply the covariances in the final result,  $\mathbf{P}_\varepsilon^U$ , by *obliquity* factors to convert from vertical to slant pierce point delay. The obliquity factor  $OF$  for a given satellite observation is computed from equation (6.3) of the last chapter, where  $\varepsilon$  is the elevation angle from user to satellite. To get the final user slant ionosphere error covariance,  $\Sigma_U$ , terms on the diagonal  $([i,i])$  of  $\mathbf{P}_\varepsilon^U$  are multiplied by  $(OF_i)^2$ , while off-diagonal  $([i,j])$  terms are multiplied by the obliquity factor product  $(OF_i OF_j)$ . This final WAAS user error matrix  $\Sigma_U$  is added to the user RMTSA covariance  $\mathbf{P}_v^{\text{user}}$  to get the “external noise covariance” matrix  $\mathbf{P}_v^*$  in (7.7). From this, the user position error covariance  $\mathbf{P}_x$  is computed.

**7.2.4.4 Numerical Challenges:** Because the matrix operations in (7.17-21) imply the numerical calculation of inverses of large matrices, numerical difficulties can arise even though the C code does not need to explicitly calculate each inverse. The WRS pierce-point covariance  $\mathbf{P}^{\hat{P}}$  is usually the most difficult to work with, as it is of size  $M_P \times M_P$ , where  $M_P$  is the total number of pierce points observed by all WRS's. The C code uses double-precision floating-point representation (16 digits), but cases arise where the *condition number* of  $\mathbf{P}^{\hat{P}}$ , which is the ratio of the largest to the smallest of the eigenvalues, or *singular values*, of  $\mathbf{P}^{\hat{P}}$ , increases above  $10^{12}$ , meaning that the covariance matrix approaches singularity to double precision. Since singular matrices are not

permitted in the covariance- propagation equations, a numerical breakdown in the calculations may result [3-12].

The source of near-singularity is the fact that most of the pierce points are located in a relatively small area. Their correlation to each other is thus quite high relative to the underlying ionospheric variation and observation errors (that is, the diagonal terms). The covariance matrix thus becomes closer to singular (i.e. invalid) as the spatial decorrelation in the off-diagonal terms decreases. We have found that the matrix condition deteriorates as more pierce points are included, since more observations reduce the effective decorrelation. For WAAS architectures (such as the FAA NSTB) where  $M_p$  generally exceeds 100, a *parsing* procedure must be employed to ensure adequate numerical conditioning.

Parsing refers to the reduction of the size of  $\mathbf{P}^{\bar{p}}$  by merging pierce points that are close together into a single observation (for ionosphere purposes only).  $\mathbf{P}^{\bar{p}}$  is computed normally using (7.14-16) for all pierce points (PP's), then a loop goes through the list of pierce points backward:  $j = M_p, \dots, 1$ . If PP  $j$  is within a certain threshold distance  $T_d$  from another PP  $i$ , where  $i < j$ , PP  $j$  is parsed out of  $\mathbf{P}^{\bar{p}}$  and PP  $i$  is moved to a new location:

$$\mathbf{x}_{\text{new}}^i = \frac{a \mathbf{x}_{\text{old}}^i + b \mathbf{x}_{\text{old}}^j}{a + b} \quad (7.22)$$

where  $\mathbf{x}$  is the relevant PP location in  $x, y, z$  Earth-centered, Earth-fixed coordinates; while  $a$  and  $b$  are integer weighting factors which count the number of PP's that have been "parsed" into the current PP locations  $i$  and  $j$ . The effect of  $a$  and  $b$  in (7.22) is to amplify the importance of "pierce points" that have resulted from previous "mergings". The base error variance of the new PP (entry  $[j, j]$  of  $\mathbf{P}^{\bar{p}}$ ) is reduced by:

$$(\sigma_{\text{new}}^i)^2 = (\sigma_{\text{old}}^i)^2 N_{\text{dec}} = (\sigma_{\text{old}}^i)^2 \frac{1}{a+1} \quad (7.23)$$

This simply models the noise “averaging” which takes place as more and more WRS’s observe a PP in the same general area. The off-diagonal terms of  $\mathbf{P}^{\bar{\mathbf{P}}}$  are not reduced.

### 7.3 WAAS Coverage Prediction Results

#### 7.3.1 Output Histograms and Computed Results

The covariance propagation method described here has been applied to a variety of existing and planned WAAS networks. Much of the algorithm development was done with variations of the 3-WRS Stanford WAAS (including some with one or two additional WRS’s). The RMTSA and ionosphere covariance parameters were derived with results from the Stanford setup in mind. The algorithm is the same (with the exception of adding parsing from Section 7.2.4.4) for the much larger FAA WAAS testbed networks, and we have been able to obtain projections for hypothesized WAAS systems located anywhere in the world.

Using MATLAB to plot the results of our C programs, the results can be presented as 2-D contours or 3-D surface plots. The program outputs any or all the following data for each user, and the overall results are plotted over a map of the relevant user geography:

- 95% and 99% vertical position error
- 95% and 99% values for *one-sigma* vertical position error
- 95% and 99% vertical dilution of precision (VDOP)
- *Availability* =  $\Pr(\text{vertical error } \sigma < 1\sigma \text{ requirement})$
- Maximum outage duration (in minutes)
- 95% and 99% vertical ionosphere delay error (UIVE)
- 95% and 99% WMS grid ionosphere vertical error (GIVE)
- 95% and 99% user differential range error (UDRE)

Note that vertical ionosphere delay error corresponds to “User Ionosphere Vertical Error” (UIVE) in the FAA’s terminology [8-5] and is contained in the elements of  $\mathbf{P}_e^u$  from (7.21) (before adjusting for obliquity). “User Differential Ranging Error” (UDRE) for each satellite in view is given by the diagonal elements of the matrix product  $\tilde{\mathbf{G}}_u \mathbf{P}^{sv} \tilde{\mathbf{G}}_u^T$  contained in (7.7) which projects the clock/ephemeris correction error covariance into the user’s geometry.

To compute the above results, histograms of vertical position error, non-availability (1 if  $\sigma_v > \sigma_v^{\text{req}}$ , 0 otherwise), and outages (add time elapsed since last geometry if in a continuing outage state) are updated at each site in the user location grid for each sample geometry. For each geometry, UIVE and UDRE are incremented  $K$  times at each user site with the computed UIVE or UDRE for each of the  $K$  satellites visible to that user for that geometry. GIVE is stored for each geometry at each point in the WMS ionosphere grid.

Computation of overall 95% and 99% position error cutoffs requires convolving the outer-loop histogram of error variance (over all satellite geometries) with the Normal distributions implied by the relevant variances. Specifically, for each  $\sigma_v$  bin in the position error histogram for a given user, a new distribution of convolved position error  $\sigma_v^*$  is incremented by the number of increments in the original  $\sigma_v$  bin times the following cumulative Gaussian probability for each  $\sigma_v^*$ :

$$P_{\text{gauss}}^{\text{cdf}}(\sigma_v^*) = 0.5 \left[ 1 - \text{erf} \left( \frac{\sigma_v^* / \sigma_v}{\sqrt{2}} \right) \right] \quad (7.24)$$

where  $\text{erf}(\bullet)$  denotes a numerical approximation of the error function defined in [3-12]. The resulting new histogram on  $\sigma_v^*$  is normalized by the number of increments in the original one as the final step. The 95% and 99% vertical error *sigma* cutoffs are taken from the histograms before convolution with the underlying Gaussian errors. All of the 95% and 99% results are computed for each grid location which has its own (original or

convolved) histogram simply by summing the cases from the worst result on down until 1% or 5% of the observations have been accounted for.

### 7.3.2 Results for Stanford WAAS Network

As introduced in Section 6.1, the current Stanford experimental WAAS network consists of a master station in the Durand building at Stanford University and three reference stations located in San Diego, CA., Arcadia, CA., and Elko, NV. An additional reference receiver exists at Stanford but is used in a “passive” mode to evaluate the quality of the position solution provided by the three WRS’s [8-18]. Plans for additional WRS’s in Hawaii and Alaska and possible relocations of the three existing WRS’s have been discussed, motivating us to compute predictions for some of these variations as well. The results shown here for the basic Stanford network do not use WRS carrier smoothing, but they also use a slightly more conservative RMTSA model (Section 7.2.2) and use the ionosphere covariance equation (7.16) without modification. These changes tend to cancel out, giving similar results for the small Stanford WAAS network.

Figure 7.3 shows a 2-D contour plot of 95% (convolved) vertical position accuracy over a user area bounded by 18° to 48° North latitude and 108° to 128° West longitude, divided into one-degree increments. A 10-degree WMS ionosphere grid is used, extending from 20° to 50° N latitude and 105° to 135° W longitude. The WRS locations are denoted on the map surface by an ‘x’ in all plots.

The results for 95% user accuracy show that the optimal performance region is relatively flat and encloses an area considerably wider than the area between the three WRS’s. Vertical accuracy degrades gracefully as one gets farther away from this zone,

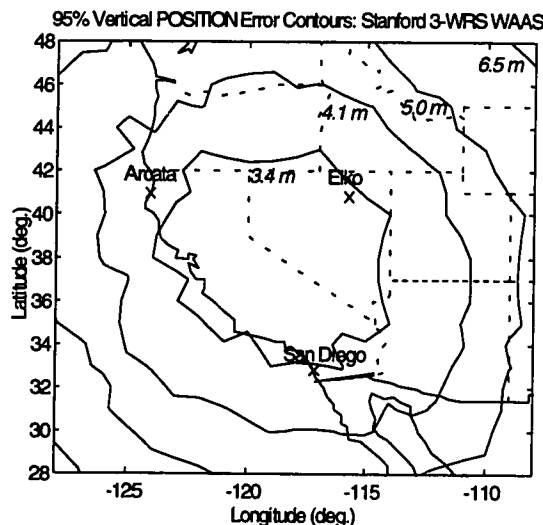
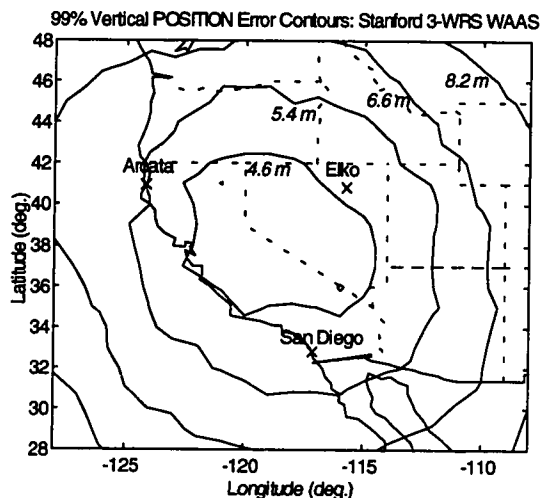


Figure 7.3: Stanford WAAS 95% Position Accuracy

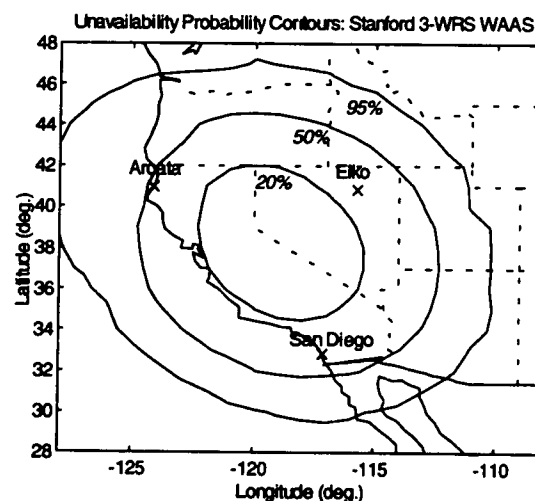
although the falloff becomes substantial if one gets far enough away, because the WRS observations have little leverage to correct the (probably different set of) satellites a far-away user can see. However, accuracy sufficient to meet the 4.1-meter 95% ILS requirement is provided for users as far away as the Oregon-Washington border and Phoenix, AZ. Note that all users use weighted least-squares to get their position fixes; thus far-away users

will optimally de-emphasize the satellites for which good WAAS corrections are not available in their position solutions. These results agree with preliminary flight-test data collected from flight tests using the Stanford WAAS at Palo Alto Airport, Livermore, CA., and Truckee (near Lake Tahoe) [8-18]. In the Palo Alto tests,  $1\sigma$  RMS vertical error was observed to be 1.7 meters, which agrees well with a 95% prediction of 3.4 m.

Figure 7.4 gives the 99% vertical position accuracy results for the same system. Note that if the underlying overall distribution were Gaussian, 95% cutoffs would represent  $2\sigma$  and 99% would result in approximately  $3\sigma$ , giving an expected multiplication of 1.5. In these results, while the contour shapes are similar, the apparent 99% performance for most user locations is better than 1.5 times the 95% performance shown in Figure 7.3. The same phenomenon is also observed for the 95% and 99% UIVE contour plots, where in the



**Figure 7.4: Stanford WAAS 99% Position Accuracy**



**Figure 7.5: Stanford WAAS User Availability wrt 4.1-m ILS Cat. I Requirement**

majority of cases the 99% results are only 10-20% worse. These results suggest a non-Gaussian or truncated-Gaussian underlying distribution, but given that our method is based on a concatenation of “Normal” conditions only, we are not confident that this is a fair representation of reality. Rather, we believe that a Gaussian distribution is valid out to 95% and that a projection of performance from  $2\sigma$  to  $3\sigma$  (i.e., multiplying by 1.5) would be more reasonable than using the 99% histogram results.

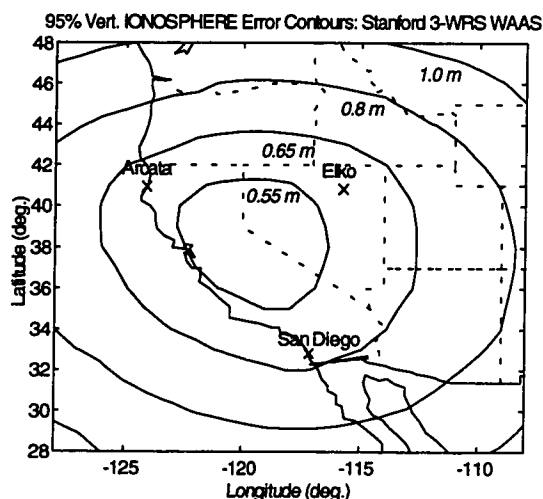


Figure 7.6: Stanford WAAS 95% UIVE

Figure 7.5 gives a 2-D contour plot of availability, relative to the 4.1-meter ILS Category I requirement, for the Stanford WAAS. Recall that “availability” here is different from that produced by geometry parsing in GPS integrity monitoring in Chapters 4-5; it is now simply a measure of how often (over *time*) the ILS accuracy requirement is met for a given user. Availability of the required *accuracy* at the beginning of an operation is thus shown. Clearly, for the ILS requirement, a small 3-WRS network cannot provide 4.1-m 95% accuracy more than 90% of the time even in the optimal coverage area, but the proposed 7.6-m limit for WAAS Cat. I approaches gives 100% availability over a region larger than that enclosed by the three WRS’s.

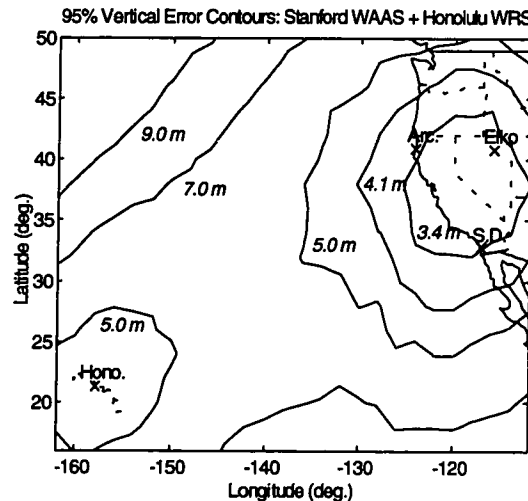
Figure 7.6 gives a 2-D contour plot of 95% UIVE, or user vertical ionosphere delay error. These contours conform very well to the shape of the WRS geographic distribution, and once again, the degradation as one moves away from CA-NV is well-behaved and gradual. At Stanford, the predicted 95% error of under 0.6 meters is difficult to verify using our current database of observed ionospheric errors, but it seems reasonable. We are devising a new method for storing ionosphere errors so that a better comparison can be made. As mentioned above, the 99% UIVE plot gives only slightly higher errors, suggesting that the use of only “normal” conditions may lead to overly optimistic predictions for rare-event ionospheric decorrelations.

### 7.3.3 Stanford WAAS and Hawaii

Various augmentations to the Stanford WAAS have been proposed. It is now expected that some of these will be incorporated into the FAA WAAS testbed (NSTB) as it is built. One key augmentation we have studied (not planned for NSTB) would be the addition of a fourth reference station in Honolulu, Hawaii. It would attempt to both provide Category I accuracy to the Hawaiian Islands and augment the

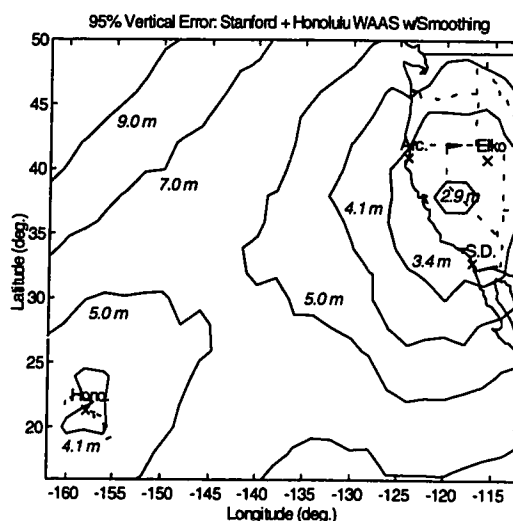
quality corrections for users in the western part of CONUS. Note that the implementation of WRS carrier smoothing can make a substantial difference here, as most GPS satellites will be visible to the Hawaii WRS first. Once these satellites are visible to users in CONUS, the Hawaii WRS will have had sufficient time to filter its own observations, and the WMS would then give the Hawaii measurements more weight in its determination of the corrections to be sent to all users. Thus, carrier smoothing is implemented for this case, along with the RMTSA in Table 7.1 and the relaxed ionosphere correlation discussed in Section 7.2.4.1. The results of these model changes can then be compared to results from the older model to help determine the cumulative effect of these changes.

Figure 7.7 is a plot of 95% vertical position errors for this 4-WRS network over an enlarged user area from 16° to 50° N latitude and 112° to 162° W longitude, divided into 2° increments. A 15-degree ionosphere grid from the Equator to 60° N and from 105° to 165° W was used for the WMS. This figure shows the results under the *old model* (no smoothing) used for the Stanford WAAS network. Given the lack of carrier smoothing and the advantages it provides, the considerable distance between Honolulu and the other three WRS's (over 4000 km) is why the Hawaii WRS only slightly improves CONUS performance (compared to Figure 7.2). The Hawaii WRS is not quite able to provide ILS Category I accuracy for the area encircling the Hawaiian Islands, but the 95% error is still



**Figure 7.7: Stanford/Hawaii Position Error (unsmoothed)**

under 5 meters, which would easily meet the relaxed RNP requirement [6-10]. Significant improvement is attainable by improving the ionosphere calibration (i.e., reducing  $\sigma_{\text{bias}}$  in Section 6.1) [8-2]. Note that the degradation of position accuracy over the Pacific between CONUS and Hawaii is graceful enough to support very precise enroute navigation for commercial air routes between the two.



**Figure 7.8: Stanford/Hawaii Position Error  
(WRS carrier smoothing)**

Figure 7.8 shows the same position error results for the case where the new WRS error models are used as described above. The improvement is substantial; notice the much larger area included in the 5.0-meter contour relative to Figure 7.6. Also note the area around the Hawaiian Islands which is now within a 4.1-meter contour and thus meets the ILS requirement at least 95% of the time. Because Hawaii is so far from CONUS, the issue of WRS service redundancy arises, since Category I accuracy would be lost if the Honolulu WRS had to shut down. If the amount of WRS internal hardware redundancy is not deemed adequate to give near-100% availability over time, a second WRS would be needed in Hawaii. The result would be better nominal accuracy in the Hawaiian Islands region when all WRS's are operational.

#### **7.3.4 FAA National Satellite Test Bed (NSTB) Results**

In advance of the planned introduction of an operational WAAS in the late 1990's, the FAA plans to build a testbed of about WRS's spread throughout CONUS. As introduced in Section 6.1, this testbed, known as *NSTB*, is designed to experimentally evaluate WAAS user performance and to provide guidance to the development of the operational system [8-11]. Stanford University is participating in developing, testing, and installing WRS and WMS hardware and software for the NSTB. The covariance methods

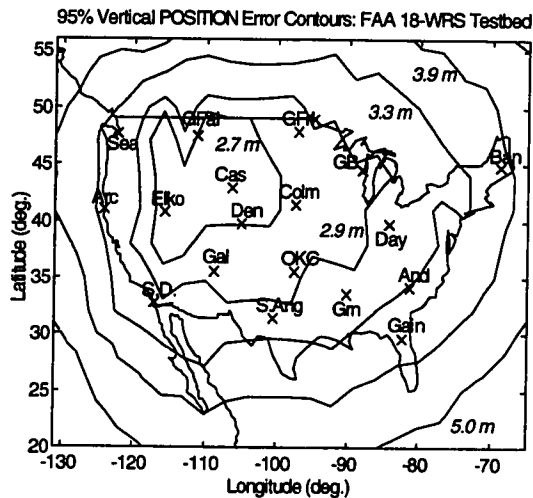
detailed here can also be used to predict NSTB user accuracies over the entire CONUS region.

The WRS network design of the NSTB has evolved in several directions in the past year. Section 7.3.4.1 presents results for the 18-WRS network proposed originally. Holes in this network over the Eastern half of the U.S., along with logistical problems in setting up reference stations at specific sites, have motivated

several changes in the CONUS network. Section 7.3.4.2 presents results, focused on users in CONUS, for the latest revision (as of October 1995) to the CONUS network. At the same time, Canadian and Mexican participation in the NSTB is being pursued, and at least one Canadian WRS is already operational. Section 7.3.4.3 thus expands the coverage area to include most of Mexico and the southern provinces of Canada while presenting results for a network of 24 WRS's, where 3 each are placed in Mexico and southern Canada.

