The Dissertation Committee for Gary Nathan Green
certifies that this is the approved version of the following dissertation:

Advanced Techniques for Safety-of-Life Carrier Phase Differential GNSS Positioning with Applications to Triplex Architectures

Committee:

Todd Humphreys, Supervisor

Jeffrey Andrews

Gustavo de Veciana

Haris Vikalo

Johnathan York

Advanced Techniques for Safety-of-Life Carrier Phase Differential GNSS Positioning with Applications to Triplex Architectures

by

Gary Nathan Green

DISSERTATION

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2017

Dedicated to my wife Emily and my father Jon.

Acknowledgments

I would like to thank Dr. Todd Humphreys, my advisor, for his willingness to work with me as a part time student. The past four years have been an adventure. I am particularly thankful for his help in disciplined editing of this manuscript. I thank my committee members, Dr. Gustavo de Veciana and Dr. Jeff Andrews for their excellent teaching in Stochastic Processes and Digital Communications, respectively. I learned much under your instruction. Thanks also to Dr. Haris Vikalo and Dr. Johnathan York for their willingness to serve on my committee.

I gratefully acknowledge my support from the Naval Air Systems Command for my research, particularly Phil Rooney, who helped me in the early days of my research, and Martin King, who encouraged me through several years of work. The conclusions herein are my own and do not represent the opinion of the Navy or any of its members. I would also like to recognize the support that I received from my employer, Coherent Technical Services, Inc. including from Tom Sanders, Carrie Willmore, Ian Gallimore, Stefan Toussaint, Peter Leget, and Surjan Dogra.

Most vitally, I praise my excellent wife, Emily, for encouraging, supporting, and enabling me to accomplish everything that I have in the past decade. Without her love and steadfastness, I would be nothing. I thank my father for teaching me to love mathematics and hard work. All thanks and praise are ultimately due to the Lord Jesus Christ, from whom all blessings flow.

Advanced Techniques for Safety-of-Life Carrier Phase Differential GNSS Positioning with Applications to Triplex Architectures

Publication No. _____

Gary Nathan Green, Ph.D.
The University of Texas at Austin, 2017

Supervisor: Todd Humphreys

Safety-of-life Carrier phase Differential Global Navigation Satellite System (CDGNSS) positioning systems must provide guarantees that their position estimates have errors that are smaller than specified levels, called alert limits (AL). These guarantees are specified as an allowable probability, called integrity risk (IR), that the error exceeds its AL. Typical values of IR are between 10^{-9} and 10^{-7}, per hour of operation. CDGNSS positioning has been demonstrated to provide centimeter-accurate estimates of a vehicle's location when the so-called integer ambiguities are resolved; however, in safety-of-life applications, the probability of incorrectly resolving the integer ambiguities frequently exceeds the allowable IR. To address this limitation, existing algorithms bound the positioning error caused by incorrectly resolved ambiguities. If such bounds satisfy the AL, then the integer-resolved, or fixed, solution can be used. Unfortunately, the positioning error from

incorrect fixing can exceed several meters, which fails to satisfy the most demanding ALs for autonomous vehicles. This dissertation offers three contributions to the science of CDGNSS positioning for safety-of-life applications.

First, a novel algorithm is developed that validates the correctness of integer ambiguity estimates. This algorithm, called Generalized Integer Aperture Bootstrapping (GIAB), establishes a rigorous, fixed-missed-detection-rate test that provides a guarantee that the integer ambiguities have been fixed correctly. GIAB also allows for partial fixing, where a subset of the ambiguities are resolved. Partial fixing allows for graceful degradation of positioning when measurement quality is poor. GIAB is derived analytically and validated via Monte Carlo simulation. Its performance is compared with existing ambiguity validation techniques.

Second, the probability density function of the positioning estimate resulting from GIAB is derived. This distribution leads to a provable bound on the IR that the estimate has errors exceeding the specified ALs. This bound allows GIAB to be used for safety-of-life application while satisfying ALs of less than a meter.

Third, triplex CDGNSS architectures, in which the vehicle position is estimated using three separate navigation systems with mid-level voting (MLV) logic, are analyzed. Such architectures are commonly used since they are robust to single equipment failures, but the integrity benefit of their fault-free performance has not previously been evaluated. It is shown that integer-fixed CDGNSS solutions improve in accuracy performance, but gain no integrity benefit. However, when the integer constraint is not enforced, the so called CDGNSS float solution benefits greatly from MLV in both accuracy and integrity performance.

Table of Contents

Acknowledgments v

Abstract vi

List of Tables xii

List of Figures xiii

Chapter 1. Introduction 1

1.1 Carrier Phase Differential Global Navigation Satellite System Positioning . . . 1
1.2 Required Navigation Performance . . . 3
1.3 CDGNSS and RNP . . . 4
1.4 Triplex CDGNSS Architectures . . . 6
1.5 Thesis Statement and Expected Contributions . . . 7
1.6 Published Works . . . 9
1.6.1 Dissertation Organization . . . 10
1.7 Nomenclature . . . 11

Chapter 2. Generalized Integer Aperture Bootstrapping 13

2.1 Introduction . . . 13
2.2 Generalized Integer Aperture Bootstrapping . . . 17
2.2.1 Integer Bootstrapping Overview . . . 17
2.2.2 Integer Aperture Bootstrapping (IAB) . . . 24
2.2.3 Generalization to Partial Ambiguity Resolution . . . 27
2.2.4 Partial Ambiguity Resolution Probabilities . . . 30
2.3 Setting the Integer Aperture Parameters . . . 33
2.3.1 Allocation from Probability of Failure . . . 34
2.3.2 Optimization for Availability of Full Ambiguity Resolution . 35

2.4 The GIAB Baseline . . . 37
2.5 Validation via Monte Carlo Simulation . . . 40
2.6 Comparison against Existing IA Methods . . . 43
2.7 Conclusions . . . 46

Chapter 3. Data-Driven Position Domain Integrity for Generalized Integer Aperture Bootstrapping 50
3.1 Introduction . . . 50
3.2 Generalized Integer Aperture Bootstrapping Overview . . . 52
3.3 Prior Distribution of the GIAB Baseline . . . 56
3.4 Conditional distributions of the Generalized Integer Aperture Baseline 57
3.4.1 Conceptual Overview . . . 57
3.4.2 Finding $f_{\bar{\boldsymbol{b}}|F^c}$ . . . 58
3.4.3 Finding $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ . . . 59
3.4.4 Finding $f_{\bar{\boldsymbol{b}},\epsilon_{c(i+1)}|Z_i}(\boldsymbol{\xi}, \varepsilon + k)$ . . . 61
3.4.5 Finding $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ . . . 63
3.4.6 Discussion . . . 64
3.4.7 GIAB variants . . . 66
3.5 Validation of Baseline Distributions . . . 70
3.6 Data-Driven Position Domain Integrity . . . 73
3.6.1 Position Domain Integrity in EPIC . . . 73
3.6.2 Position Domain Integrity in GIAB . . . 75
3.7 Performance Analysis . . . 79
3.7.1 Protection Levels . . . 79
3.7.2 Comparison to EPIC . . . 80
3.8 Conclusions . . . 84

Chapter 4. Correlation-Aware Integrity Protection for Fault-Free Federated Triplex Carrier Differential GNSS Solutions 86
4.1 Introduction . . . 86
4.2 Fault-Free Integrity for Mid-Level Voting with Arbitrary Distributions 88
4.2.1 Integrity Specification Parameters . . . 88
4.2.2 Derivation of MLV Integrity Risk . . . 89

4.3 Correlation Agnostic Integrity Risk Bound 94
4.4 Application of Mid-Level Voting to Carrier Phase Differential GNSS Solutions 95
4.4.1 Simplex Float Solution Joint Distribution 95
4.4.2 Triplex Float Solution Joint Distribution 97
4.4.3 Simplex Fixed Solution Integrity 98
4.4.4 Triplex Fixed Solution Joint Distribution 100
4.4.5 Simplex GERAFS Integrity Risk 103
4.4.6 Triplex GERAFS Joint Distribution 104
4.5 Analytical Comparison to Simplex Solutions 105
4.6 Simulation Methodology 107
4.7 Simulation Results 108
4.8 Conclusions 109

Chapter 5. Conclusion **112**
5.1 Summary 112
5.2 Future Work 113

Appendices **114**

Appendix A. CDGNSS Measurement Model **115**
A.1 GNSS Measurement Model 115
A.1.1 Undifferenced Measurement Model 116
A.1.2 Short Baseline Single Differences Measurement Model . . . 118
A.1.3 Double Difference Measurement Model 119
A.1.4 Carrier Phase Smoothed Pseudorange Measurements 120
A.1.5 Divergence-Free Smoothing 123
A.1.6 Double Difference Error Model 125

Appendix B. Linearized Least Squares Relative Position Solution **128**
B.1 Float Solution 128

Appendix C. Interpretation and Properties of the Float Ambiguity **131**
C.1 Interpretation of L and D 131
C.2 Relationships among $\hat{\boldsymbol{z}}$, $\hat{\boldsymbol{z}}_{\mathrm{c}}$, $\check{\boldsymbol{z}}$, $\check{\boldsymbol{\epsilon}}$ and $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}$ 134

Appendix D. Derivations for Data-Driven PDI **135**
D.1 Truncation of Posterior Risk Probabilities 135
D.1.1 Posterior PDI and Fixing Probabilities 135
D.1.2 Bounding Posterior Fixing Probabilities for GIAB 136
D.1.3 Binary-Tree-Based Alternative Fix Pruning 138
D.2 Distribution of Truncated Gaussian 145

Appendix E. Proof of Correlation Agnostic Bound **147**

Bibliography **149**

Vita **155**

List of Tables

2.1 Weighting Function Alternatives Considered 36

2.2 Predicted vs Simulated Event Probabilities for a Strong Model . . . 42

2.3 Predicted vs Simulated Event Probabilities for a Weak Model 43

2.4 Joint Probability Mass Function of GIAB (Rows) and Optimal IA (Columns) Fixing Decision for Weak $m = 7$ Model 45

2.5 Joint Probability Mass Function of IAB (Rows) and Optimal IA (Columns) Fixing Decision for $m = 2$ Model 48

2.6 Joint Probability Mass Function of GIAB (Rows) and Optimal IA (Columns) Fixing Decision for $m = 2$ Model 48

3.1 Integrity performance comparison between EPIC and MAP GIAB. All distance units are in meters. The left-most column indicates the result from GIAB. The next column indicates it's theoretical probability of occurrence [11]. The third column is the standard deviation of the $\bar{b}$ under the given GIAB event. For EPIC, the standard deviations from the previous row applies. The next three columns indicate the minimum, average and maximum PL produced by GIAB under each event. The final column is the value of PL produced by EPIC when it fixes the same number of integers as GIAB. 82

3.2 Integrity performance comparison between EPIC and MAP GIAB for a weak model. Columns are the same as the prior table. Even for a weak model, GIAB usually provides a lower AL than EPIC with superior accuracy. In the case that GIAB rejects all fixes, the computed PL represents the poor quality of the measurements. . . . 83

4.1 World-wide availability for varying ALs for each solution 109

4.2 Daily average 70% accuracy for each solution at 99.5% availability or maximum obtained 110

List of Figures

2.1 IAB aperture regions for a two-dimensional example model. This is a visual representation of the possible outcomes of $\mathrm{IAB}(\hat{\boldsymbol{z}} - \boldsymbol{z}, L, \beta)$. The central, darkly shaded region is the success region, in which $\check{\boldsymbol{z}} = \boldsymbol{z}$. The lightly shaded regions correspond to incorrect ambiguity fixes, in which $\check{\boldsymbol{z}} \neq \boldsymbol{z}$. The unshaded region is the fix-rejection region. 26

2.2 Regions of the float ambiguity error, $\boldsymbol{\epsilon} \in \mathbb{R}^m$, that are mapped by GIAB to failure, undecided, and success events for an example model with $m = 2$. Event F results if one or more ambiguities are fixed incorrectly. Event S_i occurs when exactly $i \in \{1, \ldots, m\}$ ambiguities are fixed and each of these is correct. Event U occurs when the first ambiguity is rejected, leaving all ambiguities unfixed. 31

2.3 Comparison of the integer aperture acceptance regions for integer aperture bootstrapping (IAB), ellipsoidal IA, GIAB, and optimal IA. All apertures allow the same expected number of incorrect fixes, but yield different rates of accepting the correct fix. Listed in ascending order of success are the ellipsoidal, IAB, GIAB, and optimal IA acceptance regions. There is a significant improvement from IAB to GIAB as many more correct fixes are admitted. The optimal IA decisions only differ from the GIAB decisions in a small fraction of cases. The scatter plot are color coded by optimal IA event: dark gray for success, light gray for undecided, and large red for failure. 47

3.1 The rejection event R_{i+1} is triggered when $\epsilon_{\mathrm{c}(i+1)}$ falls within the indicated bands. For compactness, β_{i+1} is abbreviated as β. 61

3.2 A single component of $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{b} + \boldsymbol{\xi}|\varepsilon)$ from (3.33) for float, MAP, and MMSE GIAB for $q = i < m$ and a large ambiguity residual ε. Because the distributions differ only by the variant-specific bias $\boldsymbol{\mu}_k(\varepsilon)$, the three PDFs are merely shifted versions of each other, with that of MMSE GIAB between those of float and MAP GIAB. . 69

3.3 A single component of $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{b}+\boldsymbol{\xi})$ for float, MAP, and MMSE GIAB with $q = i < m$, plotted with a log-scaled vertical axis. Float GIAB has a symmetric, bimodal PDF. MAP GIAB has a symmetric PDF with a dominant, zero-mean central mode and heavy tails due to incorrect fixes. MMSE GIAB has a lower probability of large errors than does MAP GIAB at the expense of a slight increase in the probability of moderately-sized errors. In the lightly-shaded region, MAP GIAB has higher density than MAP GIAB, and vice versa in the darker region. 70

3.4 A single component of $f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{b}+\boldsymbol{\xi})$ for float, MAP, and MMSE GIAB, plotted with a log-scaled vertical axis. Float GIAB has strong but narrow tails. Its central mode results from the probability of fixing all m ambiguities. MAP and MMSE GIAB both have a strong central mode and tails that are wider but lower than those of float GIAB. MMSE GIAB has smoother and narrower tails than MAP GIAB. 71

3.5 A single component of $f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{b}+\boldsymbol{\xi})$ for the theoretical PDF of MAP GIAB (dashed line) and the empirical (simulated) histogram (solid line), plotted in log scale for $m = 7$, $\bar{P}_F = 10^{-8}$, and a Monte Carlo sample size of 4×10^8. The underlying model has an IB probability of correct fix equal to $1 - 2 \times 10^{-5}$. As the PDFs are symmetric about zero, only the positive portion is shown. Clearly, there is strong agreement between the analytical and simulated distributions. Note that because the plot's vertical axis is log scaled, small differences are exaggerated at low probabilities. 72

4.1 The risk of excess error for a MLV solution is a function of the correlations among the three solutions. When the solutions are fully correlated, the risk is equivalent to that of a simplex solution. When the solutions are independent, the risk is greatly reduced. The example of partially correlated errors shown is for a correlation coefficient $\rho = 0.5$ between each pair among the three solutions. 93

4.2 The positive portion of the modeled PDFs of the fixed, GERAFS, and EPIC baseline errors. The fixed baseline has a Gaussian mixture distribution with biases and weights determined by the incorrect fixes. Whereas the EPIC baseline follows the fixed baseline closely for incorrect fixes of non-negligible probability, the GERAFS baseline assigns all of the non-negligible probability of incorrect fix to the fix with the largest incorrect fixing bias. 100

4.3 Comparison of availability vs. accuracy for float, GERAFS, and MLV triplex solutions for varying AL. As the AL is reduced, the GERAFS algorithm is less frequently able to fix the integer ambiguities. When the fixed solution is unavailable, both the GERAFS and MLV triplex solutions use a float solution as a backup. The performance of the GERAFS solution degrades to that of the float solution when $AL = 1.5m$ since the fixed solution is completely unavailable. Even when the MLV triplex solution uses the float solution, the integrity and accuracy benefits of MLV enable a 15% increase in availability and an 8 cm improvement in 70% accuracy. . 110

A.1 An illustration of the geometric relationships among two GNSS satellites and two GNSS receivers. A CDGNSS system is interested in computing the relative position vector, or baseline, between the two receivers. This baseline is denoted $\boldsymbol{b}$. The vectors with subscripts, e.g. $\boldsymbol{r}_i$, indicate receiver positions, and those vectors with superscripts, e.g. $\boldsymbol{r}^k$, indicate satellite positions. Vectors with both super- and subscripts, e.g. $\boldsymbol{r}_i^k$, indicate relative position vectors from satellites to receivers. 116

A.2 A block diagram of the low pass filter form of the carrier smoothed code observable. In this approach, the difference between the carrier phase and the pseudorange is smoothed with a low pass filter with a time-varying gain. This difference is then added back to the carrier phase to form a smoothed estimate of the pseudorange. . . . 121

A.3 Divergence-free smoothing is accomplished by weighting the input pseudoranges and carrier phases such that the ionospheric delay in the pseudorange is canceled by the carrier phase used in the smoother. In this way, reference and rover receivers are able to independently smooth their pseudoranges without decorrelating the ionospheric delay. 124

D.1 The set of non-negligible fix candidates can be represented as a binary tree of depth r, shown for $r = 3$. See the body text for a full interpretation of the tree. The nodes are numbered according to the order visited by an in-order traversal. Unshaded leaf-nodes correspond to alternative IB solutions. Non-leaf nodes, which are all shaded, partial IB solutions leading to the different alternatives. The shade of lines matches the shade of the nearest ancestor node that generated the corresponding IB solution leaf, e.g. node 4 and leaf-node 5. Dashed lines indicate neglected branches of the tree. . 139

D.2 Illustration of part of the process of building the binary tree of alternate fixes via an in-order tree traversal. The tree is initialized with the IB solution as the left-children of the root-node. The first node visited is the leaf corresponding to the IB solution. The second node visited, labeled 2, produces the first alternative solution, with offset $\boldsymbol{\zeta}_1$, which is stored as the right-child of node 2, and visited next. Node 4 is visited next, which leads to the second alternative solution, with offset $\boldsymbol{\zeta}_2$. This alternative fix is stored in the branch with nodes labeled 6 and 5. Node 5 is visited next and added to $\mathbb{Z}$ and $\lambda_{\mathbb{Z}}$. The alternative branches at nodes 6 and 7 are then found to be negligible, completing the tree traversal and construction. 141

Chapter 1

Introduction

1.1 Carrier Phase Differential Global Navigation Satellite System Positioning

Global Navigation Satellite System (GNSS) receivers produce two fundamental types of measurement. The first, called pseudorange, is a noisy, biased estimate of the total distance between the satellite's antenna and the receiver's antenna. Sources of pseudorange error include atmospheric delay, multipath interference from reflected signals, thermal noise in the electronics, receiver clock errors, and errors in the broadcast satellite orbit and clock models. Even when corrected using data from the Wide Area Augmentation System (WAAS), these errors exceed 2 meters in standard deviation [24, 38].

The second type of measurement, called carrier phase, is a less noisy, though still biased measurement of the change in range between the satellite and the receiver antennas from the time that carrier phase lock was established until the time of the measurement. Many of the error sources that afflict pseudoranges also impact carrier phases, including atmospheric delays and broadcast model errors, but these common errors are mitigated via double-difference processing, as described in Appendix A. Whereas the uncorrected differential pseudorange errors are several meters in magnitude, the corresponding carrier phase errors are smaller than

a centimeter. The challenge for CDGNSS is that carrier phase only measures the change in range, and so is ambiguous with respect to the full range to the satellite. Fortunately, this ambiguity is an integer multiple of the wavelength of the GNSS carrier signal.

The process of estimating the integer ambiguities is called integer ambiguity ambiguity resolution or integer fixing. Two of the most common methods for integer fixing are Integer Bootstrapping (IB) [30] and Integer Least Squares (ILS) [32]. The probability that an integer estimator correctly resolves the ambiguities is called the probability of correct fix, or P_{CF}. ILS is an optimal estimator in that it has the maximum P_{CF} of all permissible integer estimators. The ILS probability of correct fix, $P_{CF,\mathrm{ILS}}$ has no analytical formula, but its tightest lower bound is the IB probability of correct fix, $P_{CF,\mathrm{IB}} \leq P_{CF,\mathrm{ILS}}$.

Prior to estimating the ambiguities as integers, they are estimated jointly with the relative position, or baseline, $\boldsymbol{b}$, between two GNSS antennas as real-valued approximations. This joint estimate is called the float solution comprising the float baseline, $\hat{\boldsymbol{b}}$, and the float ambiguity, $\hat{\boldsymbol{a}}$, alluding to the floating-point representation of real-valued numbers in computers. The float solution is derived in Appendix B.

Once the fixed ambiguities, $\check{\boldsymbol{a}}$, are estimated, the float baseline is corrected based on the residual between the float and fixed ambiguities. The resulting baseline estimate is called the fixed baseline, denoted $\check{\boldsymbol{b}}$. If the fixed ambiguities are correct, then the fixed baseline is an unbiased estimate of the true baseline, $\boldsymbol{b}$; however, if the fixed ambiguities are incorrect, then the fixed baseline can have a bias that is excessively large, often exceeding the float baseline error. This dissertation is

concerned primarily with methods to validate the correctness of the fixed ambiguity estimate and to bound the position-domain errors in the baseline solution in the event that the fixed ambiguities cannot be fully validated.

1.2 Required Navigation Performance

Safety-of-life navigation systems are specified using the concepts of Required Navigation Performance (RNP). RNP is assessed in terms of integrity, accuracy, continuity, and availability [6]. Integrity is specified in terms of integrity risk (IR), the probability that the solution error exceeds an alert limit (AL) without warning. Accuracy can be specified in terms of quantiles of interest, such as 95% accuracy, which refers to the error volume within which 95% of solutions fall. Continuity risk (CR) is the probability that the solution will become unavailable during a critical exposure interval given that it was available at the beginning of that interval. Availability is the percentage of time that the solution satisfies its required integrity, accuracy, and continuity requirements.

RNP for CDGNSS systems has become more demanding with each new application. The ground-based augmentation system (GBAS), originally specified over a decade ago as a landing aid for large runways on land, required ALs of 10 m with IR on the order of 10^{-7} per approach. This leads to a relatively loose 95% accuracy requirement of 2 m for a zero-mean-error Gaussian-distributed solution, which can be met by a float CDGNSS solution. More recent navigation system applications, such as landing aboard an aircraft carrier or a recent demonstration of autonomous aerial refueling [1], have meter-level ALs, IR on the order of 10^{-7},

and decimeter-level accuracy requirements.

The next generation of safety-of-life CDGNSS use cases includes fully autonomous landing and refueling of large, unmanned aerial vehicles, and automated land vehicle navigation. These applications will demand centimeter- to decimeter-level position accuracy and more stringent solution integrity than all previous applications. Integrity is specified in terms of integrity risk (IR), the probability that the solution error exceeds an alert limit (AL) without warning. The percentage of time that a system meets its required navigation performance, including accuracy and IR, is called solution availability. For safety-of-life applications, IR is between of 10^{-9} and 10^{-7} per hour, with required availability in excess of 99%.

1.3 CDGNSS and RNP

The ILS probability of incorrect fix, or failure, is denoted $P_{F,\mathrm{ILS}} = 1 - P_{CF,\mathrm{ILS}}$. It is common for $P_{F,\mathrm{ILS}}$ to be orders of magnitude greater than IR. $P_{F,\mathrm{ILS}}$ is a function of the measurement quality, satellite geometry, and number of measurements used. Typically, $0.001 < P_{F,\mathrm{ILS}} < .1$ for most CDGNSS applications [3, 16]. Because $IR \ll P_{F,\mathrm{ILS}}$ frequently, safety-of-life systems have not historically used fixed solutions. To exploit the accuracy benefit of the fixed solution while protecting solution integrity, the biases induced by any incorrect fixes must be addressed. Two such methods are the Geometry Extra Redundant Almost Fixed Solutions (GERAFS) [42] and the Enforced Position-domain Integrity-risk of Cycle resolution (EPIC) [17, 18] algorithms.

In both EPIC and GERAFS, the IB algorithm is used to fix the integer ambi-

guities because there are analytic expressions for $P_{CF,\mathrm{IB}}$ as well as the probabilities that IB will incorrectly resolve the ambiguities to any particular integer offset from the true ambiguity [29]. Denote the IB fixed baseline and fixed ambiguities as $\check{\boldsymbol{b}}_{\mathrm{IB}} \in \mathbb{R}^3$ and $\check{\boldsymbol{a}}_{\mathrm{IB}} \in \mathbb{Z}^m$, respectively. Each potential integer offset between the true and IB ambiguity, $\Delta\check{\boldsymbol{a}}_k \in \mathbb{Z}^m$ would result in a particular bias in the baseline estimate, $\boldsymbol{\mu}_k$.

The total probability that the fixed baseline estimate has errors that exceed the AL in vertical or horizontal dimensions can then be computed by considering the alternative hypotheses that the true ambiguity is $\boldsymbol{a} = \check{\boldsymbol{a}}_{\mathrm{IB}} - \Delta\check{\boldsymbol{a}}_k$. Let b and $\check{b}_{\mathrm{IB}}$ denote the vertical components of $\boldsymbol{b}$ and $\check{\boldsymbol{b}}_{\mathrm{IB}}$, respectively. Similarly, VAL is the AL in the vertical direction. The IR for the IB solution is

$$P\left(\left|\check{b}_{\mathrm{IB}} - b\right| > VAL\right) = \sum_{\Delta\check{\boldsymbol{a}}_k \in \mathbb{Z}^m} P\left(\left|\check{b}_{\mathrm{IB}} - b\right| > VAL \middle| \check{\boldsymbol{a}}_{\mathrm{IB}} = \boldsymbol{a} + \Delta\check{\boldsymbol{a}}_k\right) \quad (1.1)$$

GERAFS and EPIC both bound this probability and issue an alert whenever the bound exceeds a specified IR. A similar computation must be made for both the vertical and horizontal components of the baseline estimate. Both GERAFS and EPIC use measurement models to decide *a priori* whether to use the fixed or float baseline estimate. Such methods called "model-driven."

Alternatively, data-driven methods use *a posteriori* statistical tests on the ambiguity residual , denoted $\check{\boldsymbol{\epsilon}} = \hat{\boldsymbol{a}} - \check{\boldsymbol{a}}$, to choose between the float and fixed baseline estimates. Such methods aim to reduce the probability of accepting an incorrect fix by rejecting fixes with ambiguity residuals that are large in a statistical sense. One class of data-driven methods is Integer Aperture (IA) estimators [28].

Many such methods have been proposed including the ratio test [37], the difference test [43], the projector test [39], integer aperture bootstrapping [33], ellipsoidal IA [31], and the optimal IA estimator [36]. When extended to include partial ambiguity resolution, IA estimators are called Generalized Integer Aperture (GIA) estimators [4].

Existing IA methods are insufficient for safety-of-life applications, as will be discussed in Ch. 2, but they are attractive because they validate the correctness of the integer ambiguity estimate, eliminating the need to account for the position-domain biases by incorrect fixes. If an IA method could be devised such that P_F can be proven to be less than a specified failure rate, $\bar{P}_F < IR$, then such a system could satisfy smaller ALs since the IA method would preclude large biases in the baseline estimate. This dissertation develops such a GIA method and demonstrates its performance. An important result of this work is that GIA methods imply position-domain biases when the full set of ambiguities is unable to be validated. The concept of position-domain integrity from the model-driven EPIC algorithm is extended to the novel data-driven method developed in this dissertation.

1.4 Triplex CDGNSS Architectures

A common design approach in safety-of-life systems to improve continuity performance is to have three subsystems that perform the same function. Such designs are called triplex architectures. Such architectures provide continuity of the required function with a redundant backup even when a single subsystem fails. The outputs of the three systems can be combined using mid-level voting (MLV),

in which the middle value of each output from the subsystems is selected for use by other parts of the overall system.

MLV-triplex architecture provide the additional benefit that latent faults – faults present in the system, but not yet detected or excluded – will not corrupt the final output from the MLV logic. This benefit reduces integrity risk since a latent fault in a simplex (non-redundant) architecture can cause a large positioning error. Even in a triplex architecture that averages the three solutions, a single large error can cause the final estimate to have an error that exceeds an AL without an alarm. Because the integrity requirements of current and future CDGNSS positioning systems are so demanding, it is worth considering the potential benefits of MLV-triplex architectures on fault-free integrity performance.

1.5 Thesis Statement and Expected Contributions

This dissertation defends the following thesis statement:

Integer-fixed CDGNSS positioning for demanding safety-of-life applications requires a novel data-driven integer ambiguity validation method which benefits further from mid-level voting triplex architectures.

The following is a summary of the contributions of this dissertation:

1. **Generalized Integer Aperture Bootstrapping:** The Generalized Integer Aperture Bootstrapping (GIAB) algorithm is developed to provide provable guarantees that the fixed integer ambiguities are correct with a known probability of failure. The performance of GIAB is compared to existing IA meth-

ods, and the theoretical performance of the algorithm is validated via Monte Carlo simulation. This work has been published in [11, 15].

2. **Data-Driven Position-Domain Integrity for Generalized Integer Aperture Bootstrapping:** The several important conditional probability density functions are derived for the baseline estimate produced by the GIAB algorithm. We prove that baseline estimates produced from data-driven partial ambiguity resolution methods have biased *a posteriori* error distributions even when the baseline estimate is only constrained based on correctly validated ambiguities. The *a posteriori* distribution of the GIAB baseline estimate is used to derive a data-driven position-domain integrity approach, ensuring that GIAB is fully appropriate for safety-of-life applications. The theoretical distributions derived in this paper are validated via Monte Carlo simulation. The performance of GIAB is compared to the EPIC. This work has been published in [12, 15].

3. **Correlation-Aware Integrity Protection for Fault-Free MLV-Triplex CDGNSS Solutions:** Integrity risk monitors are derived for MLV-triplex CDGNSS architectures using float, fixed, or position- domain integrity solutions. The performance of all monitors is compared using a world-wide covariance analysis tool, called the availability model (AM). Improvements for each type of CDGNSS solution are shown by comparison to simplex versions of the same algorithms. This work has been published in [10, 13, 14].

1.6 Published Works

The publications to which the author contributed during the course of carrying out the contributions described in this dissertation are as follows:

Journal Publications

1. G. Nathan Green and Todd E. Humphreys. Data-driven generalized integer aperture bootstrapping for high-integrity positioning. *IEEE Transactions on Aerospace and Electronic Systems*, 2017. Submitted for review.

2. G. Nathan Green and Todd E. Humphreys. Position-domain integrity analysis for generalized integer aperture bootstrapping. *IEEE Transactions on Aerospace and Electronic Systems*, 2017. Submitted for review.

3. G. Nathan Green and Todd E. Humphreys. Correlation-aware integrity protection for fault-free federated triplex CDGNSS solutions. *Navigation, Journal of the Institute of Navigation*, 2018. In preparation.

Conference Publications

1. G. Nathan Green, Martin King, and Todd E. Humphreys. Fault free integrity of mid-level voting for triplex differential GPS solutions. In *Proceedings of the ION GNSS+ Meeting*, Tampa, FL, 2015.

2. G. Nathan Green, Martin King, and Todd E. Humphreys. Data-driven generalized integer aperture bootstrapping for real-time high integrity applications. In *Proceedings of the IEEE/ION PLANS Meeting*, Savannah, GA, 2016.

3. G. Nathan Green and Todd E. Humphreys. World-wide triplex CDGNSS performance. In *"The Proceedings of the Royal Institute of Navigation"*, Glasgow, Scotland, UK, 2016.

1.6.1 Dissertation Organization

Chapter 2 develops the GIAB algorithm, including a method to establish the acceptance thresholds to rigorously control P_F while nearly maximizing P_{CF}. Chapter 3 derives important conditional probability density functions of the GIAB-produced baseline estimate, validates the correctness of the distributions via Monte Carlo simulation, extends model-driven position-domain integrity concepts to data-driven position-domain integrity, and demonstrates the performance improvement of GIAB over the existing state-of-the-art algorithm, EPIC. Chapter 4 develops MLV-triplex CDGNSS integrity risk monitors and compares their performance via the AM tool. Chapter 5 concludes this dissertation with a summary of the contributions.

Some of the longer derivations of important results from the contributions are collected in appendices. Appendix A provides a detailed derivation of a GNSS measurement model. Appendix B derives the linearized least-squares estimator called the float solution. Appendix C explores important properties of the float ambiguity error and the IB algorithm. Appendix D derives a method to efficiently search all non-negligible integer ambiguity alternatives while providing a provable bound on the integrity risk taken by neglecting unexplored alternatives. Appendix E derives an integrity monitor for MLV-triplex solutions when the correlations among

the three solutions are unknown.

1.7 Nomenclature

AL	Alert Limit
AM	Availability Model
BIE	Best Integer Equivariant
BRSD	Between Receiver Single Difference
CCD	Code Carrier Divergence
CDF	Cumulative Density Function
CF	Correct Fix
CR	Continuity Risk
CSC	Carrier Smoothed Code
DFS	Divergence Free Smoothing
EPIC	Enforced Position-domain Integrity-risk of Cycle resolution
EPIR	Enlarged Pull-In Region
GBAS	Ground Based Augmentation System
GERAFS	Geometry Extra Redundant Almost Fixed Solutions
GIA	Generalized Integer Aperture
GIAB	Generalized Integer Aperture Bootstrapping

GNSS	Global Navigation Satellite System
IA	Integer Aperture
IAB	Integer Aperture Bootstrapping
IB	Integer Bootstrapping
ILS	Integer Least Squares
IR	Integrity Risk
LPF	Low Pass Filter
MAP	Maximum *A Posteriori*
ML	Maximum Likelihood
MLV	Mid-Level Voting
MMSE	Miminum Mean Square Error
PAR	Partial Ambiguity Resolution
PDF	Probability Density Function
RF	Radio Frequency
RNP	Required Navigation Performance
SV	Space Vehicle
WAAS	Wide Area AugmentationSystem

Chapter 2

Generalized Integer Aperture Bootstrapping

2.1 Introduction

This chapter [1] focuses on the portion of the IR budget allocated to incorrect resolution, or fixing, of the carrier-phase integer ambiguities that are a central feature of CDGNSS positioning. This portion is specified as the acceptable probability of incorrect fix, or failure rate, $\bar{P}_F$. High-integrity CDGNSS techniques must provably satisfy demandingly low $\bar{P}_F$. Two such methods are the Geometry Extra Redundant Almost Fixed Solutions (GERAFS) [42] and the Enforced Position-domain Integrity-risk of Cycle resolution (EPIC) [17, 18] algorithms. Both of these rely exclusively on *a priori* error models to determine, before the measurements are processed, whether a fixed solution or a float backup solution will be selected. This approach is termed model-driven because the solution selection logic is entirely dependent on the prior error model. Because the EPIC and GERAFS algorithms attempt to bound IR using the *a priori* distribution, they are inherently conservative.

[1] This chapter was based on

1. G. Nathan Green and Todd E. Humphreys. Data-driven generalized integer aperture bootstrapping for high-integrity positioning. *IEEE Transactions on Aerospace and Electronic Systems*, 2017. Submitted for review.

The author contributed all derivation, simulation, and analysis in that paper and in this chapter.

Their conservatism arises from the need to protect against potentially-incorrect fixes without the benefit of conditioning on the observed carrier phase measurements.

In contrast to the model-driven approach, data-driven methods decide *a posteriori* whether to accept the fixed or float solution. Conditioning the selection on the observed measurements can reduce the risk of incorrect fixing. A subset of data-driven methods is called integer aperture (IA) estimation. In IA methods, the integer ambiguity vector is estimated using either integer bootstrapping (IB) [26] or integer least squares (ILS) [34], after which a test statistic is computed from the ambiguity residual, i.e., the difference between the float and fixed ambiguities. This test statistic is compared to a threshold to decide between the fixed and float solution.

Perhaps the simplest IA method is IA bootstrapping (IAB), which resolves the integer ambiguities via IB and then tests the fixed solution by applying IB to a scaled-up version of the ambiguity residual [33]. If the test returns the zero vector, which occurs only when the residual is small, then the fixed solution is selected; otherwise the float solution is selected. IAB is sub-optimal in the sense that IB does not always find the maximum likelihood integer ambiguity, as opposed to ILS, which is guaranteed to do so. It is also sub-optimal in the sense that it does not maximize the probability of successfully fixing the ambiguities for a given probability of incorrectly fixing them. But it has the advantage that all of these probabilities have analytically computable values, which allows the decision threshold to be set analytically as a function of $\bar{P}_F$. More generally, IAB enables the strict performance requirements that safety-of-life applications demand to be provably satisfied.

The remaining IA methods discussed in this section solve for the integer ambiguity with ILS, which is optimal in the maximum likelihood sense for Gaussian measurement noise. Ellipsoidal IA takes the covariance weighted norm of the ILS ambiguity residual as its test statistic [31]. As with IAB, the simplicity of this statistic allows the decision threshold to be set analytically, but the probability of successfully fixing the ambiguities is sub-optimal. While ellipsoidal IA can have a higher probability of success than IAB for models with a few ambiguities of approximately equal conditional variance, IAB tends to provide a higher probability of success for models in which the conditional variances of the ambiguities differ by more than about 10%, which tends to be the case for realistic measurement models.

Other ILS-based IA methods employ test statistics that are a function of the ambiguity residuals of the ILS fix and of one or more higher-cost alternate fixes. These include the ratio test [37], the difference test [43], and the optimal test [36]. Unlike IAB and ellipsoidal IA, none of these methods' test statistics has an analytical probability distribution or decision threshold [35]. In practice, decision thresholds are set based on one of a few *ad hoc* methods. The crudest of these, which applies a fixed threshold for all measurement models, does not allow one to control the actual probability of incorrect fix, P_F, for time-varying measurement models. More sophisticated methods determine the decision threshold that approximately satisfies $\bar{P}_F$ via Monte Carlo simulation, lookup tables [37], or functional approximations [3, 40]. But these techniques are inapt for safety-of-life systems because the resulting thresholds cannot be analytically proven to satisfy $\bar{P}_F$ for any particular model. At best, they incorporate sufficient conservatism to protect the solution

at the expense of decreased availability. Of course, in the limit as the number of test points becomes exceedingly large, Monte Carlo simulation for a given measurement model can yield an arbitrarily exact decision threshold, but such simulation is hardly feasible for real-time operation.

The optimal IA algorithm takes as its test statistic the *a posteriori* probability of correct fix [27]. Counterintuitively, the threshold corresponding to a particular $\bar{P}_F$ for this statistic is not analytically computable. Also, the optimal IA estimator involves an infinite sum over all possible integer ambiguities. The search can be truncated once a sufficiently large number of integer fixes has been evaluated, but the number required depends on the strength of the model and on the required $\bar{P}_F$. To satisfy the most demanding integrity requirements, the search often extends to several hundred candidate fixes in realistic scenarios, which becomes impractical for real-time applications.

This chapter's focus on IAB is motivated by the alternative IA approaches' computational complexity or lack of an analytical connection between $\bar{P}_F$ and the decision threshold.

This chapter extends the IAB technique to a generalized form in which subsets of the full set of integer ambiguities are considered for resolution if the full set cannot be resolved confidently. This generalization makes IAB a member of the family of Generalized Integer Aperture (GIA) estimators [4]. These algorithms evaluate successively smaller subsets until either a satisfactory fix is found or the float solution is applied as a last resort. Also known as partial ambiguity resolution (PAR), this technique ensures gradual degradation of performance for weak models.

In summary, to meet the increasingly stringent performance requirements of safety-of-life applications there is a need for a data-driven ambiguity resolution and validation method whose decision threshold for choosing between a fixed and float solution can be set analytically for a desired $\bar{P}_F$. To maximize availability, the method must be generalized to accommodate PAR. Extant methods in the high-integrity CDGNSS literature do not satisfy this need.

This chapter offers three contributions to address this need. First, IAB is extended to encompass PAR. The extended technique is called Generalized Integer Aperture Bootstrapping (GIAB). Second, analytical characterizations of the probability of incorrect fix, correct partial fix, and correct full fix are developed and validated. Third, a method for setting the integer aperture size and shape is developed that ensures GIAB's availability exceeds IAB's subject to a given $\bar{P}_F$. These contributions are validated with a set of Monte Carlo simulations, and algorithm performance is compared to the optimal IA, ellipsoidal IA, and IAB methods.

2.2 Generalized Integer Aperture Bootstrapping

2.2.1 Integer Bootstrapping Overview

The basic theory of IB is reproduced here from [26] with a few amplifications for ease of understanding and notational consistency. The treatment begins with the linearized, short-baseline GNSS measurement model

$$\boldsymbol{y} = B\boldsymbol{b} + A\boldsymbol{a} + \boldsymbol{\nu} \tag{2.1}$$

where $\boldsymbol{y} \in \mathbb{R}^n$ contains the "observed-minus-modeled" double-difference carrier-phase and, optionally, pseudorange measurements, $\boldsymbol{b} \in \mathbb{R}^3$ is the unknown, real-valued correction to the modeled baseline between GNSS antennas, $\boldsymbol{a} \in \mathbb{Z}^m$ holds the unknown carrier phase integer ambiguities, B and A are appropriately dimensioned measurement sensitivity matrices, and $\boldsymbol{\nu} \in \mathbb{R}^n$ is the zero-mean, double-difference measurement noise with variance $Q_{\boldsymbol{y}}$. See Appendix B for a derivation of this measurement equation and Appendix A for a derivation of an appropriate model for $Q_{\boldsymbol{y}}$.

Applying weighted least squares estimation to (2.1), with $H = [B\ A]$, produces real-valued estimates of $\boldsymbol{b}$ and $\boldsymbol{a}$:

$$\begin{bmatrix} \hat{\boldsymbol{b}} \\ \hat{\boldsymbol{a}} \end{bmatrix} = \left(H^T Q_{\boldsymbol{y}}^{-1} H\right)^{-1} H^T Q_{\boldsymbol{y}}^{-1} \boldsymbol{y} \tag{2.2a}$$

$$E\left(\begin{bmatrix} \hat{\boldsymbol{b}} \\ \hat{\boldsymbol{a}} \end{bmatrix}\right) = \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{a} \end{bmatrix} \tag{2.2b}$$

$$\operatorname{cov}\left(\begin{bmatrix} \hat{\boldsymbol{b}} \\ \hat{\boldsymbol{a}} \end{bmatrix}\right) = \begin{bmatrix} Q_{\hat{\boldsymbol{b}}} & Q_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}} \\ Q_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}}^T & Q_{\hat{\boldsymbol{a}}} \end{bmatrix} = \left(H^T Q_{\boldsymbol{y}}^{-1} H\right)^{-1} \tag{2.2c}$$

The estimates $\hat{\boldsymbol{a}} \in \mathbb{R}^m$ and $\hat{\boldsymbol{b}} \in \mathbb{R}^3$, called the float ambiguity and float baseline, ignore the integer constraint $\boldsymbol{a} \in \mathbb{Z}^m$.

Integer ambiguity resolution techniques map the float ambiguity $\hat{\boldsymbol{a}}$ to a vector of integers $\check{\boldsymbol{a}} \in \mathbb{Z}^m$. Such processes can be represented by the map

$$\check{\boldsymbol{a}} = M(\hat{\boldsymbol{a}}, Q_{\hat{\boldsymbol{a}}}) : \mathbb{R}^m \times \mathbf{S}_{++}^m \mapsto \mathbb{Z}^m \tag{2.3}$$

where $\mathbf{S}_{++}^m$ is the set of positive definite matrices of size $m \times m$. The IB variant of M operates in such a way that when $Q_{\hat{\boldsymbol{a}}}$ has non-zero off-diagonal elements, the probability that $\check{\boldsymbol{a}} = \boldsymbol{a}$ depends on the ordering of the elements of $\hat{\boldsymbol{a}}$ [26]. To ensure near-optimal IB performance, an integer-preserving transformation is applied to decorrelate, insofar as possible, the elements of $\hat{\boldsymbol{a}}$; details of this transformation, referred to as the Z-transform, may be found in [25, 32]. The decorrelated float ambiguity is $\hat{\boldsymbol{z}} = Z^T\hat{\boldsymbol{a}}$, and the transformed true ambiguity is $\boldsymbol{z} = Z^T\boldsymbol{a}$, with Z being the integer-preserving transformation matrix. Likewise, $Q_{\hat{\boldsymbol{a}}}$ and $Q_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}}$ are transformed as $Q_{\hat{\boldsymbol{z}}} = Z^T Q_{\hat{\boldsymbol{a}}} Z$ and $Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} = Q_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}} Z$. All integer-related operations hereafter will be performed in the decorrelated space, with $\hat{\boldsymbol{z}}$ referred to as the float ambiguity.

The functional map $M(\hat{\boldsymbol{z}}, Q_{\hat{\boldsymbol{z}}})$ partitions $\mathbb{R}^m$ into disjoint subsets, called pull-in regions, that collectively cover $\mathbb{R}^m$. Each region is an integer-valued translation of the subset

$$S_{\mathbf{0}} \triangleq \{\hat{\boldsymbol{z}}\,|M(\hat{\boldsymbol{z}}, Q_{\hat{\boldsymbol{z}}}) = \mathbf{0}\} \tag{2.4}$$

The pull-in region $S_{\boldsymbol{\zeta}} \subset \mathbb{R}^m$ is the set of all $\hat{\boldsymbol{z}}$ mapped by $M(\hat{\boldsymbol{z}}, Q_{\hat{\boldsymbol{z}}})$ to the integer vector $\boldsymbol{\zeta} \in \mathbb{Z}^m$:

$$S_{\boldsymbol{\zeta}} \triangleq \{\hat{\boldsymbol{z}}\,|M(\hat{\boldsymbol{z}}, Q_{\hat{\boldsymbol{z}}}) = \boldsymbol{\zeta},\, \boldsymbol{\zeta} \in \mathbb{Z}^m\} = S_{\mathbf{0}} + \boldsymbol{\zeta} \tag{2.5}$$

For IB, the pull-in regions are m-dimensional parallelotopes centered on the integers.

For presentation of the IB algorithm, it will be convenient to decompose the covariance of the float ambiguity as $Q_{\hat{\boldsymbol{z}}} = LDL^T$, where L is a unit lower triangular

matrix and D is a diagonal matrix, and to model the float ambiguity as the true ambiguity plus zero-mean Gaussian noise, $\hat{\boldsymbol{z}} = \boldsymbol{z}+\boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(0, Q_{\hat{\boldsymbol{z}}})$. Multiplication by L^{-1} transforms $\boldsymbol{\epsilon}$ into a vector whose elements are mutually uncorrelated: $\boldsymbol{\epsilon}_{\mathrm{c}} \triangleq L^{-1}\boldsymbol{\epsilon},\ \boldsymbol{\epsilon}_{\mathrm{c}} \sim \mathcal{N}(0, D)$. Letting l_{ij} denote the ijth element of L, d_i the ith element of the diagonal of D, and ϵ_i and $\epsilon_{\mathrm{c}i}$ the ith elements of $\boldsymbol{\epsilon}$ and $\boldsymbol{\epsilon}_{\mathrm{c}}$, respectively, ϵ_i and its variance can be computed from the first i components of $\boldsymbol{\epsilon}_{\mathrm{c}}$ as

$$\epsilon_i = \sum_{k=1}^{i} l_{ik}\epsilon_{\mathrm{c}k}, \quad \mathrm{var}\,(\epsilon_i) = \sum_{k=1}^{i} l_{ik}^2 d_k \tag{2.6}$$

IB can be interpreted as greedy constrained maximum likelihood estimation in which the integer constraint $\boldsymbol{z} \in \mathbb{Z}^m$ is applied sequentially. Application of the integer constraint can also be viewed as conditioning on an assumed value of $\boldsymbol{\epsilon}$. For convenience in what follows, let the shorthand notation $\boldsymbol{v}_I$ denote the vector composed of the first $i-1$ elements of any vector $\boldsymbol{v}$ of sufficient length. Thus, $\boldsymbol{\epsilon}_I = [\epsilon_1, \ldots, \epsilon_{i-1}]^T$. Let $\epsilon_j|\boldsymbol{\epsilon}_I$ represent the jth element of $\boldsymbol{\epsilon}$ conditioned on $\boldsymbol{\epsilon}_I$ being known. Starting with (2.6), and exploiting the lack of correlation in the elements of $\boldsymbol{\epsilon}_{\mathrm{c}}$, it is shown in Section C.1 that

$$\epsilon_j|\boldsymbol{\epsilon}_I \sim \mathcal{N}\left(\sum_{k=1}^{i-1} l_{jk}\epsilon_{\mathrm{c}k},\ \sum_{k=i}^{j} l_{jk}^2 d_k\right), \quad j = i, ..., m \tag{2.7}$$

Note that $\mathrm{var}(\epsilon_i|\boldsymbol{\epsilon}_I) = d_i$. Thus, d_i can be interpreted as the conditional variance of the ith ambiguity. A larger value of d_i indicates that correct integer resolution of the ith ambiguity will be more difficult.

One may alternatively find the mean of $\epsilon_j|\boldsymbol{\epsilon}_I$ via the standard expression for conditional mean. Assume $\boldsymbol{\epsilon}_I \sim \mathcal{N}(0, Q_I)$, and let $Q_{jI} \in \mathbb{R}^{1\times(i-1)}$ be the cross-

correlation matrix between ϵ_j and $\boldsymbol{\epsilon}_I$, for $j \geq i$. Then the mean of ϵ_j conditioned on knowledge of $\boldsymbol{\epsilon}_I$ is [2]

$$E[\epsilon_j|\boldsymbol{\epsilon}_I] = Q_{jI}Q_I^{-1}\boldsymbol{\epsilon}_I \tag{2.8}$$

With these preliminaries, the algorithm for a single step of IB is straightforward. Let $\boldsymbol{z}_I = [z_1, \ldots, z_{i-1}]^T$, and suppose that one assumes $\boldsymbol{z}_I = \check{\boldsymbol{z}}_I$ for some known $\check{\boldsymbol{z}}_I = [\check{z}_1, \ldots, \check{z}_{i-1}]^T \in \mathbb{Z}^{i-1}$. Then, starting from $\hat{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{\epsilon}$, the constrained maximum likelihood estimate of z_i given $\boldsymbol{z}_I = \check{\boldsymbol{z}}_I$ is

$$\hat{z}_{i|I} = \hat{z}_i - Q_{iI}Q_I^{-1}(\hat{\boldsymbol{z}}_I - \check{\boldsymbol{z}}_I) \tag{2.9}$$

Defining $\check{\boldsymbol{\epsilon}} \triangleq \hat{\boldsymbol{z}} - \check{\boldsymbol{z}}$ and $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \triangleq L^{-1}\check{\boldsymbol{\epsilon}}$, and referencing (2.7) and (2.8), one recognizes (2.9) as equivalent to

$$\begin{aligned} \hat{z}_{i|I} &= \hat{z}_i - E\left[\epsilon_i \middle| \boldsymbol{\epsilon}_I = \check{\boldsymbol{\epsilon}}_I\right] \\ &= \hat{z}_i - \sum_{k=1}^{i-1} l_{ik}\check{\epsilon}_{\mathrm{c}k} \\ &= z_i + \sum_{k=1}^{i} l_{ik}\epsilon_{\mathrm{c}k} - \sum_{k=1}^{i-1} l_{ik}\check{\epsilon}_{\mathrm{c}k} \end{aligned} \tag{2.10}$$

where the last equality makes use of $\hat{z}_i = z_i + \epsilon_i$ and (2.6). The quantities $\hat{z}_{i|I},\ i = 1, 2, ..., m$ are called the sequentially-constrained float ambiguity estimates; these are stacked to form the vector $\hat{\boldsymbol{z}}_{\mathrm{c}}$.

It is shown in Appendix C.2 that $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} = \hat{\boldsymbol{z}}_{\mathrm{c}} - \check{\boldsymbol{z}}$, which evokes an interpretation of $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}$ as the sequentially-constrained ambiguity residual. When $\check{\epsilon}_{ci}$, the ith element of $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}$, is small, this implies that the sequentially-constrained float estimate $\hat{z}_{ci} = \hat{z}_{i|I}$ is close to $\check{z}_i$, meaning the assumption $z_i = \check{z}_i$ is likely correct. If the assumption is

correct for all $z_k,\ k \in \{1, \ldots, i-1\}$, then $\epsilon_{\mathrm{c}k} = \check{\epsilon}_{\mathrm{c}k}$ for all $k \in \{1, \ldots, i-1\}$, and (2.10) simplifies to

$$\hat{z}_{i|I} = z_i + \epsilon_{\mathrm{c}i} \tag{2.11}$$

The appearance of $\epsilon_{\mathrm{c}i}$ as the sole noise element in this equation indicates that, given the true value of $\boldsymbol{z}_I$, $\hat{z}_{i|I}$ is uncorrelated with $\boldsymbol{\epsilon}_I$, and, by extension, with $\hat{\boldsymbol{z}}_I$. This important property allows the integer constraint $z_i \in \mathbb{Z}$ to be enforced directly on $\hat{z}_{i|I}$ by simple rounding, yielding the sequentially-maximum-likelihood integer estimate

$$\check{z}_i = \lfloor \hat{z}_{i|I} \rceil \tag{2.12}$$

where $\lfloor \cdot \rceil$ denotes nearest integer rounding. The set of successively-obtained integer estimates are stacked to form the vector $\check{\boldsymbol{z}} = [\check{z}_1, \ldots, \check{z}_m]^T$, which is called the fixed ambiguity, as distinguished from the float ambiguity $\hat{\boldsymbol{z}}$. Note that $\check{\boldsymbol{z}}$ is not necessarily the maximum-likelihood (ML) estimate; rather it is a greedy approximation to the ML estimate. Note also that if one or more of the elements in $\boldsymbol{z}_I$ are constrained incorrectly, i.e. the integer error vector $\Delta\boldsymbol{z}_I \triangleq \check{\boldsymbol{z}}_I - \boldsymbol{z}_I$ is nonzero, then (2.10) instead becomes

$$\hat{z}_{i|I} = z_i + \epsilon_{\mathrm{c}i} + \sum_{k=1}^{i-1} l_{ik} \Delta z_k \tag{2.13}$$

where Δz_k is the kth element of $\Delta\boldsymbol{z}_I$.

To summarize, the ith IB iteration starts by assuming $\boldsymbol{z}_I = \check{\boldsymbol{z}}_I$, calculates $\hat{z}_{i|I}$ subject to this constraint as in the center equation in (2.10), then rounds $\hat{z}_{i|I}$ to the nearest integer to obtain $\check{z}_i$. The full IB algorithm becomes clear by mention

of two additional points: (1) $\check{\boldsymbol{z}}_I$ is taken to be composed of the integer-rounded estimates from previous steps, and (2) for $i = 1$, $\hat{z}_{i|I} = \hat{z}_i$.

An efficient implementation of IB is given in pseudocode below. This implementation, which is functionally equivalent to that given in [26] although its internal details differ, is the starting point for the new algorithm developed in this chapter.

Algorithm 2.1: $\mathrm{IB}(\hat{\boldsymbol{z}}, L)$

Input : $\hat{\boldsymbol{z}} \in \mathbb{R}^m$, $L \in \mathbb{R}^{m \times m}$
Output: $\check{\boldsymbol{z}} \in \mathbb{Z}^m$

$\hat{\boldsymbol{z}}_{\mathrm{c}} = \hat{\boldsymbol{z}}$
for *i = 1:m* **do**
  $\check{z}_i = \lfloor \hat{z}_{\mathrm{c}i} \rceil$
  $\check{\epsilon}_{\mathrm{c}i} = \hat{z}_{\mathrm{c}i} - \check{z}_i$
  for *j = i+1:m* **do**
    $\hat{z}_{\mathrm{c}j} = \hat{z}_{\mathrm{c}j} - l_{ij}\check{\epsilon}_{\mathrm{c}i}$
  end
end

Once the fixed ambiguity $\check{\boldsymbol{z}}$ is computed, an integer-constrained baseline estimate, called the fixed baseline, is produced as

$$\begin{aligned} \check{\boldsymbol{b}} &= \hat{\boldsymbol{b}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} Q_{\hat{\boldsymbol{z}}}^{-1} \check{\boldsymbol{\epsilon}} \\ &= \hat{\boldsymbol{b}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} L^{-T} D^{-1} \check{\boldsymbol{\epsilon}}_{\mathrm{c}} \end{aligned} \tag{2.14}$$

The corresponding covariance matrix reflects the improved precision of the baseline estimate due to integer fixing, assuming all ambiguities were fixed correctly:

$$Q_{\check{\boldsymbol{b}}} = Q_{\hat{\boldsymbol{b}}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} Q_{\hat{\boldsymbol{z}}}^{-1} Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}}^{T} \tag{2.15}$$

The distribution of the fully-fixed baseline conditioned on a particular fixed ambiguity $\check{\boldsymbol{z}} = \boldsymbol{z} + \Delta\boldsymbol{z}$ is [32]

$$\left(\check{\boldsymbol{b}} \,|\check{\boldsymbol{z}} = \boldsymbol{z} + \Delta\boldsymbol{z}\right) \sim \mathcal{N}\left(\boldsymbol{b} + Q_{\hat{\boldsymbol{b}}\hat{\boldsymbol{z}}} Q_{\hat{\boldsymbol{z}}}^{-1} \Delta\boldsymbol{z}, Q_{\check{\boldsymbol{b}}}\right) \tag{2.16}$$

Thus, when the integer ambiguity is fixed correctly ($\Delta\boldsymbol{z} = \boldsymbol{0}$), the fully-fixed baseline has a Gaussian distribution whose mean equals the true baseline $\boldsymbol{b}$.

2.2.2 Integer Aperture Bootstrapping (IAB)

IAB extends the IB concept by adding a validation test [33]. The test statistic for IAB can be expressed as a function of the ambiguity residual $\check{\boldsymbol{\epsilon}} \triangleq \hat{\boldsymbol{z}} - \check{\boldsymbol{z}}$ and a parameter $\beta \in [0, 1]$ called the aperture parameter:

$$T(\check{\boldsymbol{\epsilon}}, L, \beta) = \left\| \mathrm{IB}\left(\frac{1}{\beta}\check{\boldsymbol{\epsilon}}, L\right) \right\|_0 \tag{2.17}$$

Here, $\|\boldsymbol{v}\|_0 \triangleq |\{i \mid v_i \neq 0\}|$ denotes the number of non-zero elements in the vector $\boldsymbol{v}$. It can be shown that $T\left(\check{\boldsymbol{\epsilon}}, L, \beta\right) = 0 \iff |\check{\epsilon}_{ci}| < \frac{\beta}{2},\ \forall i \in \{1, \ldots, m\}$ [33]. Thus, a small β ensures that $T\left(\check{\boldsymbol{\epsilon}}, L, \beta\right) = 0$ only when the sequentially-constrained ambiguity residuals are small, implying that $\check{\boldsymbol{z}} = \boldsymbol{z}$ with high probability. Accordingly, IAB accepts the integer fix produced by IB whenever $T = 0$, but otherwise rejects it. In the event that the fix is rejected, IAB resorts to the float solution $\hat{\boldsymbol{z}}$. The overall IAB process can be represented by the map

$$\mathrm{IAB}\left(\hat{\boldsymbol{z}}, L, \beta\right) \triangleq \begin{cases} \mathrm{IB}(\hat{\boldsymbol{z}}, L) & \text{if } T(\check{\boldsymbol{\epsilon}}, L, \beta) = 0 \\ \hat{\boldsymbol{z}} & \text{otherwise} \end{cases}$$

Note that, since $\mathrm{IB}(\check{\boldsymbol{\epsilon}}, L) = \mathbf{0}$, a fixed solution can be forced by choosing $\beta = 1$; likewise, a float solution is forced by $\beta = 0$.

The set of all float ambiguities mapped to the vector $\boldsymbol{\zeta}$, called $\Omega_{\boldsymbol{\zeta}}$, is a subset of the corresponding pull-in region of $\mathrm{IB}(\hat{\boldsymbol{z}}, L)$, with equality if and only if $\beta = 1$:

$$\Omega_{\boldsymbol{\zeta}} = \{\hat{\boldsymbol{z}} \mid \mathrm{IAB}(\hat{\boldsymbol{z}}, L, \beta) = \boldsymbol{\zeta}\} \subseteq S_{\boldsymbol{\zeta}} \tag{2.18}$$

Such sets are called apertures. Due to the integer invariance of IB, $\Omega_{\boldsymbol{\zeta}} = \Omega_{\mathbf{0}} + \boldsymbol{\zeta}$.

The IAB apertures have the same shape as the IB pull-in region but are scaled by a factor of β. Accordingly, gaps between integer-shifted apertures emerge whenever $\beta < 1$, as illustrated in Fig. 2.1. Three important regions can be identified in Fig. 2.1, each corresponding to a possible IAB outcome. The central, dark region corresponds to the success event in which the full ambiguity set is resolved correctly. The union of the many lightly shaded regions corresponds to the failure event in which one or more integer ambiguities are fixed incorrectly. Values of $\boldsymbol{\epsilon} = \hat{\boldsymbol{z}} - \boldsymbol{z}$ falling in the unshaded region result in the fix being rejected. This is the undecided event. The probabilities of these events are [33]

$$P_S = \prod_{i=1}^{m} \left(2\Phi\left(\frac{\beta/2}{\sqrt{d_i}}\right) - 1\right) \tag{2.19a}$$

$$P_F = \sum_{\tilde{\boldsymbol{z}} \in \mathbb{Z}^m \setminus \{0\}} \prod_{i=1}^{m} \left(\Phi\left(\frac{\frac{\beta}{2} - L^i \tilde{\boldsymbol{z}}}{\sqrt{d_i}}\right) - \Phi\left(\frac{-\frac{\beta}{2} - L^i \tilde{\boldsymbol{z}}}{\sqrt{d_i}}\right)\right) \tag{2.19b}$$

$$P_U = 1 - P_F - P_S \tag{2.19c}$$

where L^i is the ith row of L^{-1}, and $\Phi(\cdot)$ is the CDF of the standard normal random variable.

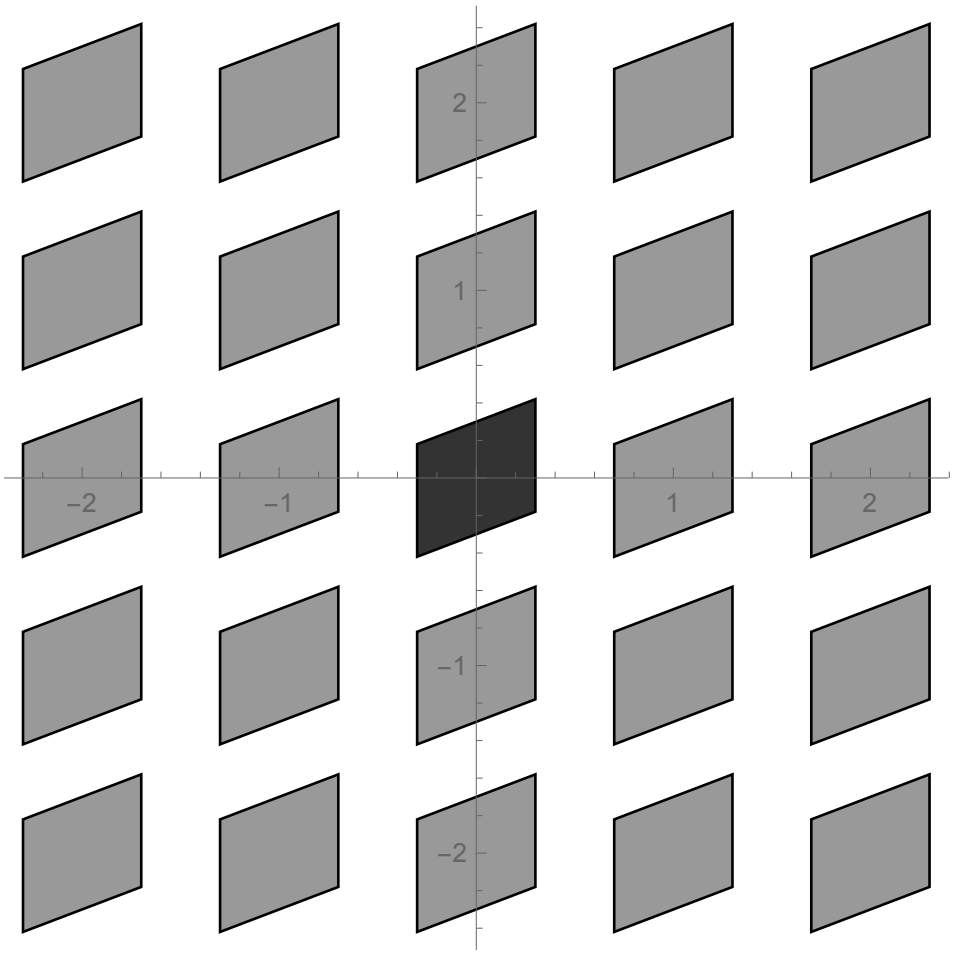

Figure 2.1: IAB aperture regions for a two-dimensional example model. This is a visual representation of the possible outcomes of $\mathrm{IAB}(\hat{\boldsymbol{z}} - \boldsymbol{z}, L, \beta)$. The central, darkly shaded region is the success region, in which $\check{\boldsymbol{z}} = \boldsymbol{z}$. The lightly shaded regions correspond to incorrect ambiguity fixes, in which $\check{\boldsymbol{z}} \neq \boldsymbol{z}$. The unshaded region is the fix-rejection region.

A few observations should be made about the event probabilities. First, calculation of P_F involves an infinite sum over all integer ambiguities other than the correct one. One can calculate an approximate P_F by summing over a large number of alternative ambiguities, but this may still be computationally expensive if the specified acceptable P_F, written $\bar{P}_F$, is small or if m is large. Second, P_F is a monotonically increasing function of β, which implies that P_F decreases as the integer aperture is made smaller. Thus, the aperture parameter β controls the failure probability. Third, P_S is also monotonically increasing in β, which implies that any increase in P_S comes at the expense of an increase in P_F.

2.2.3 Generalization to Partial Ambiguity Resolution

Accepting or rejecting the whole of $\check{\boldsymbol{z}}$, as IAB does, is an extreme approach that limits the range of useful outcomes. Consider instead a variant of IAB in which a subset of the elements of $\check{\boldsymbol{z}}$ may be accepted. IAB is well suited to such generalization from full to partial ambiguity resolution, for two reasons. First, the lack of correlation between the elements of $\check{\boldsymbol{\epsilon}}_c$ allows an aperture test to be applied separately to each element. Moreover, the test can be tailored for each element: the ith ambiguity can be tested against aperture parameter β_i, with the vector $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_m]^T$ chosen such that $P_F \leq \bar{P}_F$. A later section will discuss the benefits of such element-specific aperture sizing. Second, one need not consider every possible subset of IAB ambiguities, which, besides being computationally demanding, would involve so many aperture tests that satisfying $P_F \leq \bar{P}_F$ would require such small β_i values that P_S would become intolerably small. Instead, one can

achieve good performance even when considering only the subset corresponding to the first $q \leq m$ elements of $\hat{\boldsymbol{z}}$, where q is the number that pass the validation test. This is because any of the commonly-accepted Z transform techniques (e.g., those in [25, 32]) tend to arrange $\hat{\boldsymbol{z}}$ to greatly increase (though not necessarily maximize) P_S relative to what would have been possible with the un-transformed system[2]. And since the expected value of q can be shown to increase with P_S, attempted fixing from the first to last element of $\hat{\boldsymbol{z}}$ ensures that q will be maximized, or nearly so.

The new algorithm, generalized integer aperture bootstrapping (GIAB), is given in pseudocode below. GIAB successively fixes ambiguities until it determines that the next one cannot be fixed without P_F exceeding $\bar{P}_F$. The output q is the number of ambiguities fixed; $q < m$ implies the $(q+1)$th validation test failed, so the last $m - q$ ambiguities were left unfixed.

Whereas IAB has three outcome events (success, failure, and undecided), GIAB has $m + 2$. These are defined in terms of the random variables $\check{\boldsymbol{z}}$ and q as follows, where $\boldsymbol{z}_{1:n}$ indicates the vector composed of the first n elements of the

[2]For any of the common LAMBDA methods, the ordering ensures that $\sigma_{i|I} < \sigma_{i+1|I}$. The two variants in [25] further enhance P_S by enforcing the constraints that either $\sigma_{i|I} < \sigma_{j|I}, \forall\, j \geq i$ or $\sigma_{i|I} > \sigma_{j|I}, \forall\, j \leq i$. The first constraint means that the integers are ordered so that the each integer fixed has the lowest conditional variance from among those not yet fixed. The second constraint means that the i^{th} ambiguity fixed has the maximum conditional variance from among the first i ambiguities when conditioned on the other $i - 1$ ambiguities.

Algorithm 2.2: GIAB $(\hat{\boldsymbol{z}}, L, \boldsymbol{\beta})$

Input : $\hat{\boldsymbol{z}} \in \mathbb{R}^m$, $L \in \mathbb{R}^{m\times m}$, $\boldsymbol{\beta} \in [0,1]^m$
Output: $q \in \{0,\ldots,m\}$, $\check{\boldsymbol{z}} \in \mathbb{Z}^{\min(q+1,m)}$

$q = 0$
$\hat{\boldsymbol{z}}_{\mathrm{c}} = \hat{\boldsymbol{z}}$
for *i = 1:m* **do**
  $\check{z}_i = \lfloor \hat{z}_{\mathrm{c}i} \rceil$
  $\check{\epsilon}_{\mathrm{c}i} = \hat{z}_{\mathrm{c}i} - \check{z}_i$
  if $|\check{\epsilon}_{\mathrm{c}i}| < \frac{\beta_i}{2}$ **then**
    $q = i$
    for *j = i+1:m* **do**
      $\hat{z}_{\mathrm{c}j} = \hat{z}_{\mathrm{c}j} - l_{ji}\check{\epsilon}_{\mathrm{c}i}$
    end
  else
    break
  end
end

vector $\boldsymbol{z}$:

$$F : \check{\boldsymbol{z}}_{1:q} \neq \boldsymbol{z}_{1:q},\ q \in \{1,\ldots,m\} \tag{2.20a}$$

$$U : q = 0 \tag{2.20b}$$

$$S_i : \check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i},\ q = i \in \{1,\ldots,m\} \tag{2.20c}$$

The failure event F occurs upon acceptance of any incorrect integers. The undecided event U occurs when no ambiguity is fixed. There are m success events S_i defined for each possible number of correct integer fixes from 1 to m.

The aperture $\Omega_{\boldsymbol{\zeta}} \in \mathbb{R}^m,\ \boldsymbol{\zeta} \in \mathbb{Z}^m$, introduced earlier for IAB, can be generalized for partial ambiguity resolution as $\Omega_{i,\boldsymbol{\zeta}} \in \mathbb{R}^m,\ \boldsymbol{\zeta} \in \mathbb{Z}^i,\ i \in \{1,\ldots,m\}$. Let

$\check{\boldsymbol{z}}$ and q be the outputs of GIAB $(\hat{\boldsymbol{z}}, L, \boldsymbol{\beta})$. Then

$$\Omega_{i,\boldsymbol{\zeta}} = \{\hat{\boldsymbol{z}} \in \mathbb{R}^m \mid \check{\boldsymbol{z}}_{1:i} = \boldsymbol{\zeta}, q = i \in \{1, \ldots, m\}\} \tag{2.21}$$

In other words, $\Omega_{i,\boldsymbol{\zeta}}$ is the set of all float ambiguity vectors whose first i elements are mapped and validated by GIAB to $\boldsymbol{\zeta} \in \mathbb{Z}^i$, but whose $(i+1)$th element is not validated. Note that when $\beta_i = \beta$ for all $i \in \{1, \ldots, m\}$, then $\Omega_{m,\boldsymbol{\zeta}} = \Omega_{\boldsymbol{\zeta}}, \boldsymbol{\zeta} \in \mathbb{Z}^m$.

The success event S_i can be defined in terms of $\Omega_{i,\boldsymbol{\zeta}}$ as

$$S_i : \hat{\boldsymbol{z}} \in \Omega_{i,\boldsymbol{\zeta}},\ \zeta = \boldsymbol{z}_{1:i},\ q = i \in \{1, \ldots, m\} \tag{2.22}$$

and the failure event can be defined as

$$F : \hat{\boldsymbol{z}} \in \left\{ \bigcup_{i \in \{1,\ldots,m\}} \left(\bigcup_{\boldsymbol{\zeta} \in \mathbb{Z}^i \setminus \boldsymbol{z}_{1:i}} \Omega_{i,\boldsymbol{\zeta}} \right) \right\} \tag{2.23}$$

The regions corresponding to the F, U, and S_i events are illustrated in Fig. 2.2 for $m = 2$.