Because each of the WRS's in the NSTB sees an average of 8-10 satellites at a time, parsing of the ionosphere covariance matrix  $\mathbf{P}^{\hat{P}}$  (see Section 7.2.4.4) is necessary. Using a separation threshold of  $T_d = 500$  km, the number of rows and columns in  $\mathbf{P}^{\hat{P}}$  for the 18-WRS NSTB to be presented in Section 7.3.4.1 decreases from an average of 160 to 54. With this level of parsing, no numerical matrix singularity problems have been observed in any of the proposed NSTB networks (although  $T_d$  is increased to 600 km for networks with more than 20 WRS's).

**7.3.4.1 Original NSTB Network Results:** Figure 7.9 shows a 2-D contour plot of 95% vertical position errors for a 3° user grid from 20° to 56° North latitude and 65° to 131° West longitude. The WRS ionosphere grid goes from the Equator to 75° North and from



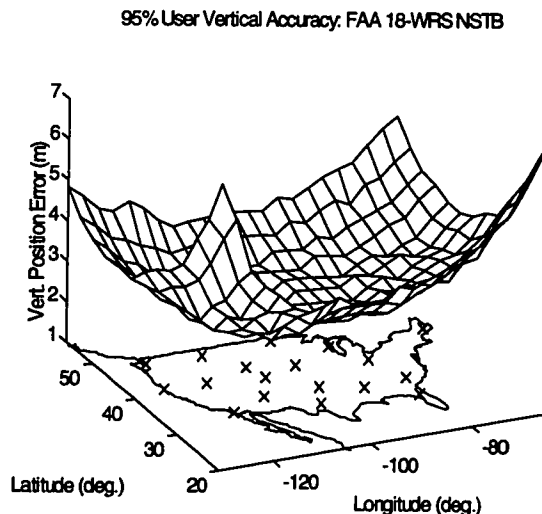
**Figure 7.9: FAA 18-WRS NSTB 95% Accuracy**

50° to 140° West in 15° increments. These original results were derived from the older error models without carrier smoothing or ionospheric decorrelation adjustment.

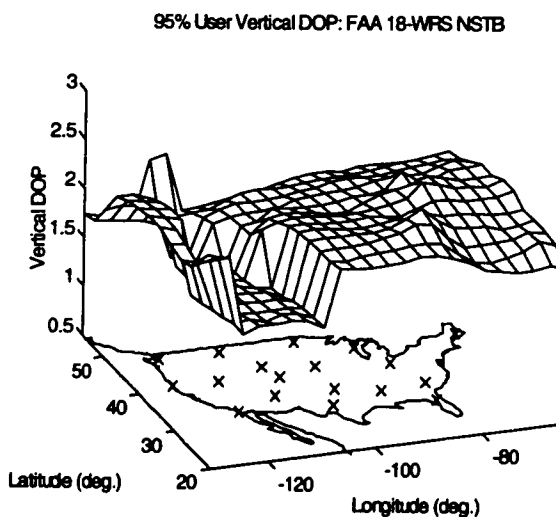
From this plot, it is clear that 18 WRS's are fully capable of providing approach accuracy close to the ILS Category I requirement throughout CONUS. A WAAS 95% "accuracy floor" of about 2.5 m is evident in the Western U.S., where the density of WRS locations is highest. WRS density in the Eastern U.S. is much lower, however, giving 95% errors of around 3.5 m. Nevertheless, all of the Eastern U.S. coastline still achieves 95% vertical accuracy of four meters or better; so it can be considered to be acceptable.

Figure 7.10 is a 3-D representation of the same position accuracy results. This plot gives a nice overview of the shape of the 95% accuracy surface, or "bowl", over CONUS. As would be expected from the lack of WRS's outside CONUS, the gradient of this surface gets worse near the edges of the user grid. Users off to the southeast or southwest of CONUS have the worst 95% accuracy, but it is still within 7 m.

Figure 7.11 is a similar 3-D plot of the 95% user VDOP surface. It is shown to display the effect on user geometry of the added geosynchronous satellites. Note that users in the



**Figure 7.10: FAA NSTB 3-D 95% Accuracy Plot**



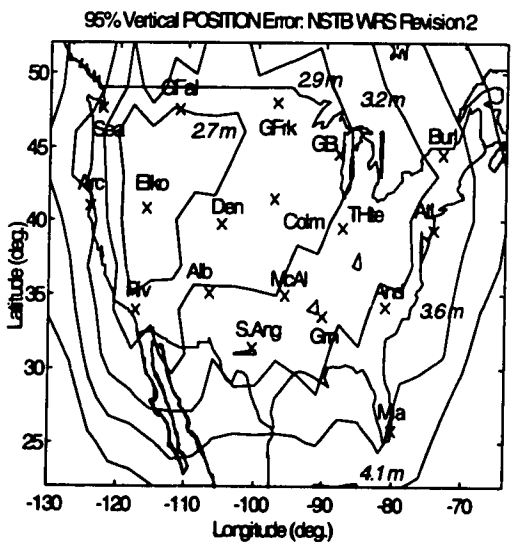
**Figure 7.11: FAA NSTB 3-D 95% VDOP Plot**



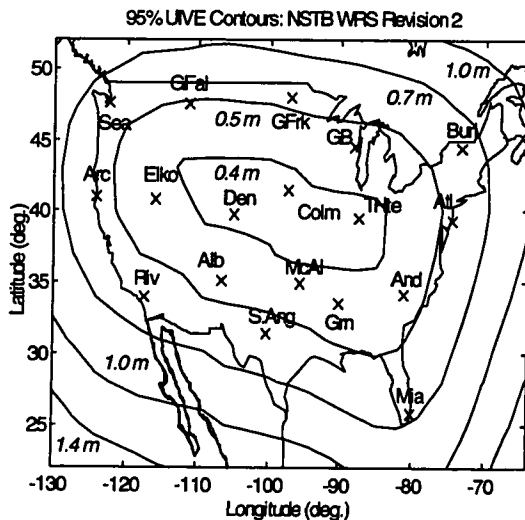
the number of WRS-observed pierce points on all sides of the user pierce points, the contours on this plot are quite regularly-shaped ovals which tend to mirror the layout of the outermost WRS locations. In the center of CONUS, the presence of WRS observations on all sides improves 95% UIVE to below 0.4 m. UIVE increases gradually and predictably as one moves away from the center, but all of CONUS is under 0.8 m. This demonstrates how well a nationwide WAAS can fit the underlying ionosphere states under “normal” ionospheric conditions. In solar maximum years, these errors should be, at worst, no more than twice that shown here [8-10].

**7.3.4.2 Revised CONUS NSTB Results:** Several revisions have been made to the NSTB WRS network within CONUS since the original layout shown in the last section was proposed. Some of the WRS locations in the interior of CONUS have moved a few hundred miles from their former locations when alternate sites proved more convenient places to “set up shop.” But many of the changes have been motivated by coverage analyses (including the results of the last section) which indicated the unbalanced nature of the WRS layout and the undesirable preponderance of WRS’s in the Western U.S. As a result, the redundant-looking WRS in Casper, WY. has been removed, and a WRS has been added at the FAA Technical Center in Atlantic City, N.J. As of October 1995, the location of one WRS (originally in Dayton, Ohio) is in limbo -- both Lansing, MI. and Terre Haute, IN. are being considered. The total number of WRS’s inside CONUS remains at 18.

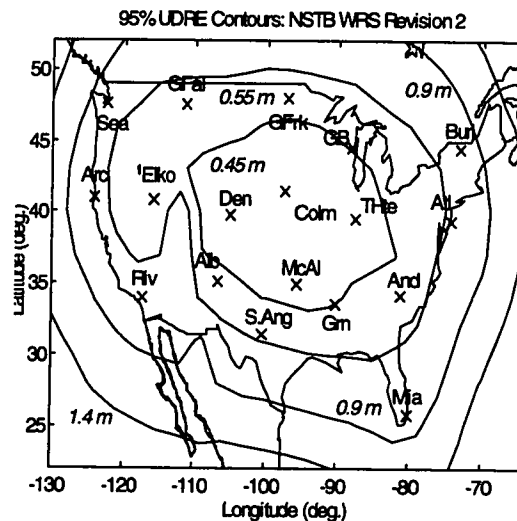
Figure 7.13 shows contours of 95% vertical position error for this revised WRS network, where the Dayton WRS has been placed in Terre Haute. The revised WAAS error models (including carrier smoothing) are used here, and the user grid is tightened to focus on CONUS. Note first that the



**Figure 7.13: Revised NSTB 95% Accuracy**



**Figure 7.14: Revised NSTB 95% UIVE**



**Figure 7.15: Revised NSTB 95% UDRE**

WRS network is now more evenly spread throughout CONUS, and the Eastern Seaboard is more densely represented by the addition of the WRS in Atlantic City. The revised noise model helps improve user performance, but this improved WRS layout is the primary reason why contours of 3.2-meter accuracy stretch much further east than in the original layout shown in Figure 7.9. Now all of CONUS is enclosed within a 95% vertical error contour of about 3.7 meters, which is an improvement of 7.5% from Figure 7.9.

Figure 7.14 shows contours of 95% UIVE for this revised network. Once again, contours of UIVE are regular and predictable, but the equivalent-error contours of this plot enclose slightly more territory than those in Figure 7.12 for the original network. Note that in Figure 7.12, the 0.7-meter UIVE contour, while otherwise very similar to that in Figure 7.14, extends only to the Florida panhandle. Moving the Gainesville, FL. WRS down to Miami, as shown in Figure 7.14, extends this 0.7-meter contour to include all of Florida. Also note that this WRS relocation does not hurt UIVE over the Southeastern U.S., as the 0.5-meter contour has in fact extended further southeast in Figure 7.14.

Figure 7.15 shows, for the first time in this chapter, contours of 95% UDRE for the revised network. We see that for most of CONUS, 95% UDRE is similar to 95% UIVE as shown in Figure 7.13. However, the deterioration of UDRE as one moves further away from CONUS is somewhat more rapid than it is for UIVE. This is a tendency that is more evident for smaller WAAS networks, such as the European WAAS

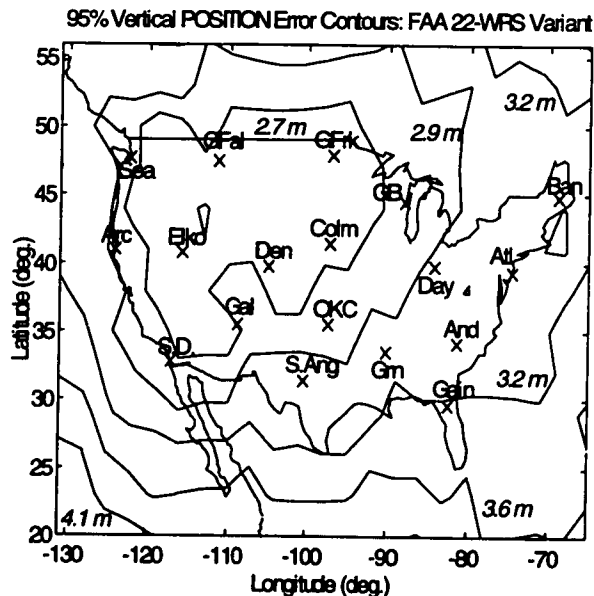
to be introduced in Chapter 8. Users outside the optimal coverage area of a given WAAS network will experience worsened UIVE due to fewer WRS pierce points being available to fit the ionosphere gridpoint estimates from which these users interpolate their corrections. Worsened UDRE, on the other hand, is due to (1) the fact that geometric cancellation of the relatively high ephemeris variances in  $P^{SV}$  in (7.7) is less effective as users move further away from the nearest WRS, and (2) the relatively poor WRS visibility of satellites near the edge of the network. WRS carrier smoothing also will have less of an effect, since an “edge” user will not have access to highly-smoothed satellite corrections as often.

The apparent variation between UIVE and UDRE spatial degradation points out the limits of the coverage prediction methodology. We have less reason to be confident that the error models developed here will hold as predictably for far-away users. Most testing of WAAS networks to date has focused on users in the primary coverage area; thus little data exists for users outside this area. The results shown here for normal conditions do make sense, but there is more statistical uncertainty for distant regions as a result. In addition, the effects on far-away users of unfavorable rare-normal or failure conditions could be magnified further than they would be for users in the primary coverage area.

Further testing from networks such as the NSTB should improve our understanding of far-away WAAS behavior by supplying more data for future error analysis.

#### 7.3.4.3 NSTB Plus “Four Corner”

**Arrangement:** The rapid worsening of the user accuracy performance at the edges in Figures 7.9, 7.10, and 7.13, along with the greater error uncertainty in these regions, motivates a study of adding

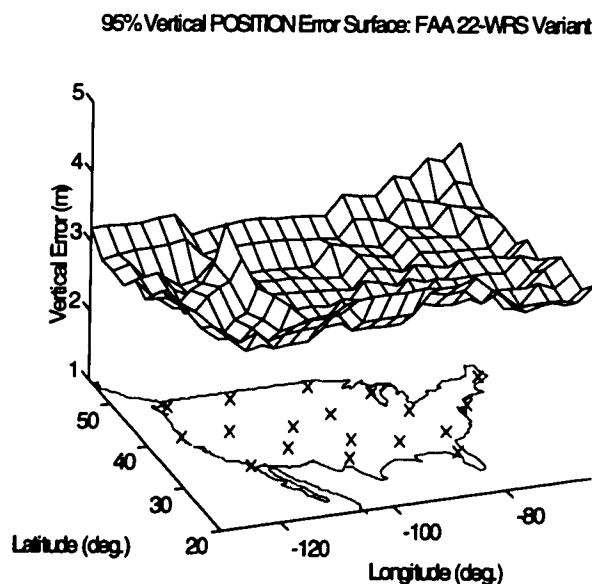


**Figure 7.16: 22-WRS Variant 95% Accuracy**

additional WRS's outside CONUS to improve WAAS performance near the borders and to provide coverage of a broader subset of North America. This might also include coverage of Alaska and Hawaii. In theory, performance near the borders of CONUS would be improved most by adding stations well outside CONUS in a "four-corners" arrangement. Figure 7.16 shows 95% user vertical accuracy for a proposed NSTB variant in which the original 18-WRS CONUS network is augmented by four additional WRS's in Honolulu, HI., Anchorage, AK., Puerto Rico, and Gander, Newfoundland. In addition, the Casper WRS shown in Figure 7.9 has been moved to Atlantic City, N.J. as shown in Figure 7.13. Note that the user and ionosphere grids are not changed; i.e., they remain limited to CONUS and environs. The original (non-smoothed) error model was used for this study.

Comparing the accuracy contours in Figure 7.16 to those in Figure 7.9 (for the original CONUS-only NSTB) shows that substantial improvement has indeed been gained near the borders. In Figure 7.16, all of CONUS is within a 95% vertical error of 3.5 meters, and only Florida is outside of 3.2 meters. This represents a 12.5% improvement over the CONUS-only network. Also note that very little was lost by the removal of the Casper WRS, while much was gained along the East Coast by the addition of a WRS in Atlantic City.

Figure 7.17 shows this improvement more clearly by providing a 3-D surface plot of 95% vertical accuracy. This surface is much flatter than the one in Figure 7.10; thus we can see that the accuracy gradient near the edges of the user grid has been lessened considerably. The worst degradation in Figure 7.17 can be seen in the southwesterly direction, since the Hawaii WRS is far from CONUS.

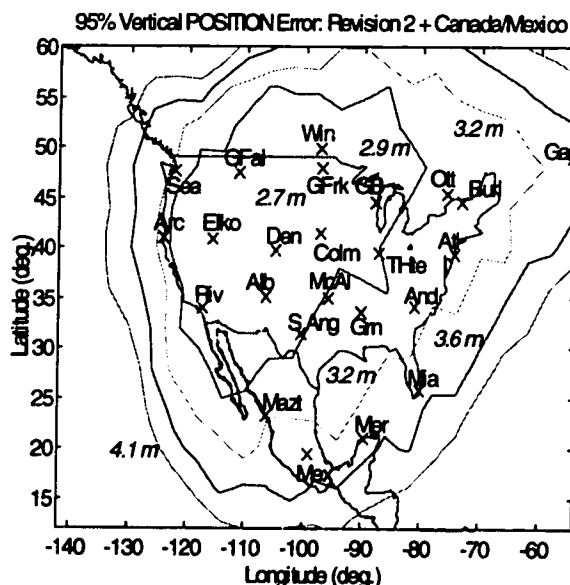


**Figure 7.17: 22-WRS Variant 3-D Accuracy Plot**

**7.3.4.4 NSTB Plus Canada/Mexico:** The FAA Satellite Program Office has recognized the desirability of augmenting its WAAS networks with WRS's outside of CONUS, but instead of the "four-corners" approach, the NSTB has considered adding stations within both Canada and Mexico to the CONUS network shown in Figure 7.13. Canada has set up NSTB-compatible WRS's in Ottawa and Winnipeg that are already supplying observations to the FAA Technical Center. A third Canadian WRS in Gander, Newfoundland (one of the "four-corners" sites) is also in the works. Similarly, inquiries have been made to authorities in Mexico as to whether they would also want to participate. Three WRS locations in Mexico have been proposed: Mexico City, Merida (in the Yucatan), and Mazatlan.

Figure 7.18 shows 95% vertical accuracy contours for a combined 24-WRS network which combines the latest CONUS distribution from Figure 7.13 with these six additional WRS locations. The WRS-carrier-smoothing error model has been used here. Note that the user grid has been extended to 12-60° N latitude and 54-142° W longitude, divided into 4° increments. Figure 7.19 emphasizes the vertical accuracy gradients at the edges of the extended coverage area by plotting the same user accuracy on a 3-D surface.

In Figure 7.18, the addition of three Canadian WRS's has extended sub-3.2-meter 95% vertical accuracy well northward into the southern provinces of Canada and has commensurably improved accuracy in the northern half of CONUS (when compared to Figure 7.13). The presence of two of these WRS's in Eastern Canada has also extended this level of coverage to all of the Eastern Seaboard from South Carolina northward. The three WRS's in Mexico extend the 3.2-meter contour well south of the



**Figure 7.18: NSTB+Canada/Mexico 95% Accuracy**

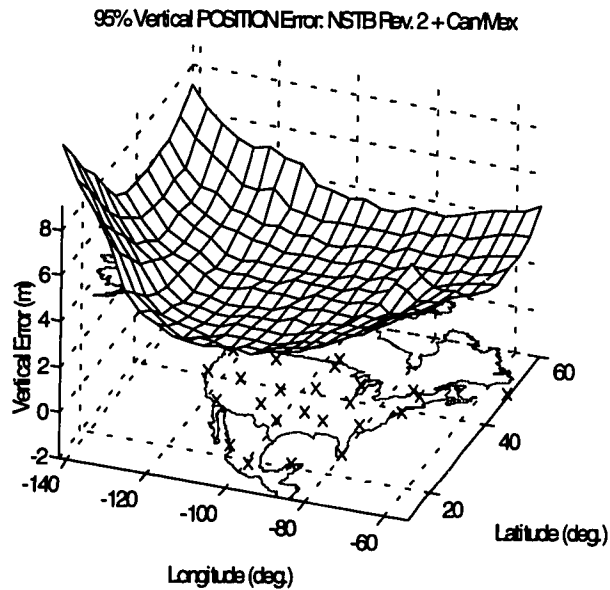
U.S.-Mexico border, and most of Mexico proper is enclosed within a 95% vertical error of 3.6 meters.

However, since the NSTB is an experimental network, it is not necessarily designed with optimal WRS placement in mind. This is evident in the locations of the three proposed WRS's in Canada (those in Mexico are clearly limited by the restricted breadth of the country).

The Gander WRS is a considerable help, as shown in Figure 7.16 (where

it is part of the four-corners variant). The Ottawa WRS, on the other hand, is clearly redundant, being quite close to the WRS in Burlington, VT. (which was moved from Bangor, ME. in the revision shown in Figure 7.13). The Winnipeg WRS is also within 300 km of the one in Grand Forks, N.D., and there is a gaping lack of coverage of Northern British Columbia and Alberta since Winnipeg is the westernmost Canadian WRS. In Figure 7.19, it is clear that the negative accuracy gradients in the direction of Hawaii and Puerto Rico are unavoidable without placing additional WRS's in those locations. The similar degradation over BC and Alberta could be reduced by simply moving the Ottawa and Winnipeg WRS's to Timmins (northern Ontario) and Edmonton, giving considerably better coverage of Canada (south of 60° N latitude) with no additional cost. The location of the Canadian WRS's shown in Figure 7.18 was decided by the Canadian aviation authorities based on existing sites with the necessary infrastructure; thus optimal coverage is not really the goal.

Regarding West Coast coverage, it is important to note that even for regions which have WRS's on the coastline, user accuracy degrades rather rapidly just offshore. This holds true for all of the cases studied which do not have additional WRS's in Alaska and Hawaii, and it is noticeably different from what is observed on the East Coast. Since most



**Figure 7.19: Canada/Mexico 3-D Accuracy Plot**

GPS satellites rise in a general west-to-east direction, WRS carrier smoothing is of limited help to West Coast users, as the nearby WRS's do not have much time to "filter" their corrections for newly-arisen satellites before users see them as well. Thus, one does not have to move too far offshore before the only WAAS corrections available are corrupted by significant WRS error.

### *7.3.5 Summary of Results*

The results presented in this section demonstrate that the covariance propagation methods developed in this chapter produce useful and sensible predictions of WAAS performance. The Stanford WAAS predictions meet our performance expectations and agree quite closely with our limited flight-test data base. The geographic performance degradation observed for the Stanford WAAS network is gradual enough to verify that the underlying concept can provide useful corrections across a very large area. The addition of a WRS in Hawaii has a limited effect on performance over CONUS since it is far away, but it provides improved accuracy to Hawaii and helps calibrate the error surface over the region of the Pacific in between.

Predictions for the FAA NSTB suggest that a WRS network of this density is able to meet the ILS Category I approach requirement throughout CONUS. A combination of nearby WRS density, geosynchronous satellite visibility, and a bias toward the best-error-fit middle of the WRS region influence a given user's performance in a logical way. Although the NSTB is not designed to provide optimal coverage, the results from early versions of the NSTB have motivated changes in the WRS layout that noticeably improve performance near the edges of the user grid. The data collected from NSTB starting in late 1996 should be the first opportunity to verify these predictions on a nationwide scale.

Several key points should be raised here regarding the limitations of this methodology for predicting WAAS utility. First, the covariance propagation steps use batch least squares fits to the GPS error surfaces. Given the assumed "white" Gaussian behavior of GPS performance in non-failure cases, a least-squares solution is generally seen to be optimal. However, execution time constraints and the numerical problems mentioned in Section 7.2.4.4 make this approach probably infeasible for a real-time

WAAS. Instead, numerically easier data fits will be used in the WMS [8-7]. Thus, our covariance method represents the assumed *underlying behavior* of GPS errors in a WAAS context. It could therefore be viewed as an algorithmic “best case bound” on WAAS performance. Instead, it should be noted that the parameters of our current error models will improve as more data is collected. Furthermore, the process of design-and-test should produce real-time algorithms that are tuned to non-ideal error performance as observed from test results. As a result, it is possible that operational WAAS systems might be able to improve on the predictions contained in this chapter.