2.2.4 Partial Ambiguity Resolution Probabilities

To assess GIAB's theoretical performance, the probability of each possible event must be computed. For the ith ambiguity reached during GIAB processing, there are three possibilities: the fix is accepted correctly, accepted erroneously, or rejected. Conditioned on the event that the first $i-1$ integers have been fixed correctly (i.e., $\check{\boldsymbol{z}}_I = \boldsymbol{z}_I$), the probabilities for these three events, for $i \in \{1, \ldots, m\}$,

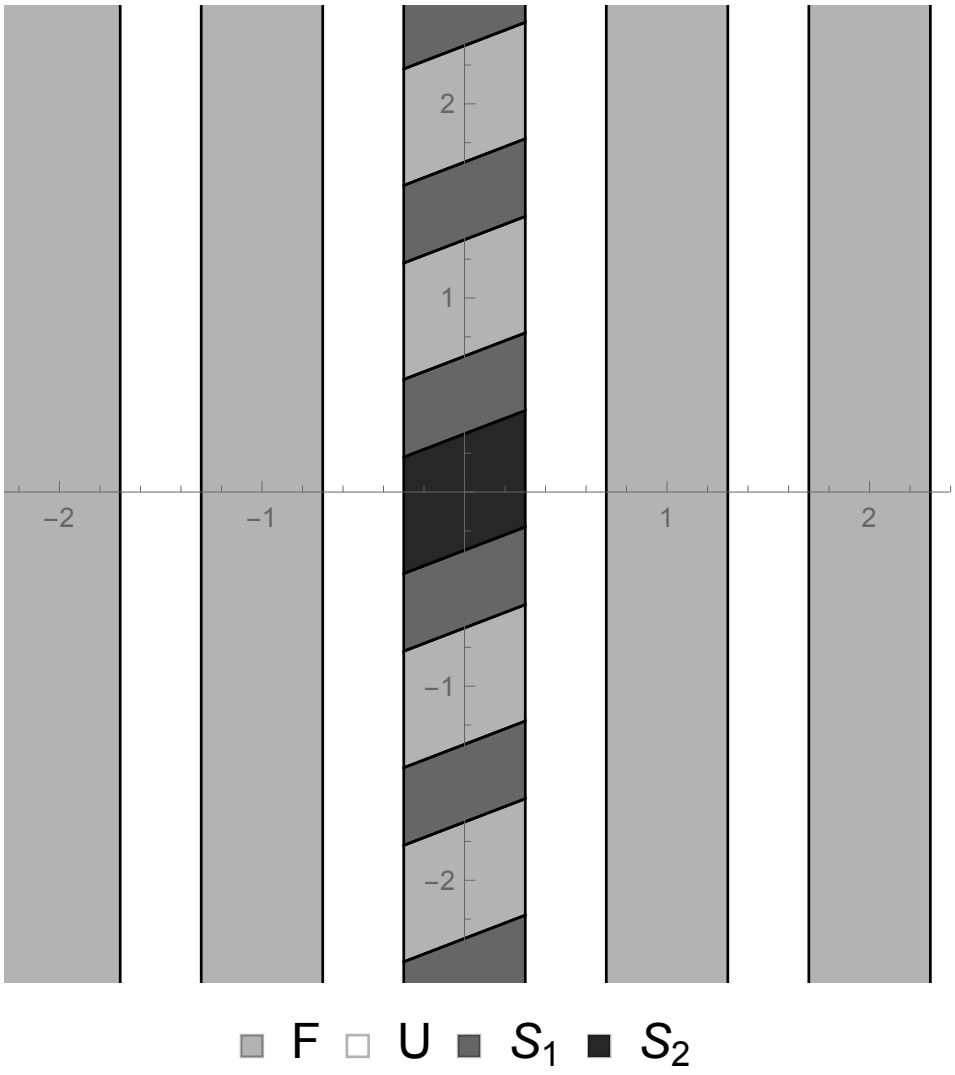

Figure 2.2: Regions of the float ambiguity error, $\boldsymbol{\epsilon} \in \mathbb{R}^m$, that are mapped by GIAB to failure, undecided, and success events for an example model with $m = 2$. Event F results if one or more ambiguities are fixed incorrectly. Event S_i occurs when exactly $i \in \{1, \ldots, m\}$ ambiguities are fixed and each of these is correct. Event U occurs when the first ambiguity is rejected, leaving all ambiguities unfixed.

follow from (2.19):

$$P_{Ci} = P\left(|\epsilon_{ci}| < \frac{\beta_i}{2}\middle|\, \check{\boldsymbol{z}}_I = \boldsymbol{z}_I\right) = 2\Phi\left(\frac{\beta_i/2}{\sqrt{d_i}}\right) - 1 \tag{2.24a}$$

$$\begin{aligned} P_{Ei} &= \sum_{\zeta\in\mathbb{Z}\setminus\{0\}} P\left(|\epsilon_{ci} - \zeta| < \frac{\beta_i}{2}\middle|\, \check{\boldsymbol{z}}_I = \boldsymbol{z}_I\right) \\ &= \sum_{\zeta\in\mathbb{Z}\setminus\{0\}} \left(\Phi\left(\frac{\frac{\beta_i}{2} - \zeta}{\sqrt{d_i}}\right) - \Phi\left(\frac{-\frac{\beta_i}{2} - \zeta}{\sqrt{d_i}}\right)\right) \end{aligned} \tag{2.24b}$$

$$P_{Ri} = P\left(\frac{\beta_i}{2} \le |\epsilon_{ci}|\middle|\, \check{\boldsymbol{z}}_I = \boldsymbol{z}_I\right) = 1 - P_{Ei} - P_{Ci} \tag{2.24c}$$

Note that, for $i = 1$, $\check{\boldsymbol{z}}_I$ and $\boldsymbol{z}_I$ become empty vectors and the conditioning has no effect.

The failure event probability, P_F, is computed by noting that one or more fixing errors entail the failure event and that, if an ambiguity is rejected, no further ambiguities are considered. Thus P_{Ei} only contributes to P_F if all previous ambiguities were fixed correctly. The probability of the ith success event, P_{S_i}, can be computed by applying similar logic. The probability of the undecided event, P_U, is simply P_{R1}. The failure, success, and undecided probabilities are thus

$$P_F = P_{E1} + \sum_{i=2}^{m} P_{Ei} \prod_{j=1}^{i-1} P_{Cj} \tag{2.25a}$$

$$P_{S_i} = \begin{cases} \prod_{j=1}^{m} P_{Cj} & i = m \\ P_{R(i+1)} \prod_{j=1}^{i} P_{Cj} & i \in \{1, \ldots, m-1\} \end{cases} \tag{2.25b}$$

$$P_U = P_{R1} \tag{2.25c}$$

Note that if $\beta_i = \beta$ for all $i \in \{1, \ldots, m\}$, then P_{S_m} is equal to P_S as defined for IAB in (2.19a).

A bound can be introduced to avoid the infinite sum in calculating P_{Ei}. Consider the region $\frac{\beta_i}{2} \leq |\epsilon_{ci}| \leq 1 - \frac{\beta_i}{2}$, which is a subset of the rejection region $\frac{\beta_i}{2} \leq |\epsilon_{ci}|$, so that $P_{Ri} \geq P\left(\frac{\beta_i}{2} \leq |\epsilon_{ci}| \leq 1 - \frac{\beta_i}{2} \middle| \check{\boldsymbol{z}}_I = \boldsymbol{z}_I\right)$. Appending to this region the correct acceptance region $|\epsilon_{ci}| < \frac{\beta_i}{2}$ from (2.24a), and working out the probability for the combined region, it follows that

$$P_{Ci} + P_{Ri} \geq 1 - 2\Phi\left(\frac{\beta_i/2 - 1}{\sqrt{d_i}}\right) \tag{2.26}$$

From this and (2.24c), one obtains the following upper bound on P_{Ei}:

$$P_{Ei} \leq 2\Phi\left(\frac{\beta_i/2 - 1}{\sqrt{d_i}}\right) \tag{2.27}$$

Provided $\bar{P}_F$ is small and the measurement model is strong enough that $\sqrt{d_i} < 0.2$, as is typical, this bound on P_{Ei} is tight. The next section invokes the bound, together with $\bar{P}_F$, to set the aperture parameters β_i.

2.3 Setting the Integer Aperture Parameters

In IAB, P_F is controlled by adjusting a single aperture parameter β. GIAB is more flexible, as it allows a tailored parameter β_i for each validation test. For any specified $\bar{P}_F$, a parameter vector $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_m]^T$ can almost always be found so that GIAB's probability of fixing all m integers,P_{S_m}, exceeds IAB's P_S. This section shows how $\boldsymbol{\beta}$ can be computed analytically to satisfy $P_F \leq \bar{P}_F$, and develops a technique that chooses $\boldsymbol{\beta}$ to nearly maximize P_{S_m}.

2.3.1 Allocation from Probability of Failure

Each validation test that GIAB performs contributes to P_F. The parameter β_i determines the amount of incorrect fixing risk that gets allocated to ith ambiguity, from an overall risk budget $\bar{P}_F$. (The word *risk* here and elsewhere in this chapter refers to the probability of an undesirable event.) Suppose $w_i\bar{P}_F$ is allocated to the ith ambiguity, where $w_i < 1$, and suppose the aperture parameters preceding β_i have all been set, which implies that P_{Cj} is known for all $j \in \{1, \ldots, i-1\}$. Then the maximum allowable β_i—the one that maximizes P_{S_i} subject to the allocation $w_i\bar{P}_F$—is found in two steps. First, P_{Ei} is written as a function of $w_i\bar{P}_F$ by isolating its contribution to P_F in (2.25a):

$$P_{Ei}(w_i\bar{P}_F) = \begin{cases} w_1\bar{P}_F & i = 1 \\ \frac{w_i\bar{P}_F}{\prod_{j=1}^{i-1} P_{Cj}} & i = 2, \ldots, m \end{cases} \tag{2.28}$$

Second, the correponding value of β_i is found by treating (2.27) as an equality and inverting it to find β_i. Applying the constraint $\beta_i \in [0, 1]$, one has $\beta_i = \beta_{\max}(P_{Ei}, d_i)$, with

$$\beta_{\max}(P_{Ei}, d_i) \triangleq \min\left[1, \max\left[0, 2\left(1 + \sqrt{d_i}\Phi^{-1}(P_{Ei}/2)\right)\right]\right] \tag{2.29}$$

Note that if any $\beta_i = 0$, then the ith and following ambiguities cannot be fixed while satisfying $P_F \leq \bar{P}_F$. Conversely, if $\beta_i = 1$ for all $i \in \{1, \ldots, m\}$, then all m ambiguities can be fixed while satisfying $P_F \leq \bar{P}_F$.

The functions $P_{Ei}(w_i\bar{P}_F)$ and $\beta_{\max}(P_{Ei}, d_i)$, which are constructed from well-known and readily-computable operations, constitute an analytical mapping

from $w_i \bar{P}_F$ to β_i. This analytical relationship is a key benefit of GIAB, as it allows data-driven partial ambiguity resolution to be applied in safety-of-life systems that must provably satisfy $P_F \leq \bar{P}_F$.

2.3.2 Optimization for Availability of Full Ambiguity Resolution

Consider how the w_i should be chosen. Assuming a nonzero risk is allocated to each ambiguity, and assuming the full risk budget $\bar{P}_F$ is to be exhausted, the w_i should satisfy

$$0 < w_i < 1, \forall i \in \{1, \ldots, m\} \quad \text{and} \quad \sum_{i=1}^{m} w_i = 1 \tag{2.30}$$

One could allocate an equal fraction of $\bar{P}_F$ to each of the m ambiguities by setting $w_i = 1/m$ for all $i \in \{1, \ldots, m\}$, but this may not be optimal in the sense of maximizing the probability P_{Sm} of correctly resolving all m ambiguities. The optimal allocation problem can be posed in terms of $\boldsymbol{\beta}$ as

$$\begin{aligned} \boldsymbol{\beta}^* &= \arg\max_{\boldsymbol{\beta}} \left[P_{Sm}\left(\boldsymbol{\beta}\right) \right] \\ \text{s.t.} \;\; & P_F\left(\boldsymbol{\beta}\right) \leq \bar{P}_F \text{ and conditions in (2.30)} \end{aligned} \tag{2.31}$$

This problem can be approached by gradient ascent, but P_{Sm} and P_F are both non-convex functions of $\boldsymbol{\beta}$ and give rise to many local maxima in the region of the global maximum. Thus, gradient ascent offers no guarantee of finding the global optimum, besides which the gradient calculation for this problem is computationally expensive.

Mercifully, a nearly-optimal choice of the w_i can be found by a simple heuristic. Because both P_{Sm} and P_F are functions of the conditional variances

d_i, it is reasonable to compute the weights as functions of d_i as well. The most general function satisfying (2.30) is

$$w_i = \frac{f(d_i)}{\sum_{j=1}^{m} f(d_j)} \tag{2.32}$$

where $f(d_i)$ is a weighting function. Guided by the intuition that more risk must be allocated to the ambiguities that are most difficult to resolve (those having the largest d_i), lest their resulting small β_i reject fixing too often, four variants of $f(d_i)$, shown in the following table, are considered: Equal-weighting, σ-weighting, σ^2-weighting, and P_E-weighting. Note that P_E-weighting simply sets $f(d_i)$ equal to P_{Ei} from (2.27) with $\beta_i = 1$.

Table 2.1: Weighting Function Alternatives Considered

	Equal	σ	σ^2	P_E
$f(d_i)$	1	$\sqrt{d_i}$	d_i	$2\Phi\left(\frac{-1/2}{\sqrt{d_i}}\right)$

When tested on a variety of models with bootstrap probability of correct fix ranging from .85 to .9999 and for a wide range of $\bar{P}_F$, it was found that P_E-weighting produces the highest P_{S_m} for all models studied, including cases of flat spectra (e.g., $\max\{d_i\}_1^m / \min\{d_i\}_1^m < 1.1$), and spectra with significant variation (e.g., $\max\{d_i\}_1^m / \min\{d_i\}_1^m > 7$). When performing gradient ascent optimization starting from the P_E-weighted $\boldsymbol{\beta}$, or starting from a large number of random initial $\boldsymbol{\beta}$ distributed across its whole range, there was never observed more than a 0.03% increase in P_{S_m}. Moreover, compared to the common-parameter case in which $\beta_i =$

β for all $i \in \{1, \ldots, m\}$, the probability P_{S_m} for P_E-weighting was never lower, and almost always higher—often by several percent. P_E-weighting can thus be considered nearly optimal, and is the recommended strategy for aperture sizing. The overall aperture sizing algorithm is given in the following pseudocode. Note that even when the algorithm's output $\boldsymbol{\beta}$ does not quite maximize P_{Sm}, it nevertheless guarantees $P_F \leq \bar{P}_F$, which is most important for safety-of-life systems.

Algorithm 2.3: $\mathrm{SetBeta}\left(\bar{P}_F, \boldsymbol{d}\right)$

Input : $\bar{P}_F \in [0,1]$, $\boldsymbol{d} \in \mathbb{R}^m$
Output: $\boldsymbol{\beta} \in [0,1]^m$

$\Sigma = 0$;
$A = 1$;
for *i = 1:m* **do**
  $P_{Ei} = 2\Phi\left(-\frac{1/2}{\sqrt{d_i}}\right)$
  $\Sigma = \Sigma + P_{Ei}$
end
for *i = 1:m* **do**
  $w_i = \frac{P_{Ei}}{\Sigma}$
  $\beta_i = \min\left(\max\left[2\left(1 + \sqrt{d_i}\Phi^{-1}\left(\frac{w_i \bar{P}_F}{2A}\right)\right), 0\right], 1\right)$
  $A = \left(2\Phi\left(\frac{\beta_i/2}{\sqrt{d_i}}\right) - 1\right) A$
end

2.4 The GIAB Baseline

Analogous to the float baseline $\hat{\boldsymbol{b}}$ and the fixed baseline $\check{\boldsymbol{b}}$, a partially-fixed baseline can be calculated from the inputs and outputs of GIAB. Let (2.14) be

rewritten as

$$\check{\boldsymbol{b}} = \hat{\boldsymbol{b}} - \sum_{j=1}^{m} Q^{j}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \frac{\check{\epsilon}_{\mathrm{c}j}}{d_j} \tag{2.33}$$

where $Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \triangleq Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} L^{-T}$ and where $Q^{j}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}}$ denotes the jth column of $Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}}$. Rewriting $\check{\boldsymbol{b}}$ in this way reveals that each element of the sequentially-constrained ambiguity residual $\check{\boldsymbol{\epsilon}}_c$ makes a separate correction to $\hat{\boldsymbol{b}}$ in the direction defined by the vector $Q^{j}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}}$. To obtain a partially-fixed baseline, one simply truncates the summation. Thus, the baseline constrained only by the first i ambiguities, denoted $\check{\boldsymbol{b}}_i$, is

$$\check{\boldsymbol{b}}_i = \hat{\boldsymbol{b}} - \sum_{j=1}^{i} Q^{j}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \frac{\check{\epsilon}_{\mathrm{c}j}}{d_j} \tag{2.34}$$

Its covariance $Q_{\check{\boldsymbol{b}}_i}$, assuming all fixed ambiguities are correctly fixed, can be derived from (2.15):

$$Q_{\check{\boldsymbol{b}}_i} = Q_{\hat{\boldsymbol{b}}} - \sum_{j=1}^{i} \frac{1}{d_j} Q^{j}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \left(Q^{j}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \right)^{T} \tag{2.35}$$

For high-integrity positioning, the probability distribution of the baseline vector is of great importance. It can be shown that the float baseline $\hat{\boldsymbol{b}} \sim \mathcal{N}(0, Q_{\hat{\boldsymbol{b}}})$. On the other hand, the fixed IB baseline from (2.14) is distributed as an infinite sum of Gaussians, though, like $\hat{\boldsymbol{b}}$, it is unbiased [26].

Analysis of the baseline resulting from GIAB is complicated by the effects of data-driven partial fixing. If, for some reason, one decides *a priori* to fix only i ambiguities (e.g., based on the strength of the model), then, given that all fixed ambiguities are fixed correctly, $\check{\boldsymbol{b}}_i$ has a simple distribution:

$$\check{\boldsymbol{b}}_i \big| \left(\check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i} \right) \sim \mathcal{N}(0, Q_{\check{\boldsymbol{b}}_i}),\ i \in \{1, \ldots, m\}$$

One might expect the same distribution to apply for GIAB when $q = i < m$. However, there is key difference between these two cases: $q = i < m$ implies that GIAB has rejected fixing the $(i+1)$th ambiguity. The data-driven (*a posteriori*) decision to reject yields a different baseline distribution than that of *a priori* partial fixing:

$$\check{\boldsymbol{b}}_i | \left(\check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i}, q = i < m \right) \nsim \mathcal{N}(0, Q_{\check{\boldsymbol{b}}_i})$$

To understand why, recall that $q = i < m$ implies GIAB rejected fixing the $(i+1)$th ambiguity upon finding that $|\check{\epsilon}_{\mathrm{c}(i+1)}| \geq \beta_{(i+1)}/2$, as fixing it would violate $P_F < \bar{P}_F$. Even so, the most likely fix for the $(i+1)$th ambiguity, given $\check{\epsilon}_{\mathrm{c}(i+1)}$ and given that $\check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i}$, is the same one that would have been produced by IB, which GIAB outputs in $\check{z}_{i+1}$. The next most likely fix, which is the next nearest integer, and its associated conditional ambiguity residual are

$$\check{z}_{i+1,\mathrm{alt}} = \check{z}_{i+1} + \mathrm{sgn}\left(\check{\epsilon}_{\mathrm{c(i+1)}}\right) \tag{2.36a}$$

$$\check{\epsilon}_{\mathrm{c}(i+1),\mathrm{alt}} = \check{\epsilon}_{\mathrm{c}(i+1)} - \mathrm{sgn}\left(\check{\epsilon}_{\mathrm{c(i+1)}}\right) \tag{2.36b}$$

Equation (2.34) indicates that if the $(i+1)$th integer were to be fixed, the adjustments to $\check{\boldsymbol{b}}_i$ in the most likely and alternate cases would be

$$\check{\boldsymbol{b}}_{i+1} - \check{\boldsymbol{b}}_i = -Q^{(i+1)}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \frac{\check{\epsilon}_{\mathrm{c}(i+1)}}{d_{i+1}} \tag{2.37a}$$

$$\check{\boldsymbol{b}}_{i+1,\mathrm{alt}} - \check{\boldsymbol{b}}_i = -Q^{(i+1)}_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \frac{\check{\epsilon}_{\mathrm{c}(i+1),\mathrm{alt}}}{d_{i+1}} \tag{2.37b}$$

It is shown in Section 3.4.6 that either $\check{\boldsymbol{b}}_{i+1}$ or $\check{\boldsymbol{b}}_{i+1,\mathrm{alt}}$ is unbiased with probability near 1. It follows that, having examined $\check{\epsilon}_{\mathrm{c}(i+1)}$, but having rejected the

correction it offers, and the correction $\check{\epsilon}_{\mathrm{c}(i+1),\mathrm{alt}}$ offers, GIAB will, with probability near 1, produce a biased $\check{\boldsymbol{b}}_i$. Therefore, for data-driven partial fixing, it is wrong—*and potentially hazardous*—to assume the resulting constrained baseline estimate is unbiased. For a complete discussion on the integrity implications of partial ambiguity resolution, and for development of the *a priori* and *a posteriori* partially-fixed baseline distributions, see Ch. 3.

2.5 Validation via Monte Carlo Simulation

To validate the GIAB event probabilities P_F, P_U, and $P_{S_i}, \forall i \in \{1, \ldots, m\}$, extensive Monte Carlo simulations were performed on float solution models with varying measurement error but the same satellite geometry. For each model, the simulation was initialized by computing the decorrelating Z-transform and using P_E-weighting to set the integer aperture parameters. Then a large sample was drawn from the distribution described by (2.2) to generate the float solution errors, the float baseline, and float ambiguities. The float ambiguities were then Z-transformed and the GIAB algorithm was applied to the transformed float ambiguity solution. Finally, the outputs were logged, including the number of correctly fixed samples, tabulated by q, the number of incorrectly fixed samples, tabulated by the first errant ambiguity, and the partially-fixed baseline error, tabulated by q.

The sample size for each simulation was chosen to ensure that a statistically significant number of failures occurred or a significant number of solutions was available for each value of q. This chapter's theoretical event probabilities were then compared to the simulated results. To examine the goodness of fit between theory

and simulation, the differences between predicted and simulated probabilities were calculated, normalized by the expected standard deviation in the measured rate.

Several models were simulated to illustrate a range of failure rates and fixing probabilities. Only small models with 7 ambiguities are presented in full detail, but similar results were obtained for $m \in \{14, 21, 28\}$. In the following tables, $\mathcal{E}$ is an event, whether F, U, or S_i for $i \in \{1, \ldots, m\}$, $P_{\mathcal{E}}$ is the predicted event probability, $\hat{P}_{\mathcal{E}}$ is the event probability as measured from the Monte Carlo simulation, and $k_{P_{\mathcal{E}}}$ is the normalized difference between the predicted and estimated event probabilities. The predicted probability of failure was computed using the bound on P_{Ei} given in (2.27). The difference is normalized by the standard deviation of the Beta distribution, $\beta\left(n_{\mathrm{MC}} P_{\mathcal{E}}, n_{\mathrm{MC}}\left(1 - P_{\mathcal{E}}\right)\right)$, which is the posterior distribution of $P_{\mathcal{E}}$ given the Monte Carlo results. Thus,

$$k_{P_{\mathcal{E}}} = \frac{\hat{P}_{\mathcal{E}} - P_{\mathcal{E}}}{\sqrt{\frac{P_{\mathcal{E}}(1-P_{\mathcal{E}})}{n_{\mathrm{MC}}}}} \tag{2.38}$$

The value of $k_{P_{\mathcal{E}}}$ is interpreted as follows: if $|k_{P_{\mathcal{E}}}| < N$, then the predicted and measured probabilities differ by no more than N standard deviations.

Table 2.2 shows the simulation results for $n_{\mathrm{MC}} = 4 \times 10^8$ Monte Carlo samples from a float distribution with a bootstrap probability of correct fix $P_{CF,\mathrm{B}} = 1 - 2 \times 10^{-5}$ for $\bar{P}_F = 10^{-8}$. This strong model was chosen to validate the event probabilities when partial fixing is rarely needed. Table 2.3 shows the simulation results for $n_{\mathrm{MC}} = 2.2 \times 10^7$ Monte Carlo samples from a float distribution with a bootstrap probability of correct fix $P_{CF,\mathrm{B}} = 0.988$ and $\bar{P}_F = 10^{-5}$. This weak model was chosen to validate the event probabilities when partial fixing must be

Table 2.2: Predicted vs Simulated Event Probabilities for a Strong Model

$\mathcal{E}$	$P_{\mathcal{E}}$	$\hat{P}_{\mathcal{E}}$	$k_{P_{\mathcal{E}}}$
F	10^{-8}	5e-9	1.0000
U	0.0007217	0.0007204	1.0098
S_1	0.0004399	0.0004396	0.2180
S_2	0.0005133	0.0005131	0.1440
S_3	0.0011997	0.0012005	-0.4574
S_4	0.0009000	0.0009023	-1.4966
S_5	0.0018919	0.0018902	0.7913
S_6	0.0003955	0.0003956	0.0896
S_7	0.9939380	0.9939380	-0.0864

employed frequently.

Table 2.3: Predicted vs Simulated Event Probabilities for a Weak Model

$\mathcal{E}$	$P_{\mathcal{E}}$	$\hat{P}_{\mathcal{E}}$	$k_{P_{\mathcal{E}}}$
F	0.00001	0.0000101	-0.2673
U	0.13872	0.138628	1.1858
S_1	0.09308	0.093210	-2.2452
S_2	0.08423	0.084117	1.9596
S_3	0.09987	0.099910	-0.8364
S_4	0.07162	0.071578	0.6782
S_5	0.08010	0.080070	0.2955
S_6	0.03362	0.033647	-0.7922
S_7	0.39878	0.398801	-0.2348

As can be seen in Tables 2.2 and 2.3, both the strong and weak model predictions match the simulation results well.

2.6 Comparison against Existing IA Methods

To demonstrate the improved performance of the GIAB aperture sizing algorithm, GIAB was compared with existing IA methods within Monte Carlo simulations for $m \in \{2, 7\}$. For the $m = 7$ simulation, a single representative satellite

geometry was used with the measurement covariance scaled in the same way as described in section 2.5 to give a weak model that would test GIAB in the least favorable circumstance for comparison with the optimal IA method. IAB and GIAB compare similarly to the $m = 2$ case, so results are not tabulated for IAB for compactness. Table 2.4 shows the results for GIAB. Note the greatest benefit of GIAB is in partial ambiguity resolution: whereas optimal IA correctly resolves the full set of ambiguities less than 84.2% of the time, GIAB correctly fixes some ambiguities almost 95% of the time, and more than half the ambiguities over 82% of time.

For the $m = 2$ simulation, the same set of 10^6 float ambiguity samples was processed using the optimal IA method, the ellipsoidal IA method, IAB, and GIAB. The results are visualized by a scatter plot of the float ambiguities in Fig. 2.3. Each point is shaded according the results of the optimal IA method: dark gray points were correctly fixed, light gray points were left floating, and large, red points were fixed incorrectly.

The apertures of the ellipsoidal, IAB, GIAB, and optimal methods are plotted over the scatter plot to illustrate the comparative probability of successfully fixing all integers for $\bar{P}_F = 10^{-5}$. The threshold for the optimal method was set using a larger Monte Carlo simulation of 10^7 samples such that exactly $n_{MC} \times \bar{P}_F - 1 = 99$ failures occur. This threshold was then used to determine the outcome of the optimal IA method for the smaller simulations. For visualization, the optimal aperture region was approximated by solving for its location along a polar grid with spacings of 0.1°.

It is visually apparent that both IAB, which applies a single threshold, and

Table 2.4: Joint Probability Mass Function of GIAB (Rows) and Optimal IA (Columns) Fixing Decision for Weak $m = 7$ Model

	S_{opt}	U_{opt}	F_{opt}	Marginal
$S_{7,\text{GIAB}}$	0.6992	0.0064	0	0.7056
$S_{6,\text{GIAB}}$	0.0009	0.0231	0	0.0239
$S_{5,\text{GIAB}}$	0.0279	0.0253	5E-6	0.0532
$S_{4,\text{GIAB}}$	0.0051	0.0360	7.5E-6	0.0411
$S_{3,\text{GIAB}}$	0.0170	0.0340	0	0.0510
$S_{2,\text{GIAB}}$	0.0227	0.0144	0	0.0370
$S_{1,\text{GIAB}}$	0.0286	0.0089	0	0.0374
U_{GIAB}	0.0403	0.0103	5.8E-5	0.0506
F_{GIAB}	0	5E-5	2.8E-5	7.8E-5
Marginal	0.8415	0.1584	9.8E-5	1

GIAB, which applies two different risk-allocated thresholds, are superior to ellipsoidal IA for the model considered. There is also a visually perceptible improvement from IAB to GIAB; the exact improvement is quantified in Tables 2.5 and 2.6. The similarity between the GIAB and optimal apertures is clear. It is not visible, but is important to note that the GIAB aperture is slightly wider than the optimal aperture in the region of highest density. Since the optimal threshold must be set by Monte Carlo simulation, it is possible that it will perform worse than GIAB in practice though it is optimal in theory. This is the case for the results shown in Table 2.6.

Table 2.5 compares the baseline performance of IAB with a single aperture threshold to that of the optimal IA estimator. This example was for a relatively lax incorrect fix risk of $\bar{P}_F = 10^{-5}$, so the performance is quite similar: the percent of samples where IAB rejects a fix that the optimal method correctly accepts is only 0.26%. Compare these results to Table 2.6, in which the percent of samples where GIAB rejects a fix that the optimal method correctly accepts is significantly lower, 0.0074%. Moreover, GIAB correctly accepts more fixes that the optimal method rejects. The main advantage of GIAB over optimal IA is that GIAB allows partial ambiguity resolution. GIAB correctly partially fixes 4.1% of all samples, all of which are rejected by the optimal method.

2.7 Conclusions

A new data-driven CDGNSS partial ambiguity resolution and validation algorithm has been developed analytically and validated with Monte Carlo simula-

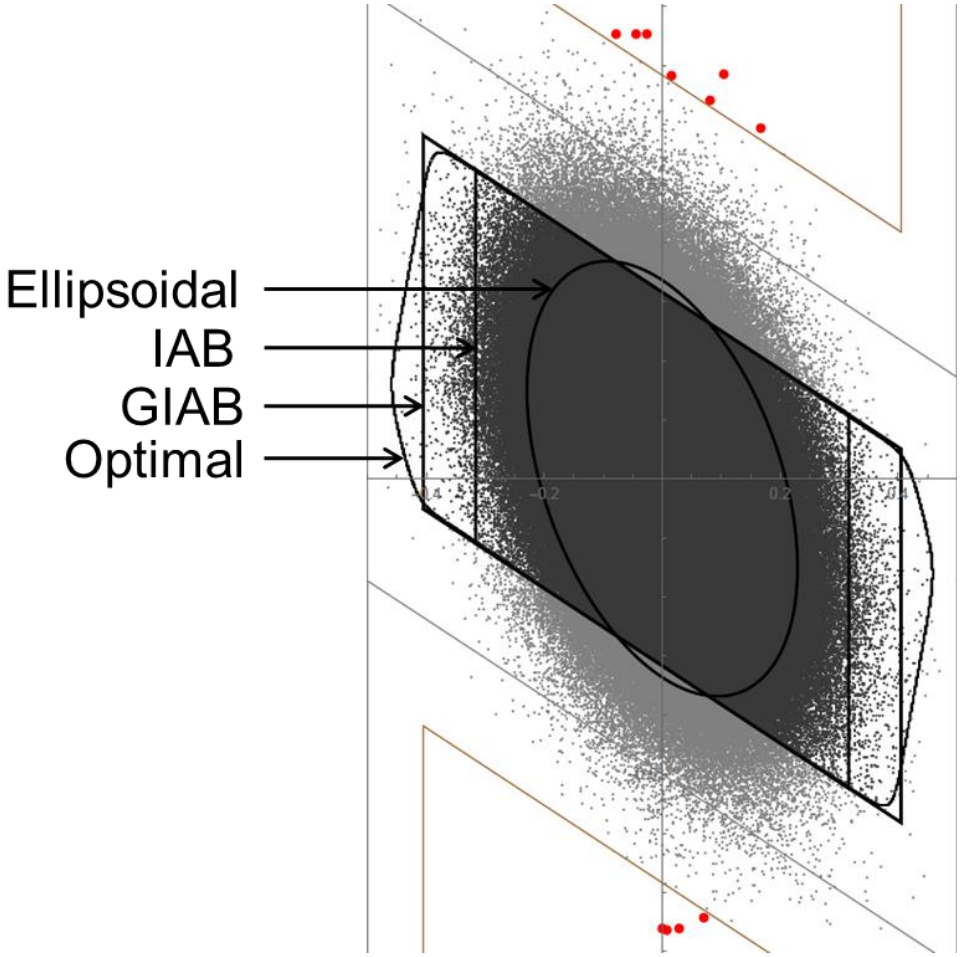

Figure 2.3: Comparison of the integer aperture acceptance regions for integer aperture bootstrapping (IAB), ellipsoidal IA, GIAB, and optimal IA. All apertures allow the same expected number of incorrect fixes, but yield different rates of accepting the correct fix. Listed in ascending order of success are the ellipsoidal, IAB, GIAB, and optimal IA acceptance regions. There is a significant improvement from IAB to GIAB as many more correct fixes are admitted. The optimal IA decisions only differ from the GIAB decisions in a small fraction of cases. The scatter plot are color coded by optimal IA event: dark gray for success, light gray for undecided, and large red for failure.

Table 2.5: Joint Probability Mass Function of IAB (Rows) and Optimal IA (Columns) Fixing Decision for $m = 2$ Model

	S_{opt}	U_{opt}	F_{opt}	Marginal
S_{IAB}	0.954108	0.001876	0	0.955984
U_{IAB}	0.002623	0.041382	0	0.044005
F_{IAB}	0	0	0.000011	0.000011
Marginal	0.956731	0.043258	0.000011	1

Table 2.6: Joint Probability Mass Function of GIAB (Rows) and Optimal IA (Columns) Fixing Decision for $m = 2$ Model

	S_{opt}	U_{opt}	F_{opt}	Marginal
S_2	0.956657	0.001791	0	0.958448
S_1	0	0.041416	0	0.041416
U_{GIAB}	0.000074	0.000051	0	0.000125
F_{GIAB}	0	0	0.000011	0.000011
Marginal	0.956731	0.043258	0.000011	1

tion. The new algorithm has advantages over the state-of-the-art in that (1) data-driven methods offer improved availability of integrity over model-driven methods, (2) the integrity risk due to incorrect fixing is precisely controlled analytically as compared to functional approximation methods used with the ratio test and similar integer aperture methods, and (3) it provides superior probability of success when compared to IAB or ellispoidal IA and approaches that of optimal IA. In simulation testing, the new algorithm was shown to provide superior performance to the current state-of-the-art methods for a range of measurement models. GIAB's partial fixing, together with its analytical connection between the allowable failure rate and its validation thresholds, make GIAB attractive for safety-of-life systems in challenging environments.

Chapter 3

Data-Driven Position Domain Integrity for Generalized Integer Aperture Bootstrapping

3.1 Introduction

[1] State-of-the-art methods in high-integrity CDGNSS enforce IR constraints in the position domain by accounting for baseline biases induced by incorrect integer fixing. Two such methods are the Geometry Extra Redundant Almost Fixed Solutions (GERAFS) [42] and the Enforced Position domain Integrity-risk of Cycle resolution (EPIC) [17, 18] algorithms. Both of these rely exclusively on *a priori* error models to determine, before the measurements are processed, whether a fixed solution or a float backup solution will be selected. This approach is termed model-driven because the solution selection logic is entirely dependent on the prior error model. Because GERAFS and EPIC attempt to bound IR using the *a priori* distribution, they are inherently conservative. Their conservatism arises from the need to protect against position domain biases induced by a large number of potentially-incorrect fixes without the benefit of conditioning on the observed carrier-phase

[1] This chapter was based on

1. G. Nathan Green and Todd E. Humphreys. Position-domain integrity analysis for generalized integer aperture bootstrapping. *IEEE Transactions on Aerospace and Electronic Systems*, 2017. Submitted for review.

The author contributed all derivation, simulation, and analysis in that paper and in this chapter.

measurements.

In contrast to the model-driven approach, data-driven methods exploit measured data to decide whether to accept the fixed or float solution. Conditioning selection on the observed measurements reduces the risk of incorrect ambiguity resolution. Foremost among data-driven techniques is the integer aperture (IA) approach [28]. In this approach, the integer ambiguity vector is first estimated by some means, e.g., integer bootstrapping (IB) [26] or integer least squares (ILS) [34], after which a test statistic is computed from the ambiguity residual, i.e., the difference between the float and fixed ambiguities. Based on this statistic, a hypothesis test decides whether to accept or reject the fixed solution.

IA bootstrapping (IAB) is a particularly simple type of IA estimation in which the integer ambiguities are fixed via IB and the test statistic is produced by a second application of IB, this time to a scaled-up version of the ambiguity residual [33]. If the statistic is the zero vector, the fixed solution is selected; otherwise the float solution is selected. IAB is sub-optimal in two respects: First, IB does not always find the maximum likelihood integer ambiguity, as opposed to ILS, which is guaranteed to do so. Second, IAB fails to maximize the probability of successfully fixing the ambiguities for a given probability of incorrectly fixing them. Although sub-optimal, IB enjoys a significant advantage: its fixing probabilities are analytically calculable, which allows the residual scaling parameter to be set analytically as a function of a desired probability of incorrect fix, or failure rate, $\bar{P}_F$. Crucially, this property enables a system to provably satisfy the strict performance requirements of safety-of-life applications.

It was noted in Section 2.4 that GIAB's partially-integer-constrained baseline estimate exhibits non-negligible biases, even when all fixed integers pass validation. But no prior work has characterized these biases or assed their effect on integrity risk.

This chapter makes four novel contributions to the literature: First, it shows that baseline estimate biases are present in any data-driven partial ambiguity resolution algorithm that corrects the float baseline with the validated fixes. Second, it develops and validates an analytical characterization of the *a priori* and *a posteriori* distributions of the GIAB baseline. Third, it extends the position domain integrity concepts originally developed for EPIC to data-driven algorithms for use in safety-of-life applications. Fourth, it validates GIAB's performance via Monte Carlo simulation and compares this with EPIC.

3.2 Generalized Integer Aperture Bootstrapping Overview

The most important elements of Ch. 2 are collected in this overview for ease of reference.

Beginning from the LAMBDA-decorrelated float solution, $\hat{\boldsymbol{b}}$ and $\hat{\boldsymbol{z}}$ (2.2), GIAB's objective is to fix and validate as many of the ambiguities as possible while ensuring that the probability that a validated ambiguity is incorrect is less than a specified level, $\bar{P}_F$. GIAB outputs the number of validated ambiguities, $q \in \{0, \ldots, m\}$, and the vector, $\check{\boldsymbol{z}} \in \mathbb{Z}^{\min(q+1,m)}$, whose first q elements are the fixed and validated ambiguities, and whose $(q+1)$th element, if $q < m$, is the first fixed but rejected ambiguity.

The outputs of GIAB can be mapped to various events defined in terms of the random variables $\check{\boldsymbol{z}}$ and q. In the following event definitions, $\boldsymbol{z}_{1:n}$ indicates the vector composed of the first n elements of the vector $\boldsymbol{z}$:

$$F : \check{\boldsymbol{z}}_{1:q} \neq \boldsymbol{z}_{1:q},\ q \in \{1, \ldots, m\} \tag{3.1a}$$

$$U : q = 0 \tag{3.1b}$$

$$S_i : \check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i},\ q = i \in \{1, \ldots, m\} \tag{3.1c}$$

$$Z_i : \check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i},\ q = i \in \{0, \ldots, m\} \tag{3.1d}$$

$$C_i : \check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i},\ i \in \{1, ..., m\} \tag{3.1e}$$

$$R_{i+1} : q = i < m \tag{3.1f}$$

The failure event F occurs upon validation of any incorrect integers. The undecided event U occurs when no ambiguity is fixed. There are m success events S_i defined for each possible number of correctly validated integer fixes from 1 to m. The event Z_i is identical to S_i except that it includes the $q = 0$ (no fixes) case. Note that the null vector $\check{\boldsymbol{z}}_{1:0}$ is assumed to be identical to $\boldsymbol{z}_{1:0}$ so that $Z_0 = U$. The correct validation event C_i occurs when the first $i \geq 1$ integers are fixed correctly, irrespective of the value of q. The rejection event R_{i+1} occurs when GIAB refuses to fix the $(i + 1)$th ambiguity.