#### **7.4 Conclusions and Further Work**

Despite the assumptions and limitations of the best GPS/WAAS error models available, it is possible to use least-squares covariance propagation to predict the performance that widely-spread-out users will be able to achieve from any candidate WAAS network. This approach has produced reasonable, internally consistent accuracy predictions for both current prototype WAAS systems and a full-scale NSTB designed to cover all of CONUS (along with parts of Canada and Mexico). It thus serves as a very useful “end-to-end” computer analysis tool to aid the layout, design, and development of future networks.

Currently, we are using this method to predict WAAS performance while varying the proposed FAA network layouts and some of the key error parameters that go into the covariance model. Predictions are also being generated for conceptual WAAS networks covering other parts of the world. In addition to being useful for development, more detailed covariance algorithms of this type could be used as part of the WMS monitoring software in “semi-real-time”. Given up-to-date GPS satellite almanacs and recent ionosphere/troposphere conditions, they could independently predict expected user accuracies over the next several hours, providing a basis for comparison to the real-time accuracy being measured at WRS’s or monitor sites. A sizable divergence between the two could serve as a warning of unusual GPS error behavior, and these warnings could be monitored by the WMS and then sent to users as needed.

A complete analysis of WAAS network performance requires integrating accuracy and integrity predictions into a unified system evaluation. The building blocks of integrity prediction and RAIM optimization were constructed in Chapters 5 and 6, and a comprehensive accuracy prediction methodology has been developed in this chapter. In the next chapter, these two algorithms are merged and fed into a new WAAS user “value model” to derive a single “bottom-line” numerical evaluation of overall network acceptability. Chapter 8 also illustrates the use of the combined evaluation method for a proposed European WAAS network. It also demonstrates that given sufficient computing power, a genetic algorithm (discussed in Chapter 4) can automatically vary the elements of a WAAS network and run the evaluation algorithms to search for and converge on an optimal system for Category I precision-approach capability in an assigned user coverage area.

## **Chapter 8: WAAS Overall System Evaluation: A European Example**

*Probabilities direct the conduct of the wise man.(Probabilia...sapientis vita regetur.)*  
-- Cicero, *De Natura Deorum*, Bk. i, Ch 5, Sec. 12.

*Insisting on perfect safety is for people who don't have the balls to live in the real world.*  
-- Mary Shafer, NASA Ames Dryden

*In every work of genius we recognize our own rejected thoughts: they come back to us with a certain alienated majesty.*  
-- Ralph Waldo Emerson, *Essays, First Series: Self-Reliance*.

### **8.1 Introduction**

Chapters 6 and 7 have demonstrated that methods exist to predict vertical position accuracy and integrity/continuity/availability (“PICA”) for a large WAAS user region. It should thus be possible to link these separate analyses to perform a unified evaluation of the acceptability of a given WAAS network. As mentioned before, standards are now being developed which specify acceptable performance in all four PICA parameters [8-16], and the results of a unified PICA analysis can be compared to these requirements to determine acceptability. However, given that WAAS development is in an early state, it may be too early to come up with a uniform set of requirements that can be agreed upon by authorities in different regions who must operate under widely varying economic, political, and meteorological contexts.

In this chapter, an approach for complete top-level analysis and optimization of wide-area augmented GPS networks is developed by uniting accuracy and integrity prediction with a system value model focused on the benefits and costs of supplying Category I precision approach capability via WAAS. An example application not yet mentioned in this thesis is the use of WAAS to provide this coverage to Europe. Section 8.2 introduces a baseline WAAS with only four reference stations and demonstrates, using the coverage prediction algorithm, that this minimalist network is sufficient to meet the Category I ILS vertical accuracy requirement for most of continental Europe.

Experimental flight-test results obtained to date can only suggest designing networks of WRS's that are both numerous enough and in close-enough proximity to ensure that users are "close enough" to the nearest WRS with near-certainty. WAAS networks are currently laid out with the help of only these preliminary guidelines. The user accuracy prediction algorithm developed in Chapter 7 provides a system-wide evaluation of 95% performance for a given WAAS network design. As demonstrated in Section 7.3, it is a very useful tool for studying the effects of moving reference stations and/or geosynchronous satellite locations. In fact, the NSTB results shown there strongly suggest that acceptable 95% user vertical position accuracy (good enough to meet the 4.1-meter ILS-derived Category I requirement) can be achieved with a network of as few as 10-12 optimally-located WRS's in CONUS.

However, the key limiting factor in WAAS network design is not so much meeting a specified 95% error requirement but instead providing sufficient *integrity* to users who rely on WRS observations and WMS corrections. In order to catch locally-correlated spurious events in the WAAS error parameters, a denser net of WRS's and/or passive monitors (which check user accuracy but do not contribute to the WMS corrections) is probably desirable. Given the uncertainty about rare-event WAAS error magnitudes discussed in Chapter 6, however, it is quite difficult to determine just how dense a net would be required. Operational (or testbed) WAAS network design will need to be completed before a complete model of such events is available.

In Chapter 6, a prior uncertainty model of WAAS ionospheric delay correction error was constructed, and its effect on user integrity and RAIM threshold selection was simulated for a specific user location. But the results in Chapter 7 make it clear that position accuracy (and hence integrity) will vary depending on a given user's location within the WRS network distribution. Integrity prediction and optimization along the lines of Chapter 6 requires many simulation trials to fully sample the set of possible failures; thus it is not possible to make numerical predictions throughout a dense user grid, as done for coverage prediction. Instead, a reasonably comprehensive prediction can be produced from a statistically representative set of locations within the user coverage area.

Section 8.3 expands on the simulation methodology used in Chapter 6 by sampling failures from tail expansion of normal-condition Gaussian distributions produced by the coverage model of Chapter 7. It also adds new categories of failures to the models for ionospheric correction error developed in Section 6.2.3 to provide a more complete “failure space” representation. The results of this integrity-prediction approach are included for the baseline European WAAS introduced in Section 8.2. This result demonstrates that a WAAS network with acceptable accuracy may not be able to meet integrity requirements for Category I airborne users.

Section 8.4 formulates a genetic algorithm (GA) to search for the optimal combination of GPS augmentation systems for a given WAAS application. Unlike the integer encoding and GA operators developed in Section 4.3.2 for spacecraft redundancy allocation, the application of a variable list of augmentations can be encoded as a simple binary (0-1) string. This allows canonical GA operators (designed for binary solution vectors) to be used, giving more confidence that convergence will eventually be obtained. Section 8.5 develops the top-level user value model that would be used to compute GA fitness for the Category I precision approach application and a European user coverage area. This is a simple, provisional benefit-minus-cost function that incorporates the benefits of Category I service, the costs of imperfect accuracy and integrity performance, and the life-cycle costs of installing a given WAAS network.

Although it has not yet been possible to run a GA to convergence on the WAAS optimal design problem due to limited computing resources, the first few generations of a typical GA search produce network combinations that give a much higher “fitness” than the 4-WRS baseline design. The results of both accuracy and integrity analyses for two of these proposed designs are given in Section 8.6. While the baseline network did not appear to provide sufficient integrity (after inner-loop RAIM optimization), the integrity picture improves considerably with the addition of a single WRS in a predictable spot, and it improves much more with the addition of one more WRS *and* a third geosynchronous satellite over Central Europe. Finally, Section 8.7 summarizes the progress made thus far on WAAS network optimization and discusses the steps remaining to make it a practical reality. The key step will be to recompile the WAAS prediction codes to run on a multi-

processor workstation to be procured as a high-speed simulation facility. This step should allow GA search to convergence for the European WAAS application.

## **8.2 Preliminary Design for a European WAAS**

### **8.2.1 *European WAAS Developments***

As noted in Chapter 6, the concept of wide-area GPS augmentation networks lends itself readily to international cooperation, as GPS corrections broadcast from geosynchronous satellites will reach a footprint that covers a significant portion of the Earth. Users who can receive and apply these corrections will need to be supported by WRS's in their own geographic region to gain the full benefits of WAAS. Thus, there is a strong incentive for aviation authorities the world over to work toward a common message framework and a globally-acceptable sharing of responsibilities. FAA plans for future international cooperation on WAAS are outlined in [8-11]. However, the WAAS concept itself is still in the experimental stage, and it appears that developments over the next decade will focus on semi-independent development of separate WAAS-like networks in other parts of the world.

While European states appear to hesitate to cooperate with FAA initiatives to spread its own WAAS concept, sufficient motivation exists for a "Tri-Partite Group" of European Space and Transport authorities to have developed a concept called European Geostationary Navigation Overlay Service (EGNOS) which is aimed at an IOC in 1999 and an FOC in 2002. EGNOS includes a set of 13 wide-area reference stations, at least two geosynchronous ranging satellites and ground stations for their control and maintenance, and a single WMS. Beyond that, the similarities between it and the FAA WAAS concept depend on how EGNOS is tied into other European plans for civil GPS, including the possibility of a proposed new civilian satellite constellation called GNSS-2. Also, research into radiobeacon broadcast of DGPS corrections in Europe suggests that much of the Continent can be covered from a single transmitting site [6-16]. However, it is expected that if NSTB can clearly demonstrate the feasibility and practicality of the

FAA's concept, EGNOS may become more similar to the FAA operational WAAS so as to facilitate future transatlantic cooperation.

### 8.2.2 Results for Baseline European WAAS

While the FAA has already more or less defined the system architecture for the NSTB and the operational WAAS, EGNOS is in an earlier stage of development. Depending on one's geographic definition of "Europe," it should be possible to provide sufficient accuracy to meet the current ILS Category I requirement with fewer than 13 WRS's, since very good coverage predictions have been obtained for 18 WRS's over a larger area in Section 7.3. A European WAAS application probably requires a smaller network, then, and the precise locations of reference stations and geostationary satellites have yet to be finalized. Thus, it is a good design problem for the end-to-end network optimization methodology to be developed in this chapter.

Figure 8.1 shows 95% vertical position error results, using the latest normal-condition error model, for a WAAS that could provide precision DGPS corrections to Europe. It has a "four corner" arrangement of WRS's in Scotland, Spain, Northern Russia, and Turkey. The AOR-E and IOR Inmarsat geosynchronous satellites mentioned in Section 5.1.1 are also presumed to be present, providing (when healthy) two additional ranging sources. Note that this minimal WRS arrangement provides sufficient accuracy to exceed the 4.1-meter vertical 95% ILS Category I precision approach requirement over almost all of Europe (The Asia Minor part of Turkey is formally in Asia but is presumed to be part of the European coverage area). Furthermore, the northern tip of Scandinavia (where the geosynchronous satellites are at low elevation and hard to receive) and the region just west of the Ural Mountains in Russia are the only places where this error

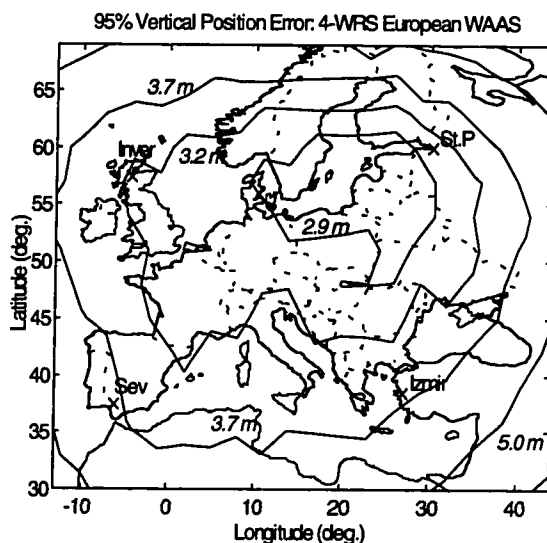
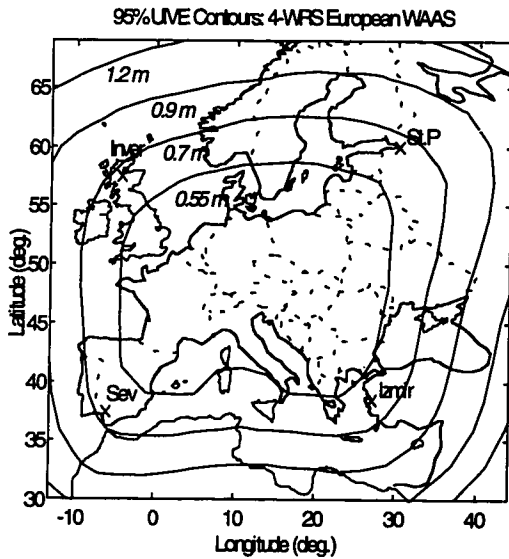
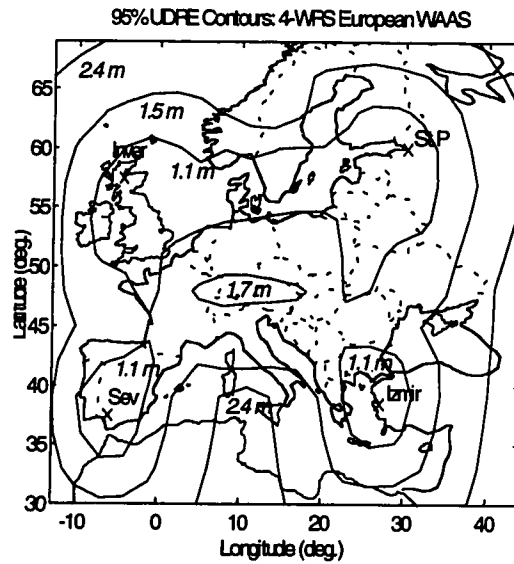


Figure 8.1: European WAAS 95% Accuracy



**Figure 8.2: European WAAS 95% UIVE**



**Figure 8.3: European WAAS 95% UDRE**

exceeds 5 m. Since Europe, unlike North America, is contiguous with other continents on a broad front, this minimal network also provides sub-5-meter coverage to the entire Mediterranean coast of North Africa.

Figures 8.2 and 8.3 show contours of 95% UIVE and 95% UDRE, as defined in Section 7.3.1, for this baseline European WAAS network. There is an interesting contrast in these two plots that is not apparent when UIVE and UDRE are plotted for a much larger network such as the NSTB. The regular, oval shape of the UIVE contours in Figure 8.2 is similar to that seen in Figures 7.11 and 7.13 for the NSTB. The ideal four-corners WRS arrangement provides a relatively flat UIVE noise floor of under 0.7 m ( $2\sigma$ ) over the area contained in an oval running through the four WRS sites. This is quite good for such a sparse network, suggesting once again that normal-condition ionospheric behavior is highly spatially-correlated. Outside of this, UIVE slowly begins to fall off in a predictable manner.

Figure 8.3, on the other hand, has quite irregular contours that are surprising at first glance. The area enclosed by the four WRS locations is comfortably within a 1.5-meter 95% UDRE contour, but the lowest error (of about 1 meter) is obtained in regions close to the two southern WRS's and in the region between (and including) the two northern ones. Note that the two northern WRS's are much closer together than the two

southern ones, a fact that is somewhat obscured by the Mercator map projection in these figures. A small region in the center of Europe seems to occupy a UDRE “hole” with errors as high as 1.7 m. It is the sparsity of WRS locations that leads to this result, as UDRE under such conditions decorrelates more rapidly than does UIVE -- a result that is not necessarily intuitive. Remember that only normal conditions are simulated here, and these have been shown to provide relatively benign UIVE. In contrast, only users close to one of the four WRS sites have confidence that they will receive good corrections for all satellites they can see. In the middle of the network, far from any single WRS, users can see satellites only visible to a subset of the four WRS's, and these users lie outside the space spanned by those WRS's often enough to give significantly poorer 95% ranging errors. This UDRE degradation is also caused by the fact that users far from the nearest WRS do not achieve the same level of “geometry cancellation” of the high ephemeris variances in the  $\tilde{\mathbf{G}}_u \mathbf{P}^{sv} \tilde{\mathbf{G}}_u^T$  term in (7.7) as do users close to a WRS. This suggests that despite the good overall position accuracy displayed in Figure 8.1, good reason exists to further augment this network with at least one centrally-located reference station.

These results show the potential of WAAS to provide high accuracy in a very cost-effective way, but the next question is whether such an arrangement also provides sufficient integrity, or user safety. Section 8.3 addresses this issue by building upon the RAIM optimization approach of Chapter 6 while utilizing the coverage prediction results to describe “normal-case” conditions.

### 8.2.3 *Single-Frequency WRS Option*

One more issue needs addressing for the case of a European WAAS. Recall that the separation of clock/ephemeris and ionosphere covariance propagation tracks in Section 7.2 relies on having dual-frequency receivers at each reference station to make measurements of ionospheric delay at their pierce points. However, because dual-frequency measurements by definition require the use of L2, which is not part of the SPS service guaranteed to civilian users, WAAS networks deployed by non-NATO agencies may choose to restrict themselves to the use of L1 measurements only [8-19]. As a result,

the possibility of using only *single-frequency* ionosphere observations should be considered when designing a wide-area network for non-U.S. users.

It is possible to extract a measurement of ionospheric delay from the code-carrier divergence on a single broadcast frequency, as described in [6-2,8-1]. But this approach takes significant time to sequentially compute an estimate, and the lack of an independent ionosphere measurement prevents the separation of ionosphere and clock/ephemeris that is key to the coverage prediction method. Thus, a rework of the coverage prediction algorithm to handle this possibility would be needed. Such a revision is feasible, but the lack of separation would make coverage prediction much more cumbersome.

For now, we can only predict how much less accurate UIVE would be in the single-frequency WRS case (compared to the UIVE dual-frequency predictions). From comparisons of single-frequency measurements to more accurate dual-frequency ones, our best current estimate is that the use of single-frequency observations would add a one-sigma *vertical* error of around 0.8 meters to the WRS ionosphere delay measurement [6-2]. However, since single-frequency receivers cannot directly separate ionosphere from other error sources, the covariance propagation method used here must be re-worked to combine the clock/ephemeris and ionosphere into one larger estimator for this case. We now expect this change to reduce 95% accuracy by about 15-30%, but the effect on integrity could be much worse.

### **8.3 Coverage-Based Integrity Simulations**

#### **8.3.1 Background and User Cost Model**

Unlike the “normal conditions” assumed by the WAAS coverage prediction model, integrity threats are hazardous events that are presumed to occur rarely but have the potential to put the user in serious danger if he or she is not promptly warned. Since WAAS includes both ground-station and user elements, detecting these events is a shared responsibility of the augmentation network and of each user. Individual users can use Receiver Autonomous Integrity Monitoring (RAIM) to provide a warning from their overdetermined set of ranging measurements if they can see more than four satellites. This

process, described in detail in Chapters 5 and 6, uses the magnitude of a residual vector as the decision statistic; when it exceeds a pre-set threshold, the user is warned that conditions are unsafe. In addition, WAAS ground stations can also monitor statistics that compare primary and redundant system measurements to each other. Warnings of unreliable satellites can then be included in the WAAS message to warn all affected users (note that corrections deemed unreliable would not be sent) [8-12,7-13].

In Section 5.3, a method of setting WAAS RAIM residual thresholds based on a *user cost model* was developed and demonstrated. This cost model measures the expected likelihood of a fatal aircraft accident based on hazardous outcomes such as aborted approaches and missed integrity threats. Section 6.4.2 contains the cost model used for WAAS that is given in Table 6.3. This relative measure of RAIM event costs allows us to optimize the detection thresholds rather than relying on the chi-square probability assumption used in [7-13]. The same optimization approach explained in Section 5.3.5 and used in Section 6.4.2 is applied here as well. The WAAS MOPS [8-13] has specified a required protection error (RPE) of 19.2 meters instead of the 15-meter number used in Section 6.4.2. The risk multiplier for approaches in Europe is twice that used in Chapter 6, or  $R_c = 10^{-4}$ . In order to maintain consistency between economic cost and passenger risk (also see Section 8.5.4), the non-availability cost is then reduced to half that of Chapter 6 ( $C_{NA} = 0.0125$ ). However, a key question that has been raised for the WAAS Category I application is how well the risk multiplier of  $10^{-4}$  applies to this protection radius when (a) the approach is not supposed to get closer than 200 feet (about 61 m) to the ground, and (b) few Category I approaches descend all the way to the 200-ft. minimum ceiling before being able to see the runway. This will be discussed further in the context of the baseline network integrity results contained in Section 8.3.4.

### 8.3.2 Revised WAAS Rare-Event Probability Models

The RAIM analyses in Chapter 6 focus on worse-than-Gaussian ionospheric spatial decorrelations, which are assumed to be possible in the worst 10% of cases based on previously published experimental data [8-10]. More severe decorrelations are possible in

the worst 2% of cases. The effects of these “non-Normal” expanded-tail decorrelations must be analyzed by Monte Carlo simulation.

The approach taken in Section 6.2.3 is to sample a standard Normal random variable and then “expand” the effective variance in rare-event samples. The expansion factors for the worst 2% of cases and the next-worst 8% of cases are given by (6.6). The same sampling procedure is conducted here except that the “Normal-case” variances are taken from coverage prediction UIVE results for each evaluated user location instead of the generic variances listed (by season) in Table 6.2 and adjusted by (6.7). It samples ionospheric delay using the Normal distribution implied by the UIVE for a given site. If a sampled standard normal variable  $|z| \leq 1.645$ , the vertical ionosphere ranging error is simply the sample  $z$  times one-half of the assumed  $2\sigma$  95% UIVE result. If the sample exceeds 1.645 (the 90% cutoff), the UIVE-based standard deviation is *multiplied* by the expansion factor  $TM$  computed from (6.6). This inflated deviation is multiplied by the already-large sample  $z$  to give the vertical delay in this case. Finally, the vertical error is multiplied by the appropriate obliquity factor (a function of the satellite elevation angle  $\epsilon$ ) from (6.3) to provide a sampled slant pseudorange error.