GIAB requires that the variance of the float ambiguity be decomposed into LDL^T form such that

$$Q_{\hat{\boldsymbol{z}}} = LDL^T \tag{3.2}$$

where L is a unit-lower triangular matrix and D is a diagonal matrix. The float ambiguity can be modeled as the true ambiguity plus zero-mean Gaussian noise,

$\hat{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(0, Q_{\hat{\boldsymbol{z}}})$. Multiplication by L^{-1} transforms $\boldsymbol{\epsilon}$ into a vector whose elements are mutually uncorrelated: $\boldsymbol{\epsilon}_{\mathrm{c}} \triangleq L^{-1}\boldsymbol{\epsilon},\ \boldsymbol{\epsilon}_{\mathrm{c}} \sim \mathcal{N}(0, D)$.The quantity $\boldsymbol{\epsilon}_{\mathrm{c}}$, called the decorrelated float ambiguity error, plays a key role in the analysis that follows.

GIAB takes as input a vector, $\boldsymbol{\beta}$, called the aperture parameter vector, that determines the validation threshold for each ambiguity. $\boldsymbol{\beta}$ also determines the probabilities of the failure, success, and undecided events, $P_F,\ P_{S_i},$ and P_U. $\boldsymbol{\beta}$ is set as a function of D that ensures $P_F < \bar{P}_F$. GIAB can be represented as the function

$$[q, \check{\boldsymbol{z}}] = \mathrm{GIAB}\left(\hat{\boldsymbol{z}}, L, \boldsymbol{\beta}\right) \tag{3.3}$$

The event probabilities, P_F, P_{S_i}, and P_U, are defined in (2.24) and (2.25). These probabilities are determined by $\boldsymbol{\beta}$, which is set as a function of D (Algorithm 2.3) that ensures $P_F < \bar{P}_F$.

Let $r = \min\{q+1, m\}$ for notational simplicity. The ambiguity residual is defined as $\check{\boldsymbol{\epsilon}} \triangleq \hat{\boldsymbol{z}}_{1:r} - \check{\boldsymbol{z}}$. Note that if $\check{\boldsymbol{z}} = \boldsymbol{z}_{1:r}$, then $\check{\boldsymbol{\epsilon}} = \boldsymbol{\epsilon}$. Denote the upper $r \times r$ sub-matrix of L as $L_{1:r,1:r}$. An important quantity, called the sequentially-constrained ambiguity residual, is defined as $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \triangleq L^{-1}_{1:r,1:r}\check{\boldsymbol{\epsilon}}$. This vector has a convenient property: if the first i integer ambiguities GIAB fixes are correctly fixed (i.e., if $\check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i}$), then the $(i+1)$th element of $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}$, denoted $\check{\epsilon}_{\mathrm{c}(i+1)}$, is uncorrelated with the previous i elements. This property will be exploited later on. GIAB decides whether to fix the ith ambiguity based on the value of $\check{\epsilon}_{\mathrm{c}i}$. It operates in such

a way that

$$\begin{aligned} |\check{\epsilon}_{ci}| &\leq \frac{\beta_i}{2} \quad \text{for} \quad i \in \{1, \ldots, q\},\ q > 0 \\ \frac{\beta_i}{2} < |\check{\epsilon}_{ci}| &\leq \frac{1}{2} \quad \text{for} \quad i = q+1,\ q < m \end{aligned} \tag{3.4}$$

In other words, all $i \in \{1, \ldots, q\}$ ambiguities that GIAB validates have small sequentially-constrained ambiguity residuals $\check{\epsilon}_{ci}$, but the $(q+1)$th ambiguity, which GIAB refuses to fix (assuming $q < m$), has $\check{\epsilon}_{c(q+1)}$ too large for GIAB to confidently fix. Note that a rounding operation within GIAB ensures $|\check{\epsilon}_{ci}| \leq 1/2$.

If the full set of ambiguities is fixed and validated, the float baseline can be constrained by the float ambiguity residual, resulting in the so-called fixed baseline estimate:

$$\begin{aligned} \check{\boldsymbol{b}} &= \hat{\boldsymbol{b}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} Q_{\hat{\boldsymbol{z}}}^{-1} \check{\boldsymbol{\epsilon}} \\ &= \hat{\boldsymbol{b}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} \left(L^{-T} D^{-1} L^{-1} \right) L \check{\boldsymbol{\epsilon}}_{\mathrm{c}} \\ &= \hat{\boldsymbol{b}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} L^{-T} D^{-1} \check{\boldsymbol{\epsilon}}_{\mathrm{c}} \\ &= \hat{\boldsymbol{b}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} D^{-1} \check{\boldsymbol{\epsilon}}_{\mathrm{c}} \end{aligned} \tag{3.5}$$

where $Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} \triangleq Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} L^{-T}$.

The distribution of the fully-fixed baseline conditioned on a particular fixed ambiguity $\check{\boldsymbol{z}} = \boldsymbol{z} + \Delta\boldsymbol{z}$ is [32]

$$\left(\check{\boldsymbol{b}} \,|\check{\boldsymbol{z}} = \boldsymbol{z} + \Delta\boldsymbol{z} \right) \sim \mathcal{N}\left(\boldsymbol{b} + Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}} Q_{\hat{\boldsymbol{z}}}^{-1} \Delta\boldsymbol{z}, Q_{\check{\boldsymbol{b}}} \right) \tag{3.6}$$

where $Q_{\check{\boldsymbol{b}}} \triangleq Q_{\hat{\boldsymbol{b}}} - Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}} D^{-1} Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}}}^{T}$. Thus, when the integer ambiguity is fixed correctly ($\Delta\boldsymbol{z} = \boldsymbol{0}$), the fully-fixed baseline has a Gaussian distribution whose mean equals the true baseline $\boldsymbol{b}$.

3.3 Prior Distribution of the GIAB Baseline

Analogous to the float baseline $\hat{\boldsymbol{b}}$ and the fixed baseline $\check{\boldsymbol{b}}$, a partially-fixed baseline, denoted $\bar{\boldsymbol{b}}$, can be calculated from the inputs and outputs of GIAB. The *a priori* and *a posteriori* distributions of $\bar{\boldsymbol{b}}$ are important performance indicators. This section derives the *a priori* distribution of $\bar{\boldsymbol{b}}$.

Let $Q_{\hat{\boldsymbol{b}}\,\hat{z}_{cj}}$ indicate the jth column of the matrix $Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_c}$, and d_j the jth entry on the diagonal of D. Because D is diagonal, (3.5) can be written

$$\check{\boldsymbol{b}} = \hat{\boldsymbol{b}} - \sum_{j=1}^{m} Q_{\hat{\boldsymbol{b}}\hat{z}_{cj}} \frac{\check{\epsilon}_{cj}}{d_j} \tag{3.7}$$

The baseline constrained by only the first i ambiguities, written $\check{\boldsymbol{b}}_i$, can be calculated by truncating the summation in (3.7) at i:

$$\check{\boldsymbol{b}}_i = \hat{\boldsymbol{b}} - \sum_{j=1}^{i} Q_{\hat{\boldsymbol{b}}\hat{z}_{cj}} \frac{\check{\epsilon}_{cj}}{d_j} \tag{3.8}$$

The event $q = i < m$ implies that GIAB could not fix the $(i+1)$th ambiguity without violating the specified probability of failure. For the moment, let $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_q$ be GIAB's partially-fixed baseline solution; alternative assignments for $\bar{\boldsymbol{b}}$ will be explored later on. Denote by F^c the complement of the failure event, F, and let $f_{\bar{\boldsymbol{b}}|F}$ and $f_{\bar{\boldsymbol{b}}|F^c}$ be the probability density functions (PDFs) of $\bar{\boldsymbol{b}}$ conditioned respectively on F and F^c. It follows from the total probability theorem that the prior (unconditioned) PDF of the partially-fixed baseline $\bar{\boldsymbol{b}}$ is

$$f_{\bar{\boldsymbol{b}}}(\boldsymbol{\xi}) = f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{\xi})(1 - P_F) + f_{\bar{\boldsymbol{b}}|F}(\boldsymbol{\xi})\, P_F \tag{3.9}$$

Since, by design, $P_F \leq \bar{P}_F \ll 1$, momentarily neglect the second term on the right-hand side of (3.9). This term is not important for the average performance of the GIAB algorithm, though it is central to position domain integrity considerations in Section 3.6. A detailed expression for $f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{\xi})$, from the first term, is derived along with other conditional PDFs in the following section.

3.4 Conditional distributions of the Generalized Integer Aperture Baseline

Various conditional distributions of $\bar{\boldsymbol{b}}$ offer valuable insight into its behavior under partial ambiguity resolution. This section presents a conceptual overview of the various distributions, followed by detailed derivations of the same.

3.4.1 Conceptual Overview

Consider $f_{\bar{\boldsymbol{b}}|C_i}(\boldsymbol{\xi})$, the PDF of $\bar{\boldsymbol{b}}$ conditioned on GIAB correctly resolving the first i ambiguities. Note that this conditioning makes no assumption that GIAB resolved *only* i ambiguities; in fact, GIAB may have resolved more than i—correctly or not. The conditioning on C_i assumes only that the first i were correctly resolved. One would expect this conditional PDF to be Gaussian with a mean of $\boldsymbol{b}$, since, as (2.16) indicates, the fully-fixed baseline $\check{\boldsymbol{b}}$ conditioned on $\Delta\boldsymbol{z} = 0$ is Gaussian with mean $\boldsymbol{b}$. Indeed, this turns out to be the case.

Now consider $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ for $i < m$. The event Z_i implies $\check{\boldsymbol{z}}_{1:i} = \boldsymbol{z}_{1:i}$ but when $i < m$ it further implies that GIAB refused to fix one or more ambiguities. Thus, conditioning on Z_i when $i < m$ indicates that the magnitude of the $(i +$

$1)$th sequentially-constrained ambiguity residual $\check{\epsilon}_{\mathrm{c}(i+1)}$ was larger than $\beta_{i+1}/2$. No assumption is made about the particular value of $\check{\epsilon}_{\mathrm{c}(i+1)}$, only that it was too large to confidently fix the corresponding integer. In this case will $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ be Gaussian with mean $\boldsymbol{b}$? The answer is no: $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ has mean $\boldsymbol{b}$ but is not Gaussian. This can be explained by considering (3.8) and recognizing that, although $\check{\epsilon}_{\mathrm{c}(i+1)}$ being large has no bearing on $\check{\epsilon}_{\mathrm{c}j}$ for $j \in \{1, \ldots, i\}$ (because these are uncorrelated with $\check{\epsilon}_{\mathrm{c}(i+1)}$ under Z_i), it *does* imply something about $\hat{\boldsymbol{b}}$, namely, that its PDF does not have a mode at $\boldsymbol{b}$: the most probable values of $\hat{\boldsymbol{b}}$ are offset from $\boldsymbol{b}$.

Finally, consider $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$, which is the PDF of $\bar{\boldsymbol{b}}$ conditioned on Z_i for $i < m$ and on the particular value of the sequentially-constrained ambiguity residual, $\check{\epsilon}_{\mathrm{c}(i+1)}$, that caused GIAB to refuse to fix the $(i+1)$th ambiguity. Somewhat surprisingly, this PDF turns out to be neither Gaussian nor of mean $\boldsymbol{b}$. This key result, unknown in the existing literature, is critical because $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ informs decision making about $\bar{\boldsymbol{b}}$: it is the best indicator of whether a particular $\bar{\boldsymbol{b}}$ will be accurate enough for a high-integrity application.

Manipulation in the following subsections leads to detailed expressions for $f_{\bar{\boldsymbol{b}}|F^{\mathrm{c}}}(\boldsymbol{\xi})$, $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$, and $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$.

3.4.2 Finding $f_{\bar{\boldsymbol{b}}|F^{\mathrm{c}}}$

The conditional PDF $f_{\bar{\boldsymbol{b}}|F^{\mathrm{c}}}$, which appears in (3.9), can be written in terms of $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$, the PDF of $\bar{\boldsymbol{b}}$ conditioned on successful validation of $q = i$ ambiguities,

as follows:

$$\begin{aligned} f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{\xi}) &= \sum_{i=0}^{m} P(Z_i|F^c) f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi}) \\ &= \sum_{i=0}^{m} \frac{P(Z_i, F^c)}{P(F^c)} f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi}) \\ &= \sum_{i=0}^{m} \frac{P_{Z_i}}{1-P_F} f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi}) \end{aligned} \tag{3.10}$$

where P_{Z_i} is the probability of the event Z_i, and where the final simplification follows from $Z_i \subset F^c$.

3.4.3 Finding $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$

The PDF $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$, which appears in (3.10), can be expressed in terms of GIAB's output $\check{\boldsymbol{z}} \in \mathbb{Z}^r$, where $r = \min(m, q+1)$. When conditioned on Z_i with $i < m$, the first i ambiguities in $\check{\boldsymbol{z}}$, are correct, but the $(i+1)$th may not be; in other words, $\check{\boldsymbol{z}} = [z_1, \ldots, z_i, z_{i+1} + \Delta z]^T$ for some $\Delta z \in \mathbb{Z}$. Recalling that $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \triangleq L^{-1}_{1:r,1:r}(\hat{\boldsymbol{z}}_{1:r} - \check{\boldsymbol{z}})$, and recognizing $L^{-1}_{1:r,1:r}$ as unit lower triangular, then given Z_i it follows that $\check{\boldsymbol{\epsilon}}_{\mathrm{c}(1:i)} = \boldsymbol{\epsilon}_{\mathrm{c}(1:i)} \triangleq L^{-1}_{1:i,1:i}(\hat{\boldsymbol{z}}_{1:i} - \boldsymbol{z}_{1:i})$ and that $\check{\epsilon}_{\mathrm{c}(i+1)} = \epsilon_{\mathrm{c}(i+1)} - \Delta z$. From standard probability theory, the PDF of the difference $\check{\epsilon}_{\mathrm{c}(i+1)} = \epsilon_{\mathrm{c}(i+1)} - \Delta z$ can be expressed in terms of the joint PDF of $\epsilon_{\mathrm{c}(i+1)}$ and Δz, which, in turn can be expressed as the product of the conditional and marginal PDFs of Δz and $\epsilon_{\mathrm{c}(i+1)}$, respectively:

$$\begin{aligned} f_{\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}(\varepsilon) &= \sum_{k\in\mathbb{Z}} f_{\epsilon_{\mathrm{c}(i+1)},\Delta z|Z_i}(\varepsilon + k, k) \\ &= \sum_{k\in\mathbb{Z}} f_{\Delta z|\epsilon_{\mathrm{c}(i+1)},Z_i}(k|\varepsilon + k)\, f_{\epsilon_{\mathrm{c}(i+1)}|Z_i}(\varepsilon + k) \end{aligned} \tag{3.11}$$

This expression can be simplified by noting from (3.4) that under Z_i the rejected sequentially-constrainted ambiguity residual $\check{\epsilon}_{\mathrm{c}(i+1)}$ satisfies

$$\frac{\beta_{i+1}}{2} < \left|\check{\epsilon}_{\mathrm{c}(i+1)}\right| \leq \frac{1}{2}$$

Expressed another way, the support of $\check{\epsilon}_{\mathrm{c}(i+1)}$ under Z_i is

$$A_{i+1} \triangleq \left\{ \varepsilon \,\middle|\, \frac{\beta_{i+1}}{2} < |\varepsilon| \leq \frac{1}{2} \right\}$$

Thus, given that $\check{\epsilon}_{\mathrm{c}(i+1)} = \epsilon_{\mathrm{c}(i+1)} - \Delta z \in A_{i+1}$, the conditioning on $\epsilon_{\mathrm{c}(i+1)} = \varepsilon + k$ in (3.11) implies $\Delta z = k$. (The condition $|\varepsilon| = 1/2$ upsets this unique mapping but happens with probability 0.) Therefore,

$$f_{\Delta z|\epsilon_{\mathrm{c}(i+1)},Z_i}\left(k|\varepsilon+k\right) = 1 \quad \forall k \in \mathbb{Z},\ \forall \varepsilon \in A_{i+1}$$

and (3.11) simplifies to

$$f_{\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}(\varepsilon) = \sum_{k\in\mathbb{Z}} f_{\epsilon_{\mathrm{c}(i+1)}|Z_i}\left(\varepsilon+k\right) \tag{3.12}$$

With these preliminaries, $f_{\bar{\boldsymbol{b}}|Z_i}\left(\boldsymbol{\xi}\right)$ can be constructed as the marginal of the joint PDF of $\bar{\boldsymbol{b}}$ and $\check{\epsilon}_{\mathrm{c}(i+1)}$, and the latter can be expressed in terms of a sum of joint PDFs with $\epsilon_{\mathrm{c}(i+1)}$ by the same reasoning that led to (3.12):

$$\begin{aligned} f_{\bar{\boldsymbol{b}}|Z_i}\left(\boldsymbol{\xi}\right) &= \int_{A_{i+1}} f_{\bar{\boldsymbol{b}},\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}\left(\boldsymbol{\xi},\varepsilon\right) d\varepsilon \\ &= \int_{A_{i+1}} \sum_{k\in\mathbb{Z}} f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|Z_i}\left(\boldsymbol{\xi},\varepsilon+k\right) d\varepsilon \end{aligned} \tag{3.13}$$

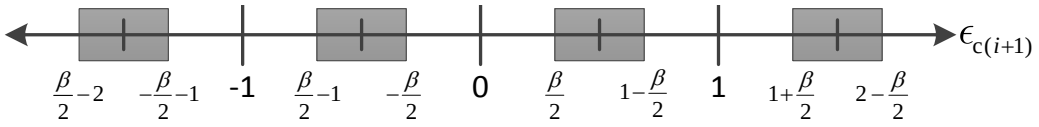

Figure 3.1: The rejection event R_{i+1} is triggered when $\epsilon_{\mathrm{c}(i+1)}$ falls within the indicated bands. For compactness, β_{i+1} is abbreviated as β.

3.4.4 Finding $f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|Z_i}(\boldsymbol{\xi},\varepsilon+k)$

To find $f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|Z_i}(\boldsymbol{\xi},\varepsilon+k)$, which appears in (3.13), it is helpful to express the rejection event R_{i+1} in terms of $\epsilon_{\mathrm{c}(i+1)}$, as follows:

$$R_{i+1}:\epsilon_{\mathrm{c}(i+1)}\in\{\varepsilon+k\mid\varepsilon\in A_{i+1},\ k\in\mathbb{Z}\}$$

Fig. 3.1 illustrates the bands of $\epsilon_{\mathrm{c}(i+1)}$ that trigger rejection. In the context of (3.13), where conditioning is on Z_i with $i<m$ [the $(i+1)$th ambiguity was rejected], Z_i is the intersection of the correct validation event C_i and the rejection event R_{i+1}. Accordingly, the PDF $f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|Z_i}(\boldsymbol{\xi},\varepsilon)$ is identical to $f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|C_i}(\boldsymbol{\xi},\varepsilon)$ but with two modifications: (1) support of $\epsilon_{\mathrm{c}(i+1)}$ is restricted to R_{i+1}, and (2) a normalization by $P_{R_{i+1}}$, given in (2.24c), is applied to ensure the PDF integrates to unity. Let $\mathbf{1}_{R_{i+1}}(\varepsilon)$ be the indicator function for the rejection event, equal to unity for those values of ε that trigger R_{i+1}, and zero otherwise. Then the joint PDF is

$$f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|Z_i}(\boldsymbol{\xi},\varepsilon)=\frac{\mathbf{1}_{R_{i+1}}(\varepsilon)}{P_{R_{i+1}}}f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|C_i}(\boldsymbol{\xi},\varepsilon)\tag{3.14}$$

To find $f_{\bar{\boldsymbol{b}},\epsilon_{\mathrm{c}(i+1)}|C_i}(\boldsymbol{\xi},\varepsilon)$ note that, under the event C_i, $\check{\boldsymbol{b}}_i$ and $\epsilon_{\mathrm{c}(i+1)}$ are jointly Gaussian:

$$\begin{bmatrix}\check{\boldsymbol{b}}_i\\ \epsilon_{\mathrm{c}(i+1)}\end{bmatrix}\sim\mathcal{N}\left(\begin{bmatrix}\boldsymbol{\xi}\\ \varepsilon\end{bmatrix};\begin{bmatrix}\boldsymbol{b}\\ 0\end{bmatrix},Q_{\check{\boldsymbol{b}}_i\epsilon_{\mathrm{c}(i+1)}}\right)\tag{3.15}$$

with

$$Q_{\check{\boldsymbol{b}}_i \epsilon_{\mathrm{c}(i+1)}} = \begin{bmatrix} Q_{\check{\boldsymbol{b}}_i} & Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} \\ Q^T_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} & d_{i+1} \end{bmatrix} \tag{3.16}$$

where $Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}}$ is the $(i+1)$th column of $Q_{\hat{\boldsymbol{b}}\hat{\boldsymbol{z}}_{\mathrm{c}}}$, introduced in (2.14), and

$$Q_{\check{\boldsymbol{b}}_i} = Q_{\hat{\boldsymbol{b}}} - \sum_{j=1}^{i} \frac{1}{d_j} Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}j}} \left(Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_{\mathrm{c}j}} \right)^T \tag{3.17}$$

is found by exploiting the independence of each element of $\boldsymbol{\epsilon}_{\mathrm{c}}$. Then the conditional mean error of $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_i$, given C_i and $\epsilon_{\mathrm{c}(i+1)} = \varepsilon + k$, follows from the standard expression for the Gaussian conditional mean:

$$\begin{aligned} \boldsymbol{\mu}_k(\varepsilon) &\triangleq E\left[\bar{\boldsymbol{b}} - \boldsymbol{b} \,\middle|\, \epsilon_{\mathrm{c}(i+1)} = \varepsilon + k, C_i \right] \\ &= Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} \left(\frac{\varepsilon + k}{d_{i+1}} \right) \end{aligned} \tag{3.18}$$

Its covariance is found by extending the summation in (3.17) to $(i+1)$:

$$\mathrm{cov}\left(\bar{\boldsymbol{b}} \,\middle|\, \epsilon_{\mathrm{c}(i+1)} = \varepsilon + k, C_i \right) = Q_{\check{\boldsymbol{b}}_{i+1}} \tag{3.19}$$

Hence the partially-fixed baseline $\bar{\boldsymbol{b}}$, when conditioned on $\epsilon_{\mathrm{c}(i+1)}$ and C_i, is Gaussian distributed and biased away from the true baseline $\boldsymbol{b}$ by $\boldsymbol{\mu}_k(\varepsilon)$:

$$f_{\bar{\boldsymbol{b}}|\epsilon_{\mathrm{c}(i+1)}, C_i}\left(\boldsymbol{\xi} | \varepsilon + k \right) = \mathcal{N}(\boldsymbol{\xi}; \boldsymbol{b} + \boldsymbol{\mu}_k(\varepsilon), Q_{\check{\boldsymbol{b}}_{i+1}}) \tag{3.20}$$

Then, recognizing that $\boldsymbol{\epsilon}_{\mathrm{c}} \sim \mathcal{N}(0, D)$ implies

$$f_{\epsilon_{\mathrm{c}(i+1)}|C_i}(\varepsilon) = \mathcal{N}(\varepsilon; 0, d_{i+1}) \tag{3.21}$$

and factoring the joint PDF in (3.14) into its conditional-times-marginal form yields this subsection's desired PDF:

$$\begin{aligned} &f_{\bar{\boldsymbol{b}}, \epsilon_{\mathrm{c}(i+1)}|Z_i}\left(\boldsymbol{\xi}, \varepsilon + k \right) = \\ &\frac{\mathbf{1}_{R_{i+1}}(\varepsilon)}{P_{R_{i+1}}} \mathcal{N}(\boldsymbol{\xi}; \boldsymbol{b} + \boldsymbol{\mu}_k(\varepsilon), Q_{\check{\boldsymbol{b}}_{i+1}}) \mathcal{N}(\varepsilon + k; 0, d_{i+1}) \end{aligned} \tag{3.22}$$

Moreover, substituting (3.22) into (3.13), yields $f_{\bar{\boldsymbol{b}}|Z_i}$, and substituting (3.13) into (3.10) yields a detailed expression for $f_{\bar{\boldsymbol{b}}|F^c}$:

$$\begin{aligned} f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{\xi}) &= \frac{P_{S_m}}{1-P_F}\mathcal{N}(\boldsymbol{\xi};\boldsymbol{b},Q_{\check{\boldsymbol{b}}_m}) \\ &+ \sum_{i=0}^{m-1}\frac{P_{Z_i}/P_{R_{i+1}}}{1-P_F} \\ &\times \sum_{k\in\mathbb{Z}}\int_{A_{i+1}}\mathcal{N}(\boldsymbol{\xi};\boldsymbol{b}+\boldsymbol{\mu}_k(\varepsilon),Q_{\check{\boldsymbol{b}}_{i+1}})\mathcal{N}(\varepsilon+k;0,d_{i+1})d\varepsilon \end{aligned}$$

3.4.5 Finding $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$

After the foregoing steps, one can find the important PDF $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ starting with

$$f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon) = \frac{f_{\bar{\boldsymbol{b}},\check{\epsilon}_{c(i+1)}|Z_i}(\boldsymbol{\xi},\varepsilon)}{f_{\check{\epsilon}_{c(i+1)}|Z_i}(\varepsilon)} \tag{3.23}$$

Substituting the integrand of (3.13) for the numerator and (3.12) for the denominator yields

$$f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon) = \frac{\sum_{k\in\mathbb{Z}} f_{\bar{\boldsymbol{b}},\epsilon_{c(i+1)}|Z_i}(\boldsymbol{\xi},\varepsilon+k)}{\sum_{j\in\mathbb{Z}} f_{\epsilon_{c(i+1)}|Z_i}(\varepsilon+j)} \tag{3.24}$$

Now substituting (3.22) and (3.21), where the normalization for the rejection event cancels out, and constraining $\varepsilon \in A_{i+1}$ to eliminate the indicator functions, yields

$$\begin{aligned} &f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon) = \\ &\quad = \frac{\sum_{k\in\mathbb{Z}}\mathcal{N}(\boldsymbol{\xi};\boldsymbol{b}+\boldsymbol{\mu}_k(\varepsilon),Q_{\check{\boldsymbol{b}}_{i+1}})\mathcal{N}(\varepsilon+k;0,d_{i+1})}{\sum_{j\in\mathbb{Z}}\mathcal{N}(\varepsilon+j;0,d_{i+1})} \end{aligned} \tag{3.25}$$

This PDF can be interpreted as a mixture of Gaussian densities with different means but equal variances. The mixture probabilities are in fact the conditional probabilities that $\Delta z = k$ given the sequentially-constrainted ambiguity residual $\check{\epsilon}_{c(i+1)}$ and

the event Z_i:

$$\begin{aligned} p_k(\varepsilon) &\triangleq P(\Delta z = k | \check{\epsilon}_{\mathrm{c}(i+1)} = \varepsilon \in A_{i+1}, Z_i) \\ &= \frac{\mathcal{N}(\varepsilon + k; 0, d_{i+1})}{\sum_{j \in \mathbb{Z}} \mathcal{N}(\varepsilon + j; 0, d_{i+1})}, \ \varepsilon \in A_{i+1} \end{aligned} \tag{3.26}$$

Simplifying (3.25) with the mixture probability notation of (3.26) yields, for $\varepsilon \in A_{i+1}$,

$$f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)}, Z_i}(\boldsymbol{\xi}|\varepsilon) = \sum_{k \in \mathbb{Z}} p_k(\varepsilon) \cdot \mathcal{N}(\boldsymbol{\xi}; \boldsymbol{b} + \boldsymbol{\mu}_k(\varepsilon), Q_{\check{\boldsymbol{b}}_{i+1}}) \tag{3.27}$$

3.4.6 Discussion

Two important observations can be drawn from the foregoing conditional distributions. First consider (3.27). Note that $p_k(\varepsilon)$ and $\boldsymbol{\mu}_k(\varepsilon)$ are evaluated only for $\varepsilon \neq 0$, since $\varepsilon \in A_{i+1}$, which does not contain the origin. From (3.18), one observes that, for $\varepsilon \neq 0$ and assuming $Q_{\check{\boldsymbol{b}}\check{z}_{\mathrm{c}(i+1)}}$ is nonzero, the bias $\boldsymbol{\mu}_k(\varepsilon)$ is nonzero for any value of $k \in \mathbb{Z}$. Thus, the means of the Gaussian PDFs that get summed in (3.27) are all shifted away from the true baseline $\boldsymbol{b}$. It is possible for a weighting function $p_k(\varepsilon)$ to be chosen to counteract this shifting and thereby restore symmetry in $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)}, Z_i}(\boldsymbol{\xi}|\varepsilon)$, but the actual weighting that applies, given by (3.26), does not do this. As a result, $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)}, Z_i}(\boldsymbol{\xi}|\varepsilon)$ is asymmetric about $\boldsymbol{b}$ with respect to $\boldsymbol{\xi}$.

To be explicitly clear, the PDF of the partially-fixed baseline $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_q$ that results from correction of the float baseline $\hat{\boldsymbol{b}}$ with GIAB-produced $\check{\epsilon}_{\mathrm{c}j}$, as in (3.8) with $i = q < m$, when conditioned on $\check{\epsilon}_{\mathrm{c}(i+1)} = \varepsilon \in A_{i+1}$, will not have a mean coincident with the true baseline $\boldsymbol{b}$ *even when all validated ambiguity fixes are correct*.

The second important observation is that, for $i < m$, $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ given by (3.13) is symmetric about $\boldsymbol{b}$ but lacks a mode at $\boldsymbol{b}$. To see this, note that $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ can be written

$$f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi}) = \int_{A_{i+1}} f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon) f_{\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}(\varepsilon)\, d\varepsilon \tag{3.28}$$

with $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ given by (3.27) and $f_{\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}(\varepsilon)$ by (3.12). The first of these, $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$, is symmetric about $\boldsymbol{b}$ when integrated over ε because both $p_k(\varepsilon)$ from (3.26) and $\boldsymbol{\mu}_k(\varepsilon)$ from (3.18) are symmetric about the origin with respect to ε when summed over all $k \in \mathbb{Z}$. The second, $f_{\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}(\varepsilon)$, is symmetric about the origin with respect to ε because the summand $f_{\epsilon_{\mathrm{c}(i+1)}|Z_i}(\varepsilon + k)$ of (3.12), with ε restricted to A_{i+1}, is simply a normalized version of $f_{\epsilon_{\mathrm{c}(i+1)}|C_i}(\varepsilon + k)$ from (3.21). Thus, since $f_{\epsilon_{\mathrm{c}(i+1)}|C_i}(\varepsilon+k)$ and A_{i+1} are symmetric with respect to ε, so is the PDF $f_{\check{\epsilon}_{\mathrm{c}(i+1)}|Z_i}(\varepsilon)$. Taken together, these facts imply that $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ is symmetric about $\boldsymbol{b}$. Critically however, the support A_{i+1} does not contain the origin, so $\boldsymbol{\mu}_k(\varepsilon) \neq 0$ for all $\varepsilon \in A_{i+1}$ and all $k \in \mathbb{Z}$. This implies that, although $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$ with $i < m$ is symmetric about $\boldsymbol{b}$, it does not have a mode at $\boldsymbol{b}$.

The above two observations can be understood intuitively as follows: GIAB refusing to fix the $(i+1)$th ambiguity indicates the float ambiguity $\hat{\boldsymbol{z}}$ is biased away from $\boldsymbol{z}$, which implies the float baseline $\hat{\boldsymbol{b}}$ is biased away from $\boldsymbol{b}$. If GIAB with $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_i$ fixes only $i = q < m$ ambiguities, the correction to the float baseline given by (3.8) is incomplete, leaving some residual bias in $\bar{\boldsymbol{b}}$. When $\bar{\boldsymbol{b}}$ is conditioned on the particular value $\check{\epsilon}_{\mathrm{c}(i+1)} = \varepsilon \in A_{i+1}$ under Z_i, the bias manifests as an ε-dependent shift of the mean away from $\boldsymbol{b}$. When $\bar{\boldsymbol{b}}$ is conditioned only on Z_i, the

bias manifests as a symmetric exodus of probability density away from $\boldsymbol{b}$, leaving no mode at $\boldsymbol{b}$. Figures of these distributions will be presented in the next subsection.

Note that the above reasoning is not unique to GIAB: the conditional PDF of $\bar{\boldsymbol{b}}$ will behave similarly for any data-driven partial ambiguity resolution algorithm that corrects the float baseline with the validated fixes.

3.4.7 GIAB variants

The foregoing conditional PDFs and discussion assume $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_i$ with $i = q$, which, according to (3.8), implies the float baseline $\hat{\boldsymbol{b}}$ is only corrected by the q sequentially-constrained ambiguity residuals $\check{\boldsymbol{\epsilon}}_{\mathrm{c}(1:q)}$ that pass validation (those satisfying $|\check{\epsilon}_{\mathrm{c}i}| \leq \beta_i/2$). If $q < m$, the next sequentially-constrained ambiguity residual, $\check{\epsilon}_{\mathrm{c}(q+1)}$, is ignored, which means that the component of $\hat{\boldsymbol{b}}$ that might have been corrected by $\check{\epsilon}_{\mathrm{c}(q+1)}$ is left unchanged at its float value. This approach, hereafter called float GIAB, is the typical practice in the existing literature on partial ambiguity resolution. However, the existing literature's calculation of integrity risk IR does not appear to recognize that $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_q$ is biased as described above [3, 4, 16].

Setting $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_q$ (thus ignoring $\check{\epsilon}_{\mathrm{c}(q+1)}$) is of course not the only way to handle the first ambiguity that fails validation. This paper considers three variants of GIAB, each distinguished by its treatment of the $(q+1)$th ambiguity. The first is float GIAB, described above. The second, called MAP GIAB by analogy to maximum *a posteriori* estimation, applies the most likely fix candidate, which, given GIAB's operation as defined in [11], is equivalent to choosing $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_{q+1}$ for $q < m$. The PDF $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ for $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_{q+1}$ is the same as that of the float variant (for which

$\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_q$), except that all $\boldsymbol{\mu}_k(\varepsilon)$ are shifted by the $(i+1)$th correction in (3.8), namely $-Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} \frac{\check{\epsilon}_{\mathrm{c}(i+1)}}{d_{i+1}}$. Recalling that $\epsilon_{\mathrm{c}(i+1)} = \check{\epsilon}_{\mathrm{c}(i+1)} + \Delta z$, one notes that the additional correction removes the fractional part from $\epsilon_{\mathrm{c}(i+1)}$, leaving

$$\boldsymbol{\mu}_k(\varepsilon) = Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} \frac{k}{d_{i+1}} \quad \text{(MAP GIAB)} \tag{3.29}$$

which is zero if $\Delta z = k = 0$. [The argument ε in $\boldsymbol{\mu}_k(\varepsilon)$ is retained for functional consistency with (3.18).] Thus, for $\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_{q+1}$, the conditional PDF $f_{\bar{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ is unbiased about $\boldsymbol{b}$ if the non-validated fix is correct. MAP GIAB monitors the effect of incorrect fixes on IR by calculating each alternate fix's probability and position domain bias. This approach is similar to the concept of position domain integrity (PDI) in the EPIC and GERAFS algorithms.

The third variant of GIAB, called MMSE GIAB by analogy to minimum mean squared error estimation, computes a weighted average of the MAP GIAB baseline solution and the alternative fixed solutions. Because the baseline corrections are applied linearly, MMSE GIAB's partially-fixed baseline $\bar{\boldsymbol{b}}$ can be written

$$\bar{\boldsymbol{b}} = \check{\boldsymbol{b}}_i - Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} \frac{\varepsilon + \sum_{j\in\mathbb{Z}} p_j(\varepsilon) j}{d_{i+1}} \quad \text{(MMSE GIAB)} \tag{3.30}$$

This baseline solution is analogous to the Best Integer Equivariant (BIE) estimator of [5, 41], with the difference that MMSE GIAB limits the number of fixes considered by sizing the aperture according to [11].

The ideal corrected baseline, which has a zero-mean-error PDF if $\Delta z = k$, is

$$\bar{\boldsymbol{b}}_{\text{ideal}} = \check{\boldsymbol{b}}_i - Q_{\hat{\boldsymbol{b}}\hat{z}_{\mathrm{c}(i+1)}} \frac{\varepsilon + k}{d_{i+1}} \tag{3.31}$$

The bias in the MMSE solution is thus

$$\begin{aligned}\boldsymbol{\mu}_k(\varepsilon) &= \bar{\boldsymbol{b}} - \bar{\boldsymbol{b}}_{\text{ideal}} \quad \text{(MMSE GIAB)} \\ &= Q_{\hat{\boldsymbol{b}}\hat{z}_{\text{c}(i+1)}} \frac{k - \sum_{j\in\mathbb{Z}} j p_j(\varepsilon)}{d_{i+1}} \\ &= Q_{\hat{\boldsymbol{b}}\hat{z}_{\text{c}(i+1)}} \frac{\sum_{j\in\mathbb{Z}} (k-j) p_j(\varepsilon)}{d_{i+1}}\end{aligned} \tag{3.32}$$

where the last equality makes use of $\sum_{j\in\mathbb{Z}} p_j(\varepsilon) = 1$.

For small values of ε, $p_0(\varepsilon) \gg p_j(\varepsilon),\ \forall j \neq 0$, meaning that the MAP fix is much more likely than the alternatives, in which case the MMSE and MAP GIAB baselines will differ only slightly. At the other extreme, in the zero-probability event that $|\check{\epsilon}_{c(i+1)}| = 1/2$, the MMSE and float GIAB baselines are equivalent.