In the spacecraft orbital simulation,  $N_j$  satellite geometries are updated in the same way as in Section 7.2.1 (using the approach outlined in Section 5.1), but unlike the coverage prediction simulation, all satellites are not assumed to be healthy. Instead, each satellite is sampled to determine if it is “out of service” and is thus not usable. These failure probabilities, derived from the Prior Probability Model of Section 5.3.1 [6-15], are:

$$\begin{aligned}\text{Pr (GPS satellite unhealthy)} &= 0.014 \\ \text{Pr (GEO satellite unhealthy)} &= 0.010\end{aligned}$$

Note that all users lose availability if they cannot receive WAAS corrections from at least one working GEO satellite. Since some of the user locations in Europe will only be able to see one GEO, the availability of those sites could not exceed 99%, the probability of any given GEO being operational. Furthermore, users near or north of 60 deg. latitude will have trouble maintaining lock on GEO’s since they would be low on the

horizon. Availability higher than 99% is expected for Category I operations; thus either more GEO's would have to be added or, as a backup source of corrections, a radiobeacon network along the lines of [6-16] would be necessary. In the simulations conducted here, a radiobeacon network with a combined coverage and availability of 90% over all of Europe is assumed to exist for this purpose.

In addition, it is assumed that a failure of the WAAS ground network could lead to errors in the broadcast clock/ephemeris corrections that are much larger than the 95% UDRE computed from the coverage prediction method. This could result from database or computation errors that are not caught by the WRS and WMS monitor systems. Because we are interested in sampling failure cases, the probability of having significantly inflated UDRE is taken to be 0.001, which is at least 10 times higher than expected of an operational WAAS [8-12]. If a given satellite experiences this type of failure, its predicted (broadcast) UDRE is increased (multiplied) by one plus a factor sampled from an *exponential* distribution with a mean  $\mu = 2.0$  to get the effective (true) UDRE.

### 8.3.3 Integrity Simulation Procedure

For each of  $N_j$  satellite geometries,  $N_K$  failure state trials are conducted. In each failure trial, each satellite is sampled to see if it is functioning normally, and unhealthy ones are removed from the user geometries. Each satellite next has the state of its ground correction (UDRE) sampled. The overall ionosphere decorrelation state, which applies to all users, is then sampled. A failure *bias*  $|z| = 1.645$  is applied to this sample to insure that all cases at least have this amount of non-standard conditions. Note that for both UIVE and UDRE tail expansion, the random system state is biased to artificially enhance the likelihood of failures in the simulated trials. This is done because the coverage analysis has already provided normal-case results, and we want as much sampling of the failure space as possible. Using a bias of 1.645 for ionospheric sampling automatically limits the sampled failure space to the worst 10% of cases. When the simulation results are evaluated, these biases can be taken out to return to the "true" output probability distribution. This is a simple form of *importance sampling* as a means of variance reduction in simulation results. Importance sampling is further explained in [2-4,3-9].

Rather than attempting to run simulations for each user in the coverage prediction grid of Section 8.2, the user population is melded into a much smaller number  $N_U$  of user locations. For each sampled failure state, a weighted position fix is carried out for each user location using (7.6). This position fix is based on the normal UIVE and UDRE for that location (plus sampled tail expansions) given by the coverage prediction method. Each user constructs a diagonal weighting matrix  $\mathbf{W}^{-1}$  from the root-sum-square (rss) of his RMTSA, UIVE, and UDRE variances (from Section 8.2), but a 25% random factor is applied to prevent the use of perfect weighting information. Because weighting matrices are used in the position solution, an adaptation to the RAIM residual statistic computed in (6.9) incorporates these relative measurement weights. Introduced as “weighted RAIM” in [7-13], the weighted residual statistic  $D_n$  is now given by:

$$D_n^2 = \mathbf{z}^T \mathbf{W}^{-1} (\mathbf{I}_n - \mathbf{G} \mathbf{G}^*) \mathbf{z} / (n - 4) \quad (8.1)$$

where  $\mathbf{I}_n$  is a  $n \times n$  identity matrix ( $n$  is the number of functioning satellites in view),  $\mathbf{G}$  is the  $(n \times 4)$  user geometry matrix, and  $\mathbf{z}$  is the  $(n \times 1)$  vector of pseudorange measurements.

After all failure simulations are completed, the 90% of cases that would exist under normal DGPS conditions are added to the  $dx$  vs.  $D$  matrix stored for each discrete VDOP bin. Under ideal conditions,  $dx$  and  $D$  are independent, with (vector)  $dx$  being Normally distributed with a covariance given by the vertical position error result from coverage prediction and  $D^2$  having a chi-square distribution with variance parameter  $D^2/\sigma_z^2$  and  $n-4$  degrees of freedom (see Section 5.2). The revised matrix is then searched to find the RAIM detection threshold  $T$  that *minimizes* the overall weighted RAIM cost (see Table 6.3) over all  $N_U$  user locations (each with a separate  $dx$  vs.  $D$  matrix). Note that only one set of thresholds is chosen for all users. If this optimal threshold  $T^*$  gives a cost that is lower than the non-availability cost in Table 6.3, RAIM is *available* for that VDOP and the system is usable. Otherwise, RAIM is *unavailable*, and the system incurs the non-availability inconvenience cost for those trials (see Section 5.3.4).

#### 8.3.4 Integrity Results for 4-WRS European WAAS

Integrity evaluation for the European WAAS is done at 11 user locations shown as ‘o’ in Figure 8.4 (discussed in Section 8.4.1). These locations receive different weights as explained in Section 8.5. For these locations and the basic 4-WRS network of Figure 8.1, 1 million failure states and position fixes were simulated from 1000 satellite geometries for each of the 11 user locations. Table 8.1 contains the results in terms of the overall weighted RAIM cost and also probabilities of various hazardous events. As mentioned before, for WAAS Category I approaches, the RPE or “required (vertical) protected error” is chosen to be 19.2 m at the 200-foot decision height. The definitions of false alarm and missed detection probabilities were given in Section 5.2.2.

<i>Overall RAIM user cost</i>	0.0019
Prob(RAIM available)	0.983
VDOP availability limit	2.9
Prob(position error > RPE)	$3.3 \times 10^{-5}$
Prob(missed detection   error > RPE)	0.111
Prob(false alarm)	0.0007
<i>Fatal Accident Prob. per approach</i>	$1.7 \times 10^{-7}$

**Table 8.1: Baseline European WAAS User Integrity**

Although it is clear from Figures 8.1 and 8.4 that this 4-WRS network meets the WAAS Category I accuracy requirements at all 11 selected user locations, it is equally clear here that this network does not provide adequate integrity given the failure uncertainty models from Section 8.3.2. Availability is not bad at 98.3%, but the probability of exceeding the RPE is too high. RAIM catches 89% of these events, but the remaining 11% that become “missed detections” translate into an unacceptably high fatal accident risk, which is computed by multiplying the part of the RAIM cost due to integrity risk (0.0017) by the risk multiplier ( $10^4$ ). Note from this value that it implies that 2.5% of all missed detections (average cost of about 250) lead to fatal crashes. While current Category I requirements do not specify a maximum acceptable fatal accident risk, the implied requirement from the RNP and the Category III requirements is  $10^{-9}$ . This

network has considerably higher risk; so additional augmentations will be required. Note again, however, that the risk multiplier used here may be quite conservative for approaches that should not require WAAS navigation below the 200-foot decision height.

### 8.3.5 Integrity Simulation “Max Slope” Geometry Variant

The integrity simulation approach developed in this thesis deliberately avoids making many of the assumptions of traditional RAIM, as discussed in Section 5.2.3. One of these is limiting the integrity analysis to single-channel (i.e., single satellite) failures only. In traditional RAIM, since failures on more than one channel are deemed to be exceedingly unlikely, the cause-and-effect study focuses on “worst case” bias errors on a particular “worst-case” satellite. Since these are the failures that most threaten user integrity, traditional RAIM sets availability limits to “ensure” that these worst-case events are detectable by the residual thresholds set for that case. Two relevant geometry statistics can be monitored for this purpose. One is simply *maximum subset DOP*, in which the relevant DOP that would result from eliminating each of the  $n$  satellites in view is computed, and the highest of these indicates which satellite is most critical. This calculation is related to  $\delta H_{\max}$  in (5.19) of Section 5.2.2, which is given by (HDOP is used here) [7-11]:

$$\delta H_{\max}^2 = \max_i \{ \delta \text{HDOP}_i^2 - \text{HDOP}^2 \} \quad (8.2)$$

where  $\delta \text{HDOP}_i$  represents HDOP with the  $i^{\text{th}}$  ( $i = 1, \dots, n$ ) satellite removed.

In [7-11], it has been shown that  $\delta H_{\max}$  as an availability statistic is mathematically equivalent to the second geometry measure, which is known as *maximum (satellite) slope*. This latter statistic is given by (again for HDOP):

$$HSlope_{\max}^2 = \max_i \{ (G_{1i}^2 + G_{2i}^2) / S_{ii} \} \quad (8.3)$$

where the subscript “1” refers to the  $[1,i]$  element of matrix  $\mathbf{G}^*$  as defined in (5.9), and the matrix  $\mathbf{S}$  is computed from (5.14). Since we are concerned with vertical error here, the corresponding maximum slope parameter is calculated by:

$$VSlope_{\max}^2 = \max_i \{ \mathbf{G}_{3i}^{*2} / \mathbf{S}_{ii} \} \quad (8.4)$$

To transform these last two equations into the “weighted-RAIM” domain as described in Section 8.3.3, the numerators inside the  $\max\{\bullet\}$  term are multiplied by the relevant (elevation-dependent) ranging error standard deviation  $\sigma_i$  (taken from the square roots of the diagonal elements of the  $n \times n$  weighting matrix inverse  $\mathbf{W}^{-1}$ ) for satellite  $i$  [7-13]. Also note that the definition of  $\mathbf{G}^*$  changes in the weighted position fix calculation as shown in (7.5).

A physical interpretation of the maximum slope parameter is visible in the  $\delta x$  vs.  $\delta r$  (same as  $D$ ) plots in Figure 5.2. In this plane, each satellite has a “failure line” with a slope given by (8.3) or (8.4). Higher bias errors on a given single satellite move the ellipse of normal  $\delta x$  vs.  $\delta r$  performance further up the relevant failure line. Lines that have the highest slope will violate the missed detection region of Figure 5.2 with the smallest bias error. Thus, failures on the “max slope” satellite are the hardest to detect via traditional RAIM. A maximum-slope limit for availability is set in the context of Figure 5.2 by eliminating geometries that have a satellite with a failure slope large enough to prevent the integrity and continuity requirements from both being achieved for the worst possible failure bias. Using single-channel failure assumptions and the very-accurate IBLS carrier-phase landing system, a detailed illustration of the application of maximum slope to RAIM availability and threshold selection is given in [7-9] for the LAAS Category III precision landing application.

As discussed in Section 5.3, this whole edifice of RAIM thresholds and limits rests on tenuous “worst-case” single-failure assumptions. Thus, the availability limits set by the minimal user cost approach are referenced to DOP by itself instead of these single-failure parameters. However, since for the failure probability models constructed in this thesis, it

is still likely that most failures will be of the single-channel type, it would be interesting to see if the use of maximum slope as given by (8.4) as the availability variable instead of VDOP makes an appreciable difference.

Table 8.2 gives integrity results for the same WAAS failure model and baseline European WAAS network using maximum slope from (8.4) instead of VDOP. The basic event being simulated (WAAS-corrected GPS position fixes) is the same; thus we would expect no difference between the base probabilities of exceeding RPE for different RAIM simulations (since these are independent of integrity monitoring). The difference of 3% seen between Tables 8.1 and 8.2 is thus due to noise in the simulation results. However, it is apparent that the Max slope geometry criterion does give slightly better results. Note that both the false alarm rate (0.0006) and the missed detection probability (0.0865) improve over the values in Table 8.1 (0.0007 and 0.111 respectively). The “bottom line” integrity cost improvement of about 15% is significant but not decisive; the overall efficacy of RAIM is still limited by the uncertainty in the WAAS failure models. Since the maximum slope criterion (or, by inference, maximum subset VDOP) appears to be better than plain VDOP, it seems that RAIM effectiveness is helped by the relative predominance of single-channel failures in the prior failure space. Nevertheless, the results here still fall short of those predicted by weighted RAIM with traditional assumptions [7-13].

<i>Overall RAIM user cost</i>	0.0016
Prob(RAIM available)	0.982
Max Slope availability limit	5.5
Prob(position error > RPE)	$3.4 \times 10^{-5}$
Prob(missed detect.   error > RPE)	0.0865
Prob(false alarm)	0.0006
<i>Fatal Accident Prob. per approach</i>	$1.4 \times 10^{-7}$

**Table 8.2: Baseline User Integrity using Max Slope**

This exercise illustrates a key distinction mentioned in Chapter 1 between top-down and bottom-up design approaches. Traditional RAIM follows the “bottom-up” philosophy of systems engineering: a relatively detailed problem is studied by making

assumptions as needed, then an algorithm or set of hardware is crafted to handle this specific model. As noted before in this thesis, this conventional approach often neglects much of the uncertainty surrounding the “true” problem and focuses the solution too narrowly on a small part of the problem space. In contrast, the top-down simulation-based design heuristic builds an uncertainty model of the problem space as broadly as possible and samples from it to represent the effects of the “problem”. Broad-based intelligent search then takes selected monitoring statistics and fits them to the failure space as best as possible. As discussed in Chapter 1, this has the feature of being optimal in a decision-theoretic sense.

In addition, top-down design can be pursued further into the domain of details used by traditional design approaches. In the optimal RAIM case, the next step is to store the sampled failure-space parameters for all cases where a bad position (especially a missed detection) results. It will then be possible to examine the sets of failures that pose the greatest risk to user integrity (i.e., those with the highest product of occurrence probability and dependent likelihood of integrity failure) to determine:

(1) *Specific detection strategies* - Once the most threatening sets of failures are known, detailed design of detection algorithms targeted on these events can be done. This level of design is analogous to what is done in traditional RAIM as described above.

(2) *Value of failure probability information* - System-level evaluation of the value of more statistical certainty regarding failure source and effect probabilities is possible, since the likelihood of the most threatening failure sets occurring will depend on the uncertainty of the prior probability model. This comes from the “value of information” concept discussed in Chapter 1. Once this is determined, a strategy for further experimentation and data-collection can be devised that optimizes the cost/value tradeoff for attaining better information in the future.

This illustrates the course of future research for the optimal RAIM application. Both WAAS and LAAS augmentation architectures will be studied, and the different failure set sensitivities for each can be determined to see which augmentation options best

mitigate the respective failure spaces. Furthermore, it lays out the future development of detailed integrity algorithm design while the optimization method simultaneously (in a development sense) searches for the best top-level system architectures.

## **8.4 Genetic Algorithm Optimization Model**

By combining the coverage prediction model and integrity simulations, it is possible to generate overall evaluations for any GPS augmentation architecture. If the user population (or government agency) can derive a function that computes a top-level “figure of merit” based on the predicted geographic spread of accuracy and integrity performance, optimization of entire networks becomes possible. However, the use of complex covariance and simulation models to generate these evaluations requires a flexible optimal-search approach that does not require well-defined, deterministic problem formulations.

As introduced in Chapters 2 and 4, evolutionary algorithms now provide this capability. Several specific methods, including Simulated Annealing and Genetic Algorithms, have been used to solve a wide variety of problems. In general, they attempt to “evolve” better solutions over time by perturbing the best solutions found up to that point using semi-random operators that can avoid being “trapped” by local maxima or minima, although there is no guarantee of finding the global optimum in finite time [2-2,5,17]. They can also tolerate the noisy evaluations given by complex simulation models. Simulated Annealing has proven to be successful in handling both the aircraft control problem in Chapter 2 and the spacecraft redundancy design problem in Chapter 4. Because the selection of options from a list of possible augmentations fits nicely into a binary solution format, a genetic algorithm is developed here for use in optimal WAAS network design.

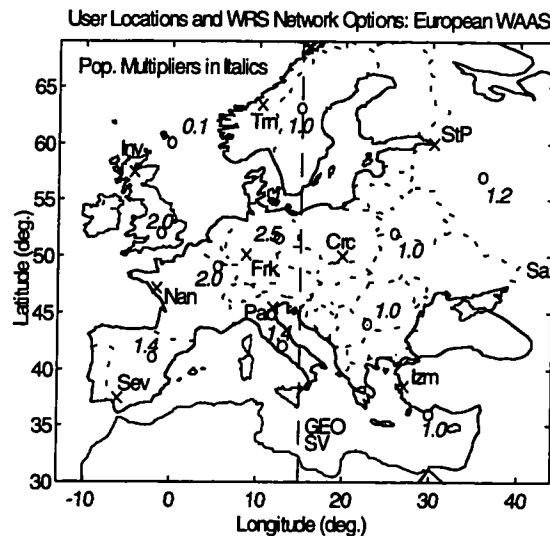
### **8.4.1 WAAS Network Design Encoding**

Much of the work in designing an evolutionary search method for a specific application lies in tailoring the search to fit a natural encoding of the design space. For

WAAS network optimization, the design variables can be expressed in a vector of binary (0/1) elements, or *genes*, which makes it possible to apply a standard *genetic algorithm* (GA) to evolve toward the optimal solution.

A computer search alone cannot design an optimal network -- the input of design engineers is crucial. In this case, we rely on human designers to provide a list of possible WAAS augmentation elements for the

GA to consider. This list may include reference station sites, provision for independent monitor sites, and additional geosynchronous spacecraft to broadcast corrections and add redundant ranging measurements.



**Figure 8.4: European WAAS Locations and Options**

No.	Augmentation Option	Incremental Cost (\$ K)
1	Inverness WRS (UK)	2000
2	Seville WRS (SP)	2000
3	St. Petersburg WRS (RU)	2000
4	Izmir WRS (TU)	2000
5	Padua WRS (IT)	2000
6	Trondheim WRS (NO)	2000
7	Saratov WRS (RU)	2000
8	Nantes WRS (FR)	2000
9	Frankfurt WRS (GE)	2000
10	Cracow WRS (PO)	2000
11	Single-Freq. WRS RCR's.	- 90 $N_{WRS}$
12	add'l. GEO SV at 15° E	25,000

**Table 8.3: European WAAS Augmentation Options**

Essentially, it can include any option that can be modeled in the GPS/WAAS accuracy and safety prediction algorithms.

Table 8.3 lists the 12 augmentation options for the European WAAS application that we shall consider in this chapter. Ten of these are potential WRS locations, which are shown as 'x' in Figure 8.4. Next is the use of single-frequency ionosphere corrections (see Section 8.2.3) instead of dual-frequency ones, and the last is a third geosynchronous satellite placed over Central Europe at 15° E longitude (shown by a dashed line in Figure 8.4). Given this list of options, a design solution is simply a vector of  $N_d = 12$  0-1 entries, where a 1 represents the *presence* of the relevant option and a 0 represents its *absence*. Note that the addition of new options thought up by the human designers can be handled simply by increasing the length of the design vector. This flexibility is important, as the results of early evaluations and optimization runs may motivate the designers to think of new augmentation options.

#### 8.4.2 GA Population Evolution Operators

A genetic algorithm evolves a population of  $N_p$  design solutions (10 are used here) from one generation to the next. The population is initialized by combining baseline solutions chosen by the designers (such as the 4-WRS network shown in Figure 8.1) and randomly generated solutions in a 50-50 ratio. This becomes the “zeroth” generation of the GA search. An accuracy and integrity evaluation of each is then conducted to provide an objective-function value for each initial network design.

The canonical GA used here evolves the next-generation design solutions based on the current-generation members and objective values. As described in Section 4.3.2, three operators are used. The first is *reproduction*, in which a percentage  $P_R = 60\%$  of the current solution members are chosen as parents of the next generation according to their objective value, or *fitness*. This is done by a variant of *roulette-wheel selection*, in which the parents are randomly chosen with probabilities that are proportional to their linearly normalized fitnesses. In addition, the best solution is automatically copied directly into the next generation (*elitism*) [2-2]. This is the same approach used for the spacecraft problem

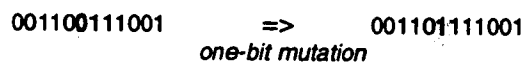
in Section 4.3.2. The following two operators will be different because, in this case, canonical operators can be used on the binary solution encoding.

Parents selected by this process are “mated” together two at a time in the *crossover* operator, which simulates sexual reproduction. In the basic *one-point* crossover operator, a location between 1 and  $N_d$  is sampled from a Uniform distribution, and the two parents swap their genes before and after that point to make up two child solutions for the next generation:



Another possible choice is *uniform* crossover, in which two parents combine to produce one child. In this case, for each gene location, if the two parents have the same gene, the child gets that gene as well. If the two parents disagree, the child’s gene is chosen by a 50-50 random sample [2-2]. One-point crossover is used here, but it is possible to combine these different operators to yield a faster and more robust search.

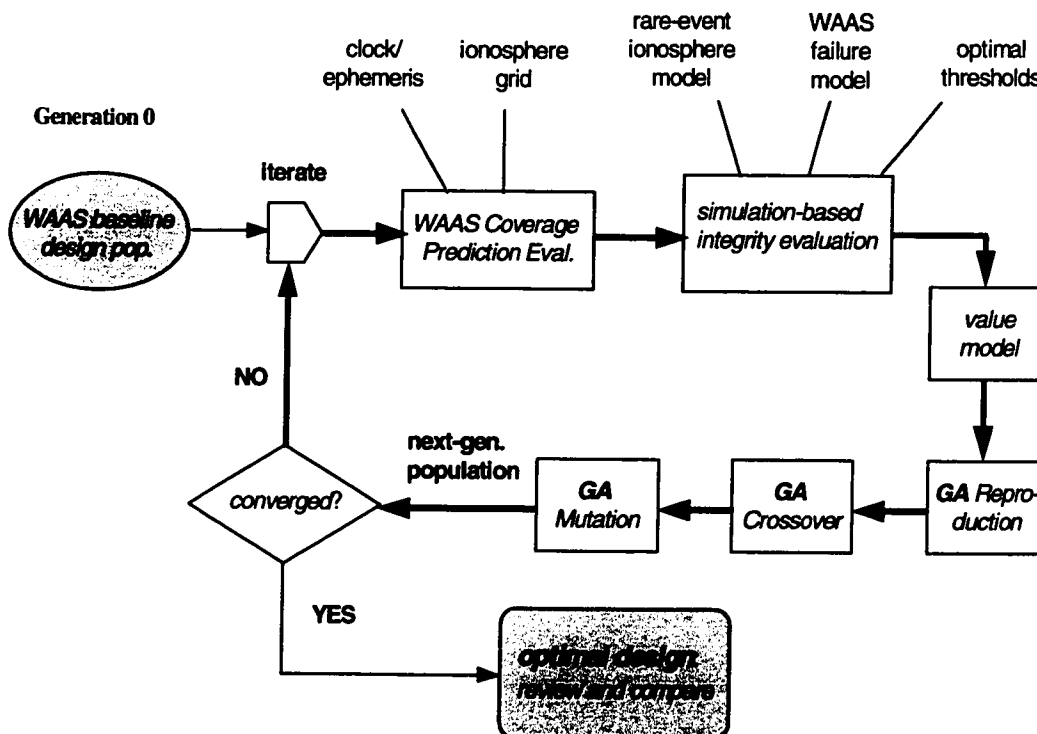
*Mutation* is the final canonical GA operator. Once the  $N_n$  solutions that make up the next generation have been chosen by reproduction and crossover, a Uniform random sample is made for each gene in each solution. If this sample is lower than a chosen mutation probability  $P_M$ , that gene (bit) is flipped to its binary complement (e.g., 0  $\rightarrow$  1, 1  $\rightarrow$  0). This process, akin to rare genetic mutations in biological organisms, helps maintain the genetic diversity of the solution population, preventing a small set of apparently good solutions from achieving premature dominance (i.e., a local optimum). Normally,  $P_M$  is chosen to be  $\leq 0.01$  (we have chosen 0.01), but higher mutation rates (inducing more diversity) have been successful for other problems [2-2].