Analysis has shown that when $P_{C_{i+1}} > 0.7$, which is attainable for even relatively weak models, neglecting all but the two most likely fix candidates for the $(i+1)$th ambiguity raises the integrity risk by less than $P_{E_{i+1}}$ for all values of d_{i+1}. This makes the third most likely fix, and all less likely fixes, negligibly likely. In particular, when $P_{C_{i+1}} > 0.7$ the three highest values of $p_k(\varepsilon)$ are $p_0(\varepsilon) > p_{-\text{sgn}(\varepsilon)}(\varepsilon) \gg p_{\text{sgn}(\varepsilon)}(\varepsilon)$. Neglecting all but the two most likely fixes, the partially-fixed baseline $\bar{\boldsymbol{b}}$ for each of the three GIAB variants can be approximated by the following conditional PDF, with $\boldsymbol{\mu}_k(\varepsilon)$ given by (3.18), (3.29), or (3.32), as appropriate:

$$f_{\check{\boldsymbol{b}}|\check{\epsilon}_{\text{c}(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon) \approx \sum_{k\in\{0,-\text{sgn}(\epsilon)\}} p_k(\varepsilon) \cdot \mathcal{N}(\boldsymbol{\xi}; \boldsymbol{b} + \boldsymbol{\mu}_k(\varepsilon), Q_{\check{\boldsymbol{b}}_{i+1}}) \tag{3.33}$$

The PDFs of (3.33), (3.13), and (3.10) are illustrated in Figs. 3.2, 3.3, and 3.4, respectively. The PDFs have been shifted so the horizontal axis's origin coincides with the component value of the true baseline $\boldsymbol{b}$. Note that the MMSE GIAB

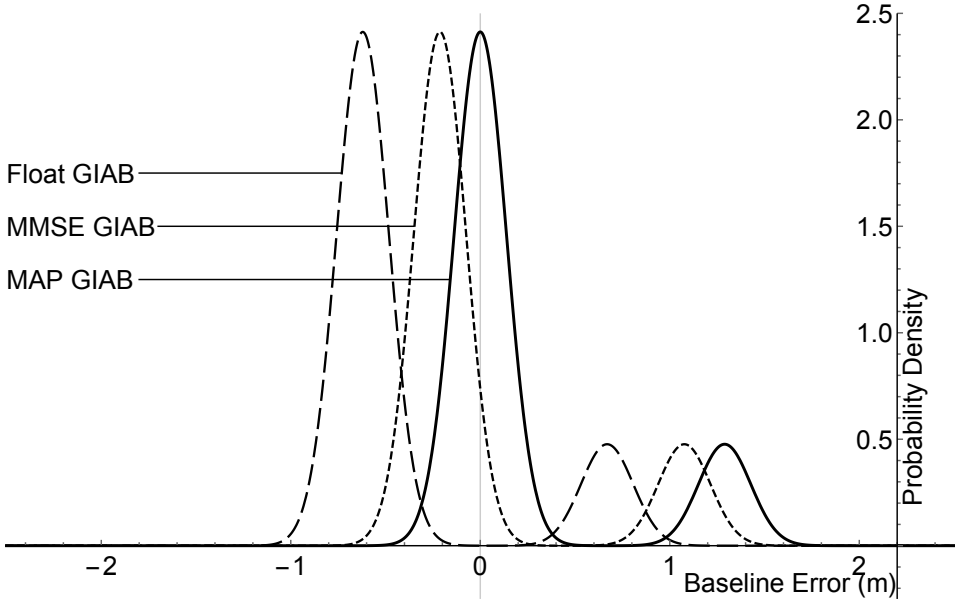

Figure 3.2: A single component of $f_{\check{\boldsymbol{b}}|\check{\epsilon}_{\mathrm{c}(i+1)},Z_i}(\boldsymbol{b}+\boldsymbol{\xi}|\varepsilon)$ from (3.33) for float, MAP, and MMSE GIAB for $q = i < m$ and a large ambiguity residual ε. Because the distributions differ only by the variant-specific bias $\boldsymbol{\mu}_k(\varepsilon)$, the three PDFs are merely shifted versions of each other, with that of MMSE GIAB between those of float and MAP GIAB.

PDF in Fig. 3.2 lies between those of the MAP and float variants. Note also that the float GIAB PDF in Fig. 3.3 is bimodal, with both modes shifted away from zero, whereas both MAP and MMSE GIAB have strong modes at zero but wider tails. Finally, observe from Figs. 3.3 and 3.4 that MMSE GIAB has narrower tails than MAP GIAB. This is because MAP GIAB does not change its baseline estimate for large residuals like MMSE GIAB does.

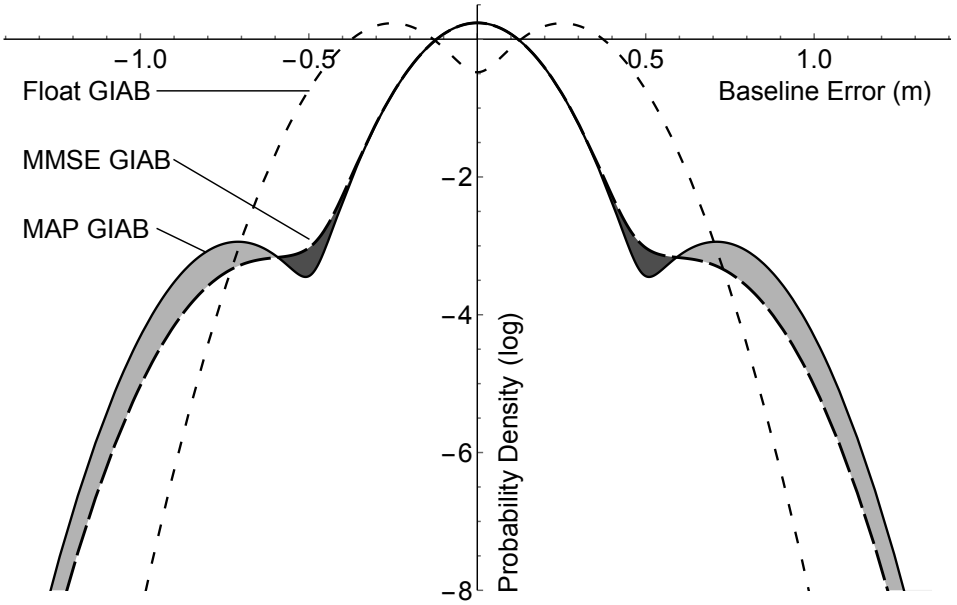

Figure 3.3: A single component of $f_{\bar{\boldsymbol{b}}|Z_i}\left(\boldsymbol{b}+\boldsymbol{\xi}\right)$ for float, MAP, and MMSE GIAB with $q = i < m$, plotted with a log-scaled vertical axis. Float GIAB has a symmetric, bimodal PDF. MAP GIAB has a symmetric PDF with a dominant, zero-mean central mode and heavy tails due to incorrect fixes. MMSE GIAB has a lower probability of large errors than does MAP GIAB at the expense of a slight increase in the probability of moderately-sized errors. In the lightly-shaded region, MAP GIAB has higher density than MAP GIAB, and vice versa in the darker region.

3.5 Validation of Baseline Distributions

In recognition of the possibility that the derived conditional PDFs of $\bar{\boldsymbol{b}}$ suffer from some error in reasoning or probabilistic book-keeping, extensive Monte Carlo simulations were conducted to cross-check the analytical expressions. A float solution model was chosen with eight satellites above a 5° mask. The simulation was initialized by computing the decorrelating Z-transform and setting the integer aperture according to the optimization described in [11]. A total of 4×10^8 Monte Carlo

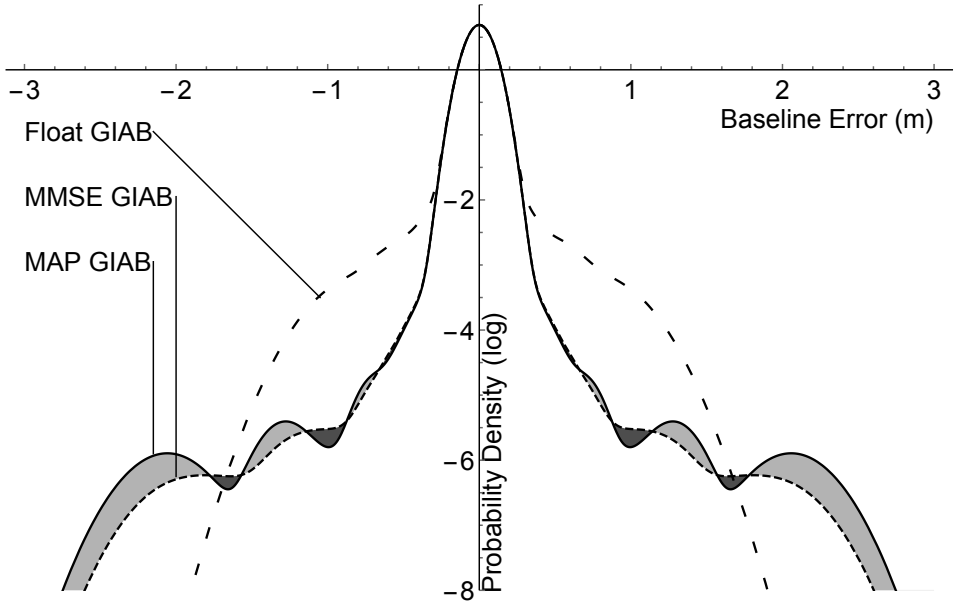

Figure 3.4: A single component of $f_{\bar{\boldsymbol{b}}|F^c}\left(\boldsymbol{b}+\boldsymbol{\xi}\right)$ for float, MAP, and MMSE GIAB, plotted with a log-scaled vertical axis. Float GIAB has strong but narrow tails. Its central mode results from the probability of fixing all m ambiguities. MAP and MMSE GIAB both have a strong central mode and tails that are wider but lower than those of float GIAB. MMSE GIAB has smoother and narrower tails than MAP GIAB.

samples were then drawn from the float distribution described by (2.2) to generate the float solution errors, including the float baseline and float ambiguities. Each sample float ambiguity vector was then Z-transformed and fed through the GIAB algorithm. All GIAB outputs were logged, including the number of correctly fixed samples, the number of incorrectly fixed samples, and the partially-fixed baseline error tabulated by q.

Appropriate histograms of the simulated outcomes were then compared with the analytical PDFs for $f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{\xi})$, $f_{\bar{\boldsymbol{b}}|Z_i}(\boldsymbol{\xi})$, and $f_{\bar{\boldsymbol{b}}|\bar{\varepsilon}_{c(i+1)},Z_i}(\boldsymbol{\xi}|\varepsilon)$ and for float,

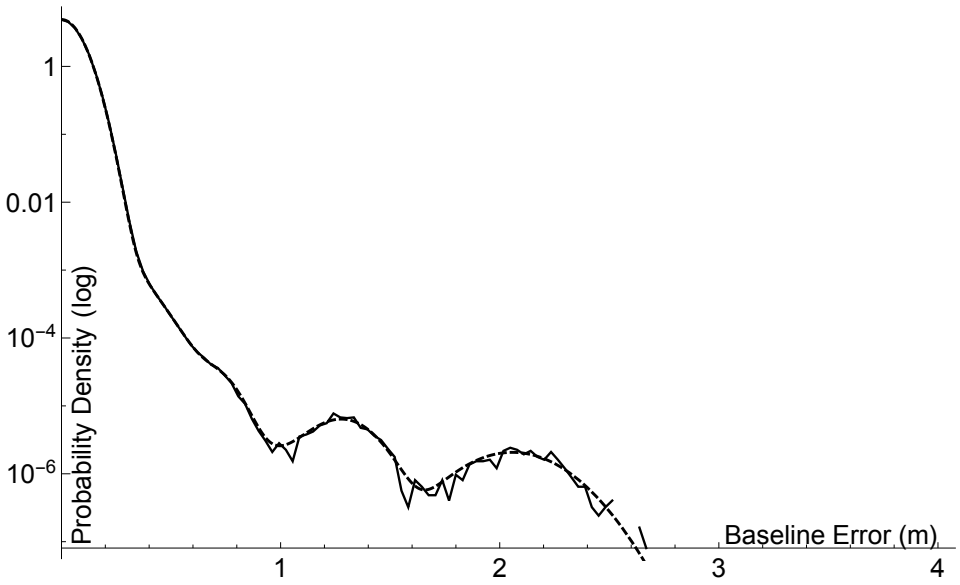

Figure 3.5: A single component of $f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{b}+\boldsymbol{\xi})$ for the theoretical PDF of MAP GIAB (dashed line) and the empirical (simulated) histogram (solid line), plotted in log scale for $m = 7$, $\bar{P}_F = 10^{-8}$, and a Monte Carlo sample size of 4×10^8. The underlying model has an IB probability of correct fix equal to $1 - 2 \times 10^{-5}$. As the PDFs are symmetric about zero, only the positive portion is shown. Clearly, there is strong agreement between the analytical and simulated distributions. Note that because the plot's vertical axis is log scaled, small differences are exaggerated at low probabilities.

MAP, and MMSE GIAB. Fig. 3.5 shows excellent agreement between the empirical (simulated) and analytical $f_{\bar{\boldsymbol{b}}|F^c}(\boldsymbol{\xi})$ for MAP GIAB. Similarly good agreement was found with the other two distributions and the other two GIAB variants. The model underlying Fig. 3.5 is relatively strong: its integer bootstrapping probability of correct fix is $1 - 2 \times 10^{-5}$ for a specified failure probability $\bar{P}_F = 10^{-8}$. Several weak models and other strong models were also studied, all of varying geometry. Each case showed excellent agreement with the derived PDFs.

3.6 Data-Driven Position Domain Integrity

The defining characteristic of high-integrity CDGNSS techniques appropriate for safety-of-life systems is their ability to strictly bound the probability of large position domain errors (errors in the estimate of the baseline vector $\boldsymbol{b}$) even in the event of incorrect fixes. This is the essence of position domain integrity (PDI). For each component of $\boldsymbol{b}$, the risk R that the component's error exceeds the AL must be monitored. Let b represent a particular component of $\boldsymbol{b}$ and $\bar{b}$ its estimate, whether fully or partially fixed. Then R is defined as

$$R \triangleq P\left(\left|\bar{b} - b\right| > AL\right) \tag{3.34}$$

If $R > \bar{IR}$, where $\bar{IR}$ is a specified integrity risk, an alert must be raised.

3.6.1 Position Domain Integrity in EPIC

The EPIC algorithm protects solution integrity by evaluating the *a priori* conditional IR for the case that the ambiguities are fixed correctly and for s cases of incorrect fix [17, 18]. EPIC produces tighter bounds on IR than GERAFS for any error model, so EPIC is considered to the exclusion of GERAFS in this paper. Define $\mathcal{E}_0$ as the event that the chosen ambiguity fix is correct, and $\mathcal{E}_k$ as the event that the kth alternative fix is correct. Let $R_k = P\left(\left|\bar{b} - b\right| > AL \middle| \mathcal{E}_k\right) \leq 1$ be the conditional risk of excess error given the event $\mathcal{E}_k$. The total risk is then

$$R = \sum_{k=0}^{s} R_k P\left(\mathcal{E}_k\right) + \sum_{k=s+1}^{\infty} R_k P\left(\mathcal{E}_k\right) \tag{3.35}$$

Define $\mathcal{E}_\infty$ as the event that the correct fix was neither the chosen fix nor

among the s alternative fix candidates; i.e.,

$$\mathcal{E}_\infty \triangleq \bigcup_{k=s+1}^{\infty} \mathcal{E}_k = \left(\bigcup_{k=0}^{s} \mathcal{E}_k \right)^c \tag{3.36}$$

where $(\cdot)^c$ indicates the set complement.

A bound on the risk of excess error can be derived by conservatively assuming that any incorrect fix not among the s considered will cause excess error, i.e., assuming $R_k = 1, \forall k > s$. This leads to the bound used by EPIC to monitor risk of excess error:

$$\begin{aligned} R_{\text{EPIC}} &\leq P\left(\mathcal{E}_\infty\right) + \sum_{k=0}^{s} R_k P\left(\mathcal{E}_k\right) \\ &\leq 1 - \sum_{k=0}^{s} P\left(\mathcal{E}_k\right) + \sum_{k=0}^{s} R_k P\left(\mathcal{E}_k\right) \\ &\leq 1 - \sum_{k=0}^{s} \left(1 - R_k\right) P\left(\mathcal{E}_k\right) \end{aligned} \tag{3.37}$$

In the EPIC algorithm, the event probabilities are the *a priori* fixing probabilities for IB:

$$\begin{aligned} &P\left(\mathcal{E}_k\right) = \\ &\prod_{j=1}^{m} \left(\Phi\left(\frac{\frac{1}{2} - L_j^{-1} \Delta \boldsymbol{z}_k}{\sqrt{d_j}} \right) - \Phi\left(\frac{-\frac{1}{2} - L_j^{-1} \Delta \boldsymbol{z}_k}{\sqrt{d_j}} \right) \right) \end{aligned} \tag{3.38}$$

where $\Delta \boldsymbol{z}_k$ is the kth candidate fix ambiguity error vector and L_j^{-1} is the jth row of the matrix L^{-1}. Assuming zero-mean Gaussian measurements, the conditional PDI risk for excess error in a particular direction is

$$R_k = \Phi\left(\frac{AL - \mu_k}{\sigma_{\bar{b}}} \right) - \Phi\left(\frac{-AL - \mu_k}{\sigma_{\bar{b}}} \right) \tag{3.39}$$

where μ_k is the desired component of the bias in (2.16) for fix error vector $\Delta \boldsymbol{z}_k$, and $\sigma_{\bar{b}}^2$ is the variance of that component.

3.6.2 Position Domain Integrity in GIAB

Position domain integrity in the GIAB framework enjoys a key advantage over that for EPIC, namely, that GIAB's event probabilities and baseline distributions are *a posteriori* rather than *a priori*. Conditioning on the observed measurements allows GIAB to satisfy the same $\bar{IR}$ as EPIC but with tighter margins. This subsection develops a PDI strategy for GIAB based on the posterior baseline distributions derived previously.

It is important to understand the subtle distinctions in the probabilities of incorrect fix under various conditions. If GIAB's results were only deemed critical in an average sense, then protection based on the *a priori* probability of failure would be sufficient. This is the probability $P_F = P(F) = P(\check{\boldsymbol{z}}_{1:q} \neq \boldsymbol{z}_{1:q}, q > 0)$ of validating any incorrect ambiguity. GIAB manages P_F by design, since its aperture vector $\boldsymbol{\beta}$ is chosen to satisfy $P_F \leq \bar{P}_F$,

But safety-of-life systems are concerned not only with average behavior but also with each estimation epoch's potential for dangerously large errors. Employing a measurement-conditioned distribution allows a clearer assessment of the risk at each epoch. Thus, one might wish to make the posterior distribution in (3.33) the basis for PDI monitoring. But this distribution is conditioned on the event Z_i, requiring $P(Z_i|q = i) = 1 - P(F|q = i)$ be known for operational use. Consider $P(F|q = i)$, which can be written

$$P(F|q = i) = \frac{P(F, q = i)}{P(F^c, q = i) + P(F, q = i)}, \quad i > 0 \tag{3.40}$$

The first term in the denominator may be recognized as the probability of correctly

validating the first i ambiguities, or P_{S_i}. The other term, $P(F, q = i)$, is the probability that one or more of i validated ambiguities are incorrect. Since the event $(F, q = i)$ is a subset of the failure event F, it must have a lower probability; thus $P(F, q = i) = \alpha P_F$, where $0 < \alpha < 1$ for $i > 0$. Then

$$P(F|q = i) = \frac{\alpha P_F}{P_{S_i} + \alpha P_F} \approx \frac{\alpha P_F}{P_{S_i}}, \quad i > 0 \tag{3.41}$$

where the approximation follows from $P_{S_i} \gg P_F$, which is typical for high-integrity systems. For $q = m$ and a strong model, P_{S_m} is very close to unity, so $P(F|q = m)$ remains close to P_F. However, for $q = i < m$ and a strong model, P_{S_i} might itself be quite small, say, less than 10^{-3}, making $P(F|q = i)$ orders of magnitude larger than P_F. In other words, for a strong model, conditioning only on $q < m$ makes an incorrect fix in the q validated ambiguities appear too likely. This would almost certainly cause $IR > \bar{IR}$, triggering an alert and rendering the solution useless.

A more precise assessment of position integrity in such a situation requires a different approach, one based on examination of the full *a posteriori* probabilities of both the correct fix and a large number of potential incorrect fixes. Denote the posterior fixing probability as

$$P\left(\mathcal{E}_{\boldsymbol{\zeta}}\,|\boldsymbol{\varepsilon}, i\right) \triangleq P(\Delta \boldsymbol{z} = \boldsymbol{\zeta}|\check{\boldsymbol{\epsilon}}_{\mathrm{c}} = \boldsymbol{\varepsilon}, q = i) \tag{3.42}$$

Note that, unlike (3.26), this expression is not conditioned on Z_i; i.e., it does not assume that validated fixes are correct. As argued earlier, under Z_i, only two alternative fixes need be considered to approximate the conditional baseline distribution as (3.26). But in operation, one does not know whether Z_i holds, and so must consider two alternatives *for each ambiguity*, assuming at each stage that the preceding

ambiguities were fixed correctly. Since GIAB's output $\check{\boldsymbol{z}}$ contains r ambiguities, at least $2^r - 1$ alternatives must be evaluated, or their probabilities, as determined by (3.42), bounded rigorously.

It is straightforward to derive an expression for $P\left(\mathcal{E}_{\boldsymbol{\zeta}} \,|\boldsymbol{\varepsilon}, i\right)$. Let $\{\boldsymbol{\zeta}_k\}$ be an ordered, indexed set of all members of $\mathbb{Z}^r$, with $\boldsymbol{\zeta}_0 \triangleq \mathbf{0}$. Then by the definition of conditional probability,

$$\begin{aligned} &P\left(\mathcal{E}_{\boldsymbol{\zeta}_k}|\boldsymbol{\varepsilon}, i\right) \\ &\quad = \lim_{d\boldsymbol{\varepsilon}\to 0} \frac{P(\boldsymbol{\varepsilon} - d\boldsymbol{\varepsilon} < \check{\boldsymbol{\epsilon}}_{\mathrm{c}} \leq \boldsymbol{\varepsilon}, \Delta\boldsymbol{z} = \boldsymbol{\zeta}_k|q = i)}{\sum_{\boldsymbol{\zeta}\in\mathbb{Z}^r} P(\boldsymbol{\varepsilon} - d\boldsymbol{\varepsilon} < \check{\boldsymbol{\epsilon}}_{\mathrm{c}} \leq \boldsymbol{\varepsilon}, \Delta\boldsymbol{z} = \boldsymbol{\zeta}_k|q = i)} \end{aligned} \tag{3.43}$$

Noting that the measured random variable $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \in \mathbb{R}^r$ and the fixing error $\Delta\boldsymbol{z} \in \mathbb{Z}^r$ are related to the zero-mean random variable $\boldsymbol{\epsilon}_{\mathrm{c}(1:r)} \triangleq L^{-1}_{1:r,1:r}\boldsymbol{\epsilon}_{1:r} \sim \mathcal{N}(\mathbf{0}, D_{1:r,1:r})$ as

$$\begin{aligned} \boldsymbol{\epsilon}_{\mathrm{c}(1:r)} &\triangleq L^{-1}_{1:r,1:r}\boldsymbol{\epsilon}_{1:r} = L^{-1}_{1:r,1:r}(\hat{\boldsymbol{z}}_{1:r} - \boldsymbol{z}_{1:r}) \\ &= L^{-1}_{1:r,1:r}(\hat{\boldsymbol{z}}_{1:r} - (\check{\boldsymbol{z}} - \Delta\boldsymbol{z})) \\ &= L^{-1}_{1:r,1:r}(\check{\boldsymbol{\epsilon}} + \Delta\boldsymbol{z}) \\ &= \check{\boldsymbol{\epsilon}}_{\mathrm{c}} + L^{-1}_{1:r,1:r}\Delta\boldsymbol{z} \end{aligned} \tag{3.44}$$

and noting that the additional conditioning on the event $q = i$ in (3.43) restricts the support of $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}$, but that this only affects the normalization of the PDF, not its form, then by recognizing equivalent events, (3.43) can be rewritten as

$$\begin{aligned} P\left(\mathcal{E}_{\boldsymbol{\zeta}_k}|\boldsymbol{\varepsilon}, i\right) =& \frac{\mathcal{N}\left(\boldsymbol{\varepsilon} + L^{-1}_{1:r,1:r}\boldsymbol{\zeta}_k; \mathbf{0}, D_{1:r,1:r}\right)}{\sum_{\boldsymbol{\zeta}\in\mathbb{Z}^r} \mathcal{N}\left(\boldsymbol{\varepsilon} + L^{-1}_{1:r,1:r}\boldsymbol{\zeta}; \mathbf{0}, D_{1:r,1:r}\right)} \\ =& \frac{\exp\left(-\frac{1}{2}\left\|\boldsymbol{\varepsilon} + L^{-1}_{1:r,1:r}\boldsymbol{\zeta}_k\right\|^2_{D_{1:r,1:r}}\right)}{\sum_{\boldsymbol{\zeta}\in\mathbb{Z}^r} \exp\left(-\frac{1}{2}\left\|\boldsymbol{\varepsilon} + L^{-1}_{1:r,1:r}\boldsymbol{\zeta}\right\|^2_{D_{1:r,1:r}}\right)} \end{aligned} \tag{3.45}$$

where $\|\boldsymbol{x}\|_Q^2 \triangleq \boldsymbol{x}^T Q^{-1} \boldsymbol{x}$. Note that (3.45) can be regarded as generalization of (3.26) to include additional alternative fix candidates.

The final conditional PDF for $\bar{\boldsymbol{b}}$ under MAP GIAB, but without conditioning on the correctness of the validated fixes, is a generalization of (3.33) that includes alternative fixes for more than just the rejected ambiguity, and for each fix, its corresponding position domain bias:

$$f_{\bar{\boldsymbol{b}}|\check{\boldsymbol{\epsilon}}_c,q}(\boldsymbol{\xi}|\boldsymbol{\varepsilon},i) = \sum_{\boldsymbol{\zeta}_k \in \mathbb{Z}^r} P(\mathcal{E}_{\boldsymbol{\zeta}_k}|\boldsymbol{\varepsilon},i)\, \mathcal{N}\left(\boldsymbol{\xi}; \boldsymbol{b} + \boldsymbol{\mu}_k, Q_{\check{\boldsymbol{b}}_{i+1}}\right) \tag{3.46}$$

Each term in the summation corresponds to the event that one of the infinite possible alternative fixes is correct, and accounts for the conditional PDI risk given that event. Each event's baseline remains normally distributed, but with additional mean error caused by the integer offset, as in (2.16):

$$\boldsymbol{\mu}_k = Q_{\hat{\boldsymbol{b}}\,\hat{\boldsymbol{z}}_c} D^{-1}_{1:r,1:r} L^{-1}_{1:r,1:r} \boldsymbol{\zeta}_k \tag{3.47}$$

A similar expression for $f_{\bar{\boldsymbol{b}}|\check{\boldsymbol{\epsilon}}_c,q}(\boldsymbol{\xi}|\boldsymbol{\varepsilon},i)$ can be obtained for float and MMSE GIAB. See Appendix D of [9] for details on how to determine which incorrect fixes must be accounted for in (3.46) and when to truncate the infinite summation in the denominator of (3.45). The maximum required set of size 2^r, including the IB solution, is obtained by considering the nearest two integers for each ambiguity in a branching tree of alternative solutions. In practice, far fewer than 2^r alternatives need be considered. This represents a significant reduction in computational effort when compared to EPIC.

To compare the relative computational effort of EPIC and GIAB for equivalent requirements, an implementation of EPIC was developed that included early

termination of the search once the integrity requirements are met. To further limit unnecessary computation, the alternative fixes are ordered to roughly minimize search time by sorting the alternatives by the metric $\Delta \boldsymbol{z}^T L^{-T} D^{-1} L^{-1} \Delta \boldsymbol{z}$. This metric is strongly correlated with the *a priori* probability that IB will produce the associated incorrect fix. The average execution time for EPIC and GIAB was compared for a range of models and integrity requirements. Depending on the number of alternative fixes considered by EPIC, GIAB showed between an $8\times$ and $64\times$ speed up over EPIC.

Let $s \leq 2^r - 1$ be the number of non-negligible alternative fixes considered and the index 0 represent the chosen fix. The IR can be bounded by the following expression, with R_k defined by (3.39) and with means defined by (3.47) for MAP GIAB:

$$R_{\text{GIAB}} = 1 - \sum_{k=0}^{s} (1 - R_k) \, P\left(\mathcal{E}_{\boldsymbol{\zeta}_k} | \boldsymbol{\varepsilon}, i\right) \tag{3.48}$$

3.7 Performance Analysis

3.7.1 Protection Levels

Integrity requirements are specified in terms of an integrity risk, $\bar{IR}$, that the baseline estimation error will exceed the AL threshold without warning. $\bar{IR}$ is derived from an overall risk requirement, such as probability of loss of aircraft, and is typically a fixed value for a given system use case. The AL is related to physical obstacle clearance requirements, which are constant for a particular land based runway and a given aircraft. However, obstacle clearance margins are not

constant when landing on a moving platform, such as an aircraft carrier at sea. Because the risk of excess error is frequently evaluated against a time-varying AL, it is useful to determine a protection level PL that bounds the estimation error to the required level of risk.

PL can be thought of as the minimum AL that could be met by a navigation system or algorithm for a given value of $\bar{IR}$. In terms of statistical hypothesis testing, $\bar{IR}$ corresponds to the desired confidence level, AL corresponds to the decision threshold, and PL to a prediction interval. If the risk of excess error is expressed as a function of AL, then PL can be defined as

$$PL \triangleq \min_{AL} \left\{ AL \,\middle|\, R\left(AL\right) \leq \bar{IR} \right\} \tag{3.49}$$

PL for EPIC or for any version of GIAB can be computed by using a root solving method to solve (3.49) with $R(AL)$ defined by (3.37) or (3.48), as appropriate.

3.7.2 Comparison to EPIC

To demonstrate the performance of GIAB compared to the state-of-the-art high-integrity algorithm, the performance of EPIC and MAP GIAB will be compared for the measurement models previously examined. MAP GIAB is chosen because it provides better accuracy than float GIAB and is simpler to analyze than MMSE GIAB. If MMSE GIAB were used, it would compare even more favorably with EPIC because MMSE GIAB always produces smaller PL values than MAP GIAB.

Because it uses an *a priori*, model driven approach to validation, EPIC will always produce the same PL for the same number of integers fixed with a given measurement model and $\bar{IR}$. Conversely, GIAB is a data-driven algorithm for which PL is a random variable. PL has a finite support because it is driven by $|\breve{\epsilon}_{ci}| \leq \frac{1}{2}$.

PL values produced by EPIC will be compared to the minimum, maximum, and average PL produced by GIAB for each number of integers fixed, along with the probability that GIAB will fix that number of integers for each model considered. As shown in Table 3.2, GIAB is able to provide smaller PL values than EPIC most of the time. Note that PL computed for U by EPIC is simply the PL of the float solution with no incorrect fixing bias. The event U for GIAB corresponds to the case where the measurements are so poor that no integers can be fixed successfully. The worst case PL computed for any $q > 0$ by MAP GIAB, which has the largest PLs of any of the GIAB implementations, is better than the best PL computed by EPIC.

GIAB provides lower PLs because it is able to reject and exclude most of the incorrect fixes that EPIC must protect against. This implies that GIAB will also provide superior availability of integrity for models similar to those examined in this paper. It is expected that this will be the case in general because the *a posteriori* alternate candidate fix used in GIAB will virtually always be among the candidates considered *a priori* by EPIC. This implies that any decrease in PL computed by EPIC as compared to GIAB will result only when the incorrect fixing bias of the GIAB alternative fix is the same as the largest incorrect fixing bias considered by EPIC, which will be a rare event.

Table 3.1: Integrity performance comparison between EPIC and MAP GIAB. All distance units are in meters. The left-most column indicates the result from GIAB. The next column indicates it's theoretical probability of occurrence [11]. The third column is the standard deviation of the $\bar{b}$ under the given GIAB event. For EPIC, the standard deviations from the previous row applies. The next three columns indicate the minimum, average and maximum PL produced by GIAB under each event. The final column is the value of PL produced by EPIC when it fixes the same number of integers as GIAB.

$\mathcal{E}$	$P_{\mathcal{E}}$	$\sigma\,(m)$	$PL_{\min}$	$\mathbb{E}\,[PL]$	$PL_{\max}$	PL_{EPIC}
U	0.00072	0.310	2.79	2.92	3.35	1.77
S_1	0.00043	0.221	1.31	1.32	1.37	2.62
S_2	0.00051	0.196	1.63	1.74	2.13	2.61
S_3	0.00119	0.138	1.75	1.83	2.09	2.49
S_4	0.00090	0.115	1.09	1.16	1.38	3.03
S_5	0.00189	0.084	0.99	1.04	1.20	2.91
S_6	0.00039	0.081	0.48	0.53	0.69	2.99
S_7	0.99393	0.081	0.46	0.46	0.46	2.95

Table 3.2: Integrity performance comparison between EPIC and MAP GIAB for a weak model. Columns are the same as the prior table. Even for a weak model, GIAB usually provides a lower AL than EPIC with superior accuracy. In the case that GIAB rejects all fixes, the computed PL represents the poor quality of the measurements.

$\mathcal{E}$	$P_{\mathcal{E}}$	$\sigma(m)$	$PL_{\min}$	$\mathbb{E}[PL]$	$PL_{\max}$	PL_{EPIC}
U	0.138	0.465	2.60	2.93	3.57	2.03
S_1	0.093	0.332	1.54	1.56	1.59	2.86
S_2	0.084	0.294	1.51	1.78	2.33	2.86
S_3	0.099	0.207	1.68	1.87	2.24	2.91
S_4	0.071	0.173	1.05	1.20	1.50	3.26
S_5	0.080	0.126	0.97	1.08	1.29	3.14
S_6	0.033	0.121	0.57	0.60	0.77	3.31
S_7	0.398	0.121	0.54	0.54	0.54	3.33

Note that PL values computed by GIAB and EPIC do not increase or decrease uniformly with the number of validated fixes. For example, the maximum PL increases from 1.37 m to 2.13 m when transitioning from the first successful fix to the second successful fix. Recall that PL is driven primarily by the bias between the most likely fix and the incorrect fixes of non-negligible probability. Because these biases depend on the relationships among the various integers and the baseline directions of interest (e.g. vertical error), the biases can change dramatically from one integer fix to the next.

It is tempting to think that in the case of successfully fixing only two integers in the example above would be better to only fix one integer because that would yield a lower protection level. It may in fact be preferable to do so, but only if the bias induced in the solution by leaving the second integer floating produces acceptable accuracy performance. That is, the reduction in PL is obtained only at the expense of a biased solution that degrades average accuracy. The impact to average accuracy can be seen in the difference between the strength of the central modes of Float and MAP GIAB in Fig. 3.3.

3.8 Conclusions

A new data-driven CDGNSS PAR and validation algorithm has been developed analytically and validated with Monte Carlo simulation. The new algorithm has advantages over the state-of-the-art in that (1) data-driven methods offer improved availability of integrity over model-driven methods such as EPIC, (2) the integrity risk due to incorrect fixing is precisely controlled analytically as compared to

functional approximation methods used with the ratio test and similar integer aperture methods, (3) it correctly accounts for the integrity risk of PAR in the position domain that existing GIA methods neglect, and (4) it requires less computational burden than EPIC because it eliminates the search for many alternate fix candidates. The new algorithm has been shown to provide superior performance to the current state-of-the-art method for a range of measurement models.

Chapter 4

Correlation-Aware Integrity Protection for Fault-Free Federated Triplex Carrier Differential GNSS Solutions

4.1 Introduction

[1] Hardware redundancy is one straight forward approach to satisfying the strict requirements of safety-of-life applications: the CDGNSS solutions from multiple rover receivers with separate antennas and independent multipath can be combined to improve availability for a given set of requirements. This chapter's focus is on the popular federated triplex architecture, where each of three rover receivers computes its own solution using a common set of reference measurements. This is in contrast to an integrated triplex architecture in which measurements from all rover receivers are processed together to yield a single CDGNSS solution. The resulting solutions may be combined by averaging or by mid-level voting (MLV), which selects the median of the three values. MLV is often the preferred approach since it is more robust than averaging to single solution faults.

[1]This chapter was based on

1. G. Nathan Green, Martin King, and Todd E. Humphreys. Fault free integrity of mid-level voting for triplex differential GPS solutions. In *Proceedings of the ION GNSS+ Meeting*, Tampa, FL, 2015.

The author contributed all derivation, simulation, and analysis in that paper and in this chapter.

MLV architectures are commonly used and well understood for components that are corrupted by independent errors [20]. But when subsystem outputs have correlated errors, the performance of MLV depends on the degree of correlation. Because the three CDGNSS solutions use common reference receivers, they will be correlated due to common reference receiver noise and multipath. Additional correlations arise from common atmospheric errors and common survey errors. If the three solutions are treated as independent, then the IR will be computed optimistically, exposing the system to unacceptable risk. If the three solutions are assumed to be perfectly correlated; i.e., no integrity credit is taken in the fault free case for the fact that there are three solutions, then the potential integrity benefit of MLV triplex solutions is squandered, reducing availability.

There is a large body of work on validation of CDGNSS solutions [4, 5, 27, 29, 32, 34, 37, 41, 43]. The theory of integer bootstrapping is used to predict the probability that the correct integer ambiguity is found. The implications of incorrect ambiguity resolution on IR are accounted for in the Geometry Extra Redundant almost-fixed Solutions (GERAFS) [42] and the Enforced Position-domain Integrity-risk of Cycle resolution (EPIC) [17, 18] algorithms. Moreover, previous work has analyzed the risk posed by reference receiver faults [19]. But, to the best of the authors' knowledge, there exists no prior work on IR evaluation for federated triplex CDGNSS with full treatment of correlation for multiple rover receivers.

This chapter makes three contributions to account for the integrity implications of MLV for CDGNSS solutions. First, expressions are obtained for MLV algorithm IR for arbitrarily distributed and for Gaussian distributed solutions. Second,

this theory is applied to float solutions, fixed solutions, and almost-fixed solutions. Third, the performance improvements provided by MLV for each type of solution are described theoretically and demonstrated via a world-wide covariance simulation. Results of the simulation indicate that MLV provides significant integrity and accuracy improvements for triplex float solutions when compared to simplex float solutions.

4.2 Fault-Free Integrity for Mid-Level Voting with Arbitrary Distributions

4.2.1 Integrity Specification Parameters

A few preliminary definitions are needed before MLV can be evaluated. First, IR is defined as the probability that the error in the relative navigation solution exceeds a threshold, called an alert limit (AL), without a warning. The algorithms considered in this chapter monitor IR for fault free performance *a priori* based on models that have been validated to bound the errors in the actual system [7]. Since these monitors operate *a priori*, they are deterministic values for a given satellite geometry, hardware configuration, measurement set, and carrier phase track duration.

Stated symbolically, $IR = P\left((|\epsilon| \geq AL) \wedge \bar{W}\right)$, where ϵ is the solution error, and $\bar{W}$ is the event that no warning is given. If there exists a function of the measurement models R_{mon} such that $R_{\text{mon}} \geq P\left(|\epsilon| \geq AL\right)$, then a warning can be given when $R_{\text{mon}} \geq IR_{\text{spec}}$, where IR_{spec} is a specified tolerable level of IR. Given such a monitor, $IR \leq IR_{\text{spec}}$.

4.2.2 Derivation of MLV Integrity Risk

Consider a random vector $\boldsymbol{X} = [X_1, X_2, X_3]^T$ of three distinct estimates of a scalar parameter, x. Let $\boldsymbol{X}$ be corrupted by errors $\boldsymbol{\epsilon}$ with any well defined joint probability density function, $f_{\boldsymbol{\epsilon}}(\boldsymbol{\varepsilon})$. The notation $X_{(i)}$ indicates the i^{th} order statistic of $\boldsymbol{X}$, sorted from least to greatest.

The use of a MLV algorithm ensures that the final estimate of x is $X_{(2)}$ with error $\epsilon_{(2)} = X_{(2)} - x$. To protect the integrity of the navigation solution, a monitor function is needed such that

$$R_{\text{mon}} \geq R_{\text{MLV}} = P\left(\left|\epsilon_{(2)}\right| \geq AL\right) \tag{4.1}$$

The probability can be recast in terms of the original random variables by recognizing that

$$R_{\text{MLV}} = P\left(\epsilon_{(2)} \geq AL\right) + P\left(\epsilon_{(2)} \leq -AL\right) \tag{4.2}$$

and that $\epsilon_{(2)} \geq AL$ if any two or more of $\epsilon_i \geq AL$. Define A_j as the event that the two estimates other than X_j have large positive errors, with no constraint on the errors in X_j. B_j is the similar event, but with large negative errors.

$$A_j = \{\epsilon_i \geq AL, \epsilon_k \geq AL\} \tag{4.3a}$$

$$B_j = \{\epsilon_i \leq AL, \epsilon_k \leq AL\} \tag{4.3b}$$

with i, j, k distinct elements of $\{1, 2, 3\}$.

The event that the MLV estimate has a large positive error is the union of the events A_j:

$$\begin{aligned}\left\{\epsilon_{(2)} \geq AL\right\} &= \cup_{j=1}^{3} \left\{\epsilon_i \geq AL, \epsilon_k \geq AL\right\} \\ &= \cup_{j=1}^{3} \left(A_j\right)\end{aligned} \tag{4.4}$$

Substituting (4.4) into (4.2) with similar logic for B_j, and recognizing that $A_j \cap B_j = \emptyset$:

$$\begin{aligned}R_{\text{MLV}} =& P\left(\cup_{j=1}^{3} A_j\right) + P\left(\cup_{j=1}^{3} B_j\right) \\ =& \sum_{j=1}^{3} \left[P\left(A_j\right) - P\left(A_i \cap A_k\right)\right] + P\left(\cap_{j=1}^{3} A_j\right) \\ &+ \sum_{j=1}^{3} \left[P\left(B_j\right) - P\left(B_i \cap B_k\right)\right] + P\left(\cap_{j=1}^{3} B_j\right)\end{aligned} \tag{4.5}$$

For the events A_j, the intersections of any two or more such events are equivalent, i.e.

$$\begin{aligned}\cap_{j=1}^{3} A_j &= A_i \cap A_j = A_i \cap A_k = A_k \cap A_j \\ &= \left\{\epsilon_i \geq AL, \epsilon_j \geq AL, \epsilon_k \geq AL\right\}\end{aligned} \tag{4.6}$$

Substituting (4.6) into (4.5), with similar logic for B_j, and collecting like terms yields:

$$\begin{aligned}R_{\text{MLV}} =& \sum_{j=1}^{3} \left[P\left(A_j\right) + P\left(B_j\right)\right] \\ &- 2P\left(\cap_{j=1}^{3} A_j\right) - 2P\left(\cap_{j=1}^{3} B_j\right)\end{aligned} \tag{4.7}$$

If $\boldsymbol{\epsilon}$ has probability density function $f_{\boldsymbol{\epsilon}}\left(\boldsymbol{\varepsilon}\right)$, R_{MLV} is defined by integrals over the appropriate volumes.

$$R_{\text{MLV}} = \sum_{j=1}^{3} \left[\iiint_{V_j} f_{\boldsymbol{\epsilon}}\left(\boldsymbol{\varepsilon}\right) d\boldsymbol{\varepsilon}\right] - 2 \iiint_{V_{\text{int}}} f_{\boldsymbol{\epsilon}}\left(\boldsymbol{\varepsilon}\right) d\boldsymbol{\varepsilon} \tag{4.8}$$

with volumes defined as

$$\begin{aligned} V_j &\triangleq A_j \cup B_j = \{|\epsilon_i| \geq AL, |\epsilon_k| \geq AL\} \\ V_{\text{int}} &\triangleq \left(\cap_{j=1}^{3} A_j\right) \cup \left(\cap_{j=1}^{3} B_j\right) \\ &= \{\boldsymbol{\epsilon} \succeq AL \cdot \mathbf{1}\} \cup \{\boldsymbol{\epsilon} \preceq -AL \cdot \mathbf{1}\} \end{aligned}$$

where $\preceq$ and $\succeq$ are vector inequalities and $\mathbf{1}$ is a vector of all ones.

Equation (4.3) shows that the individual events A_j and B_j do not depend on the value of ϵ_j, so the MLV risk integrals can be carried out on the marginal distributions for the V_j regions.

$$\begin{aligned} R_{\text{MLV}} = &\sum_{j=1}^{3} \iint_{AL}^{\infty} f_{\epsilon_i,\epsilon_k}(\varepsilon_i, \varepsilon_k) d\varepsilon_i d\varepsilon_k \\ &+ \sum_{j=1}^{3} \iint_{-\infty}^{-AL} f_{\epsilon_i,\epsilon_k}(\varepsilon_i, \varepsilon_k) d\varepsilon_i d\varepsilon_k \\ &- 2 \iiint_{AL}^{\infty} f_{\boldsymbol{\epsilon}}(\boldsymbol{\varepsilon})\, d\boldsymbol{\varepsilon} - 2 \iiint_{-\infty}^{-AL} f_{\boldsymbol{\epsilon}}(\boldsymbol{\varepsilon})\, d\boldsymbol{\varepsilon} \end{aligned} \tag{4.9}$$

In the case that the underlying random variables are *iid* zero-mean Gaussian, the above risk is greatly simplified by factoring the joint distributions into their independent parts and expressing the result in terms of the PDF of the standard normal distribution, $\phi(\cdot)$. Let σ be the standard deviation of ϵ_j:

$$\begin{aligned} R_{\text{MLV}} = &\, 3 \left[\left(\int_{-\infty}^{-\frac{AL}{\sigma}} \phi(\zeta)\, d\zeta \right)^2 + \left(\int_{\frac{AL}{\sigma}}^{\infty} \phi(\zeta)\, d\zeta \right)^2 \right] \\ &- 2 \left[\left(\int_{-\infty}^{-\frac{AL}{\sigma}} \phi(\zeta)\, d\zeta \right)^3 + \left(\int_{\frac{AL}{\sigma}}^{\infty} \phi(\zeta)\, d\zeta \right)^3 \right] \end{aligned} \tag{4.10}$$

Define the simplex risk as

$$R_{\text{simplex}} \triangleq \int_{-\infty}^{-\frac{AL}{\sigma}} \phi(\zeta)\, d\zeta + \int_{\frac{AL}{\sigma}}^{\infty} \phi(\zeta)\, d\zeta = 2\int_{\frac{AL}{\sigma}}^{\infty} \phi(\zeta)\, d\zeta \tag{4.11}$$

Substituting (4.11) into (4.10) yields

$$\begin{aligned} R_{\text{MLV}} =& 3\left[\left(\frac{R_{\text{simplex}}}{2}\right)^2 + \left(\frac{R_{\text{simplex}}}{2}\right)^2\right] \\ & - 2\left[\left(\frac{R_{\text{simplex}}}{2}\right)^3 + \left(\frac{R_{\text{simplex}}}{2}\right)^3\right] \\ =& \frac{3}{2}R_{\text{simplex}}^2\left(1 - \frac{R_{\text{simplex}}}{3}\right) \\ \approx& \frac{3}{2}R_{\text{simplex}}^2 \end{aligned} \tag{4.12}$$

Alternatively, if the solutions are completely correlated Gaussian random variables with equal variance, then the conditional distributions are degenerate, and the multiple integrals over the joint distributions reduce to single integrals over a single variable:

$$\begin{aligned} R_{\text{MLV}} =& 3\left(\int_{-\infty}^{-\frac{AL}{\sigma}} \phi(\zeta)\, d\zeta + \int_{\frac{AL}{\sigma}}^{\infty} \phi(\zeta)\, d\zeta\right) \\ & - 2\left(\int_{-\infty}^{-\frac{AL}{\sigma}} \phi(\zeta)\, d\zeta + \int_{\frac{AL}{\sigma}}^{\infty} \phi(\zeta)\, d\zeta\right) \\ =& R_{\text{simplex}} \end{aligned} \tag{4.13}$$

This implies that perfectly correlated solutions yield the same IR as a simplex solution of the same quality. It is important to note that the actual IR will vary as a

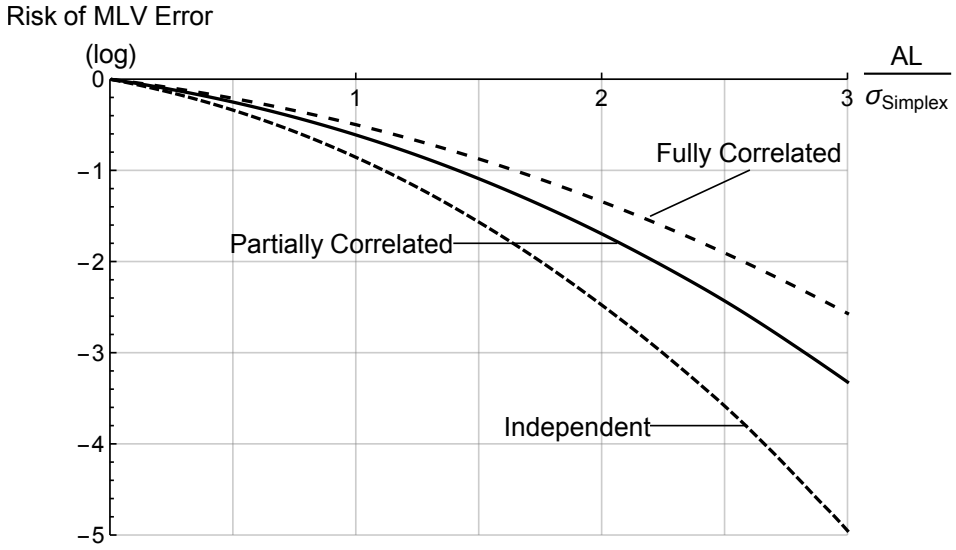

Figure 4.1: The risk of excess error for a MLV solution is a function of the correlations among the three solutions. When the solutions are fully correlated, the risk is equivalent to that of a simplex solution. When the solutions are independent, the risk is greatly reduced. The example of partially correlated errors shown is for a correlation coefficient $\rho = 0.5$ between each pair among the three solutions.

function of the degree of correlation among the three solutions. The actual risk is bounded below by equation (4.12) and bounded above by equation (4.13). The importance of accounting for the correlations among solutions when computing MLV IR is illustrated in Fig. 4.1. The partially correlated result plotted has correlation coefficients of 0.5 among the three solutions, which have equal variance. If partially correlated solutions are treated as fully correlated, then the result is overly conservative, and availability will degrade. If the solutions are incorrectly assumed to be independent, then IR is not being bounded correctly.

4.3 Correlation Agnostic Integrity Risk Bound

In the case that there is insufficient knowledge of the correlations to accurately model the joint distribution of the three solutions, a simple *a priori* bound can be computed for the selected solution based upon the MLV criteria and the simplex IRs of the individual solutions without regard for their correlations:

$$\begin{aligned} R_{\mathrm{MLV}} &= P\left(\left|\epsilon_{(2)}\right| > AL\right) \\ R_{\mathrm{MLV}} &\leq 1 - P\left(\left|\epsilon_{(1)}\right| < AL, \left|\epsilon_{(3)}\right| < AL\right) \end{aligned} \tag{4.14}$$

Take $R_j = P\left(|\epsilon_j| > AL\right)$. Let i to be the MLV solution in use and j and k to be the other two solutions. A bound on R_{MLV} can be obtained independent of the distribution of X_i:

$$\begin{aligned} R_{\mathrm{MLV}} &\leq 1 - \int_{-AL}^{AL}\int_{-AL}^{AL} f_{\epsilon_j,\epsilon_k}\left(\xi,\zeta\right) d\zeta d\xi \\ &\leq R_j + R_k \end{aligned} \tag{4.15}$$

This upper bound for R_{MLV}, which does not depend on the correlation among the solutions, is derived by in Appendix E.

This demonstrates that even if a single one of the solutions has poor integrity, it can be assured that the IR from the selected MLV solution can be limited to the sum of the IRs of the other two solutions. This is of no use when all three solutions are of similar quality, so it provides little benefit under nominal circumstances for fault free integrity. The bound can, however, be used to protect the integrity of the selected MLV solution in the presence of a single latent fault or a single large IR.

4.4 Application of Mid-Level Voting to Carrier Phase Differential GNSS Solutions

To apply the analytical tools developed in Section 4.2 to triplex CDGNSS solutions, the joint distributions of the three solutions must be derived for all solutions of interest. This chapter will examine the float solution, the fixed solution, and the GERAFS algorithm [42]. The EPIC algorithm [18] is not evaluated. As will be shown, it is not appropriate to consider the joint probabilities of ambiguity resolution among the federated solutions.

4.4.1 Simplex Float Solution Joint Distribution

Each individual float solution is formed by solving the linearized, least squares, double-difference relative baseline solution. The solution is formed by linearizing the measurement model about an initial estimate of the baseline vector, $\bar{\underline{b}}$. The measurements comprise a set of carrier phase observables and some prior information, such as pseudoranges or geometry free estimates of the integer ambiguities. To form a weighted least squares solution, the joint covariance of the measurements and the prior information must be known. For the case that the prior information is a set of smoothed pseudoranges, the linearized measurement model for the i^{th} federated solution is

$$\boldsymbol{y}_i = \underbrace{\begin{bmatrix} \mathbf{G}_i & \boldsymbol{\Lambda}_i \\ \mathbf{G}_i & 0 \end{bmatrix}}_{\mathbf{H}_i} \underbrace{\begin{bmatrix} \delta \boldsymbol{b} \\ \boldsymbol{a}_i \end{bmatrix}}_{\boldsymbol{x}_i} + \boldsymbol{\varepsilon}_i \tag{4.16}$$

with

$\boldsymbol{\varepsilon}_i \sim \mathcal{N}(\mathbf{0}, \Sigma_i)$

Σ_i: Measurement covariance matrix

$$\mathbf{G}_i = \begin{bmatrix} \left(\boldsymbol{u}_i^1 - \boldsymbol{u}_i^l\right)^T \\ \vdots \\ \left(\boldsymbol{u}_i^{m_i} - \boldsymbol{u}_i^l\right)^T \end{bmatrix}$$

$\boldsymbol{u}_i^k$: unit vector from space vehicle (SV) k to receiver i

l: key SV used in double differences

m_i: number of double differences

$\boldsymbol{\Lambda} = Diag\begin{pmatrix} \lambda_1 & \dots & \lambda_{m_i} \end{pmatrix}$ wavelength matrix

Each float solution is computed and the solution matrix, $\mathbf{S}_i$, is retained for later use:

$$\begin{aligned} \hat{\boldsymbol{x}}_i &= \begin{bmatrix} \delta\hat{\boldsymbol{b}}_i \\ \\ \hat{\boldsymbol{a}}_i \end{bmatrix} = \left(\mathbf{H}_i^T \Sigma_i^{-1} \mathbf{H}_i\right)^{-1} {\mathbf{H}_i}^T \Sigma_i^{-1} \boldsymbol{y}_i = \mathbf{S}_i \boldsymbol{y}_i \\ \hat{\boldsymbol{b}}_i &= \bar{\boldsymbol{b}} + \delta\hat{\boldsymbol{b}}_i \ : \text{ Final baseline estimate} \end{aligned} \tag{4.17}$$

4.4.2 Triplex Float Solution Joint Distribution

Each of these float solution matrices is combined with the total system joint measurement covariance to form the total solution covariance:

$$\begin{aligned}
\Sigma_{\text{float,triplex}} &= \text{COV}\left(\left[\hat{\boldsymbol{b}}_1\,\hat{\boldsymbol{a}}_1\middle|\hat{\boldsymbol{b}}_2\,\hat{\boldsymbol{a}}_2\middle|\hat{\boldsymbol{b}}_3\,\hat{\boldsymbol{a}}_3\right]^T\right) \\
&= \begin{bmatrix} \mathbf{S}_1 & 0 & 0 \\ 0 & \mathbf{S}_2 & 0 \\ 0 & 0 & \mathbf{S}_3 \end{bmatrix} \begin{bmatrix} \Sigma_{y,1} & \Sigma_{y,2,1}^T & \Sigma_{y,3,1}^T \\ \Sigma_{y,2,1} & \Sigma_{y,2} & \Sigma_{y,3,2}^T \\ \Sigma_{y,3,1} & \Sigma_{y,3,2} & \Sigma_{y,3} \end{bmatrix} \begin{bmatrix} \mathbf{S}_1 & 0 & 0 \\ 0 & \mathbf{S}_2 & 0 \\ 0 & 0 & \mathbf{S}_3 \end{bmatrix}^T \\
&= \begin{bmatrix} \mathbf{S}_1\Sigma_{y,1}\mathbf{S}_1^T & \mathbf{S}_1\Sigma_{y,2,1}^T\mathbf{S}_2^T & \mathbf{S}_1\Sigma_{y,3,1}^T\mathbf{S}_3^T \\ \mathbf{S}_2\Sigma_{y,2,1}\mathbf{S}_1^T & \mathbf{S}_2\Sigma_{y,2}\mathbf{S}_2^T & \mathbf{S}_2\Sigma_{y,3,2}^T\mathbf{S}_3^T \\ \mathbf{S}_3\Sigma_{y,3,1}\mathbf{S}_1^T & \mathbf{S}_3\Sigma_{y,3,2}\mathbf{S}_2^T & \mathbf{S}_3\Sigma_{y,3}\mathbf{S}_3^T \end{bmatrix}
\end{aligned} \tag{4.18}$$

The off-block-diagonal terms of the measurement covariance matrix are populated by reference receiver errors, atmospheric errors, and lever arm errors used to translated the three solutions to a common reference point. The resulting triplex covariance can be broken down into several pieces: covariance among the various baseline estimates $\left(\Sigma_{b_i b_j}\right)$, among the various real-valued float integer estimates $\left(\Sigma_{N_i N_j}\right)$, and among the baseline estimates and the integer estimates $(\Sigma_{b_i N_i})$. This decomposition of the matrix is denoted as follows, with the elided upper triangular

portion being the transpose of the lower triangular portion shown:

$$\left[\begin{array}{cc|cc|cc} \Sigma_{\hat{\boldsymbol{b}}_1} & & & & & \\ \Sigma_{\hat{\boldsymbol{a}}_1\hat{\boldsymbol{b}}_1} & \Sigma_{\hat{\boldsymbol{a}}_1} & & & & \\ \hline \Sigma_{\hat{\boldsymbol{b}}_2\hat{\boldsymbol{b}}_1} & \Sigma_{\hat{\boldsymbol{b}}_2\hat{\boldsymbol{a}}_1} & \Sigma_{\hat{\boldsymbol{b}}_2} & & & \\ \Sigma_{\hat{\boldsymbol{a}}_2\hat{\boldsymbol{b}}_1} & \Sigma_{\hat{\boldsymbol{a}}_2\hat{\boldsymbol{a}}_1} & \Sigma_{\hat{\boldsymbol{a}}_2\hat{\boldsymbol{b}}_2} & \Sigma_{\hat{\boldsymbol{a}}_2} & & \\ \hline \Sigma_{\hat{\boldsymbol{b}}_3\hat{\boldsymbol{b}}_1} & \Sigma_{\hat{\boldsymbol{b}}_3\hat{\boldsymbol{a}}_1} & \Sigma_{\hat{\boldsymbol{b}}_3\hat{\boldsymbol{b}}_2} & \Sigma_{\hat{\boldsymbol{b}}_3\hat{\boldsymbol{a}}_2} & \Sigma_{\hat{\boldsymbol{b}}_3} & \\ \Sigma_{\hat{\boldsymbol{a}}_3\hat{\boldsymbol{b}}_1} & \Sigma_{\hat{\boldsymbol{a}}_3\hat{\boldsymbol{a}}_1} & \Sigma_{\hat{\boldsymbol{a}}_3\hat{\boldsymbol{b}}_2} & \Sigma_{\hat{\boldsymbol{a}}_3\hat{\boldsymbol{a}}_2} & \Sigma_{\hat{\boldsymbol{a}}_3\hat{\boldsymbol{b}}_3} & \Sigma_{\hat{\boldsymbol{a}}_3} \end{array}\right] \tag{4.19}$$

By extracting the desired components from this overall joint solution covariance matrix, MLV can be performed on the vertical and lateral components of the relative baseline solution.

4.4.3 Simplex Fixed Solution Integrity

Many methods exist to fix the integer ambiguities. Among these are integer rounding [29], integer bootstrap [32], and integer least squares (ILS) [34]. Each algorithm has its advantages and disadvantages. Integer rounding is the simplest, but it has the lowest probability of correctly fixing the integers. Bootstrap has improved probability of success and has a convenient way to predict probability of correct fix, $P_{CF} \triangleq P\left(\breve{\boldsymbol{a}} = \boldsymbol{a}\right)$, but this probability is suboptimal and is sensitive to the order and combination in which ambiguities are resolved. ILS is an optimal method in terms of P_{CF}, but entails a grid search of the integer space and has no analytical expression for P_{CF}.

High integrity CDGNSS systems typically use the bootstrap method together with the ambiguity decorrelation adjustment of the LAMBDA method. By using the decorrelated ambiguities, the bootstrap algorithm fixes successive integer ambiguities in the order of maximum conditional P_{CF}. This set of algorithms provides high integrity with predictable probability of correct integer fixing.

Due to the nonlinear rounding operation, the fixed baseline solution does not havea Gaussian distribution. Rather, it is a mixture of multivariate Gaussian distributions of equal covariance, but differing means [32]. This distribution is illustrated in Fig. 4.2 and described mathematically in (4.20). Let the accent mark ($\check{}$) denote a quantity conditioned on the fixed integer estimate $\check{\boldsymbol{a}}$.

$$\check{\boldsymbol{b}} = \hat{\boldsymbol{b}} - \Sigma_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}}\Sigma_{\hat{\boldsymbol{a}}}^{-1}\left(\hat{\boldsymbol{a}} - \check{\boldsymbol{a}}\right) \tag{4.20a}$$

$$\Sigma_{\check{\boldsymbol{b}}} = \Sigma_{\hat{\boldsymbol{b}}} - \Sigma_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}}\Sigma_{\hat{\boldsymbol{a}}}^{-1}\Sigma_{\hat{\boldsymbol{a}}\hat{\boldsymbol{b}}} \tag{4.20b}$$

$$\check{\boldsymbol{b}} \sim \sum_{\boldsymbol{z}\in\mathbb{Z}^m} P\left(\check{\boldsymbol{a}} = \boldsymbol{a} + \boldsymbol{z}\right)\mathcal{N}\left(\boldsymbol{b} + \Sigma_{\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}}\Sigma_{\hat{\boldsymbol{a}}}^{-1}\boldsymbol{z}, \Sigma_{\check{\boldsymbol{b}}}\right) \tag{4.20c}$$

The probability $P\left(\check{\boldsymbol{a}} = \boldsymbol{a} + \boldsymbol{z}\right)$ is a deterministic function of $\Sigma_{\hat{\boldsymbol{a}}}$ and the decorrelating transform used [32]. Conditioned on the event that $\check{\boldsymbol{a}} = \boldsymbol{a}$, the baseline estimate is once again a zero-mean multivariate Gaussian random vector. The IR in making this assumption is $P_{IF} \triangleq 1 - P_{CF}$. The covariance of the fixed solution is smaller than the float covariance since the errors are driven by the carrier phase measurements rather than by the pseudoranges. The IR associated with the vertical component of a fixed solution is bounded by the following:

$$\begin{aligned} R_{\text{fixed,V}} &= P\left(\left|\check{b}_{\text{V}} - b_{\text{V}}\right| > AL_{\text{V}}\right) \\ &\leq P_{IF} + P_{CF}P\left(\left|\check{b}_{\text{V}} - b_{\text{V}}\right| > AL_{\text{V}} | \check{\boldsymbol{a}} = \boldsymbol{a}\right) \end{aligned} \tag{4.21}$$

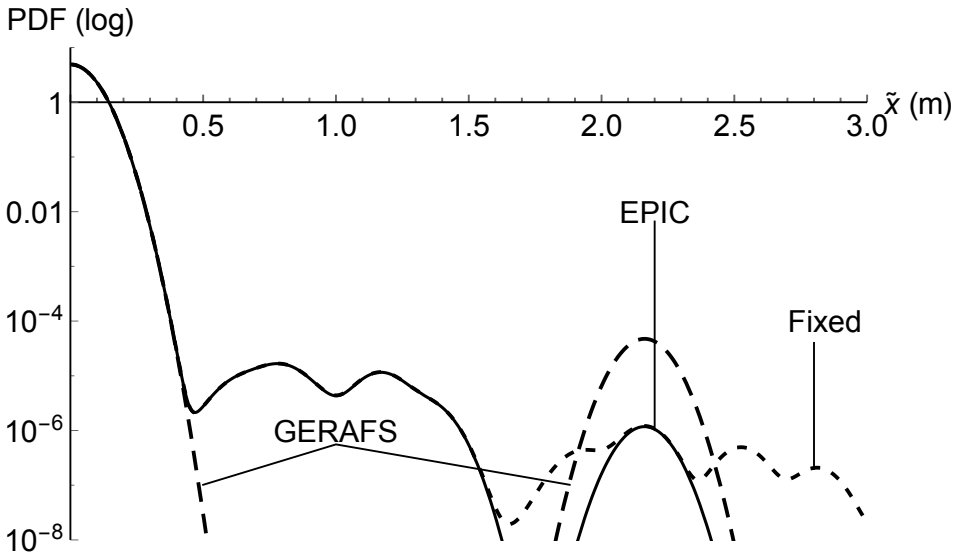

Figure 4.2: The positive portion of the modeled PDFs of the fixed, GERAFS, and EPIC baseline errors. The fixed baseline has a Gaussian mixture distribution with biases and weights determined by the incorrect fixes. Whereas the EPIC baseline follows the fixed baseline closely for incorrect fixes of non-negligible probability, the GERAFS baseline assigns all of the non-negligible probability of incorrect fix to the fix with the largest incorrect fixing bias.

4.4.4 Triplex Fixed Solution Joint Distribution

The transition from simplex to triplex for the float solution is simply a matter of extracting the appropriate portions of the system float covariance matrix to form the baseline covariance, but the fixed triplex solution is more complicated. Consider the triplex fixed baseline covariance:

$$\Sigma_{\breve{\boldsymbol{B}},\text{triplex}} = \begin{bmatrix} \Sigma_{\hat{\boldsymbol{b}}_1|\hat{\boldsymbol{a}}_1} & & \\ \Sigma_{\hat{\boldsymbol{b}}_2|\breve{\boldsymbol{a}}_2,\hat{\boldsymbol{b}}_1|\breve{\boldsymbol{a}}_1} & \Sigma_{\hat{\boldsymbol{b}}_2|\breve{\boldsymbol{a}}_2} & \\ \Sigma_{\hat{\boldsymbol{b}}_3|\breve{\boldsymbol{a}}_3,\hat{\boldsymbol{b}}_1|N_1} & \Sigma_{\hat{\boldsymbol{b}}_3|\breve{\boldsymbol{a}}_3,\hat{\boldsymbol{b}}_2|N_2} & \Sigma_{\hat{\boldsymbol{b}}_3|\breve{\boldsymbol{a}}_3} \end{bmatrix} \tag{4.22}$$

Where the off diagonal terms are computed from parts of the system float covariance and the block diagonal terms are the results of the individual fixed solutions. Take $\check{\boldsymbol{\epsilon}}_i = (\hat{\boldsymbol{a}}_i - \check{\boldsymbol{a}}_i)$ and assume $\check{\boldsymbol{a}}_i = \boldsymbol{a}_i$. Defining $\delta\hat{\boldsymbol{b}} \triangleq \hat{\boldsymbol{b}} - \boldsymbol{b}$, the off diagonal blocks can be computed as

$$\begin{aligned}
&\Sigma_{\hat{\boldsymbol{b}}_i|\check{\boldsymbol{a}}_i,\hat{\boldsymbol{b}}_j|\check{\boldsymbol{a}}_j} = \mathbb{E}\left[\left(\check{\boldsymbol{b}}_i - \boldsymbol{b}\right)\left(\check{\boldsymbol{b}}_j - \boldsymbol{b}\right)^T\right] \\
=&\mathbb{E}\left[\left(\delta\hat{\boldsymbol{b}}_i - \Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{a}}_i}\Sigma_{\hat{\boldsymbol{a}}_i}^{-1}\check{\boldsymbol{\epsilon}}_i\right)\left(\delta\hat{\boldsymbol{b}}_j - \Sigma_{\hat{\boldsymbol{b}}_j\hat{\boldsymbol{a}}_j}\Sigma_{\hat{\boldsymbol{a}}_j}^{-1}\check{\boldsymbol{\epsilon}}_j\right)^T\right] \\
=&\mathbb{E}\left[\delta\hat{\boldsymbol{b}}_i\delta\hat{\boldsymbol{b}}_j^T\right] + \mathbb{E}\left[\Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{a}}_i}\Sigma_{\hat{\boldsymbol{a}}_i}^{-1}\check{\boldsymbol{\epsilon}}_i\left(\Sigma_{\hat{\boldsymbol{b}}_j\hat{\boldsymbol{a}}_j}\Sigma_{\hat{\boldsymbol{a}}_j}^{-1}\check{\boldsymbol{\epsilon}}_j\right)^T\right] \\
&- \mathbb{E}\left[\delta\hat{\boldsymbol{b}}_i\left(\Sigma_{\hat{\boldsymbol{b}}_j\hat{\boldsymbol{a}}_j}\Sigma_{\hat{\boldsymbol{a}}_j}^{-1}\check{\boldsymbol{\epsilon}}_j\right)^T\right] - \mathbb{E}\left[\Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{a}}_i}\Sigma_{\hat{\boldsymbol{a}}_i}^{-1}\check{\boldsymbol{\epsilon}}_i\delta\hat{\boldsymbol{b}}_j^T\right] \\
=&\Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{b}}_j} + \Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{a}}_i}\Sigma_{\hat{\boldsymbol{a}}_i}^{-1}\Sigma_{\hat{\boldsymbol{a}}_i\hat{\boldsymbol{a}}_j}\Sigma_{\hat{\boldsymbol{a}}_j}^{-1}\Sigma_{\hat{\boldsymbol{b}}_j\hat{\boldsymbol{a}}_j}^T \\
&- \Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{a}}_j}\Sigma_{\hat{\boldsymbol{a}}_j}^{-1}\Sigma_{\hat{\boldsymbol{b}}_j\hat{\boldsymbol{a}}_j}^T - \Sigma_{\hat{\boldsymbol{b}}_i\hat{\boldsymbol{a}}_i}\Sigma_{\hat{\boldsymbol{a}}_i}^{-1}\Sigma_{\hat{\boldsymbol{a}}_i\hat{\boldsymbol{b}}_j}
\end{aligned} \tag{4.23}$$

The integrity of the MLV algorithm for the triplex fixed solutions depends upon all three of the fixes being correct. Conditioned on this event, the three fixed baselines are distributed as a single, correlated, zero-mean multivariate Gaussian random vector. Assuming that any incorrect fix will result in errors exceeding the AL, the risk of excess error for a fixed triplex solution can be counded as

$$\begin{aligned}
R_{\text{fixed,triplex}} \leq\ & 1 - P\left(CF_1 \cap CF_2 \cap CF_3\right) \\
& + P\left(CF_1 \cap CF_2 \cap CF_3\right) \\
& \times R_{\text{MLV}}\left(AL, \mathbf{0}, \Sigma_{\check{\boldsymbol{B}},\text{triplex}}\right)
\end{aligned} \tag{4.24}$$

To upper bound the IR, the probability of correctly fixing all three integer solutions must be upper bounded. This bound is obtained by assuming that the events are perfectly correlated; i.e., one correct fix implies that all others are

correctly fixed, and one incorrect fix implies all others are incorrectly fixed. Additionally, the risk that any of the fixes is incorrect must be bounded since $1 - P\left(CF_1 \cap CF_2 \cap CF_3\right) = P\left(IF_1 \cup IF_2 \cup IF_3\right)$:

$$\begin{aligned} &P\left(\bigcap_i CF_i\right) = P\left(CF_1 | CF_2, CF_3\right) P\left(CF_2 | CF_3\right) P\left(CF_3\right) \\ &\text{Perfect correlation} \Rightarrow P\left(CF_j | CF_k\right) = 1, \forall j, k \\ &\therefore P\left(\bigcap_i CF_i\right) \leq \min_i \left(P_{CF_i}\right) \end{aligned} \tag{4.25}$$

Also, the probability of a union of events must be less than or equal to the sum of the probabilities of the events.

$$P\left(IF_1 \cup IF_2 \cup IF_3\right) \leq \sum_i P_{IF_i} \tag{4.26}$$

Substituting equations (4.25) and (4.26) into equation (4.24), the final bound is obtained.

$$\begin{aligned} R_{\text{fixed,MLV}} &\leq \sum_i P_{IF_i} \\ &+ \min_i \left(P_{CF_i}\right) \times R_{\text{MLV}}\left(AL, \mathbf{0}, \Sigma_{\check{\boldsymbol{B}},\text{triplex}}\right) \end{aligned} \tag{4.27}$$

It is evident from examining the above expression that this solution is only available if $\sum_i P_{IF_i} < IR_{\text{spec}}$. This implies that the demands on the probability of correct fix for triplex MLV are more stringent than for the simplex case. This may cause the overall system availability to be even lower for a federated triplex fixed solution than for the simplex solution if the P_{IF} is near the required IR.

4.4.5 Simplex GERAFS Integrity Risk

If a fixed solution does not produce a sufficiently high probability of correct fix, then an alternative is to use an "almost-fixing" solution. Almost-fixing solutions control IR by assessing the risk induced by biases that result from incorrect fixes near the fixed solution. Two such algorithms are GERAFS and EPIC.

In the almost-fixed case, the fixed baseline has the same underlying multi-modal mixture distribution as the fixed solution, but the integrity impact of modes other than the chosen fixed solution is accounted for in the position domain. Both GERAFS and EPIC examine a set of candidate integer fixes that differ from $\breve{\boldsymbol{a}}$ by an integer offset $\boldsymbol{d} \in \Delta \subset \mathbb{Z}^m \setminus \{\mathbf{0}\}$. The deterministic bias that would result if $\breve{\boldsymbol{a}} = \boldsymbol{a} + \boldsymbol{d}$ is computed for all $\boldsymbol{d} \in \Delta$. The set of all float ambiguities mapped to $\{\boldsymbol{a} + \boldsymbol{d} | \boldsymbol{d} \in \Delta\}$ is called the enlarged pull-in region (EPIR).