### 8.4.3 GA Optimization Procedure

Figure 8.5 gives a flow chart of the procedure by which the GA “breeds” new generations of solutions and evaluates their fitnesses. This chart follows the same format

as Figure 4.4 for the spacecraft redundancy problem. Generation 0 is initialized as mentioned in Section 8.4.1, then a loop of generations begins. Given a generation  $n$ , the fitnesses of each of its  $N_p$  solution members are evaluated using both coverage (Section 8.2) and integrity (Section 8.3) analyses fed into the cost model of Section 8.5. Reproduction, crossover, and mutation are then applied to generate the new generation  $n+1$ . The GA evolution can be stopped when the population (or the value of its best solution) stops improving, or it can be ended after a set number of generations. Each re-evaluation of a given network is added to those conducted previously; thus statistical significance increases with each new evaluation. Once a given network evaluation converges to within an uncertainty tolerance, no further accuracy/integrity evaluations are needed. Therefore, later GA generations will run faster on the computer than earlier ones.



**Figure 8.5: Genetic Algorithm Optimization Procedure**

## 8.5 WAAS Network Objective Function

Each of the possible solutions generated by GA evolution needs a *fitness* evaluation, or a measure of its relative “goodness”. Because GA optimization is very flexible, there are no mathematical constraints on the form of this system objective function. We can thus construct a “value model” that attempts to express the system’s top-level utility for the total user population. This is a key driver of the optimization process, as the GA evolution will tend to exploit any inconsistency or “hole” in the fitness model. For this reason, the elements of the objective function should be carefully considered, and the results of early GA runs may motivate changes in the value model.

The value model developed here is a provisional attempt to weigh user benefits and system costs in as wide a framework as possible, knowing that substantial revisions may be necessary as more designer and user input is received. Beyond its use as a fitness measure for GA optimization, a coherent and consistent value model provides a cost/benefit justification for investing resources in a new navigation infrastructure. As a result, the assumptions which go into the tally of costs and benefits are likely to be controversial at both the engineering and political decision-making levels. This fact further emphasizes the need to see the following value model as a proposal which will be subject to substantial updating at several levels of the decision hierarchy. However, the form of the model is of quite general applicability and could also be modified to evaluate the performance of a set of Local-Area Augmentation Systems (LAAS) over a similarly diverse set of users.

The overall objective function  $F(n)$  to be maximized is given by:

$$F_{\text{WAAS}}(n) = \sum_{u=1}^{11} PM_u [B_{\text{air}}^u f_{\text{acc}}^u - f_{\text{integ}}^u] - LCost_n \quad (8.5)$$

where  $f_{\text{acc}}^u$  and  $f_{\text{integ}}^u$  represent evaluations of coverage and integrity performance respectively for user location  $u$ ,  $B_{\text{air}}^u$  is the Category I user benefit for a given user location,  $PM_u$  is a “population multiplier” which measures the size of the user population

near that location, and  $LCost_n$  is the acquisition cost of a given WAAS network solution  $n$ , which includes the procurement cost and four years of OEM (operations and maintenance). Note that (8.5) represents a (benefit - cost) evaluation instead of a benefit/cost ratio. Either approach is acceptable, and (8.5) could easily be converted to ratio representation of acceptability.

#### 8.5.1 Population Multiplier

The basic definition for the population multiplier is:

$$PM_i = \begin{cases} \sqrt{p_i / p_c} & \text{where } p_i > p_c \\ 1 & \text{otherwise} \end{cases} \quad (8.6)$$

where  $p_i$  is the user population (which could be total population, number of air passengers, etc.) and  $p_c$  is a “critical value” which insures that all areas covered by WAAS get a minimum base priority. Locations which exceed this critical value do get a higher priority, but it does not scale linearly. The values of  $PM$  for the 11 user locations selected for the European WAAS is shown (in italics) in Figure 8.4. Note that the single location over the North Sea is valued at 10% of the overland site values since precision approaches cannot be done there. The maximum value of 2.5 given to the Leipzig, Germany user location implies a critical value for overall population ( $p_c$ ) of about 8 million.

#### 8.5.2 Network Acquisition Costs

The system acquisition cost for all WAAS networks assumes a well-equipped triply-redundant hardware setup at all ground stations. It includes a WMS procurement cost estimated at \$6 million and four years of OEM at \$2 million/year, giving a WMS acquisition cost of around \$14 million. The incremental WRS cost is estimated to be \$1.1 million, which includes a \$0.5 million procurement cost and a \$150 thousand per-year OEM cost. The cost saving obtained by using single-frequency receivers in the WRS's is estimated at 75% of the cost of a dual-frequency receiver set multiplied by the number of WRS's in a given solution. For all ground augmentations, an 80% administrative and

indirect cost factor is added, giving conservative final life cycle costs of \$25 million for a WMS and \$2 million for each WRS (as shown in Table 8.3). This is based on the high overall cost estimates for the FAA WAAS given in [6-1]. Finally, the cost of providing an additional geosynchronous satellite is assumed to be \$25 million, the estimated cost of an inexpensive satellite designed for just this purpose. As in the Inmarsat case, leasing a GEO transponder may be an option, but the high value of the 15° E location suggests even a lease cost will be much higher than the \$2-5 million/year paid by the FAA. The sensitivity of the optimal result to this cost will be examined further in Section 8.6.

### *8.5.3 User Benefit Estimates*

The calculation of benefits provided by Category I to precision approach users requires making significant assumptions. According to [6-5], WAAS is expected to increase the number of Category I-equipped approaches in the U.S. from 765 (in 1994) to over 5,000. It also suggests an overall user benefit for WAAS Cat. I to be \$992 million, or about \$200,000 per approach. In Europe, we estimate that this life-cycle per-approach benefit will be doubled due to the poorer weather there. In [6-8], Europe is estimated to have 326 Category I ILS facilities (1994), and we conservatively assume that WAAS will allow this to grow to 1200, giving a total user benefit of \$480 million.

An estimate of the per-approach benefit of having Cat. I available is estimated by [6-8] as saving 2 minutes. Converted to aircraft per-hour fuel and direct operating costs of a weighted mix of passenger aircraft (about \$4800), the benefit (conservatively) becomes an average of \$160 per approach. Given 1200 Cat. I approaches each providing benefits of \$400,000 on average over a four-year life cycle, approximately 3 million Category I approaches in Europe are expected to result during this time frame.

A second user benefit to WAAS is removing the need to support and maintain the 326 current ILS facilities that now provide Category I capability. This cost is estimated by [6-8] to be \$400,000 per ILS facility (per life cycle), which, multiplied by 326, gives an added benefit to WAAS of \$130 million. While it can be argued [8-19] that the current ILS network has been recently upgraded and represents a "sunk cost," the continual maintenance of it would no longer be necessary after WAAS becomes operational. Under

this model, the total life-cycle benefit of WAAS Category I is \$610 million, if it were to perform perfectly in both the accuracy (i.e., 95% error approaching zero) and integrity (i.e., availability approaching one and fatal accident risk approaching zero) domains.

#### 8.5.4 Accuracy and Integrity Evaluation

The WAAS accuracy evaluation  $f_{acc}^u$  is simply a percentage of the benefit for each user location, which is broken down from the \$610 million total based on the population multiplier for that site. Perfect navigation (i.e., zero position error) gets 100% credit, a  $2\sigma$  vertical error of 2.1 m gets 99%, 4.1 m (the ILS requirement) gets 90%, and 7.6 m (the WAAS RNP requirement) gets only 20% (since it is at the outermost limit of acceptability). A cubic polynomial fit gives, for a resulting  $2\sigma$  vertical accuracy  $a$  (m):

$$f_{acc}^u = 1 - 0.005a + 0.0052a^2 - 0.0024a^3 \quad (8.7)$$

where  $f_{acc}^u$  is in decimal terms (i.e. from 0 to 1).

Converting the RAIM user cost of Section 8.3.1 to this value framework requires two further assumptions. Recall that in Probabilistic Risk Assessment (PRA), it is considered valid to assign cost values to fatalities if the underlying risk is sufficiently small (below  $10^{-4}$  - see [1-1]), which it is for the level of integrity provided by GPS. Assuming an average (based on the breakdown of aircraft sizes for Cat. I approaches) of 100 fatalities per fatal incident and a conservative “value per life” of \$10 million (which is actually assessed as \$10 per  $10^{-6}$  mortality risk), each fatal accident incurs a loss of about \$1 billion. Since 3 million approaches are foreseen over a 4-year life cycle, and a fatal accident implies a cost of  $1/R_c = 10,000$  in the RAIM cost model of Section 6.4.2), we can convert from RAIM cost ( $C_{RAIM}$ ) to overall value ( $f_{integ}^{tot}$ ):

$$f_{integ}^{tot} = \frac{(\$1 \times 10^9)(3 \times 10^6)}{10^4} C_{RAIM} = 3 \times 10^{11} C_{RAIM} \quad (8.8)$$

Note that this calculation is also broken down by user location and population multiplier within the RAIM user cost optimization (Section 8.3.3). Also note that the non-availability cost per approach (0.0125) from Section 8.3.1, which is included in the integrity evaluation, implies a nuisance equivalent to about an hour of added aircraft cost, including all consequent delays.

#### 8.5.5 *Value of 4-WRS Baseline European WAAS*

The accuracy of the baseline 4-WRS European WAAS network (shown in Figure 8.1) translates into a accuracy multiplier (weighted by *PM*) of 0.958, giving an overall user benefit of \$584 million. However, the RAIM user cost of 0.0019 from Table 8.1 translates (using (8.8)) into an integrity cost of \$570 million, which is about equal to the user benefit. Subtracting the acquisition cost of \$33 million gives a final value of -\$19 million; thus this network is clearly insufficient. Note that the acquisition cost is dwarfed by the benefits and costs that result, indicating that additional augmentations would be very cheap relative to the possible performance improvement. Also, the fact that a 4-WRS WAAS network cannot provide sufficient integrity suggests that proposed augmented-GPS systems for large regions of Europe that are based on one or two DGPS sites would be insufficient as well, even though they may meet the Category I accuracy requirements [6-16].

### 8.6 “First-Generation” WAAS Results

In attempting to run the GA optimization code on the European WAAS problem, it was discovered that the software needs to be re-written for parallel processing and that a computer with sufficient available processors will be needed to evolve a population of networks toward optimal convergence. However, it is possible to conduct a first-generation evolution using the GA operators and manually investigate some of the networks that result. Results for two of these variants are shown here.

Figure 8.6 shows 95% vertical accuracy contours for a network coded [111110000000], which is simply the base 4-WRS network plus a fifth WRS in Padua, Italy, in south-central Europe. Compared to Figure 8.1, accuracy over highly-populated

central Europe is somewhat better, resulting in an accuracy benefit of \$590.5 million. In monetary terms, this benefit is small because the accuracy shown in Figure 8.1 is already good enough to support Category I operations, at least in principle. More importantly, integrity risk (see Table 8.4 below) has decreased by a factor of 5.6 to give a cost of \$134.7 million from (8.8). The acquisition cost is still only \$35 million, giving a final user value of \$420.8 million. The addition of a single WRS in a beneficial location thus has resulted in a design that appears feasible.

Figure 8.7 shows vertical accuracy for a network coded [111110000101]. This adds a sixth WRS in Cracow, Poland, and it also uses the additional GEO at 15° E, giving an acquisition cost of \$62 million. Although the accuracy contours continue to improve, the benefit has only slightly increased to \$593 million. However, the addition of the GEO satellite has made a further substantial improvement to integrity. All failure trials were available, and the integrity cost from Table 8.4 and (8.8) has fallen to just \$22.3 million. The total value for this network is \$508.7 million, which is \$87.9 million higher than for the 5-WRS system. This result demonstrates that the addition of the 15° E GEO satellite is desirable even if (as is likely) its acquisition cost is much higher than \$25 million. The

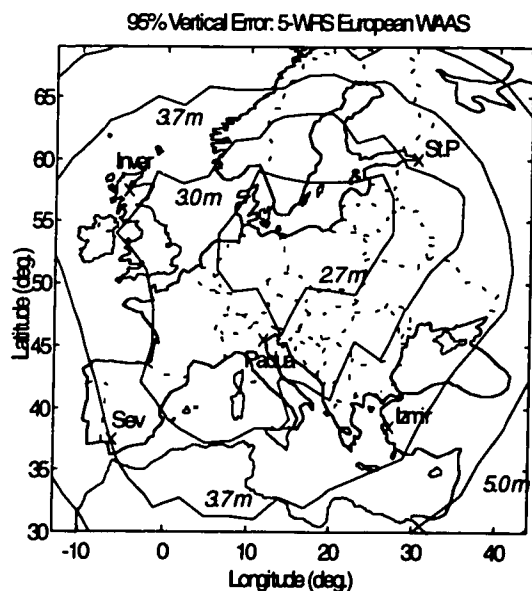


Figure 8.6: Accuracy of 5-WRS Euro. Network

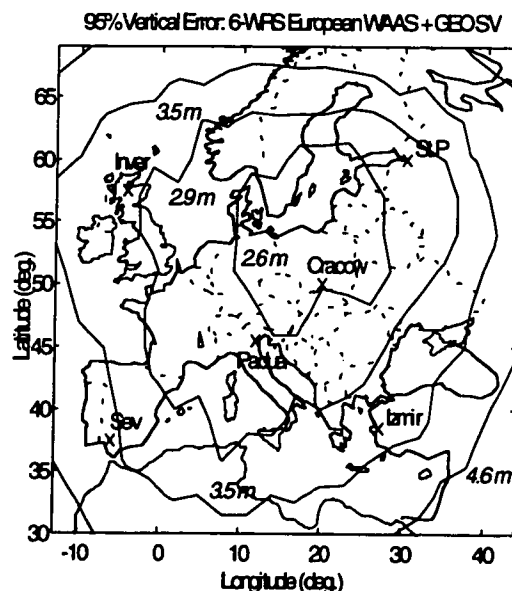


Figure 8.7: Accuracy of 6-WRS + GEO Network

“break-even” point between these two network designs would be reached if the additional cost of the GEO and the sixth WRS together were \$115 million instead of the \$27 million forecast here (this is obtained simply by adding \$27 million and \$88 million).

Figures 8.8 and 8.9 display 95% UIVE and UDRE, respectively, for this 6-WRS + GEO network variant. In Figure 8.8, the addition of two WRS's and a geostationary ranging source toward the center of Europe (compared to Figure 8.2) has improved UIVE in that region substantially -- a 95% UIVE floor of under 0.4 meters results, which is about a 20% improvement. On the other hand, UIVE toward the fringes of the network (e.g., north of Scotland) has improved hardly at all. In Figure 8.9, it is clear that the additions in Central Europe have removed the strange UDRE “hump” over southern Germany that appeared in Figure 8.3. Instead, the 95% UDRE contours have smoothed out into a regular oval format as seen in the UIVE plots, and a wide swath of Central Europe now has a 95% UDRE of under 0.75 meters. As in Figure 8.8, improvement (relative to Figure 8.3) along the edges of the plotted user coverage area is minimal.

Due to both the UIVE and UDRE improvements shown in Figures 8.8 and 8.9, overall user accuracy over Central Europe in Figure 8.7 is significantly better (about 10-15%) than for the baseline design in Figure 8.1. The accuracy component of the user

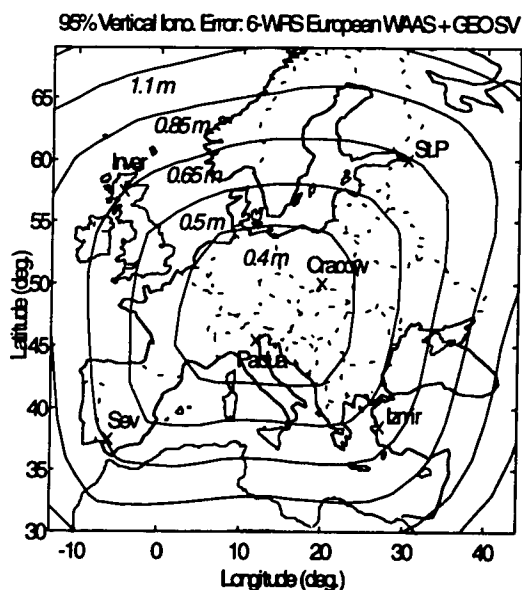


Figure 8.8: UIVE for 6-WRS + GEO Network

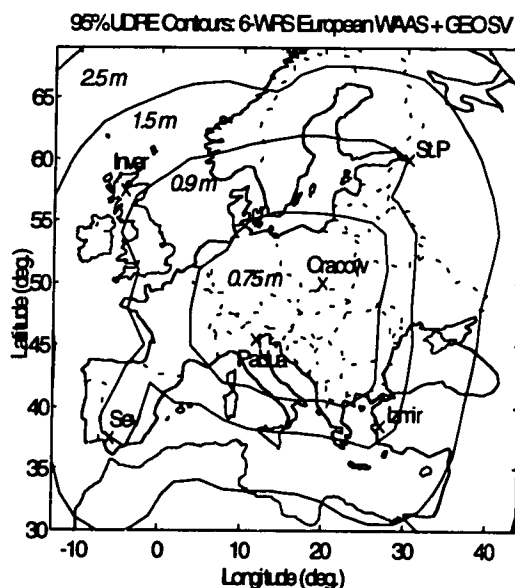


Figure 8.9: UDRE for 6-WRS + GEO Network

value function in Section 8.5.4 does not place great value on these improvements, but they are definitely significant in the integrity evaluation, as evidenced by the dramatic integrity cost reductions mentioned above. This points out the importance of designing networks whose accuracy results are both sufficiently small and well-behaved (i.e. simple contour shapes). Limited as we are to a tail-probability inflation way of simulating rare-normal and failure events in Section 8.3, integrity acceptability depends on having good accuracy (better than the 95% error requirements) throughout the areas of significant importance to the value function.

As mentioned above, Table 8.4 gives the relevant integrity parameters for both of these network designs. The results shown here are for the original simulation approach, where user geometries are classified by VDOP. The integrity improvement as the specified additional augmentations are added is both consistent and impressive. Note that availability, bad position probability, false alarm rate, and missed detection probability all improve from the baseline design (Table 8.1) to the 5-WRS design *and* from the 5-WRS to 6-WRS + GEO design. Based on the discussion and results of Section 8.3.5, we may expect these results to improve as much as 10-15% further if maximum slope were used as the availability parameter.

This research will progress further as the computer capability necessary to run more GA generations becomes available. Currently, more evaluations of the first and

Category	5-WRS	6-WRS + 15° GEO
<i>RAIM user cost</i>	0.000449	0.0000743
Prob(available)	0.9998	1.0
VDOP availability limit	4.6	N/A
Pr(error > RPE)	$1.4 \times 10^{-5}$	$6 \times 10^{-6}$
Pr(MDlerror > RPE)	0.055	0.020
Pr(false alarm)	0.0002	$3.2 \times 10^{-5}$
<i>Fatal Accident Prob/approach</i>	$4.5 \times 10^{-8}$	$7.4 \times 10^{-9}$

**Table 8.4: Integrity for European WAAS Variants**

second-generation GA designs are being run and compared to manually-designed alternatives. The point of “diminishing returns” beyond which further augmentations are not cost-effective has not yet been conclusively found. However, given that the GEO (which is clearly a desirable addition) may actually cost \$50-100 million, it seems that the gap in accuracy-minus-integrity value of only \$120 million between the two networks evaluated here suggests that the 6-WRS + GEO network is approaching that point.

## **8.7 Conclusions and Further Work**

Given the current state of information about augmented DGPS systems (WAAS in particular), it is difficult to make predictions regarding WAAS system-level performance from which network design decisions can be made. We have succeeded in doing so by developing algorithms that combine covariance propagation to determine position accuracy for large areas of potential users with failure-case simulations that incorporate the best available current knowledge. Further improvements in these prediction methods are possible, including fitting more detailed error models to the rapidly-growing Stanford WAAS database. Better models of ground integrity can also be developed, allowing us to add detailed ground integrity monitor optimization to our current optimal-RAIM integrity prediction algorithm. Finally, the wealth of data to be collected by the FAA’s NSTB starting in 1997 should dramatically reduce our uncertainty about potential failure sources, most notably including ionospheric spatial decorrelation.

The augmented-GPS network optimization results we have achieved to date are impressive. We have demonstrated the policy-level feasibility and desirability of using WAAS to provide Category I precision approach capability to Europe with the network designs of Section 8.6, and we are continuing to search for the best possible combination of WRS’s and geosynchronous satellites to accomplish this. The 6-WRS+GEO (3 GEO’s total) combination looks very promising, as it meets all Category I requirements and provides a value benefit of over \$500 million, depending on the cost of the GEO.

It is of interest to compare the “bottom-line” results achieved through value optimization to what is expected by WAAS requirements published to date. Table 8.5

compares the predicted performance of the 6 WRS + GEO network to the relevant accuracy and integrity requirements chosen by the FAA in the WAAS Specification [8-16]. The “service volume” defined there is taken to include an oval that passes through the outermost of the 11 evaluation sites plus a 10% additional area on the edges of this oval. Note that the requirements focus on GIVE rather than UIVE, which was plotted in Figure 8.8. GIVE can be extracted from the coverage prediction outputs, but this has not yet been done for the European case. Since UIVE from Figure 8.8 (which includes user gridpoint extrapolation error in addition to grid estimate error) would meet the GIVE requirement, this is assumed to be met.

The key discrepancy between the requirements and the optimal network results comes from the tri-axis integrity/continuity/availability tradeoff. The requirements listed in the last three rows of Table 8.5 come from a combination of ILS heritage and augmented-GPS performance expectations. They do not necessarily represent a well-thought-out tradeoff between these three performance classes. In this case, the continuity requirement can only be met if RAIM is used as a single snapshot check as the aircraft approaches the ceiling height. RAIM will probably operate throughout the approach, however, and it is not clear whether sufficient time correlation will be present to allow it to be met in practice. What is clear is that the per-epoch HMI (“Hazardously Misleading Information”) result, obtained by multiplying the bad position and missed detection probabilities from Table 8.4, is three times the listed requirement for the entire approach.

Requirement Class	WAAS Spec.	Europe Perform.	Req't. Met?
vert. position accuracy (95%)	7.6 m	$\leq 3.5$ m	YES
UDRE (99.9%)	1.5 m	$\approx 1.5$ m	YES
GIVE (99.9%)	1.5 m	$\approx 1.3$ m (UIVE)	YES (probably)
Prob{availability}	99.9%	100%	YES
Prob{continuity loss}	$5.5 \times 10^{-5}$ / appch.	$3.8 \times 10^{-5}$ / epoch	YES*
Prob{HMI - hazardous info.}	$4 \times 10^{-8}$ / appch.	$1.2 \times 10^{-7}$ / epoch	NO

\* depends on number of RAIM decision points per 150-second approach

**Table 8.5: European WAAS Requirements Comparison**

The allowed HMI probability in the WAAS specification is motivated by the risk allocation done in the RNP [6-10] discussed in Section 5.3.4, but the allocation of  $4 \times 10^{-8}$  for Category I WAAS (taken from the ORD [6-11]) assumes that the same risk applies as when an approach is taken all the way to the ground (Cat. IIIB). It should once again be noted that the definition of HMI as a vertical navigation error exceeding 19.2 meters does not introduce as much consequent risk to the aircraft when it should not continue the approach below the 200-ft decision height. Thus, the significance of not meeting the HMI number is questionable. If the RAIM cost model for WAAS introduced in Section 6.4.2 and revised in Section 8.3.1 is a fair representation of the original RNP risk allocation, the result obtained from the optimal-RAIM approach better captures the intent of the requirement than the requirement itself does. Otherwise, if the FAA requirement is taken as gospel, then the RAIM non-availability cost could be lowered (since it now gives 100% availability) until the specified HMI probability is achieved. The overall requirement would then be met if the resulting availability probability remains above 99.9%.

Current plans include expanding the applicability of our optimization approach by revising the assumptions of European value model for networks in North America and the rest of the world. Furthermore, the same basic approach will be used for optimal LAAS architecture design beginning in 1996. The LAAS case will focus more on single-airport operations, but the same fundamental cost-versus-performance tradeoffs will need to be made. As noted before, our ability to make this vision of augmented-GPS evolutionary optimization a reality requires implementing the coverage prediction and integrity simulation software on a multi-processor computer. This is intuitively easy because the evaluation of accuracy or integrity for each user location is a similar process that can be done simultaneously for as many locations as there are available processors. Stanford's GPS research groups plan to acquire a workstation with at least 16 fast processors by early 1996. This computer will be used for extensive simulations of both LAAS and WAAS architectures, as Stanford is tasked by the FAA to evaluate the cost-benefit performance and certifiability of various competing LAAS systems. This work will utilize and further develop the GPS evaluation and optimization techniques reported in this chapter.

## Chapter 9: Summary and Recommendations

*We are in an age that assumes the narrowing trends of specialization to be logical, natural, and desirable.... Advancing science has now discovered that all the known cases of biological extinction have been caused by over-specialization, whose concentration of only selected genes sacrifices general adaptability.... Specialization has bred feelings of isolation, futility, and confusion in individuals. It has also resulted in the individual's leaving responsibility for thinking and social action to others. Specialization breeds biases that ultimately aggregate as international and ideological discord, which, in turn, leads to war.*

- R. Buckminster Fuller (1894-1983)

*One machine can do the work of fifty ordinary men. No machine can do the work of one extraordinary man.*

- Elbert Green Hubbard (1856-1915)

### 9.1 Systems Design and the Decision Paradigm

Despite covering a lot of ground and jumping among several applications, this thesis has at its center the goal of validating the use of decision-theoretic concepts on practical aerospace design problems from a top-down perspective. In each application studied, a probability model of performance and/or risk was developed at the mission level, where the events can clearly be linked to gains and losses of the end-state user. This model contained within it measures of the uncertainty present in the parameter values applied to a given application. These uncertainties were given probability distributions of their own. As a result, the same Monte Carlo simulation that evaluated system performance could sample from this distribution as well, giving a reasonably complete model of the total uncertainty picture.

Optimization techniques that allow system evaluation by Monte Carlo simulation (with the consequent statistical uncertainty) make it possible to search the conceptual “decision tree” of system and subsystem design options for the globally optimal solution. While there is no guarantee of finding the global optimum, methods based on evolutionary optimization provide this capability. In this thesis, algorithms based on Simulated Annealing and Genetic Algorithms have been demonstrated to be effective in optimizing both spacecraft redundancy design and GPS augmentation network design. Both methods

can be applied to almost any design problem because they allow creative encoding of the design solution space. This encoding is important because it can dramatically improve the efficiency of the search process through its incorporation of problem-specific knowledge.

Performing design optimization based on Monte Carlo sampling of a complete system risk model effectively results in the solution of the master “decision tree” illustrated in Figure 1.4. In this way, an optimal decision (design choice) is made that satisfies all of the requirements of the theory of decision making under uncertainty introduced in Chapter 1. In principle, this approach is superior to the conventional method of deterministic worst-case engineering design. For both the spacecraft and GPS applications, it was demonstrated that decision-theoretic solutions offer practical benefits over deterministic approaches. Chapter 2 of this thesis introduced these concepts with a simplified control/estimation problem utilizing WAAS as the primary position sensor. Even for this linear model, the addition of non-ideal failure modes and non-quadratic cost terms changed the optimal result significantly.

The key limiting factor in handling simulations of a large uncertainty space (in addition to system performance simulations) and then using the results to perform optimal solution search is the computer hardware and software now available. Modern workstations and PC's have made samples of millions of trials feasible. The next step is to parallelize the top-level optimization code (and the simulation code where desirable) to allow the search to utilize many independent processors either in an internally-parallel machine or by farming out simulation tasks over a network. Software to ease the coding of uncertainty and performance simulations continues to improve and become more user-friendly, and intelligent compilers that parallelize code written for single processors are now on the horizon. The end goal is a comprehensive simulation-based optimization package that can be used by engineers who do not need to be familiar with its internal workings.

## 9.2 Reliability-Based Spacecraft Design Optimization

In Chapter 3, methods of modifying traditional reliability and FMEA assessment techniques for spacecraft were developed. These provide a much better picture of the global uncertainty picture for a given spacecraft mission by (1) representing the fact that component failure rates change over the mission lifetime, (2) modeling tabulated failure rates that contain significant statistical uncertainty as lognormal probability distributions which are sampled in simulation-based reliability evaluation, (3) providing a means of representing statistical dependence among components of the same type via uncertainty sampling, and (4) allowing for reliability assessment for components without tabulated failure rates using the probability distributions implied by FMEA analyses. Instead of giving a single system failure probability (as a function of time), the simulation instead gives a histogram of reliability uncertainty at each point in the mission timeline. While this uncertainty picture may be harder to explain to the end user, it provides the capability to make optimal design decisions without sacrificing design effort in a conservative worst-case approach.

Chapter 4 illustrated that this potential is real by using simulation-based reliability models and a variant of Simulated Annealing to perform optimal redundancy allocation for the Gravity Probe-B (GP-B) spacecraft bus. Instead of merely maximizing mean reliability at the end of the primary mission, the optimal design was tailored to value functions that modeled the mission success priorities of both the spacecraft developer (LMSC) and the “customer” (Stanford/NASA). The two optimal designs that resulted demonstrated that many tenets of traditional spacecraft design theory, such as the avoidance of all single-point failures, may be sub-optimal when evaluated within the context of a realistic uncertainty model. For the Stanford value function, the optimal design found was able to improve reliability slightly while reducing the total component cost by 10%.

The key difference between this optimization approach and spacecraft redundancy design as currently practiced is that the traditional approach implicitly presumes a high degree of risk aversion and a uniform level of reliability uncertainty (due to the inaccuracies of traditional reliability calculations) across all component classes. Under

these conditions, it is evident that one should incorporate high minimum levels of redundancy across-the-board. With value functions tailored to each user's preferences and a more discriminating uncertainty model, it becomes possible to do a superior design job on a mission-by-mission basis. This is becoming more and more critical as pressure builds on spacecraft developers to provide current (and future) on-orbit capabilities in much less time and at a much lower cost.

Recall that the master decision tree in Figure 1.4 implied a tie-in between spacecraft-level redundancy optimization and detailed design of each of the relevant subsystems. This only requires linking the top-level optimization code with lower-level performance simulations of each of the subsystems. For example, the guidance-and-control subsystem performance could be simulated in the context of failures sampled at the top level. The redundancy design vector could then be augmented by the key design variables for each subsystem in a single global search, or a greater degree of autonomy could be set up in which each subsystem simulation has its own failure sampling and performs its own "inner-loop" optimization. This latter approach has the advantage of more easily implementing already existing optimization codes, such as the MATLAB optimal control toolbox for guidance-and-control design. This concept will be discussed further in Section 9.4.

### **9.3 GPS Performance Prediction and Optimization**

The concept of design optimization based on simulation of an uncertainty model was easily extended to civilian applications of the Global Positioning System (GPS). Chapter 5 considered the problem of navigation error integrity monitoring for users of the uncorrected Standard Positioning Service (SPS). Receiver Autonomous Integrity Monitoring (RAIM) works reasonably well for this application, since airborne SPS users normally can tolerate errors of at least 550 meters horizontally (the 95% SPS horizontal error specification is 100 m). Traditional RAIM usually assumes specific "worst-case" single-channel failure modes and statistical independence, thereby allowing decision thresholds to be set according to the chi-square distribution to meet specific requirements

for “false alarm” and “missed detection” probabilities. Chapter 5 addressed the adequacy of this approach in a high-uncertainty environment by introducing a “prior probability model” (PPM) for GPS satellite and receiver error performance. As in Chapter 3, the PPM incorporated statistical dependence (and thus the increased likelihood of multiple failures) by sampling from uncertain satellite failure rates which applied to all GPS spacecraft for that trial. In addition, thresholds were set to minimize a mission-level user cost function which, instead of enforcing arbitrary probabilities, allows for trade-offs to be chosen to lower the overall risk of an accident. Monte Carlo simulation was used to sample from both the PPM and the GPS satellite geometries for a fixed user position, and the results were stored such that the optimal thresholds could be chosen by a simple line search.

The results of this effort demonstrated that RAIM optimization based on a complex value model was possible and that reasonably consistent output thresholds were obtained. The optimized RAIM algorithm produced reasonable results for availability and false alarm probability. However, they were higher than published SPS integrity requirements in some cases, especially for the missed detection probability (given an error exceeding the safe limit), which lies between 2-25% for all cases studied in this thesis. This result illustrates that probabilities computed from the chi-square equations and assumptions will likely be very optimistic when compared to those from a realistic failure model. This conclusion was confirmed by a statistical test that demonstrated that the predictions of traditional RAIM equations are unlikely to accurately describe the consequences of the more realistic PPM failure model. However, it should be remembered that, for a minimal-risk approach, the bottom line is that the overall accident risk be within the system-level risk allocation taken from the RNP. This has been shown to be feasible in the SPS RAIM case and to be within reach for applications that require much greater accuracy.

Chapter 6 extended the minimal-cost RAIM approach to the Wide Area Augmentation System (WAAS) Category I precision approach application. A simulation of WAAS corrections and the resulting performance was constructed for a single user location (randomly sampled within CONUS), and a “fault” model was built from

ionospheric delay error data that suggested that the tails of this distribution are much worse than Gaussian. With a revised risk (cost) model to minimize and after changing the satellite geometry measure to VDOP alone, these two models were mated into a large simulation, then optimal thresholds were selected using the line-search procedure used in Chapter 5. The results indicated that published WAAS integrity requirements will be difficult to meet given the current level of ionospheric uncertainty. Several years of ionospheric data collected by experimental WAAS networks will help bring this uncertainty into a manageable zone as the next solar cycle waxes toward a peak around the year 2000.

Chapter 7 moves the focus away from failure uncertainty modeling in order to build a comprehensive model of normal-condition accuracy for the wide geographic distribution of users supported by large-scale WAAS networks. GPS satellite geometries (augmented by geosynchronous communication and ranging satellites) were simulated as before, but the assumption of i.i.d. Gaussian error conditions made it possible to propagate linear-quadratic error covariance matrices forward from reference station (WRS) to master station (WMS) to a collection of user locations. This propagation can be done separately for clock/ephemeris errors and ionosphere errors if the reference stations have dual-frequency receivers.

Results of this “user coverage prediction” method were obtained for small networks such as the Stanford experimental WAAS (and variations of it) and, with the addition of matrix parsing algorithms, the much larger National Satellite Test Bed (NSTB) network of 18-24 WRS’s spread throughout the Continental U.S. (and perhaps Canada and Mexico). The results for 95% vertical accuracy demonstrate that current and proposed Category I accuracy requirements can rather easily be met with relatively sparse WRS networks. Results broken down into clock/ephemeris (UDRE) and ionosphere (UIVE) error sources show UIVE degrades gracefully as users move farther away from the WRS network (*under normal conditions*). UDRE, on the other hand, degrades more quickly for sparse networks. The NSTB is relatively dense; so it provides very good coverage for users over an area much larger than that spanned by the WRS layout itself.

This coverage-prediction method has also shown itself to be applicable to proposed WAAS networks anywhere on the globe. One particular non-U.S. application is the focus of the combined accuracy/integrity WAAS network optimization developed in Chapter 8. A sparse baseline WAAS network designed to cover Europe is proposed and then studied by a combined coverage and integrity analysis, where the coverage accuracy predictions are used to represent “normal conditions” in the tail-expansion failure sampling done by the RAIM optimization code. A system-level user value model for Category I precision approaches was then constructed that places a monetary value on supplying Category I coverage over a wide area, subtracts the cost of the network, and then penalizes the result by a cost based on post-RAIM accident risk. This value model was utilized in a genetic algorithm-based search for the optimal combination of reference-station and geostationary-satellite WAAS augmentations.

Although only a few generations of the GA could be completed in a reasonable time, they resulted in at least two feasible designs. One of these, which included 6 WRS's and a third geostationary satellite at 15° E, gave the best result and appeared able to meet the FAA WAAS specifications for accuracy, availability, and continuity. Once again, an optimal-RAIM approach was not able to meet published integrity requirements, but the risk implied by the minimized integrity cost fits well with the RNP risk allocation, suggesting that the specified integrity numbers are simply too tight. They were chosen partly because traditional RAIM calculations suggested that they could be met. As in Chapter 5, the more generalized optimal-RAIM approach can point out where the assumptions of traditional RAIM are partially invalid while making its own evaluation based on more fundamental measures of risk.

The next step in pursuing GPS system evaluation and optimization under uncertainty, in addition to improving the error and failure models, is to run the overall operation on a multi-processor workstation (as discussed in Section 9.1). There are many parts of the overall task which are easily separated and can run independently and simultaneously if multiple processors are available. Two orders of magnitude of increase in effective computing speed are expected from a state-of-the-art workstation with at least 16 floating-point-optimized processors. This would make it possible to run the GA search

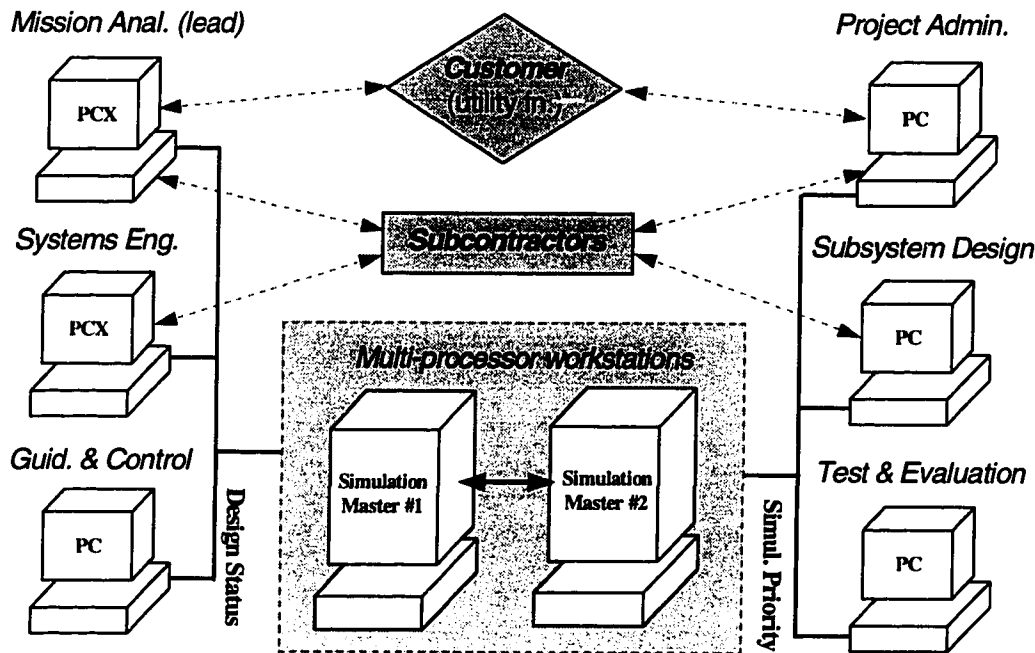
of Chapter 8 to convergence. At the same time, the error-propagation algorithms can be modified to both evaluate LAAS performance and do a cost-benefit optimization of the elements of such a system for Category II/III aircraft precision approaches and landings.

#### **9.4 The Future of Aerospace Design**

In the foreseeable future, it is likely that the traditional systems design approach will remain recognizable but will gradually be modified to take advantage of the spread of high-powered computer analysis capabilities. For one, extensive simulation can now be done on personal computers using the latest simulation software tools and powerful Pentium® or PowerPC® processors. Another example is the use of network tools such as Lotus Notes® to link distant engineers in an on-line environment where cooperative effort is made to develop requirements (and later specifications) for new technology development. Boeing Commercial Airplane Group is using such a system to link its engineers with subcontractors who are working together on requirements for augmented-GPS precision landing systems.

A key question for the future is whether these developments, taken together, will allow aerospace technology development and actual hardware design to be undertaken by small start-up companies. Firms of this type have revolutionized the electrical and computer fields and produce a staggering array of new products despite very low cost structures. In contrast, aerospace system development requires very costly effort by many hundreds of engineers to handle detailed requirements and specification compliance at many interlocking levels of detail in addition to the basic work needed to develop a new product.

Figure 9.1 shows an outline for a start-up-sized company focusing on all levels of aerospace design, either as an independent company with manufacturing subcontractors or as the design branch of a larger company. It shows a set of personal computers surrounding a core of multi-processor simulation workstations. The central computer core not only runs simulations constantly on its own processors; it manages the flow of information to the individuals responsible for intermediate and final decisions. Two key



**Figure 9.1: A Future Aerospace Design Firm**

individuals are required to handle *mission analysis* (interfacing with customers and defining the mission to design for) and *system engineering architecture* (connecting the subsystem design elements into a unified whole). These individuals will have advanced graphical PC's (denoted as PCX) which will allow them to monitor the current status of simulation-based evaluation and optimization in real-time. They will then have the capability to inject new or revised value models and/or design constraints to help redirect the search according to their own or their clients' experience and qualitative preferences. The mission analysis person can also prioritize the simulation jobs waiting to be run.

The computer network shown in Figure 9.1 aims to allow an unparalleled level of second-by-second interaction between the work of several small design teams. The top-level simulation will effectively budget varying amounts of performance and expenditure to each of the subsystems based on the feedback of the application-specific optimizations being run simultaneously for each one. This means that all of the project engineers will have and be able to utilize up-to-date information about all of the systems with which their design responsibility interacts. Furthermore, the computer network will be able to handle

lower-level changes much more rapidly than teams of human engineers. Research on combining dissimilar top-level and subsystem-specific optimization codes into a uniform search process is reported in [2-3,18].

In addition, the workstation core, which has primary responsibility for the system-level simulations, will be able to assign simulation jobs to any of the other networked computers that is currently underutilized. By taking advantage of the processing power of modern PC's sitting on everyone's desk, the simulation controller will have the equivalent of dozens of Pentium-class processors available for simulation tasks, especially at night when most of these computers would otherwise sit idle.

It is believed that a company centered on an interactive simulation-based optimization computer network will be able, by fully evaluating the master decision tree of Figure 1.4, to achieve with 10-20 people and 2-6 months what it takes hundreds of people and 3-5 years to do today. This may be the critical factor in bringing the conservative aerospace design establishment into the "modern age" so that price and time-competitive aerospace products can be developed to compete with other areas of research. Innovations that provide new capabilities to society in less than a year and at hundreds of dollars per person are now capturing most of society's allocation of money to new technology development. Considering the time and money required to develop aerospace systems today, many have argued that further development of aerospace technology will not provide the same return to society as an equivalent investment in other enabling technologies (i.e., computers, biotechnology). The challenge has thus been presented -- aerospace design must learn to adapt to a world with that insists on realistic risk/return tradeoffs and that cannot provide a guarantee of safety, or else it will face an ever-growing sense of obsolescence in the 21st Century.

## **Appendix A**

### **Axioms of Decision Theory and Required Assumptions**

This section summarizes the background assumptions required by the theory of Decision Analysis so that its methods produce the optimal decision for a given problem [1-3]. The following postulates and axioms form the structure of a decision-analysis problem. In particular, the so-called “Rules of Actional Thought” that follow ensure that the decision maker’s preferences among possible outcomes can be ordered in a logical way. Decision theory tries to avoid value judgments regarding the “sanity” of the decision maker’s preferences. For example, it allows for a person who prefers less money to more so long as consistency is maintained. Decision makers may conceivably have preferences that do not fit these rules. An “optimal decision” could still exist for such a person, but the underlying inconsistencies prevent this optimal choice from being arrived at through the decision-analysis procedures used in this thesis.

#### ***The Rules of Actional Thought***

(1) ***Probability Rule***: The decision maker is required to express his information and judgment as to the possible future outcomes related to his decision in the form of *possibilities* and *probabilities*. In other words, the outcome space within which the decision will be made should be modeled as a realm of possible results, each having a probability of occurrence assigned by the model to express the available information as accurately as possible. We can illustrate this model by a canonical *decision tree* of possibilities and probabilities, using a sports-event weather prediction problem as an example (see Figure A.1).