The GERAFS algorithm addresses the IR by assuming that the bias from any incorrect fix is equal to the worst case bias induced by any candidate fix. The magnitude of the worst case bias is called the incorrect fixing bias, IFB. The EPIC algorithm assesses the IR from each incorrect fix candidate separately, yielding higher availability than GERAFS. See Fig. 4.2 for an example of the distribution models used by EPIC and GERAFS compared with the associated fixed baseline distribution. Given these definitions, and defining

$$P_{EPIR} \triangleq \sum_{\boldsymbol{d} \in \Delta} P\left(\breve{\boldsymbol{a}} = \boldsymbol{a} + \boldsymbol{d}\right) \tag{4.28a}$$

$$P_{AF} \triangleq P_{CF} + P_{EPIR} \tag{4.28b}$$

$$P_{NAF} \triangleq 1 - P_{AF} \tag{4.28c}$$

the risk of excess error for a simplex GERAFS solution is bounded

$$\begin{aligned} R_{\mathrm{GERAFS,V}} &= P\left(|b_{\mathrm{V}} - b_{\mathrm{V}}| > AL_{\mathrm{V}}\right) \\ &\leq P_{NAF} + P_{CF} P\left(|b_{\mathrm{V}} - b_{\mathrm{V}}| > AL_{\mathrm{V}} | N = \check{\boldsymbol{a}}\right) \\ &+ P_{EPIR} P\left(|b_{\mathrm{V}} - b_{\mathrm{V}}| > AL_{\mathrm{V}} | N \in EPIR\right) \end{aligned} \tag{4.29}$$

4.4.6 Triplex GERAFS Joint Distribution

When conditioned an a particular candidate ambiguity, the covariance of an almost-fixed solution is the same as that of a fixed solution, so all that remains is to assess the integrity ramifications of almost-fixing. The development is similar to the fixed case with P_{AF} often taking the role of P_{CF} for the fixed case. The MLV integrity monitor must also be modified to reflect the worst case biases in the positive and negative directions. The modified risk monitoring function is denoted as $\bar{R}_{MLV}$:

$$\begin{aligned} \bar{R}_{\mathrm{MLV}} = &\sum_{j} \iint_{AL-IFB}^{\infty} f_{\epsilon_{i_1},\epsilon_{i_2}}\left(\varepsilon_{i_1}, \varepsilon_{i_2}\right) d\varepsilon_{i_1} d\varepsilon_{i_2} \\ &+ \sum_{j} \iint_{-\infty}^{IFB-AL} f_{\epsilon_{i_1},\epsilon_{i_2}}\left(\varepsilon_{i_1}, \varepsilon_{i_2}\right) d\varepsilon_{i_1} d\varepsilon_{i_2} \\ &- 2 \iiint_{AL-IFB}^{\infty} f_{\boldsymbol{\epsilon}}\left(\boldsymbol{\varepsilon}\right) d\boldsymbol{\varepsilon} - 2 \iiint_{-\infty}^{IFB-AL} f_{\boldsymbol{\epsilon}}\left(\boldsymbol{\varepsilon}\right) d\boldsymbol{\varepsilon} \end{aligned} \tag{4.30}$$

Where the IFB in the integral limits is the worst case incorrect fixing bias for the variable of integration.

This is identical to the previous definition of MLV integrity except that the magnitudes of the means are applied in the direction that maximizes risk. That is, IFB is added when evaluated against positive errors and subtracted when evaluated against negative errors. This ensures conservatism in the integrity bound:

$$\begin{aligned} R_{\text{GERAFS,MLV,V}} \leq & \sum_i P_{NAF_i} \\ & + \min_i \left(P_{CF_i}\right) \times R_{\text{MLV}}\left(AL, \Sigma_{\check{\boldsymbol{B}}_V,\text{triplex}}\right) \\ & + \left\{ \begin{array}{l} \min_i \left(P_{EPIR_i}\right) \\ \times \bar{R}_{\text{MLV,GERAFS}}\left(AL, IFB, \Sigma_{\check{\boldsymbol{B}}_V,\text{triplex}}\right) \end{array} \right\} \end{aligned} \tag{4.31}$$

4.5 Analytical Comparison to Simplex Solutions

The important metrics to be considered when comparing the simplex and triplex solutions are daily average accuracy and availability of integrity. Daily average accuracy is assessed for each location in a global grid and averaged over 24 hours at 5 minute intervals. Availability of integrity is the percentage of time that a particular system can meet the specified integrity requirements in operation. The IR equations for each type of solution provide the basis for making these assessments.

For the float solution, the MLV risk is simply a set of integrals with no risk due to integer fixing. Because of this there is no automatic integrity penalty for assuming that the integer ambiguities have been correctly resolved. Equations (4.9), (4.12), and (4.13) demonstrate that availability of integrity for triplex MLV float solutions can only be better than that of the simplex solution, unless the solutions are completely correlated.

The accuracy of the MLV float solution can be computed for the $(1-\alpha)$ percentile by numerically solving:

$$\varepsilon_\alpha = \inf\left\{\varepsilon | R_{\mathrm{MLV}}\left(\varepsilon, \Sigma_{\breve{\boldsymbol{B}},\mathrm{triplex}}\right) \leq \alpha\right\} \tag{4.32}$$

By the same logic applied to availability of integrity, the MLV equation demonstrates that the accuracy of federated triplex float solutions will be better than the accuracy of the simplex float solutions. Depending on the specified IR and ALs, MLV may provide sufficient performance improvement so that a triplex float solution would satisfy the requirements when a simplex float solution would not.

For the federated triplex fixed solution, MLV is not usually advantageous. Typically, the IR for a fixed solution is limited by P_{IF}, not by the accuracy of the fixed solution. Examination of equation (4.24) shows that all three solutions are required to have simultaneously correct fixes. This divides the individual required P_{IF} requirement by three. If a fixed solution is unavailable, or marginally available for the simplex case, it is less available for virtually all conditions in the federated triplex case.

For GERAFS solutions with the IR and AL requirements considered in this chapter, MLV provides no integrity benefit. The reduced allocation of P_{NAF} required for the MLV solution increases the IFB so that the MLV GERAFS solution is less available than the simplex GERAFS solution. To gain the most benefit possible from the triplex architecture, each GERAFS solution will protect its own integrity risk, and then MLV will be used to improve the accuracy performance of the final result.

4.6 Simulation Methodology

A numerical covariance analysis tool was used to assess the performance of the federated triplex GERAFS solutions as compared to the simplex GERAFS solution. This tool is an updated version of the same availability model (AM) that was originally used to assess the availability of the GERAFS algorithm [42]. The model was updated to include the MLV solution and a more stringent integrity requirement to reflect the needs of unmanned air vehicles. All solutions are computed using the wide lane carrier phase and narrow lane code combination.

For each location, availability of accuracy is zero if the daily average accuracy at that location is worse than the requirement. Availability of integrity is assessed for each time step, at each location. The AM evaluates algorithm performance on a worldwide grid of latitude and longitude and at 5 minute intervals over a 24 hour period.

Because integrity requirements must be satisfied at all times without the benefit of averaging, there are $288 \times N_{GridPoints}$ assessments of availability of integrity. Alternatively, accuracy is a daily average for each location, so there are only $N_{GridPoints}$ assessments of availability of accuracy. Let $\mathcal{T}$ be the set of times considered by the simulation. Each worldwide grid point is given an availability value defined as follows with world wide availability computed as the average over

all grid points:

$$A_{\mathrm{GP}} = \begin{cases} \frac{|\{t \in \mathcal{T} | IR_{\mathrm{GP}}(t) < IR_{\mathrm{spec}}\}|}{|\mathcal{T}|} & \varepsilon_{\mathrm{avg}} \leq \varepsilon_{\mathrm{spec}} \\ \\ 0 & \varepsilon_{\mathrm{avg}} > \varepsilon_{\mathrm{spec}} \end{cases} \tag{4.33}$$

where $|\{\cdot\}|$ is the number of elements in the set.

The error models used for the simulation include thermal noise, multipath, and antenna bias model for carrier phase and pseudorange, lever arm translation errors, and propagation effects due to latency of reference receiver data. Both simplex and triplex GERAFS implementations use a float solution as a backup when the P_{CF} and P_{AF} requirements are not satisfied. This allows for graceful degradation of system average accuracy as the GERAFS algorithm is unable to fix with integrity.

4.7 Simulation Results

The results of the original simplex GERAFS algorithm in [42] have been reproduced for an IR requirement reduced to reflect the needs of an unmanned landing. The results show the availability of the solutions as a function of the specified accuracy for varying ALs. There is a trend for all solutions that availability decreases with the AL. The simplex GERAFS and triplex GERAFS solutions also have decreasing accuracy as the AL decreases. This trend results from the less accurate float solution being used when GERAFS is unable to satisfy the required IR for the given AL.

For the least restrictive AL, the simplex GERAFS solution is almost always able to operate in the almost-fixed mode. However, as the AL is reduced, the IFB computed by GERAFS exceeds the AL more frequently, resulting in the use of the backup float solution. The direct results of the reduced AL are degraded average accuracy and reduced availability of integrity. For the specified IR used in these simulations, GERAFS is usually unavailable for an AL of 1.5 m, and the backup float solution is used. As a result, for this combination of requirements, simplex GERAFS performance is roughly equivalent to simplex float performance.

Table 4.1: World-wide availability for varying ALs for each solution

Solution Type $\backslash^{AL}$	1.5 m	2.5 m	3.5 m	4.4 m
Float	.8239	.9977	.9999	.9999
Simplex GERAFS	.8117	.9929	.9984	.9990
Triplex	.9629	.9995	.9999	.9999

For the triplex GERAFS solution, the probability of any of the three fixes being incorrect is lower bounded by the sum of the probabilities that each is incorrect. For the fixed solution to be used in these simulations, the P_{CF} is required to be greater than .998. This implies that the 95% accuracy of the MLV GERAFS solution will be as accurate as the simplex GERAFS solution or better.

4.8 Conclusions

Federated triplex solutions offer significant benefits to improve accuracy and integrity of float solutions, but for likely levels of IR and ALs the additional

Table 4.2: Daily average 70% accuracy for each solution at 99.5% availability or maximum obtained

Solution Type $\backslash^{AL}$	1.5 m	2.5 m	3.5 m	4.4 m
Float	32.7 cm	35.6 cm	36.0 cm	36.0 cm
Simplex GERAFS	32.7 cm	28.6 cm	21.9 cm	20.1 cm
Triplex	24.7 cm	20.6 cm	16.0 cm	14.6 cm

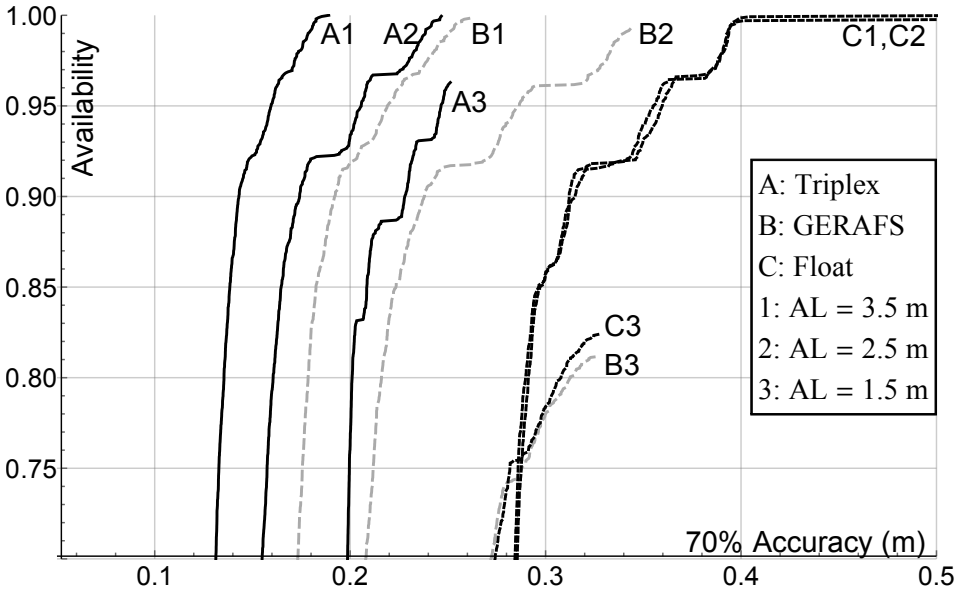

Figure 4.3: Comparison of availability vs. accuracy for float, GERAFS, and MLV triplex solutions for varying AL. As the AL is reduced, the GERAFS algorithm is less frequently able to fix the integer ambiguities. When the fixed solution is unavailable, both the GERAFS and MLV triplex solutions use a float solution as a backup. The performance of the GERAFS solution degrades to that of the float solution when $AL = 1.5m$ since the fixed solution is completely unavailable. Even when the MLV triplex solution uses the float solution, the integrity and accuracy benefits of MLV enable a 15% increase in availability and an 8 cm improvement in 70% accuracy.

burden of correctly fixing all three sets of integers prevents performance improvement for fixed or almost-fixed solutions. The degree of improvement afforded by federated triplex float solutions makes them competitive with simplex GERAFS solutions. Unfortunately, this solution requires three active rover receivers at all times which would require even more receivers to be used to ensure system continuity. MLV for the GERAFS algorithm still provides improved accuracy even when no additional integrity credit is claimed. To alleviate the continuity risk and enhance integrity, future studies will examine integrated architecture alternatives which will provide performance improvements with fewer rover receivers.

Chapter 5

Conclusion

5.1 Summary

The ability to reliably us fixed-integer CDGNSS positioning in safety-of-life applications will open up new horizons in autonomous vehicle navigation. Centimeter-accurate position estimates with sub-meter error integrity assurance will allow for unmanned air vehicles to land and refuel reliably during daily operation. Autonomous cars and advanced driver assistance features will be able to track the lane position of the vehicle with respect to curated maps which were themselves created with integrity assurance. Such remarkable feats will only be possible if the integer estimates can be trusted. This dissertation defends the following thesis:

Integer-fixed CDGNSS positioning for demanding safety-of-life applications requires a novel data-driven integer ambiguity validation method which benefits further from mid-level voting triplex architectures.

The following section offers a summary of the contributions proving this thesis statement.

- Chapter 2 presents the derivation and validation of the Generalized Integer Aperture Bootstrapping (GIAB) algorithm that validates the correctness of

the integer ambiguities with provable integrity, including for partial ambiguity resolution. The performance of the algorithm was validated via Monte Carlo simulation. GIAB was shown to be nearly as effective as the optimal IA method, while providing integrity sufficient for safety-of-life applications.

- Chapter 3 further extends the GIAB algorithm to provide position-domain integrity protection. This is particularly important for the case of partial ambiguity resolution, because it was proven in chapter 3 that the posterior PDF of a data-driven, partially-fixed baseline estimate is biased away from the true baseline even when constrained only using correctly validated fixes. The derived PDFs were validated with Monte Carlo simulations. The performance of the algorithm was shown to exceed that of the EPIC algorithm.

- Chapter 4 describes how MLV-triplex CDGNSS architectures can be used to improve the accuracy and integrity performance of positioning. Integrity monitors for float, fixed, and GERAFS solutions are developed. MLV-triplex float CDGNSS solutions benefit significantly both in terms of integrity and accuracy performance for the fault-free case. Fixed and GERAFS solutions benefit in fault-free accuracy performance, but receive no benefit in fault-free integrity benefit from a MLV-triplex architecture.

5.2 Future Work

Future work will include analysis of real-world CDGNSS data using GIAB and integration of GIAB into a complete safety-of-life navigation system.

Appendices

Appendix A

CDGNSS Measurement Model

A.1 GNSS Measurement Model

GNSS receivers provide two types of measurements for each signal that they track. The first measurement is called a pseudorange. A pseudorange is a noisy, biased estimate of the geometric range between the receiver antenna and the satellite antenna. The measurement is corrupted by a number of error sources including atmospheric delays, thermal noise, and multipath interference. The total effect of these errors for a single pseudorange measurement is typically on the order of 10s of meters.

The second type of GNSS measurement is called the carrier phase. The carrier phase provides an estimate of the change in the distance between the receiver antenna and the satellite antenna. Because the carrier phase only estimates the change in distance, it is ambiguous with respect to the full geometric range between the satellite and receiver antennas. However, it is a much smoother measurement than the pseudorange, with noise on the order of a few millimeters. It is also corrupted by atmospheric delays, similar to those of the pseudorange. The subsequent subsections will develop a model for these measurements as they are used with CDGNSS systems over short baselines, of a few kilometers.

A.1.1 Undifferenced Measurement Model

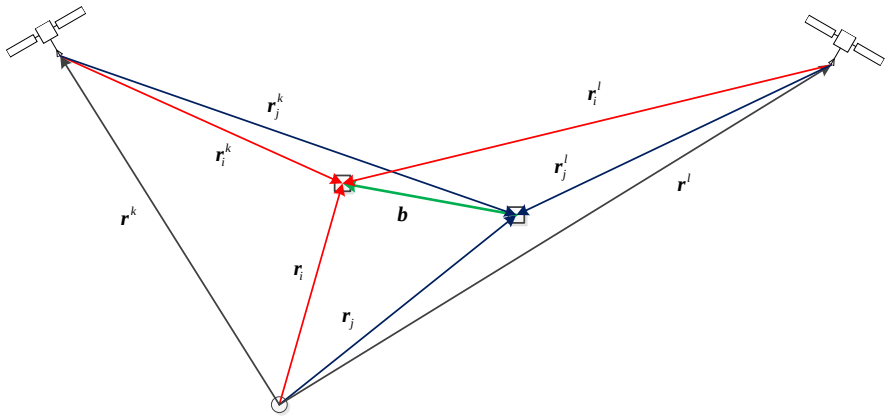

Figure A.1: An illustration of the geometric relationships among two GNSS satellites and two GNSS receivers. A CDGNSS system is interested in computing the relative position vector, or baseline, between the two receivers. This baseline is denoted $\boldsymbol{b}$. The vectors with subscripts, e.g. $\boldsymbol{r}_i$, indicate receiver positions, and those vectors with superscripts, e.g. $\boldsymbol{r}^k$, indicate satellite positions. Vectors with both super- and subscripts, e.g. $\boldsymbol{r}_i^k$, indicate relative position vectors from satellites to receivers.

The pseudorange measurement can be modeled as the sum of the geometric range between the satellite and receiver, (r_i^k), and several error terms. Refer to figure A.1 for an illustration of the vectors used in this section. The error terms include the effect of clock errors at the receiver $c\delta t_{R_i}$ and at the satellite $(c\delta t_S^j)$, ionospheric group delay (I_i^k), tropospheric delay (T_i^k), and a lumped error term that includes both white noise and time-correlated multipath errors $(w_{\rho_i}^k)$.

$$\rho_i^k = r_i^k + c(\delta t_{R_i} - \delta t_S^k) + I_{\rho_i}^k + T_i^k + w_{\rho_i}^k \tag{A.1}$$

For a single pseudorange measurement, the total error from all of these

sources is typically between 5 m and 50 m [23]. Most of the error comes from the ionospheric and tropospheric delay. The receiver clock error is solved for directly in the navigation solution since it is only a single unknown. The satellite clock error is corrected using a broadcast model with residual errors of approximately 2 m.

As GNSS signals pass through the ionosphere, they are delayed by the equivalent of between 2 m and 10 m for satellites at zenith. This delay is increased as the satellite elevation decreases toward the horizon, i.e. as the signal passes through more of the atmosphere. For very low elevation angles, the delay inceases by a factor of 3. By using the broadcast ionospheric correction model, approximately 60% of the ionospheric delay can be corrected [8]. Because the ionosphere is a dispersive medium, i.e. it induces a frequency dependent group delay, dual-frequency GNSS receivers can estimate the delay with uncertainty of approximately 1 m.

The troposphere causes a similar delay of approximately 2.5 m at zenith [21]. Because the troposphere is much thicker in extend than the ionosphere, low elevation satellites suffer a delay increase of a factor of 10 over those at zenith. By using models that reflect average meteorological conditions, these effects can compensated to an uncertainty of 0.1 m to 1 m at zenith. The troposphere is a non-dispersive medium, so dual-frequency measurements are not useful for estimating the tropospheric delay.

The multipath error depends on the radio frequency (RF) environment near the antennas. In “clean” multipath environments with few reflective surfaces, multipath errors on the pseudorange are about a meter in magnitude. In more challenging

environments, mulipath errors can be as large as 30 m to 150 m, depending on receiver architecture. Thermal noise is a much smaller contributor than multipath, typically less than half a meter.

There is a similar measurement model for the carrier phase:

$$\lambda \cdot \phi_i^k = r_i^k + c(\delta t_{R_i} - \delta t_S^k) + \lambda(\gamma_{0_i}^k - \psi_0^k) + I_{\phi_k}^j + T_i^k + \lambda \cdot w_{\phi_i}^k \tag{A.2}$$

In this equation, the carrier phase error is actually equal and opposite to the pseudorange error. That is:

$$I_i^k = I_{\rho_i}^k = -I_{\phi_i}^k \tag{A.3}$$

Also, the term $\lambda(\gamma_{0_i}^k - \psi_0^k)$ models the initial carrier phase offset of the receiver as it acquires the k^{th} satellite signal $\left(\gamma_{0_i}^k\right)$, and the unknown initial phase of the transmitted signal at the satellite $\left(\psi_0^k\right)$. The carrier phase noise $\left(w_{\phi_i}^k\right)$ is typically a small fraction of carrier phase cycle, which results in a range error of a few millimeters. This error is much smaller than that of the pseudorange noise and mulitpath, but the carrier phase is ambiguous since it only tracks the change in range from the time that phase lock was established.

A.1.2 Short Baseline Single Differences Measurement Model

When a pair of receivers is being used, it is common to form an observable by differencing the pseudoranges and carrier phases from the two receivers to each satellite that is tracked by both receivers. This forms the so called between receiver single difference (BRSD) observable. Since atmospheric errors are spatially correlated, much of the ionospheric and tropospheric errors cancel out leaving a much

smaller residual error. Also, the SV clock bias cancels out completely. The notation used to indicate a the difference between two measurement model terms for the same satellite and different receivers is

$$(\cdot)_{ij}^{k} = (\cdot)_{i}^{k} - (\cdot)_{j}^{k}\,,\ i:\ \text{Rover (Air) Receiver},\ j\text{: Reference (Ship) Receiver} \quad \text{(A.4)}$$

Applying this notation to the pseudorange and carrier phase measurement equations (A.1) and (A.2), the BRSD measurement model are

$$\rho_{ij}^{k} = r_{ij}^{k} + I_{ij}^{k} + T_{ij}^{k} + w_{\rho_{ij}}^{k} + b_{ij} \quad \text{(A.5a)}$$

$$\lambda\phi_{ij}^{k} = r_{ij}^{k} - I_{ij}^{k} + T_{ij}^{k} + \lambda\gamma_{0_{ij}}^{k} + w_{\phi_{ij}}^{k} + b_{ij} \quad \text{(A.5b)}$$

The ionospheric and tropospheric delays for receivers that are a few kilometers apart are highly correlated. This causes the residual atmospheric delay in the BRSD measurement to be significantly reduced. For such short baselines, the residual tropospheric delay drops from roughly 2.4 m to about 0.2 m. Similarly, the ionospheric error can be reduced to 0.2 m for even longer baselines in the absence of heightened solar activity. The remaining errors are not correlated between receivers, so receiver clock biases, multipath, and thermal noise errors are combined.

A.1.3 Double Difference Measurement Model

By differencing BRSDs from two satellites, a double difference measurement is formed with several attractive properties. All clock biases are eliminated and all atmospheric errors are reduced to residuals. The dominant error terms are the code and carrier multipath. Furthermore, the remaining phase uncertainty is

now integer valued, which will allow for the use of the integer ambiguity resolution techniques which enable precision relative GNSS solutions. The measurement model is

$$\rho_{ij}^{kl} = r_{ij}^{kl} + I_{ij}^{kl} + T_{ij}^{kl} + w_{\rho_{ij}}^{kl} \tag{A.6a}$$

$$\lambda\phi_{ij}^{kl} = r_{ij}^{kl} - I_{ij}^{kl} + T_{ij}^{kl} + \lambda\gamma_{0_{ij}}^{kl} + w_{\phi_{ij}}^{kl} \tag{A.6b}$$

$$\gamma_{0_{ij}}^{kl} \in \mathbb{Z} \tag{A.6c}$$

A.1.4 Carrier Phase Smoothed Pseudorange Measurements

Recall that the noise and multipath that corrupt the pseudorange measurements can be several meters in magnitude, but the same type of errors in the carrier phase measurements are on the order of a centimeter. Unfortunately, the carrier phase is ambiguous, so it cannot simply be used in place of the pseudorange measurement to directly solve for the position of a receiver without prior information. To gain some benefit from the carrier phase measurements, the pseudorange observables can be smoothed using the carrier phase observables. The resultant value is called the carrier smoothed code (CSC) measurement. The CSC has less noise than the raw pseudorange, but the temporal correlation of the multipath reduces the effectiveness of the smoothing. Further, since the ionospheric effect delays the pseudorange and advances the carrier phase measurement, over time, the code and carrier measurements diverge from each other. This effect is called code carrier divergence (CCD). Because the ionosphere is a dispersive medium, the delay/advance is related to the frequency of the carrier signal. When L1 and L2 measurements are available, the CCD effect can be mitigated by dual-frequency divergence free

smoothing (DFS) [22].

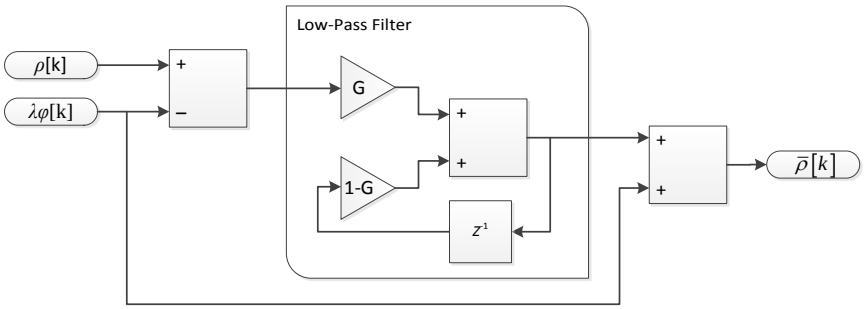

Figure A.2: A block diagram of the low pass filter form of the carrier smoothed code observable. In this approach, the difference between the carrier phase and the pseudorange is smoothed with a low pass filter with a time-varying gain. This difference is then added back to the carrier phase to form a smoothed estimate of the pseudorange.

One popular form of CSC smoothing is to use a low pass filter (LPF) with a time varying gain to smooth the difference between the pseudorange and the carrier phase measurements. This smoothed difference is then added back to the already low noise carrier phase to yield a smoothed pseudorange estimate. Denoting the CSC after k uninterrupted measurements as $\bar{\rho}\left[k\right]$, this can be expressed as

$$\bar{\rho}\left[k\right] = \left(1 - G\left[k\right]\right)\left(\bar{\rho}\left[k-1\right] - \lambda\phi\left[k-1\right]\right) + G\left[k\right]\left(\rho\left[k\right] - \lambda\phi\left[k\right]\right) + \lambda\phi\left[k\right] \quad \text{(A.7)}$$

With the gain is defined as

$$G\left[k\right] = \begin{cases} \frac{1}{k} & k \leq k_{max} \\ \frac{1}{k_{max}} & k > k_{max} \end{cases}, \quad \text{(A.8)}$$

This is how the smoother is computed practically, but to analyze the performance of the filter, it is useful to develop a closed form, non-iterative expression for $\bar{\rho}\left[k\right]$.

For $k < k_{max}$, the CSC expression simplifies to the sum of the current phase and the time average of the code-minus-carrier:

$$\bar{\rho}\left[k\right] = \lambda\phi\left[k\right] + \frac{1}{k}\sum_{j=1}^{k}\left(\rho\left[k\right] - \lambda\phi\left[k\right]\right), \forall 1 \leq k \leq k_{max} \tag{A.9}$$

For $k > k_{max}$, the time average is replaced by a constant gain LPF, and the resulting equation is a bit more complex.

$$\begin{aligned}\bar{\rho}\left[k\right] =& \lambda\phi\left[k\right] + K^{k-k_{max}}\left(\bar{\rho}\left[k_{max}\right] - \lambda\phi\left[k_{max}\right]\right) \\ &+ \frac{1}{k_{max}}\sum_{j=k_{max}+1}^{k} K^{k-j}\left(\rho\left[k\right] - \lambda\phi\left[k\right]\right), \forall k > k_{max}\end{aligned} \tag{A.10}$$

with $K = 1 - \frac{1}{k_{max}}$. In either case, the CSC is equal to the current carrier phase plus a weighted average of all previous carrier-minus-code observables. When this smoothing is applied to the measurement models in equations A.1 and A.2, a model for the CSC is obtained. The time index will be changed to n below to avoid confusion with the satellite index.

$$\lambda\phi_i^k\left[n\right] - \bar{\rho}_i^k\left[n\right] = \lambda\left(\gamma_{0_i}^k + \psi_0^k\right) - \left(2\bar{I}_i^k + \bar{w}_{\rho,i}^k - \lambda\bar{w}_{\phi,i}^k\right) \tag{A.11a}$$

$$\bar{\rho}_i^k\left[n\right] = r_i^k + c(\delta t_{R_i} - \delta t_S^k) + \left(\bar{I}_i^k + \bar{w}_{\rho,i}^k - \lambda\bar{w}_{\phi,i}^k\right) + T_i^k + \lambda\cdot w_{\phi_i}^k \tag{A.11b}$$

This expression implies that the difference between the carrier phase and the CSC comprises the carrier phase ambiguity, double the time averaged ionospheric delay, the time averaged pseudorange noise and multipath, and the time averaged carrier phase noise and multipath. If these observables are being used in a double difference in which the smoothing filters in the two receivers use different values of k_{max} or have smoothed different numbers of measurements, the time averaging

of the ionospheric delay causes the BRSD ionospheric residuals to be significantly larger than otherwise expected. This effect can be mitigated by dual-frequency divergence-free smoothing.

A.1.5 Divergence-Free Smoothing

Because the ionosphere is a dispersive medium for GNSS frequencies, the ionospheric delay is well approximated by an inverse-square law [23].

$$I \approx \frac{40.3\,TEC}{f^2} \tag{A.12}$$

By carefully combining measurements from both L1 and L2 pseudoranges and carrier phases, the ionospheric contribution of the code-minus-carrier observable can be removed prior to smoothing. This preserves the correlation between the ionospheric delay experienced by a rover and a reference receiver when using double difference or BRSD measurements.

For full details of the derivation of the constants used in divergence-free smoothing, the reader is referred to [22]. The key results are presented here. The four parameters, α_1, α_2, β_1, and β_2, are determined from the inverse square law and the constraints that $\alpha_1+\alpha_2 = 1$ and $\beta_1+\beta_2 = 1$ so that the geometric range component is preserved. Also, the ionospheric error component from the combination of carrier phase must cancel out the ionospheric error in the pseudorange combination. This ensures that no contribution from the ionospheric error is provided as input to

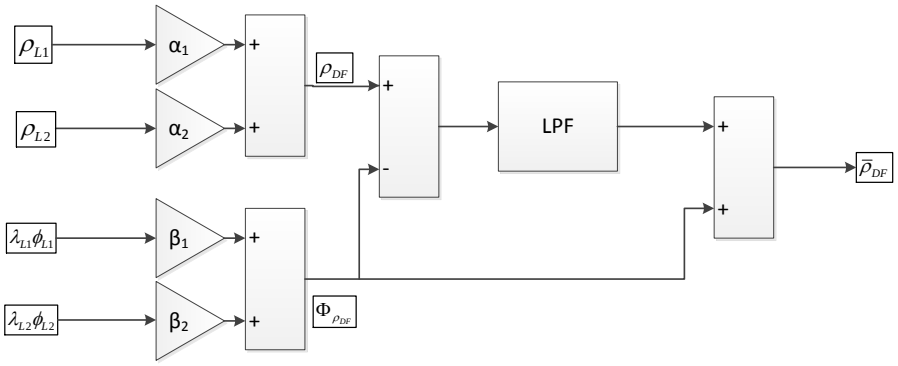

Figure A.3: Divergence-free smoothing is accomplished by weighting the input pseudoranges and carrier phases such that the ionospheric delay in the pseudorange is canceled by the carrier phase used in the smoother. In this way, reference and rover receivers are able to independently smooth their pseudoranges without decorrelating the ionospheric delay.

the LPF. For an arbitrary choice of α_1, the parameters are

$$\alpha_2 = 1 - \alpha_1 \tag{A.13a}$$

$$\beta_1 = \frac{2f_1^2}{f_1^2 - f_2^2} - \alpha_1 \tag{A.13b}$$

$$\beta_2 = \alpha_1 - \frac{f_1^2 + f_2^2}{f_1^2 - f_2^2} \tag{A.13c}$$

For legacy civil GPS signals where L2 does not have a pseudorange measurement, divergence-free smoothing can still be applied to the L1 measurement by setting $\alpha_1 = 1$ and $\alpha_2 = 0$. This leads to the combination

$$\Phi_{\rho_{L1,DF}} = \lambda_1 \frac{f_1^2 + f_2^2}{f_1^2 - f_2^2}\phi_{L1} - \lambda_2 \frac{2f_2^2}{f_1^2 - f_2^2}\phi_{L1} \tag{A.14a}$$

$$\bar{\rho}_{L1,DF} = \Phi_{\rho_{L1,DF}} + LPF\left(\rho_{L1} - \Phi_{\rho_{L1,DF}}\right) \tag{A.14b}$$

The notation $LPF\left(\cdot\right)$ indicates that the time series of measurements is low

pass filtered according to equation A.7. When full dual frequency measurements are available, L2 divergence-free smoothed pseudorange can be computed as

$$\Phi_{\rho_{L2,DF}} = \lambda_1 \frac{2f_1^2}{f_1^2 - f_2^2}\phi_{L1} - \lambda_2 \frac{f_1^2 + f_2^2}{f_1^2 - f_2^2}\phi_{L1} \tag{A.15a}$$

$$\bar{\rho}_{L2,DF} = \Phi_{\rho_{L1,DF}} + LPF\left(\rho_{L1} - \Phi_{\rho_{L1,DF}}\right) \tag{A.15b}$$

A common measurement pairing is the so called narrow-lane code – wide-lane carrier combination. This pair also happens to be a divergence-free combination:

$$\rho_{NL} = \frac{f_1}{f_1 + f_2}\rho_{L1} + \frac{f_2}{f_1 + f_2}\rho_{L2} \tag{A.16a}$$

$$\Phi_{WL} = \lambda_1 \frac{f_1}{f_1 - f_2}\phi_{L1} - \lambda_2 \frac{f_2}{f_1 - f_2}\phi_{L1} \tag{A.16b}$$

$$\bar{\rho}_{NL} = \Phi_{WL} + LPF\left(\rho_{NL} - \Phi_{WL}\right) \tag{A.16c}$$

A.1.6 Double Difference Error Model

Combining the double difference measurement model in equation (A.6b) on page 120 with the smoothed pseudorange measurement model (A.11b), the model for a pair of smoothed pseudorange and carrier phase measurements is:

$$\bar{\rho}_{ij}^{kl} = r_{ij}^{kl} + I_{ij}^{kl} + T_{ij}^{kl} + \bar{w}_{\rho_{ij}}^{kl} + \lambda\left(w_{\phi,ij}^{kl} - \bar{w}_{\phi,ij}^{kl}\right) \tag{A.17a}$$

$$\lambda\phi_{ij}^{kl} = r_{ij}^{kl} - I_{ij}^{kl} + T_{ij}^{kl} + \lambda\gamma_{0_{ij}}^{kl} + w_{\phi_{ij}}^{kl} \tag{A.17b}$$

$$\gamma_{0_{ij}}^{kl} \in \mathbb{Z} \tag{A.17c}$$

Using $Q_{(\cdot)}$ to indicate the covariance of the error sources, treating BRSD atmospheric residuals as bounded by a zero-mean Gaussian error for short baselines

which are independent between satellites, and assuming noise and multipath are independent among receivers and satellites, the covariances of double difference pseudoranges and carrier phases can be constructed as

$$Q_{\bar{\rho}_{ij}^{kl}} = \begin{array}{l} Q_{I_{ij}^{k}} + Q_{I_{ij}^{l}} + Q_{T_{ij}^{k}} + Q_{T_{ij}^{l}} + Q_{\bar{w}_{\rho_{ij}}^{kl}} \\ +\lambda^2 \left(Q_{w_{\phi,ij}^{kl}} + Q_{\bar{w}_{\phi,ij}^{kl}} - 2\text{cov}\left(w_{\phi,ij}^{kl}, \bar{w}_{\phi,ij}^{kl} \right) \right) \end{array} \tag{A.18a}$$

$$\lambda^2 Q_{\phi_{ij}^{kl}} = Q_{I_{ij}^{k}} + Q_{I_{ij}^{l}} + Q_{T_{ij}^{k}} + Q_{T_{ij}^{l}} + Q_{w_{\phi_{ij}}^{kl}} \tag{A.18b}$$

The variance of residual atmospheric errors can be modeled as a linear function of between receiver separation. The variance of the smoothed multipath and noise parameters is a function of the smoothing duration and the autocorrelation function of the underlying multipath processes. The covariance between the current carrier phase noise and multipath and the smoothed version of the same is

$$\text{cov}\left(w_{\phi,ij}^{kl}, \bar{w}_{\phi,ij}^{kl} \right) = \frac{1}{n_{max}} Q_{w_{\phi,ij}^{kl}} + \varepsilon_\tau \tag{A.19}$$

where n_{max} is the number of samples smoothed and ε_τ is the effect of time correlation in the carrier phase multipath, which is neglected.