In this case, the possibility of a game postponement is a function of the probability of good weather. From this tree, it is easy to compute the marginal probability of postponement:

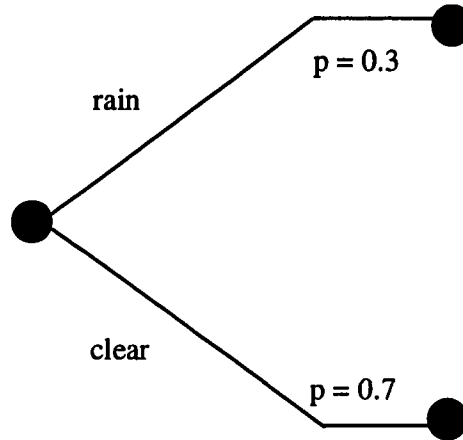
$$\text{Pr}(\text{postpone}) = (0.7)(0.05) + (0.3)(0.6) = 0.215$$

This framework is used to structure our thinking about probabilities and consequences as developed by the ensuing methodology.

(2) **Order Rule:** This rule simply requires that a decision maker order his or her preferences among the possible outcomes in an unambiguous way from best to worst. Continuing with the game postponement problem, let us suppose that the decision maker

must decide whether or not to go to the game (before finding out what the game-time weather will be or whether the game will be postponed). Using postponement as the main outcome variable, the decision maker may order his preferences as follows:

<b>Best:</b>	attend game, no postponement
	do not attend, postponement
	do not attend, no postponement
<b>Worst:</b>	attend game, postponement



**Figure A.1: Probability Tree for Weather Problem**

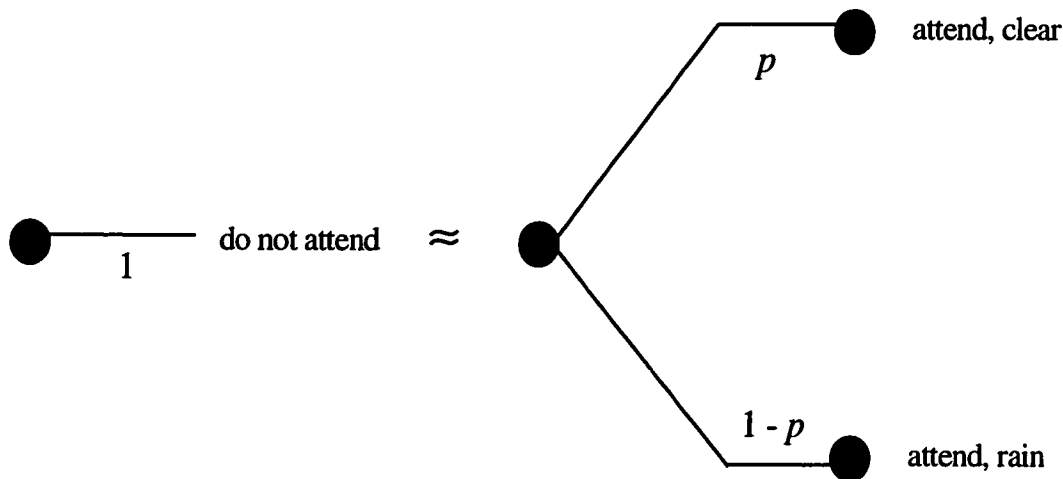
This is all that is required to satisfy the order rule. Note that if one were *indifferent* to whether or not the game was postponed assuming one did not attend, the two “do not attend” cases would effectively merge into one, and the order rule would still be satisfied.

What is expressly forbidden by the order rule is a circular list of preferences. If a person prefers outcome B to outcome A and outcome C to outcome B, his list of preferences should be {C,B,A}. If he then stated that he preferred A to C, it would become impossible to clearly distinguish which outcome is most preferred. This would

make one a so-called “money pump,” since one would in principle be willing to trade something to exchange A for B (since B is preferred over A), B for C, *and* C for A; thus being susceptible to a never-ending series of payouts in exchange for no long-term benefit. Since this fallacy is obvious, the order rule is usually satisfied by normal common sense. Domains in which circular preferences may seem to be evident do exist, however. One example is investment markets in which the same underlying asset exchanges hands repeatedly at ever-higher prices due to the so-called “greater fool” theory, in which the apparent upward trend in the value of the asset causes investors to steadily upgrade its worth to them [1-2]. Even in this case, there is normally a set preference order at each decision step. The apparent upward trend simply causes the investor to modify his preferences, causing him to buy the asset back at a higher price than he sold it for. The order rule is thus satisfied at each step, even though the overall trend of preference changes may look questionable.

(3) *Equivalence Rule*: This rule states that a decision maker should value his preferences in a way that is consistent with the order rule. Specifically, given the preference order {C,B,A}, the decision maker should be able to choose a *preference probability* ( $p$ ) such that he would be indifferent between (i) accepting the middle outcome B with certainty and (ii) accepting a lottery in which he received C (the best outcome) with probability  $p$  and A (the worst) with probability  $1-p$ . Note that higher values of  $p$  represent a lesser implied value of the best outcome (C) relative to the middle outcome (B), as the decision maker is insisting on a higher probability of getting the best outcome in return for foregoing a guarantee of receiving the middle one. This choice of a preference probability thus specifies an outcome value model that can be used to evaluate all alternatives.

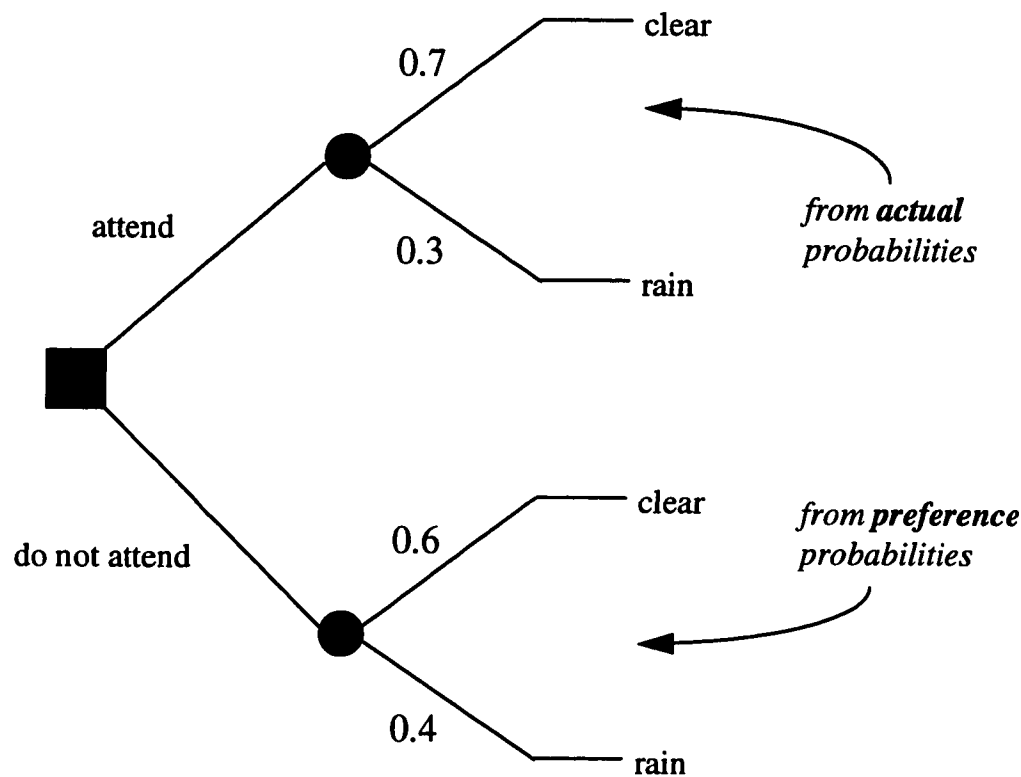
For our game postponement example problem, let us suppose that the decision maker’s preference for the “do not attend” outcome is independent of whether or not the game is actually postponed. He thus has three separate outcome preferences, with “do not attend” being in the middle. This preference situation is illustrated in Figure A.2.



**Figure A.2: Weather Problem Preference Probability Assignment**

Let us suppose that our decision maker chooses  $p = 0.6$  as his preference probability. Thus, he would prefer to go to the game if the chance of no postponement were better than 60%. If it were lower, he would prefer to stay home. Since the decision maker is indifferent between the options on the left and right-hand sides of Figure A.2, we define the middle outcome “do not attend” as the *certain equivalent* of the lottery shown on the right side (with this user-chosen preference probability). Clearly, we can now compare this preference probability to our uncertainty-model probabilities indicated in Figure A.1 (from which we can compute the marginal probability of the game being postponed) to determine which course of action our decision maker should take. The following rule formalizes this procedure.

(4) ***Substitution Rule:*** This rule allows us to *substitute* the preference-probability lottery shown on the right side of Figure A.2 for any representation of the decision “do not attend” in a formal decision tree. In essence, this allows us to cast the *outcome value* of this decision in terms of probabilities of the other two more extreme outcomes. The optimal decision (either to go or not to go) can then be determined in “probability space” from the resulting decision tree shown in Figure A.3 (in which Figure A.2 is substituted into Figure A.1).



**Figure A.3: Completed Decision Tree for Weather Problem**

(5) **Choice Rule:** In the situation represented by Figure A.3, we see that on the top half (for the “attend game” decision), the probability of no game postponement (clear -- the best outcome) is 70%. In the “do not attend” case on the bottom, preference probabilities determined by the decision maker equate to a 60% chance of the best outcome. We now compare the actual probability result on the top to the preference probability on the bottom (substituted for the “do not attend” decision). The choice rule simply requires that we accept the alternative with the higher probability of our ideal outcome as the preferred decision. Since our assignment of preference probabilities directly implies a user “value model” for this situation, this means that the decision maker must choose the alternative that *maximizes* his value model (or *minimizes* his “cost model”) for the uncertainty model he has constructed for this problem.

This rule just requires the decision maker remain consistent and choose the best alternative identified by our decision-under-uncertainty methodology. Of course, the decision-maker may be surprised by the result and determine that his value function did not properly model his preferences. He may thus revise his value model (and/or revise his uncertainty model) and repeat the decision-tree analysis. This iterative process is key to the practical application of decision analysis theory, but one must be careful not to fool oneself by altering the model to justify a preconceived decision. These “rules of actional thought” contain no protections against self-delusion. They simply represent a framework of assumptions under which decisions can be shown to be optimal given the specified user preferences and the best possible uncertainty model of the available information.

## **Appendix B**

### **Values and Tools for Longitudinal Aircraft Control Model**

#### ***B.1 Matrix Values for 747 Longitudinal Control Problem***

Section 2.2.1 introduced the problem of designing a linear feedback controller and estimator for the longitudinal motion of a Boeing 747 near sea level based on a linearization about steady level flight. This example is taken from [9-2,3]. The six states, two control inputs, and three measurements (one of which has been altered to represent the WAAS glideslope deviation signal) are described in that section. All units are expressed in terms of feet, seconds, and centiradians (crad). Using the continuous state-space format of (2.1), the plant and input parameter matrices are:

$$\mathbf{A} = \begin{bmatrix} -.021 & .122 & 0 & -.322 & 0 & 1 \\ -.209 & -.530 & 2.21 & 0 & 0 & -.044 \\ .017 & -.164 & -.412 & 0 & 0 & .544 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 2.21 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -.25 \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} .010 & 0 \\ -.064 & 0 \\ -.378 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & .25 \end{bmatrix} \quad (\text{B.1})$$

The wind vector has spectral density (covariance)  $\mathbf{W}_c = 0.9 \text{ ft}^2/\text{sec}$ . Note that for wind gust disturbances,  $\mathbf{B}_w$  is the negative of the first two columns of  $\mathbf{A}$  except  $\mathbf{B}_w(5,2) = 0$ . The sensor error covariance is estimated to be the following diagonal matrix:

$$\mathbf{V} = \begin{bmatrix} 2^2 & 0 & 0 \\ 0 & 2^2 & 0 \\ 0 & 0 & 8^2 \end{bmatrix} \quad (\text{B.2})$$

where the third entry is based on an assessment of mean WAAS  $1\sigma$  vertical position error of 8 ft, or about 2.45 meters. This level of accuracy under normal conditions should be achievable over a large area (see Chapter 7).

The random state initial condition  $x(t = 0)$  has the following diagonal covariance matrix under normal conditions:

$$\mathbf{X}_0 = \begin{bmatrix} 0.23^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.16^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.14^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.47^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.32^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.23^2 \end{bmatrix} \quad (\text{B.3})$$

The model in [9-2,3] can be discretized for a  $T_s = 1$  Hz sample rate using the MATLAB "c2d" function [9-2,7]. The result is expressed in terms of (2.2):

$$\mathbf{A}_d = \begin{bmatrix} .968 & .122 & 0 & -.322 & 0 & 1 \\ -.138 & -.530 & 2.21 & 0 & 0 & -.044 \\ .025 & -.164 & -.412 & 0 & 0 & .544 \\ 0.012 & -.058 & .777 & .999 & 0 & .217 \\ 0.088 & -.779 & .152 & 2.20 & 1 & .067 \\ 0 & 0 & 0 & 0 & 0 & .779 \end{bmatrix}; \quad \mathbf{B}_d = \begin{bmatrix} .011 & .114 \\ -.346 & .026 \\ -.290 & .054 \\ -.160 & .019 \\ .012 & .004 \\ 0 & .221 \end{bmatrix} \quad (\text{B.4})$$

In order to translate the covariance  $\mathbf{W}_c$  to the discrete model of (2.3), the discrete state covariance at the sample times  $k = n T_s$  ( $n = 0, 1, 2, 3, \dots$ ) must be the same as that given by the underlying continuous Gauss-Markov process. The Van Loan algorithm is used in the MATLAB function "cvrtq()" to give the following result for this case [9-3]:

$$\mathbf{W}_d = \begin{bmatrix} .009 & -.042 & -.009 & -.005 & .015 & 0 \\ -.042 & .269 & .043 & .026 & -.113 & 0 \\ -.009 & .043 & .010 & .004 & -.013 & 0 \\ -.005 & .026 & .004 & .004 & -.013 & 0 \\ .015 & -.113 & -.013 & -.013 & .065 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (\text{B.5})$$

$\mathbf{W}_d$  in this case incorporates  $\mathbf{B}_w$  and affects five of the six states, and the  $5 \times 5$  sub-matrix  $\mathbf{W}_d(1:5, 1:5)$  is a valid positive-definite covariance matrix.

### B.2 LQR Cost Determination from Covariance Matrix Propagation

It is possible to evaluate the true cost of any stable design by propagating state and estimate error covariance matrices forward in time as given by the step-ahead "prediction" estimator in [9-3]:

$$\bar{\mathbf{X}}(k+1) = [\mathbf{A}_d + \mathbf{B}_d \mathbf{K}] \bar{\mathbf{X}}(k) [\mathbf{A}_d + \mathbf{B}_d \mathbf{K}]^T + \mathbf{A}_d [\bar{\mathbf{P}}(k) - \hat{\mathbf{P}}(k)] \mathbf{A}_d^T; \quad \bar{\mathbf{X}}(0) = \mathbf{X}_0 \quad (\text{B.6})$$

$$\hat{\mathbf{P}}(k) = \bar{\mathbf{P}}(k) - \bar{\mathbf{P}}(k) \mathbf{C}_s^T [\mathbf{V} + \mathbf{C}_s \bar{\mathbf{P}}(k) \mathbf{C}_s^T]^{-1} \mathbf{C}_s \bar{\mathbf{P}}(k) \quad (\text{B.7})$$

$$\bar{\mathbf{P}}(k+1) = \mathbf{A}_d \hat{\mathbf{P}}(k) \mathbf{A}_d^T + \mathbf{W}_d \quad (\text{B.8})$$

where  $\mathbf{W}_d$  and  $\mathbf{V}$  are the constant discrete noise covariance matrices given above,  $\mathbf{X}_0$  is the covariance of the initial state vector, and  $\bar{\mathbf{P}}$  and  $\hat{\mathbf{P}}$  are the state estimate error covariances before and after the LQE Kalman filter measurement update, respectively. For each time step,  $\bar{\mathbf{P}} \geq \hat{\mathbf{P}}$  (i.e., the measurement  $z(k)$  always adds information) such that (B.6) is well-posed. These covariance matrices tend to a steady-state for the time-invariant LQG formulation, where the steady-state covariance  $\bar{\mathbf{X}}_{ss}$  can be obtained by solving the *discrete Lyapunov equation* formed when  $\bar{\mathbf{X}}_{ss}$  is substituted for  $\bar{\mathbf{X}}(k)$  and  $\bar{\mathbf{X}}(k+1)$  in (B.6) (see [9-3] for further details).

The expected-value LQR cost at each step can then be computed by plugging  $\mathbf{X}(k)$  into the following general equation for  $k = 1, \dots, N_t$ :

$$\bar{J}_{\text{LQR}} \equiv E[J_{\text{LQR}}] = \frac{1}{N_t} \sum_{k=1}^{N_t} \text{trace}\{(\mathbf{Q} + \mathbf{K}^T \mathbf{R} \mathbf{K}) \mathbf{X}(k)\} \quad (\text{B.9})$$

where:  $\mathbf{X}(k) = \bar{\mathbf{X}}(k) + \bar{\mathbf{P}}(k) \quad (\text{B.10})$

This method of calculating cost is not used in Chapter 2 because the intent of that chapter is to examine the capability of simulation-based optimization in the presence of non-Gaussian system failures, for which the covariance approach will not work.

## **Bibliography**

### **1.0 Decision and Risk Analysis**

- [1-1] Graham, John D. and Vaupel, James W., "The Value of a Life: What Difference Does It Make?", *Risk Analysis*, Vol. 1, No. 1, 1981, pp. 89-95.
- [1-2] Howard, Ronald A., and Matheson, James E., Eds., *Readings on the Principles and Applications of Decision Analysis*. Menlo Park, CA.: Strategic Decisions Group, 1983.
- [1-3] Howard, Ronald A., *Class Notes for EES 231: Decision Analysis*. Stanford University, Department of Engineering-Economic Systems, Draft, 1991.
- [1-4] Keeney, Ralph L., and Raiffa, Howard, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. New York: Wiley, 1976.
- [1-5] Pate-Cornell, M.E., *Class Notes for IE 240: Engineering Risk Analysis*. Stanford University, Department of Industrial Engineering, Draft, 1991.
- [1-6] Pate-Cornell, M.E., "Warning Systems in Risk Management", *Risk Analysis*. Vol. 6, No. 2, 1986, pp. 223-34.
- [1-7] Pate-Cornell, M.E. and Fischbeck, P., "Probabilistic Risk Analysis and Risk-Based Priority Scale for the Tiles of the Space Shuttle", *Reliability Engineering and System Safety*. Vol. 40, No. 3, 1993, pp. 221-38.

### **2.0 System Optimization and Artificial Intelligence**

- [2-1] Aizawa, A.N. and Wah, B.W., "Dynamic Control of Genetic Algorithms in a Noisy Environment", *Proceedings of the Fifth International Conference on Genetic Algorithms*. Urbana-Champaign, IL., 1993, pp. 48-55.
- [2-2] Davis, Lawrence, Ed., *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.

- [2-3] Gage, P.J., "New Approaches to Optimization in Aerospace Conceptual Design", Ph.D. Dissertation, Stanford University, Department of Aeronautics and Astronautics, 1994.
- [2-4] Glynn, Peter W., "Optimization of Stochastic Systems", *Proceedings of the 1986 Winter Simulation Conference*, 1986, pp. 52-59.
- [2-5] Goldberg, David E., *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA.: Addison-Wesley Publishing Co., Inc., 1989.
- [2-6] Grefenstette, J.J., "Optimization of Control Parameters for Genetic Algorithms", *IEEE Transactions on Systems, Man, and Cybernetics*. Vol. SMC-16, No. 1, Jan./Feb. 1986, pp. 122-128.
- [2-7] Grefenstette, J.J. and Fitzpatrick, J.M., "Genetic Search with Approximate Function Evaluations", *Proceedings of the First International Conference on Genetic Algorithms*. Pittsburgh, PA. 1985, pp. 112-120.
- [2-8] Himmelblau, David M., *Applied Nonlinear Programming*. New York: McGraw-Hill, 1972.
- [2-9] Holland, John H., "Adaptation in Natural and Artificial Systems", Ph.D. Dissertation, University of Michigan, 1975.
- [2-10] Karimi, A., Sebald, A.V., and Isaka, S., "Use of Simulated Annealing in Design of Very High Dimensioned Minimax Adaptive Controllers", *Asilomar Conf. on Signals, Systems, and Computers*. Pacific Grove, CA. 1989, pp. 116-118.
- [2-11] Kim, J.H., "Heuristic Method for Solving Redundancy Optimization Problems in Complex Systems", *IEEE Transactions on Reliability*. Vol. 42, No. 4, December 1993, pp. 572-78.
- [2-12] Krishnakumar, K., and Goldberg, David E., "Control System Optimization Using Genetic Algorithms", *AIAA Journal of Guidance, Control, and Dynamics*. Vol. 15, No. 3, May-June 1992, pp. 735-740.

- [2-13] Painton, L., "Genetic Algorithms in Optimization of System Reliability", *IEEE Transactions on Reliability*. Vol. 44, No. 2, June 1995, pp. 172-78.
- [2-14] Papalambros, P., and Wilde, D., *Principles of Optimal Design: Modeling and Computation*. Cambridge, Eng.: Cambridge University Press, 1988.
- [2-15] Rich, Elaine, and Knight, Kevin, *Artificial Intelligence*, 2nd Edition. New York: McGraw-Hill, 1991.
- [2-16] Vanderbilt, D. and Louie, S.G., "A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables". *Journal of Computational Physics*, Vol. 56, 1984, pp. 259-271.
- [2-17] Van Laarhoven, P.J.M., and Aarts, E.H.L., *Simulated Annealing: Theory and Applications*. Dordrecht, Netherlands: D. Reidel Publishing Co., 1987.
- [2-18] Sobieski, J., "MDO can help Resolve the Designer's Dilemma", *Aerospace America*. Vol. 29, No. 9, September 1991, pp. 32-35, 63.
- [2-19] Yan, D., and Mukai, H., "Discrete Optimization with Estimation", *Proceedings of the 20<sup>th</sup> IEEE Conference on Decision and Control*. Tampa, FL., December 1989, pp. 2463-2468.

### **3.0 Probability Modeling and Monte Carlo Methodology**

- [3-1] Bain, Lee J., and Englehardt, Max, "A Review of Model Selection Procedures Relevant to the Weibull Distribution", *Communications of Statistics - Theory and Methods*. Vol. 12, No. 5, 1983, pp. 589-609.
- [3-2] Billinton, Roy, and Allan, Ronald N., *Reliability Evaluation of Engineering Systems: Concepts and Techniques*, 2nd Ed. New York: Plenum Press, 1992.
- [3-3] Casella, George, and Berger, Roger L., *Statistical Inference*. Pacific Grove, CA.: Wadsworth and Brooks, 1990.

- [3-4] Clarotti, C.A., and Spizzichino, F., "The Bayes Predictive Approach in Reliability Theory", *IEEE Transactions on Reliability*. Vol. R-38, No. 3, August 1989, pp. 379-382.
- [3-5] Dhillon, B.S., *Reliability Engineering in Systems Design and Operation*. New York: Van Nostrand Reinhold Co., 1983.
- [3-6] Glynn, Peter W., "Stochastic Approximation for Monte Carlo Optimization", *Proceedings of the 1986 Winter Simulation Conference*. 1986, pp. 356-365.
- [3-7] Henley, Ernest J., and Kumamoto, Hiromitsu, *Reliability Engineering and Risk Assessment*. Englewood Cliffs, N.J.: Prentice-Hall, Inc., 1981.
- [3-8] Landers, Thomas L., Taha, Hamdy A., and King, Charles L., "A Reliability Simulation Approach for Use in the Design Process", *IEEE Transactions on Reliability*. Vol. R-40, No. 2, June 1991, pp. 177-181.
- [3-9] Law, Averill M., and Kelton, W. David, *Simulation Modeling and Analysis*, 2nd Edition. New York: McGraw-Hill, 1991.
- [3-10] Leemis, L.M., *Reliability: Probabilistic Models and Statistical Methods*. Englewood Cliffs, N.J.: Prentice Hall, 1995.
- [3-11] Moore, Albert H., Harter, H. Leon, and Snead, Robert C., "Comparison of Monte Carlo Techniques for Obtaining System-Reliability Confidence Limits", *IEEE Transactions on Reliability*. Vol. R-29, No. 4, Oct. 1980, pp. 327-331.
- [3-12] Press, W.H., Teukolsky, S.A., Vetterling, W.T., and Flannery, B.P., *Numerical Recipes in C: The Art of Scientific Computing*, Second Edition. Cambridge, England: Cambridge University Press, 1992.
- [3-13] Rubinstein, R.Y., *Simulation and the Monte Carlo Method*. New York: J. Wiley & Sons, 1981.

- [3-14] Singpurwalla, Nozer D., and Wilson, Simon P., "Models for Assessing the Reliability of Computer Software", *Naval Research Review*. Office of Naval Research, Vol. XLIV/XLV, No. 4/1, 1992/1993, pp. 21-29.
- [3-15] Specht, D.F., "Probabilistic Neural Networks", *Neural Networks*. Vol. 3, 1990, pp. 109-118.

#### **4.0 Satellite Design and Reliability**

- [4-1] Barnett, E., "Demonstrated Orbital Reliability of TRW Spacecraft" (74-2286.142). Redondo Beach, CA.: TRW Systems Group, Reliability Assurance Dept., December 1974.
- [4-2] Greenberg, J.S. and Hertzfeld, H.R., Eds., *Space Economics*. Progress in Aeronautics and Astronautics, Vol. 144. Washington, D.C.: American Institute of Aeronautics and Astronautics, 1992.
- [4-3] Hecht, Herbert and Myron, "Reliability Prediction for Spacecraft" (AD/A164-747). Los Angeles, CA.: SoHar Inc., December 1985.
- [4-4] Murray, Charles, and Cox, Catherine Bly, *Apollo: The Race to the Moon*. New York: Simon and Schuster, 1989.
- [4-5] Nash, F.R., *Estimating Device Reliability: Assessment of Credibility*. Boston: Kluwer Academic Publishers, 1993.
- [4-6] Parish, G. and Sollfrey, W., "A Preliminary Analysis of the Effect of Work-Arounds on Space System Performance and Procurement Requirements -- A Proposal", The RAND Corporation, N-1260-AF, March 1980.
- [4-7] *Procedures for Performing a Failure Modes, Effects, and Criticality Analysis* (MIL-HDBK-1629A). Washington, D.C.: Department of Defense, 1980.
- [4-8] *Nonelectric Parts Reliability Data 1995* (NPRD-95). Rome, N.Y.: Reliability Analysis Center, 1995.

- [4-9] Regulinski, T.L., and Gupta, Y.P., "Reliability Cost Estimation: Managerial Perspectives", *IEEE Transactions on Reliability*. Vol. R-32, No. 3, August 1983, pp. 276-281.
- [4-10] *Reliability Growth Management* (MIL-HDBK-189). Washington, D.C.: Department of Defense, February 1981.
- [4-11] *Reliability Modeling and Prediction* (MIL-STD-756B). Washington, D.C.: Department of Defense, November 1981.
- [4-12] *Reliability Prediction of Electronic Equipment* (MIL-HDBK-217F). Washington, D.C.: Department of Defense, 1991.
- [4-13] Sloane, E.A., "Demonstrated Orbital Reliability of TRW Spacecraft" (75-2286.148). Redondo Beach, CA.: TRW Systems Group, Reliability Assurance Dept., December 1975.
- [4-14] Steen, A. and Swain, D., "Demonstrated Orbital Reliability of TRW Spacecraft" (81-2286.155). Redondo Beach, CA.: TRW Defense and Space Systems Group, Reliability & Safety Dept., December 1981.
- [4-15] Wertz, J.R. and Larson, W.J., Eds., *Space Mission Analysis and Design*. Dordrecht, Netherlands: Kluwer Academic Publishers, 1991.

## **5.0 Gravity Probe-B**

- [5-1] Axelrad, P., "A Closed Loop GPS Based Orbit Trim System for Gravity Probe B", Ph.D. Dissertation, SUDAAR 598, Stanford University, Department of Aeronautics and Astronautics, October 1990.
- [5-2] Bardas, D., *et.al.*, "Gravity Probe B: II. Hardware Development; Progress Towards the Flight Instrument", *Proceedings of the Sixth Marcel Grossmann Meeting on General Relativity*. Singapore: World Scientific 1992, pp. 382-89.

- [5-3] C.W.F. Everitt: "Gravity Probe B: I. The Scientific Implications", *Proceedings of the Sixth Marcel Grossmann Meeting on General Relativity*. Singapore: World Scientific, 1992, pp. 1632-44.
- [5-4] Kasdin, N.J., "Precision Pointing Control of the Spinning Gravity Probe B Spacecraft", Ph.D. Dissertation, SUDAAR 606, Stanford University, Department of Aeronautics and Astronautics, March 1991.
- [5-5] *Proposal for Gravity Probe B Science Mission Spacecraft Development: Management Proposal* (Volume III). Sunnyvale, CA.: Lockheed Missiles & Space Company, Inc., May 28, 1993.
- [5-6] Wiktor, P.J., "The Design of a Propulsion System Using Vent Gas from a Liquid Helium Crogenic System", Ph.D. Dissertation, SUDAAR 624, Stanford University, Department of Aeronautics and Astronautics, June 1992.
- [5-7] Xiao, Y.M., *et.al.*, "Gravity Probe B: III. The Precision Gyroscope", *Proceedings of the Sixth Marcel Grossmann Meeting on General Relativity*. Singapore: World Scientific, 1992, pp. 394-398.

## **6.0 Global Positioning System: General Resources**

- [6-1] *A Technical Report to the Secretary of Transportation on a National Approach to Augmented GPS Services*. U.S. Department of Commerce, NTIA Special Publication 94-30, December 1994.
- [6-2] Cohen, C., Pervan, B., and Parkinson, B., "Estimation of Absolute Ionospheric Delay Exclusively through Single-Frequency GPS Measurements", *Proceedings of ION GPS-92*, Albuquerque, NM., Sept. 16-18, 1992, pp. 325-330.
- [6-3] Conley, R., Fryt, M., and Scott, S., "GPS Performance Characteristics and Trends", *Proceedings of ION GPS-95*, Palm Springs, CA., Sept. 12-15, 1995, pp. 1363-71.

- [6-4] Durand, J-M. and Caseau, A. "GPS Availability, Part II: Evaluation of State Probabilities for 21 Satellite and 24 Satellite Constellations", *Navigation*, Vol. 37, No. 3, Fall 1990, pp. 285-297.
- [6-5] FAA Headquarters - APO, "Projected GNSS Cat I/II/III Precision Landing Operations", *Proceedings of ICAO Comm. /Ops. Meeting*. Montreal, Canada: March 27 - April 7, 1995. Item 1.
- [6-6] Green, G.B., Massatt, P.D., and Rhodus, N.W., "The GPS 21 Primary Satellite Constellation", *Navigation*, Vol. 36, No. 1, Spring 1989, pp. 9-24.
- [6-7] Hatch, R., "The Synergism of GPS Code and Carrier Measurements", *Proceedings of the 3<sup>rd</sup> International Geodetic Symposium on Satellite Doppler Positioning*, Las Cruces, NM., Feb. 1982, pp. 1213-1232.
- [6-8] ICAO Secretariat, "Economic Evaluations of the Main Options for Precision Approach and Landing Systems", *Proceedings of ICAO Comm./Ops. Meeting*. Montreal, Canada: March 27 - April 7, 1995, Item 2-3, WP/23.
- [6-9] *International Standards, Recommended Practices and Procedures for Air Navigation Services -- Annex 10*. International Civil Aviation Organization (ICAO), April 1985.
- [6-10] Kelly, R.J. and Davis, J.M., "Required Navigation Performance (RNP) for Precision Approach and Landing with GNSS Application," *Navigation*, Vol. 41, No. 1, Spring 1994, pp. 1-30.
- [6-11] "Operational Requirements Document: Local Area Augmentation System", Washington, D.C.: FAA Satellite Program Office, February 28, 1995.
- [6-12] Pace, S., Frost, G., Lachow, I., Frelinger, D., Fossum, D., Wassem, D.K., and Pinto, M., *The Global Positioning System: Assessing National Policies*. RAND Critical Technologies Institute, MR-614-OSTP, 1995.

- [6-13] Parkinson, B.W., "The Origins, Evolution and Future of Satellite Navigation", AIAA 1996 Von Karman Paper, *Proceedings of the 34th AIAA Aerospace Sciences Meeting*. Reno, NV., January 15-18, 1996.
- [6-14] Parkinson, B.W. and Spilker, J.J., Eds., *The Global Positioning System: Theory and Applications*. Washington, D.C.: American Institute of Aeronautics and Astronautics, 1995.
- [6-15] Phlong, W. and Elrod, B., "Availability Characteristics of GPS and Augmentation Alternatives." *Navigation*, Vol. 40, No. 4, Winter 1993-94, pp. 409-428.
- [6-16] Schanzer, G., "Satellite Navigation for Precision Approach: Technological and Political Benefits and Risks", *Proceedings of ISPA '95*. Braunschweig, Germany, Feb. 21-24, 1995, pp. 25-30.
- [6-17] Spilker, J.J., "GPS Signal Structure and Performance Characteristics", *Navigation*, Vol. 1, 1980, pp. 29-54.
- [6-18] *GPS Minimal Operation Performance Standards*. Draft of SC-159 Report. Washington, D.C.: RTCA, January 1990.

## **7.0 GPS Integrity Monitoring and Safety Assessment**

- [7-1] Bancroft, S. and Chen, S.S., "Integrity Monitoring Using Bayes' Rule", *Proceedings of ION GPS-91*. Albuquerque, NM., Sept. 11-13, 1991, pp. 781-787.
- [7-2] Brown, R.G., "A Baseline GPS RAIM Scheme and a Note on the Equivalence of Three RAIM Methods", *Navigation*, Vol. 39, No. 3, Fall 1992, pp. 301-316.
- [7-3] *Department of Defense Global Positioning System Integrity Study Final Report*. Gaithersburg, MD.: IBM (now Loral) Federal Systems, December 1991.
- [7-4] Farrell, J.L. and Von Graas, F., "Statistical Validation for GPS Integrity Test", *Proc. of ION GPS-91*. Albuquerque, NM., Sept. 11-13, 1991, pp. 773-79.

- [7-5] Frank, P.M., "Fault Diagnosis in Dynamic Systems using Analytical and Knowledge-based Redundancy: A Survey and some New Results", *Automatica*. Vol. 26, No. 3, May 1990, pp. 459-74.
- [7-6] Gower, A.G., "Putting a Number on GPS Integrity: The IBM GPS Integrity Study for the DoD", *Proceedings of ION GPS-91*. Albuquerque, NM., Sept. 11-13, 1991, pp. 753-760.
- [7-7] McBurney, P.W. and Brown, R.G., "Self-Contained GPS Integrity Monitoring Using a Censored Kalman Filter", *Proceedings of ION GPS-88*. Colorado Springs, CO., Sept. 19-23, 1988, pp. 441-450.
- [7-8] Parkinson, B.W. and Axelrad, P., "Autonomous GPS Integrity Monitoring Using the Pseudorange Residual", *Navigation*, Vol. 35, No. 2, Summer 1988, pp. 255-274.
- [7-9] Pervan, B.S., Cohen, C.E., and Parkinson, B.W., "Autonomous Integrity Monitoring for Precision Approach using DGPS and a Ground-Based Pseudolite", *Proc. of ION GPS-93*. Salt Lake City, UT., Sept. 22-24, 1993.
- [7-10] Sturza, M.A., "Navigation System Integrity Monitoring Using Redundant Measurements", *Navigation*, Vol. 35, No. 4, Winter 1988-1989, pp. 483-501.
- [7-11] Sturza, M.A. and Brown, A.K., "Comparison of Fixed and Variable Threshold RAIM Algorithms", *Proceedings of ION GPS-90*. Colorado Springs, CO., Sept. 19-21, 1990, pp. 437-443.
- [7-12] Walter, T., Pervan, B., Enge, P., Herendeen, J., and Levin, P., "Autonomous Integrity Monitoring and Wide Area DGPS," *Proceedings of ION NTM-94*. San Diego, CA., Jan. 24-26, 1994, pp. 485-496.
- [7-13] Walter, T., Enge, P., and Van Graas, F., "Integrity for the Wide Area Augmentation System," *Proc. of DSNS-95*. Bergen, Norway, April 24-28, 1995, No. 38.
- [7-14] Willsky, A.S., "A Survey of Design Methods for Failure Detection in Dynamic Systems", *Automatica*. Vol. 12, 1976, pp. 601-611.

- [7-15] "Local Area Augmentation System (LAAS) Integrity Panel Written Report: Integrity Beacon Landing System (IBLS)." Department of Aeronautics and Astronautics, Stanford University, December 15, 1995.

## **8.0 Wide Area Augmentation System (WAAS)**

- [8-1] Ashkenazi, V., Hill, C.J., and Nagel, J., "Wide Area Differential GPS: A Performance Study", *Proceedings of ION GPS-92*, Albuquerque, NM., Sept. 16-18, 1992, pp. 589-598.
- [8-2] Chao, Y.C., Tsai, Y.J., Walter, T., Kee, C., Enge, P., and Parkinson, B., "An Algorithm for Inter-Frequency Bias Calibration and Application to WAAS Ionosphere Modeling", *Proceedings of ION GPS-95*, Palm Springs, CA., Sept. 12-15, 1995, pp. 639-646.
- [8-3] Chao, Y.C., Tsai, Y.J., Walter, T., Kee, C., Enge, P., and Parkinson, B., "The Ionospheric Model Improvement for the Stanford WAAS Network," *Proceedings of ION NTM-95*, Anaheim, CA., Jan. 18-20, 1995, pp. 531-538.
- [8-4] Conker, R., El-Arini, M.B., Albertson, T., Klobuchar, J., and Doherty, P., "Development of Real-Time Algorithms to Estimate the Ionospheric Error Bounds for WAAS", *Proceedings of ION GPS-95*, Palm Springs, CA., Sept. 12-15, 1995, pp. 1247-58.
- [8-5] El-Arini, M.B., Hegarty, C.J., Fernow, J.P., and Klobuchar, J.A., "Development of an Error Budget for a GPS Wide-Area Augmentation System (WAAS)," *Proceedings of ION NTM-94*, San Diego, CA., Jan. 24-26, 1994, pp. 927-936.
- [8-6] El-Arini, M.B. and Wisser, T.C., "The FAA Wide Area Differential GPS (WADGPS) Static Ionospheric Experiment," *Proceedings of ION NTM-93*, Anaheim, CA., Jan. 22-24, 1993, pp. 485-496.
- [8-7] Enge, Per and Van Dierendonck, A.J., "The Wide Area Augmentation System," *Proc. 8<sup>th</sup> International Flight Inspection Symposium*, Denver, CO., June 1994.

- [8-8] Enge, P., Van Dierendonck, A.J., and Kinal, G. "A Signal Design for the GIC Which Includes Capacity for WADGPS Data", *Proceedings of ION GPS-92*. Albuquerque, NM., Sept. 16-18, 1992, pp. 875-884.
- [8-9] Kee, C., "Wide Area Differential GPS (WADGPS)", Ph.D. Dissertation, SUDAAR 642, Stanford University, Department of Aeronautics and Astronautics, December 1993.
- [8-10] Klobuchar, J.A., Doherty, P.H., and El-Arini, M.B., "Potential Ionospheric Limitations to Wide-Area Differential GPS", *Proceedings of ION GPS-93*, Salt Lake City, UT., Sept. 22-24, 1993, pp. 1245-1254.
- [8-11] Loh, R., "Seamless Aviation: FAA's Wide Area Augmentation System," *GPS World*, April 1995, pp. 20-30.
- [8-12] Loh, R. and Fernow, J.P., "Integrity Concepts for a GPS Wide-Area Augmentation System (WAAS)", *Proceedings of ION NTM-94*, San Diego, CA., Jan. 24-26, 1994, pp. 127-134.
- [8-13] *Minimum Operational Performance Standards for Sensors Using Global Positioning System/Wide Area Augmentation System*, Draft 6. Washington, D.C.: RTCA, July 24, 1995.
- [8-14] Mueller, T., Hamry, B., and Johnson, A., "WADGPS Ionospheric Correction Model Performance Simulation", *Proceedings of ION GPS-95*, Palm Springs, CA., Sept. 12-15, 1995, pp. 1237-46.
- [8-15] Poor, W., Albertson, T., and Yen, P., "A Wide Area Augmentation System Service Volume Model and its Use in Evaluating WAAS Architectures and Design Sensitivities", *Proceedings of ION GPS-95*, Palm Springs, CA., Sept. 12-15, 1995, pp. 629-637.
- [8-16] *Specification for the Wide Area Augmentation System (WAAS)*. Washington, D.C.: Federal Aviation Administration, FAA-E-2892A, August 2, 1995.

- [8-17] Tsai, Y.J., "Snapshot WMS Algorithms for Ephemeris and Clock Estimation," Stanford University, Unpublished Manuscript, Feb. 13, 1995.
- [8-18] Walter, T., Kee, C., Chao, Y.C., Tsai, Y.J., *et.al.*, "Flight Trials of the Wide Area Augmentation System (WAAS)," *Proceedings of ION GPS-94*, Salt Lake City, UT., Sept. 21-23, 1994.
- [8-19] Lechner, W., private conversation, Stanford GPS Research presentation, August 11, 1995.

## 9.0 Controller/Estimator Algorithms

- [9-1] Birdwell, J.D., Castanon, D.A., and Athans, M., "On Reliable Control System Design", *IEEE Transactions on Systems, Man, and Cybernetics*. Vol. SMC-16, No. 5, Sept./Oct. 1986, pp. 703-711.
- [9-2] Bryson, A.E., *Control of Spacecraft and Aircraft*. Princeton, N.J.: Princeton University Press, 1994.
- [9-3] Bryson, A.E., "Dynamic Optimization under Uncertainty," Notes for E207C, Stanford University, March 30, 1993.
- [9-4] Bryson, A.E. and Ho, Y.C., *Applied Optimal Control: Optimization, Estimation, and Control*, Revised Printing. New York: Hemisphere Publishing Corp., 1975.
- [9-5] Djavdan, P., Tulleken, H.J.A.F., Voetter, M.H., Verbruggen, H.B. and Olsder, G.J., "Probabilistic Robust Controller Design", *Proceedings of the 28<sup>th</sup> IEEE Conference on Decision and Control*. Tampa, FL.: Dec. 1989, pp. 2164-72.
- [9-6] Doyle, J.C., Francis, B.A., and Tannenbaum, A.R., *Feedback Control Theory*. New York: Macmillan Publishing Co., 1992.
- [9-7] Franklin, G.F., Powell, J.D., and Emami-Naeini, A., *Feedback Control of Dynamic Systems, 3rd Edition*. Menlo Park, CA.: Addison-Wesley, 1994.

- [9-8] Franklin, G.F., Powell, J.D., and Workman, M.L., *Digital Control of Dynamic Systems, 2nd Edition*. Menlo Park, CA.: Addison-Wesley, 1990.
- [9-9] Gopal, M., *Modern Control System Theory, 2nd Edition*. New York: John Wiley & Sons, 1993.
- [9-10] Kailath, T., *Linear Systems*. Englewood Cliffs, N.J.: Prentice-Hall, 1980.
- [9-11] Ly, U.L., "A Design Algorithm for Robust Low-Order Controllers", Ph.D. Dissertation, SUDAAR 536, Stanford University, Department of Aeronautics and Astronautics, November 1982.
- [9-12] Marrison, C. and Stengel, R., "The Use of Random Search and Genetic Algorithms to Optimize Stochastic Robustness Functions," *Proceedings of the American Control Conference*, Baltimore, MD., June 1994, pp. 1484-89.
- [9-13] Mills, R.A., and Bryson, A.E., "Parameter-Robust Control Design Using a Minimax Method", *AIAA Journal of Guidance, Control, and Dynamics*. Vol. 15, No. 5, Sept.-Oct. 1992, pp. 1068-75.
- [9-14] Ray, Laura Ryan, and Stengel, Robert F., "Application of Stochastic Robustness to Aircraft Control Systems", *AIAA Journal of Guidance, Control, and Dynamics*. Vol. 14, No. 6, Nov.-Dec. 1992, pp. 1251-1259.
- [9-15] Shakarian, A., "Application of Monte-Carlo Techniques to the 757/767 Autoland Dispersion Analysis by Simulation", AIAA 83-2193, *Proceedings of AIAA Guidance and Control Conference*. Gatlinburg, TN., August 15-17, 1983, pp. 181-194.
- [9-16] Stengel, R., Ray, L., and Marrison, C., "Probabilistic Evaluation of Control System Robustness," *International Journal of Systems Science*, Vol. 26, No. 7, 1995, pp. 1363-82.
- [9-17] Whittle, Peter, *Optimal Control: Basics and Beyond*. Chichester, Eng.: John Wiley & Sons Ltd., 1996.