The covariance between two pseudorange double differences is

$$\begin{aligned}
\operatorname{cov}\left(\bar{\rho}_{i_1j_1}^{k_1l_1}, \bar{\rho}_{i_2j_2}^{k_2l_2}\right) = & \left(Q_{I_{i_1j_1}^{k_1}} + Q_{T_{i_1j_1}^{k_1}}\right)\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) \\
& + \left(Q_{I_{i_1j_1}^{l_1}} + Q_{T_{i_1j_1}^{l_1}}\right)\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right) \\
& + \left(Q_{\bar{w}_{\rho_{i_1}}^{k_1}} + \lambda^2\left(\frac{n_{max}-2}{n_{max}}Q_{w_{\phi,i_1}^{k_1}} + Q_{\bar{w}_{\phi,i_1}^{k_1}}\right)\right)\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right)\delta_{i_1i_2} \\
& + \left(Q_{\bar{w}_{\rho_{i_1}}^{l_1}} + \lambda^2\left(\frac{n_{max}-2}{n_{max}}Q_{w_{\phi,i_1}^{l_1}} + Q_{\bar{w}_{\phi,i_1}^{l_1}}\right)\right)\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right)\delta_{i_1i_2} \\
& - \left(Q_{\bar{w}_{\rho_{j_1}}^{k_1}} + \lambda^2\left(\frac{n_{max}-2}{n_{max}}Q_{w_{\phi,j_1}^{k_1}} + Q_{\bar{w}_{\phi,j_1}^{k_1}}\right)\right)\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right)\delta_{j_1j_2} \\
& - \left(Q_{\bar{w}_{\rho_{j_1}}^{l_1}} + \lambda^2\left(\frac{n_{max}-2}{n_{max}}Q_{w_{\phi,j_1}^{l_1}} + Q_{\bar{w}_{\phi,ji_1}^{l_1}}\right)\right)\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right)\delta_{j_1j_2}
\end{aligned} \tag{A.20}$$

Similarly, the covariance between two carrier phase double differences is

$$\begin{aligned}
\lambda^2\operatorname{cov}\left(\phi_{i_1j_1}^{k_1l_1}, \phi_{i_2j_2}^{k_2l_2}\right) = & \left(Q_{I_{i_1j_1}^{k_1}} + Q_{T_{i_1j_1}^{k_1}}\right)\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) \\
& + \left(Q_{I_{i_1j_1}^{l_1}} + Q_{T_{i_1j_1}^{l_1}}\right)\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right) \\
& + \lambda^2\left(Q_{w_{\phi,i_1}^{k_1}}\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) + Q_{w_{\phi,i_1}^{l_1}}\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right)\right)\delta_{i_1i_2} \\
& - \lambda^2\left(Q_{w_{\phi,j_1}^{k_1}}\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) + Q_{w_{\phi,j_1}^{l_1}}\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right)\right)\delta_{j_1j_2}
\end{aligned} \tag{A.21}$$

Finally, the covariance between a carrier phase double differences and a pseudorange double difference is

$$\begin{aligned}
\operatorname{cov}\left(\lambda\phi_{i_1j_1}^{k_1l_1}, \bar{\rho}_{i_2j_2}^{k_2l_2}\right) = & \left(Q_{I_{i_1j_1}^{k_1}} + Q_{T_{i_1j_1}^{k_1}}\right)\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) \\
& + \left(Q_{I_{i_1j_1}^{l_1}} + Q_{T_{i_1j_1}^{l_1}}\right)\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right) \\
& + \lambda^2\frac{n_{max}-2}{n_{max}}\left(Q_{w_{\phi,i_1}^{k_1}}\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) + Q_{w_{\phi,i_1}^{l_1}}\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right)\right)\delta_{i_1i_2} \\
& - \lambda^2\frac{n_{max}-2}{n_{max}}\left(Q_{w_{\phi,j_1}^{k_1}}\left(\delta_{k_1k_2} - \delta_{k_1l_2}\right) + Q_{w_{\phi,j_1}^{l_1}}\left(\delta_{l_1l_2} - \delta_{l_1k_2}\right)\right)\delta_{j_1j_2}
\end{aligned} \tag{A.22}$$

Appendix B

Linearized Least Squares Relative Position Solution

The purpose of a CDGNSS solution is to determine the position of a rover receiver relative to a reference receiver. In the event that the reference receiver is at a static, well-known position, the absolute position of the rover receiver can also be determined with commensurate accuracy and precision. However, even in the case that the absolute position of the reference receiver is only known to several tens of meters, the relative position can still be extremely useful. This section develops the linearized least-squares estimate of the relative position, or baseline, between the receivers that neglects the integer nature of the double difference ambiguity.

B.1 Float Solution

Given the non-linear measurement and error models defined in Appendix A, a linearized, weighted least-squares estimate of the relative position between the two antennas. Given initial estimates of the two antenna positions, $\bar{\boldsymbol{r}}_i$ and $\bar{\boldsymbol{r}}_j$, and initial estimate of their relative position, or baseline, is given as $\bar{\boldsymbol{b}} = \bar{\boldsymbol{r}}_i - \bar{\boldsymbol{r}}_j$. The initial position estimates may come from a survey of the reference station or the absolute position estimate from each set of measurements. From these initial estimates, an linearization point for the double differenced geometric range can be

obtained. The true baseline is $\boldsymbol{b} = \bar{\boldsymbol{b}} + \delta\boldsymbol{b}$.

$$\begin{aligned} r_{ij}^{kl} &= \left\|\boldsymbol{r}^k - \boldsymbol{r}_i\right\| - \left\|\boldsymbol{r}^k - \boldsymbol{r}_j\right\| - \left\|\boldsymbol{r}^l - \boldsymbol{r}_i\right\| + \left\|\boldsymbol{r}^l - \boldsymbol{r}_j\right\| \\ &= \left\|\boldsymbol{r}^k - (\boldsymbol{r}_j + \boldsymbol{b})\right\| - \left\|\boldsymbol{r}^l - (\boldsymbol{r}_j + \boldsymbol{b})\right\| + \left\|\boldsymbol{r}^l - \boldsymbol{r}_j\right\| - \left\|\boldsymbol{r}^k - \boldsymbol{r}_j\right\| \\ &\approx r_{ij}^{kl}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} + \frac{\partial}{\partial \boldsymbol{b}} r_{ij}^{kl} \cdot \delta\boldsymbol{b} \end{aligned} \tag{B.1}$$

The partial derivative of a vector norm is the unit vector in that direction, so

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{b}} r_{ij}^{kl} &= \frac{\partial}{\partial \boldsymbol{b}} \left\|\boldsymbol{r}^k - (\boldsymbol{r}_j + \boldsymbol{b})\right\| - \frac{\partial}{\partial \boldsymbol{b}} \left\|\boldsymbol{r}^l - (\boldsymbol{r}_j + \boldsymbol{b})\right\| \\ &= -\frac{\boldsymbol{r}^k - \boldsymbol{r}_i}{\|\boldsymbol{r}^k - \boldsymbol{r}_i\|} + \frac{\boldsymbol{r}^l - \boldsymbol{r}_i}{\|\boldsymbol{r}^l - \boldsymbol{r}_i\|} \end{aligned} \tag{B.2}$$

Defining the unit vector from satellite k to antenna i as

$$\boldsymbol{u}_i^k = -\frac{\boldsymbol{r}^k - \boldsymbol{r}_i}{\|\boldsymbol{r}^k - \boldsymbol{r}_i\|} \tag{B.3}$$

The geometric range can then be linearized as

$$r_{ij}^{kl} \approx r_{ij}^{kl}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} + \left(\boldsymbol{u}_i^k - \boldsymbol{u}_i^l\right)^T \cdot \delta\boldsymbol{b} \tag{B.4}$$

Using (B.4) in the double difference measurement equation (A.17), lumping the error terms into a single error term for compactness, and representing the carrier phase ambiguity as $a \in \mathbb{Z}$:

$$\bar{\rho}_{ij}^{kl} - r_{ij}^{kl}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} = \left(\boldsymbol{u}_i^k - \boldsymbol{u}_i^l\right)^T \cdot \delta\boldsymbol{b} + \varepsilon_{\rho_{ij}^{kl}} \tag{B.5a}$$

$$\lambda\phi_{ij}^{kl} - r_{ij}^{kl}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} = \left(\boldsymbol{u}_i^k - \boldsymbol{u}_i^l\right)^T \cdot \delta\boldsymbol{b} + \lambda a_{ij}^{kl} + \varepsilon_{\phi_{ij}^{kl}} \tag{B.5b}$$

For m double differences all using a common reference satellite (i.e. $l_i = l\ \forall i$), the linearized measurements can be stacked in vectors. Define

$$\boldsymbol{y}_\rho \triangleq \begin{bmatrix} \bar{\rho}_{ij}^{1l} - r_{ij}^{1l}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} \\ \vdots \\ \bar{\rho}_{ij}^{ml} - r_{ij}^{ml}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} \end{bmatrix} \tag{B.6}$$

$$\boldsymbol{y}_\phi \triangleq \begin{bmatrix} \lambda\phi_{ij}^{1l} - r_{ij}^{1l}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} \\ \vdots \\ \lambda\phi_{ij}^{ml} - r_{ij}^{ml}\big|_{\boldsymbol{b}=\hat{\boldsymbol{b}}} \end{bmatrix} \tag{B.7}$$

$$G \triangleq \begin{bmatrix} \left(\boldsymbol{u}_i^1 - \boldsymbol{u}_i^l\right)^T \\ \vdots \\ \left(\boldsymbol{u}_i^n - \boldsymbol{u}_i^l\right)^T \end{bmatrix} \tag{B.8}$$

$$\Lambda \triangleq \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_m \end{bmatrix} \tag{B.9}$$

$$\begin{aligned} \boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_\rho \\ \boldsymbol{y}_\phi \end{bmatrix} &= \begin{bmatrix} G & 0_{m\times m} \\ G & \Lambda \end{bmatrix} \begin{bmatrix} \delta\boldsymbol{b} \\ \boldsymbol{a} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\varepsilon}_\rho \\ \boldsymbol{\varepsilon}_\phi \end{bmatrix} \\ \boldsymbol{y} &= H \begin{bmatrix} \delta\boldsymbol{b} \\ \boldsymbol{a} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\varepsilon}_\rho \\ \boldsymbol{\varepsilon}_\phi \end{bmatrix} \end{aligned} \tag{B.10}$$

The float solution can be obtained by carefully constructing the double difference measurement covariance matrix, $Q_{\boldsymbol{y}}$ according to equations (A.20), (A.21), and (A.22), and computing a weighted least squares solution to the preceding linearized equations.

$$\hat{\boldsymbol{x}} = \begin{bmatrix} \delta\hat{\boldsymbol{b}} \\ \hat{\boldsymbol{a}} \end{bmatrix} = \left(H^T Q_{\boldsymbol{y}}^{-1} H\right)^{-1} H^T Q_{\boldsymbol{y}}^{-1} \boldsymbol{y} \tag{B.11}$$

The errors in this correction to the initial baseline estimate and carrier phase ambiguity vector are dominated by the smoothed pseudorange multipath in most applications. Also, the vector $\hat{\boldsymbol{a}}$ is real-valued, or floating point. If the integer values of $\boldsymbol{a}$ can be determined from the floating point values of $\hat{\boldsymbol{a}}$, then the correction to the initial baseline estimate can be further refined. This is called the fixed solution. It will be described further in the next section.

Appendix C

Interpretation and Properties of the Float Ambiguity

C.1 Interpretation of L and D

It will be convenient to decompose the covariance of the float ambiguity as $Q_{\hat{\boldsymbol{z}}} = LDL^T$, where L is a unit lower triangular matrix and D is a diagonal matrix, and to model the float ambiguity as the true ambiguity plus zero-mean Gaussian noise, $\hat{\boldsymbol{z}} = \boldsymbol{z} + \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(0, Q_{\hat{\boldsymbol{z}}})$. Multiplication by L^{-1} transforms $\boldsymbol{\epsilon}$ into a vector whose elements are mutually uncorrelated: $\boldsymbol{\epsilon}_{\mathrm{c}} \triangleq L^{-1}\boldsymbol{\epsilon},\ \boldsymbol{\epsilon}_{\mathrm{c}} \sim \mathcal{N}(0, D)$. Letting l_{ij} denote the ijth element of L, d_i the ith element of the diagonal of D, and ϵ_i and $\epsilon_{\mathrm{c}i}$ the ith elements of $\boldsymbol{\epsilon}$ and $\boldsymbol{\epsilon}_{\mathrm{c}}$, respectively, ϵ_i and its variance can be computed from the first i components of $\boldsymbol{\epsilon}_{\mathrm{c}}$ as

$$\epsilon_i = \sum_{k=1}^{i} l_{ik}\epsilon_{\mathrm{c}k}, \quad \mathrm{var}\,(\epsilon_i) = \sum_{k=1}^{i} l_{ik}^2 d_k \tag{C.1}$$

Note that, because L and L^{-1} are upper triangular, vectors of the first $i-1$ elements of $\boldsymbol{\epsilon}_{\mathrm{c}}$, $\boldsymbol{\epsilon}_{\mathrm{c}I} \triangleq [\epsilon_{\mathrm{c}1}, \ldots, \epsilon_{\mathrm{c}i-1}]^T$ can be constructed from $\boldsymbol{\epsilon}_I$ with the elements of L^{-1} denoted as ℓ_{jk}.

$$\epsilon_{\mathrm{c}j} = \sum_{k=1}^{j} \ell_{jk}\epsilon_k \tag{C.2}$$

This fact allows us to give the equivalence

$$\begin{aligned} E\left[\epsilon_j \left| \boldsymbol{\epsilon}_I \right.\right] &= E\left[\epsilon_j \left| \boldsymbol{\epsilon}_{\mathrm{c}I} \right.\right] \\ &= E\left[\sum_{k=1}^{j} l_{jk}\epsilon_{\mathrm{c}k} \left| \boldsymbol{\epsilon}_{\mathrm{c}I} \right.\right] \end{aligned} \tag{C.3}$$

Since subsequent values of $\epsilon_{\mathrm{c}k}$ are independent of the preceding values, the conditioning can be removed for $k \geq i$. Also, the expectation of the given value is simply the given value.

$$\begin{aligned} E\left[\epsilon_j \left| \epsilon_{\mathrm{c}1}, \ldots, \epsilon_{\mathrm{c}i-1} \right.\right] &= \sum_{k=i}^{j} l_{jk}\cancelto{0}{E\left[\epsilon_{\mathrm{c}k}\right]} + \sum_{k=1}^{i-1} l_{jk}\epsilon_{\mathrm{c}k} \\ &= \sum_{k=1}^{i-1} l_{jk}\epsilon_{\mathrm{c}k} \end{aligned} \tag{C.4}$$

The conditional variance is then derived by the following steps:

$$\mathrm{var}\left(\epsilon_j \left| \boldsymbol{\epsilon}_I \right.\right) = E\left[\left(\epsilon_j - E\left[\epsilon_j \left| \boldsymbol{\epsilon}_I \right.\right]\right)^2 \left| \boldsymbol{\epsilon}_I \right.\right] \tag{C.5a}$$

$$= E\left[\left(\sum_{k=1}^{j} l_{jk}\epsilon_{\mathrm{c}k} - \sum_{k=1}^{i-1} l_{jk}\epsilon_{\mathrm{c}k}\right)^2 \left| \boldsymbol{\epsilon}_I \right.\right] \tag{C.5b}$$

$$= E\left[\left(\sum_{k=i}^{j} l_{jk}\epsilon_{\mathrm{c}k}\right)^2\right] \tag{C.5c}$$

$$= E\left[\sum_{k=i}^{j}\sum_{h=i}^{j} l_{jk}l_{jh}\epsilon_{\mathrm{c}k}\epsilon_{\mathrm{c}h}\right] \tag{C.5d}$$

$$= \sum_{k=i}^{j}\sum_{h=i}^{j} l_{jk}l_{jh}E\left[\epsilon_{\mathrm{c}k}\epsilon_{\mathrm{c}h}\right] \tag{C.5e}$$

$$= \sum_{k=i}^{j}\sum_{h=i}^{j} l_{jk}l_{jh}d_k\delta_{hk} \tag{C.5f}$$

$$= \sum_{k=i}^{j} l_{jk}^2 d_k \tag{C.5g}$$

(C.5a) by the definition of conditional variance; (C.5b) by substituting the conditional mean (C.4); (C.5c) by canceling the first $i-1$ terms; (C.5d) by expanding the square into a summation; (C.5e) by the linearity of expectation; (C.5f) by the variance of ϵ_{ck}; (C.5g) by canceling the Kronecker delta in the summation.

As a step toward interpreting l_{ij}, define the conditional variances and covariances as

$$\sigma^2_{ji|I} \triangleq \operatorname{cov}\left(\epsilon_{j|I}, \epsilon_{i|I}\right), \text{ for } j > i \tag{C.6a}$$

By definition, the conditional covariance is

$$\sigma^2_{ji|I} = E\left[\left(\epsilon_j - E\left[\epsilon_j \left| I\right.\right]\right)\left(\epsilon_i - E\left[\epsilon_i \left| I\right.\right]\right)\right| I] \tag{C.7}$$

Substitution of (C.1) and the means of (C.4) into (C.7) and simplification yields

$$\begin{aligned} \sigma^2_{ji|I} &= E\left[\left(\sum_{k=i}^{j} l_{jk}\epsilon_{ck}\right)\left(\epsilon_{ci}\right)\middle| I\right] \\ &= \sum_{k=i}^{j} l_{jk} E\left[\epsilon_{ck}\epsilon_{ci}\right| I] \end{aligned} \tag{C.8}$$

Since ϵ_{ck} and ϵ_{ci} are independent zero-mean variables:

$$\begin{aligned} \sigma^2_{ji|I} &= \sum_{k=i}^{j} l_{jk} d_i \delta_{ik} \\ &= l_{ji} d_i \end{aligned} \tag{C.9}$$

The value d_i has already been identified as the variance of $\epsilon_i|\boldsymbol{\epsilon}_I$, $\sigma^2_{i|I}$. Thus, l_{ji} can be interpreted as

$$l_{ji} = \begin{cases} \sigma^2_{ji|I}\sigma^{-2}_{i|I} & j > i \\ 1 & j = i \\ 0 & j < i \end{cases} \tag{C.10}$$

which comports with the fact that L is a unit lower triangular matrix.

C.2 Relationships among $\hat{\boldsymbol{z}}$, $\hat{\boldsymbol{z}}_{\mathrm{c}}$, $\check{\boldsymbol{z}}$, $\check{\boldsymbol{\epsilon}}$ and $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}$

Adopting the notation from Section 2.2.1, the sequentially-constrained float ambiguity vector, $\hat{\boldsymbol{z}}_{\mathrm{c}}$, is defined such that its ith element is $\check{z}_{\mathrm{c}i} = \check{z}_{i|I}$. Each $\check{z}_{\mathrm{c}i}$ is computed as

$$\hat{z}_{\mathrm{c}i} = \begin{cases} \hat{z}_i & i = 1 \\ \hat{z}_i - \sum_{k=1}^{i-1} l_{ik}(\hat{z}_{\mathrm{c}k} - \check{z}_k) \, i \in \{2, \ldots, m\} \end{cases} \tag{C.11}$$

where l_{ik} is the i, k entry in the matrix L. Because L is unit lower triangular, (C.11) can be expressed in vector form as

$$\hat{\boldsymbol{z}}_{\mathrm{c}} = \hat{\boldsymbol{z}} - (L - I)\,(\hat{\boldsymbol{z}}_{\mathrm{c}} - \check{\boldsymbol{z}}) \tag{C.12}$$

Rearranging and collecting terms in L yields

$$\hat{\boldsymbol{z}} - \check{\boldsymbol{z}} = L\,(\hat{\boldsymbol{z}}_{\mathrm{c}} - \check{\boldsymbol{z}}) \tag{C.13}$$

By definition, $\check{\boldsymbol{\epsilon}} = \hat{\boldsymbol{z}} - \check{\boldsymbol{z}}$, so

$$\check{\boldsymbol{\epsilon}} = L\,(\hat{\boldsymbol{z}}_{\mathrm{c}} - \check{\boldsymbol{z}}) \tag{C.14}$$

Multiplication by L^{-1} produces

$$L^{-1}\check{\boldsymbol{\epsilon}} = \hat{\boldsymbol{z}}_{\mathrm{c}} - \check{\boldsymbol{z}} \tag{C.15}$$

Also by definition, $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} = L^{-1}\check{\boldsymbol{\epsilon}}$, which produces the result:

$$\check{\boldsymbol{\epsilon}}_{\mathrm{c}} = \hat{\boldsymbol{z}}_{\mathrm{c}} - \check{\boldsymbol{z}} \tag{C.16}$$

Appendix D

Derivations for Data-Driven PDI

D.1 Truncation of Posterior Risk Probabilities

D.1.1 Posterior PDI and Fixing Probabilities

Consider a set of alternative fix candidates that includes the IB ambiguity estimate $\mathcal{Z} = \{\check{\boldsymbol{z}} - \Delta z \,|\Delta z \in \{\mathbf{0}, \boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_\nu\}\}$. Denote the IB solution in this set as $\check{\boldsymbol{z}}^0 \triangleq \check{\boldsymbol{z}}$. Denote any other element of this set as $\check{\boldsymbol{z}}^k \triangleq \check{\boldsymbol{z}}^0 - \boldsymbol{\zeta}_k$, with $k \in \{1, \ldots, \nu\}$. The sequentially-constrained ambiguity residual under the hypothesis that $\check{\boldsymbol{z}}^k = \boldsymbol{z}$ is defined, in agreement with (3.44), as

$$\begin{aligned} \check{\boldsymbol{\epsilon}}_{\mathrm{c}}^k &= \check{\boldsymbol{\epsilon}}_{\mathrm{c}} + L_{1:r,1:r}^{-1} \boldsymbol{\zeta}_k \\ &= \check{\boldsymbol{\epsilon}}_{\mathrm{c}} + L_{1:r,1:r}^{-1} \left(\check{\boldsymbol{z}}^0 - \check{\boldsymbol{z}}^k\right) \end{aligned} \tag{D.1}$$

The final expression of (D.1) can be recognized as the argument of the norms in (3.45). Define the likelihood functions

$$\lambda_k \triangleq \exp(-\frac{1}{2} \left\|\check{\boldsymbol{\epsilon}}_{\mathrm{c}}^k\right\|_{D_{1:r,1:r}}^2) = \prod_{j=1}^{r} \exp\left(-\frac{(\check{\epsilon}_{\mathrm{c}j}^k)^2}{2d_j}\right) \tag{D.2}$$

Using the λ_k notation, the posterior fixing probabilities are

$$P\left(\mathcal{E}_{\boldsymbol{\zeta}_k} | \boldsymbol{\varepsilon}, i\right) = \frac{\lambda_k}{\sum_{j=0}^{\infty} \lambda_j} \tag{D.3}$$

and the posterior PDI risk is bounded by

$$R = 1 - \sum_{k=0}^{s} \left(1 - R_k\right) \frac{\lambda_k}{\sum_{j=0}^{\infty} \lambda_j} \tag{D.4}$$

To obtain an upper bound on the risk, a lower bound is needed on (D.3). Break the set of all alternative fixes into sets $\mathcal{Z}$ and $\mathcal{Z}^c$, indexed by $j \in \{0, \ldots, \nu\}$ and $j \in \{\nu+1, \ldots, \infty\}$, respectively.

$$P\left(\mathcal{E}_{\boldsymbol{\zeta}_k} | \boldsymbol{\varepsilon}, i\right) = \frac{\lambda_k}{\sum_{j=0}^{\nu} \lambda_j + \sum_{j=\nu+1}^{\infty} \lambda_j} \tag{D.5}$$

Define $\lambda_\infty \triangleq \sum_{j=\nu+1}^{\infty} \lambda_j$. For any $\lambda^* \in \{\lambda | \lambda_\infty \leq \lambda\}$, an upper bound on the neglected fixes not considered, a lower bound on (D.3) is

$$\frac{\lambda_k}{\sum_{j=0}^{\nu} \lambda_j + \lambda^*} \leq P\left(\mathcal{E}_{\boldsymbol{\zeta}_k} | \boldsymbol{\varepsilon}, i\right) \tag{D.6}$$

To ensure integrity for safety-of-life applications, a value of λ^* must be found that provably bounds λ_∞. To reduce computational complexity, it is desirable to reduce the size of the set of alternatives considered by minimizing ν.

D.1.2 Bounding Posterior Fixing Probabilities for GIAB

When the GIAB aperture is set appropriately as in [11] with sufficiently high probability of success, and when the preceding ambiguity fixes are correct, the probability that next fix has error $\Delta z \notin \{0, -\mathrm{sgn}(\epsilon)\}$ is less than P_{E_i}. This implies that for the first ambiguity, $\check{z}_1$, there are two alternative fixes to consider. For each of these alternatives, there are two alternatives for $\check{z}_2$, and so forth, yielding a total of $\nu = 2^r - 1$ non-negligible alternatives to the IB solution.

The probability associated with the neglected alternative fixes can be given an additional allocation, p_{neg}, from an overall integrity risk budget. The neglected fixes have posterior probability that must be bounded above by the allocation to

assure safety

$$p_\infty = \frac{\lambda_\infty}{\sum_{j=0}^{2^r-1} \lambda_j + \lambda_\infty} \leq P_F \tag{D.7}$$

Isolating λ_∞ yields the bound:

$$\lambda_\infty \leq \frac{P_F}{1 - P_F} \sum_{j=0}^{2^r-1} \lambda_j \tag{D.8}$$

Though this bound is rigorous, its best-case computational complexity is exponential in m, so it is desirable to further reduce the number of alternative fixes that must be considered. If an additional allocation from the overall integrity risk budget is set aside to account for further neglected alternative fixes, ν can be reduced.

Separate the summation in the numerator of (D.5) now into three parts: $\lambda_{\mathcal{Z}} \triangleq \sum_{j=0}^{\nu} \lambda_j$, $\lambda_{\text{neg}} \triangleq \sum_{j=\nu+1}^{2^r-1} \lambda_j$, and $\lambda_\infty \triangleq \sum_{j=2^r}^{\infty} \lambda_j$. If an allocation of P_{neg} is made for the additionally neglected fixes, then λ_{neg} must satisfy

$$\frac{\lambda_{\text{neg}}}{\lambda_{\mathcal{Z}} + \lambda_{\text{neg}} + \lambda_\infty} \leq P_{\text{neg}} \tag{D.9}$$

Isolating λ_{neg} as in (D.8) yields

$$\lambda_{\text{neg}} \leq \frac{P_{\text{neg}}}{1 - P_{\text{neg}}} \left(\lambda_{\mathcal{Z}} + \lambda_\infty\right) \tag{D.10}$$

Substituting (D.8) with $\sum_{j=0}^{2^r-1} \lambda_j = \lambda_{\mathcal{Z}} + \lambda_{\text{neg}}$ and further isolating λ_{neg} yields

$$\begin{aligned} \lambda_{\text{neg}} &\leq \frac{P_{\text{neg}}}{1 - P_{\text{neg}}} \left(\lambda_{\mathcal{Z}} + \frac{P_F}{1 - P_F} \left(\lambda_{\mathcal{Z}} + \lambda_{\text{neg}}\right)\right) \\ &\leq \frac{P_{\text{neg}}}{(1 - P_{\text{neg}})(1 - P_F)} \left(\lambda_{\mathcal{Z}} + P_F \lambda_{\text{neg}}\right) \\ &\leq \frac{P_{\text{neg}}}{1 - P_{\text{neg}} - P_F} \lambda_{\mathcal{Z}} \end{aligned} \tag{D.11}$$

which establishes the required bound on the total likelihood of the finite set of neglected alternative fixes not inherently precluded by the GIAB aperture. The final bound on $P\left(\mathcal{E}_{\boldsymbol{\zeta}_k}|\boldsymbol{\varepsilon}, i\right)$ can be obtained after some tedious algebra, assuming (D.8) and (D.11) hold

$$\frac{(1 - P_{\text{neg}} - P_F)\lambda_k}{\lambda_{\mathcal{Z}}} \leq P\left(\mathcal{E}_{\boldsymbol{\zeta}_k}|\boldsymbol{\varepsilon}, i\right) \tag{D.12}$$

D.1.3 Binary-Tree-Based Alternative Fix Pruning

The set of all 2^r fixes not excluded by the GIAB aperture can be represented by the leaves of a binary tree of depth r. Each node represents a decision point in the IB process. The left-child of a node at depth i corresponds to the nearest integer to $\hat{z}_{\text{c}(i+1)}$ when conditioned on the previous assumed fixes. The right-child corresponds to the next-nearest integer.

For example, in Fig. D.1, node 4 corresponds to the IB solution $\check{z}_1^{\text{left}} = \lfloor \hat{z}_1 \rceil$ and $\check{\epsilon}_{\text{c}1}^{\text{left}} = \hat{z}_1 - \check{z}_1^{\text{left}}$, while node 12 corresponds to $\check{z}_1^{\text{right}} = \check{z}_1^{\text{left}} + \text{sgn}\left(\check{\epsilon}_{\text{c}1}^{\text{left}}\right)$ and $\check{\epsilon}_{\text{c}1}^{\text{right}} = \check{\epsilon}_{\text{c}1}^{\text{left}} - \text{sgn}(\check{\epsilon}_{\text{c}1}^{\text{left}})$. Each descendant of node 12 constrains the subsequent ambiguities assuming $z_1 = \check{z}_1^{\text{right}}$. Similarly, node 1 corresponds to the GIAB solution, $\check{\boldsymbol{z}}^0 = \check{\boldsymbol{z}}$, and node 3 corresponds to $\check{\boldsymbol{z}}^1 = \check{\boldsymbol{z}} + [0, \ldots, 0, \text{sgn}(\check{\epsilon}_{\text{c}r})]^{\text{T}}$.

To save computational effort, the binary tree can be constructed recursively while keeping a running total of the considered and neglected likelihoods. Define the partial likelihood function by truncating the products in (D.2)

$$\lambda_h^k \triangleq \prod_{j=1}^{h} \exp\left(-\frac{(\check{\epsilon}_{\text{c}j}^k)^2}{2d_j}\right) \tag{D.13}$$

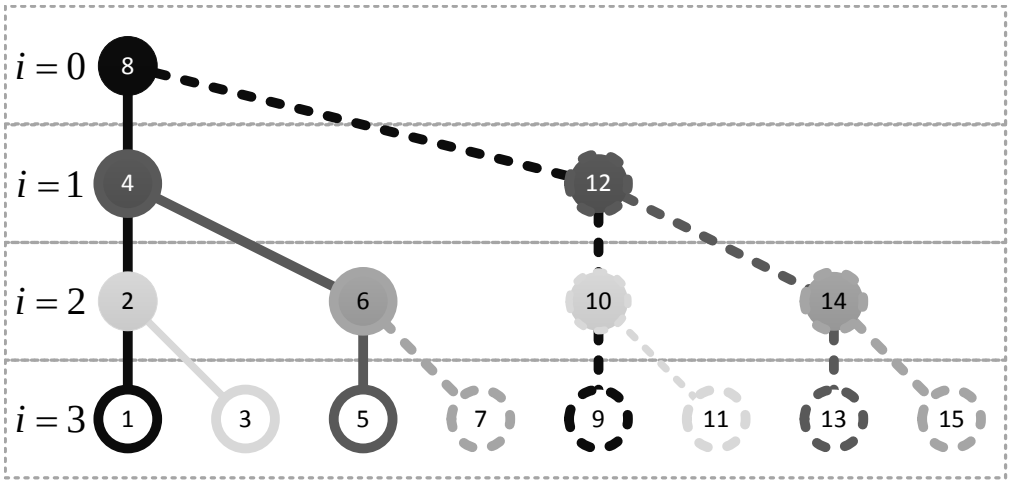

Figure D.1: The set of non-negligible fix candidates can be represented as a binary tree of depth r, shown for $r = 3$. See the body text for a full interpretation of the tree. The nodes are numbered according to the order visited by an in-order traversal. Unshaded leaf-nodes correspond to alternative IB solutions. Non-leaf nodes, which are all shaded, partial IB solutions leading to the different alternatives. The shade of lines matches the shade of the nearest ancestor node that generated the corresponding IB solution leaf, e.g. node 4 and leaf-node 5. Dashed lines indicate neglected branches of the tree.

Note that for $h > 1$,

$$\lambda_h^k = \exp\left(-\frac{(\check{\epsilon}_{cj}^k)^2}{2d_h}\right)\lambda_{h-1}^k \leq \lambda_{h-1}^k \tag{D.14}$$

Also, the likelihood of an alternative fix is $\lambda_r^k = \lambda_k$ as defined in (D.2). The λ_h^k are stored in a vector $\boldsymbol{\lambda}^k$.

Nodes in the binary tree at depth i store a 4-tuple, $\left(i, \check{z}_i^k, \check{\epsilon}_{ci}^k, \lambda_i^k\right)$. Note that any descendants of a node at depth i have the first i elements of their candidate fixes in common, i.e. $\check{\boldsymbol{z}}_{(1:i)}^k = \check{\boldsymbol{z}}_{(1:i)}^j$ when both $\check{\boldsymbol{z}}^k$ and $\check{\boldsymbol{z}}^j$ correspond to leaves of a common ancestor node at depth i.

The root of the tree is indicated by its value $(0, \emptyset, \emptyset, 1)$, where $\emptyset$ indicates a null value. The tree is initialized from the IB solution by setting each left-child from the root at depth i to $(i, \check{z}_i^0, \check{\epsilon}_{ci}^0, \lambda_i^0)$. This is the left-most branch of the trees in Fig. D.2. Initialization is captured in Algorithm D.2.

Algorithm D.1: buildBranch(&parent, $\check{\boldsymbol{z}}^k, \check{\boldsymbol{\epsilon}}_c^k, \boldsymbol{\lambda}^k, i$)

Input : parent $\in$ BinaryTree$\langle(\mathbb{Z}, \mathbb{Z}, \mathbb{R}, \mathbb{R})\rangle$, $\check{\boldsymbol{z}}^k \in \mathbb{Z}^r$, $\check{\boldsymbol{\epsilon}}_c^k \in \mathbb{R}^r$, $\boldsymbol{\lambda}^k \in \mathbb{R}^r$, $i \in \{1, \ldots, r\}$

```
node = BinaryTree((i, ž_i^k, ε̌_ci^k, λ_i^k))
node.parent = parent
if (i ≠ length(ž^k)) then
    buildBranch(node, ž^k, ε̌_c^k, λ^k, i + 1)
end
if parent ≠ NULL then
    parent.left = node
end
```

The tree is constructed via an in-order tree traversal, illustrated in Fig. D.2, during which every node is visited after its left-children but before its right-children.

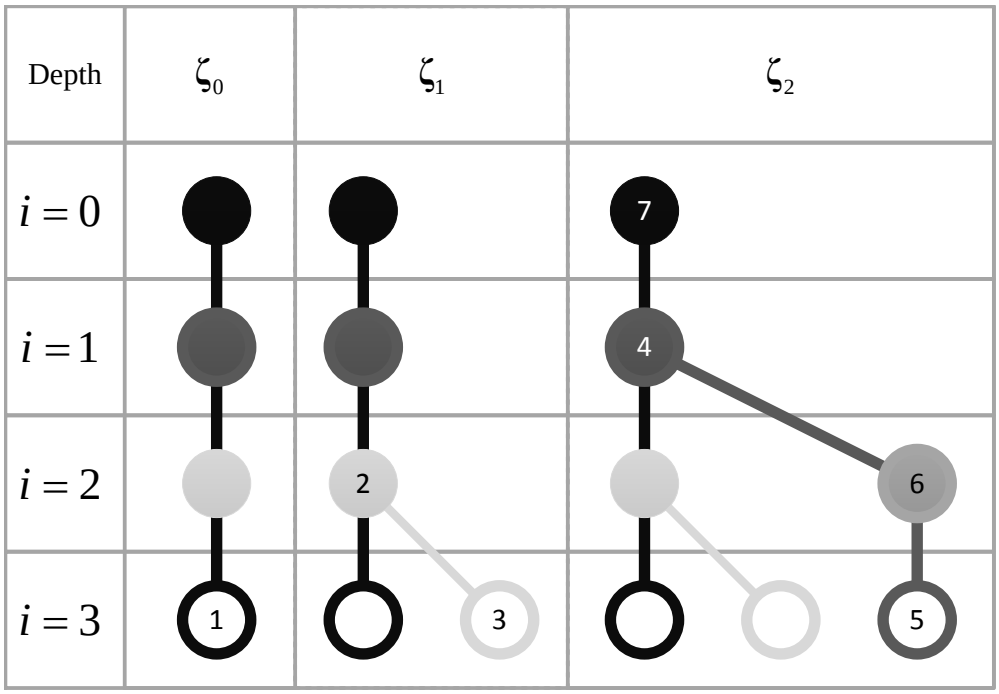

Figure D.2: Illustration of part of the process of building the binary tree of alternate fixes via an in-order tree traversal. The tree is initialized with the IB solution as the left-children of the root-node. The first node visited is the leaf corresponding to the IB solution. The second node visited, labeled 2, produces the first alternative solution, with offset $\boldsymbol{\zeta}_1$, which is stored as the right-child of node 2, and visited next. Node 4 is visited next, which leads to the second alternative solution, with offset $\boldsymbol{\zeta}_2$. This alternative fix is stored in the branch with nodes labeled 6 and 5. Node 5 is visited next and added to $\mathcal{Z}$ and $\lambda_{\mathcal{Z}}$. The alternative branches at nodes 6 and 7 are then found to be negligible, completing the tree traversal and construction.

Algorithm D.2: initIBTree$(\check{\boldsymbol{z}}, \check{\boldsymbol{\epsilon}}_{\mathrm{c}}, \boldsymbol{\lambda})$

Input : $\check{\boldsymbol{z}} \in \mathbb{Z}^r$, $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \in \mathbb{R}^r$, $\boldsymbol{\lambda} \in \mathbb{R}^r$

Output: root $\in$ BinaryTree$\langle(\mathbb{Z}, \mathbb{Z}, \mathbb{R}, \mathbb{R})\rangle$

1 root = BinaryTree$((0, \emptyset, \emptyset, 1))$

2 buildBranch(root, $\check{\boldsymbol{z}}, \check{\boldsymbol{\epsilon}}_{\mathrm{c}}, \boldsymbol{\lambda}, 1)$

Algorithm D.3: $\mathrm{altIB}(\check{\boldsymbol{z}}, \check{\boldsymbol{\epsilon}}_{\mathrm{c}}, \boldsymbol{\lambda}, L^{-1}, \boldsymbol{d}, i)$

Input : $\check{\boldsymbol{z}} \in \mathbb{Z}^r$, $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \in \mathbb{R}^r$, $\boldsymbol{\lambda} \in \mathbb{R}^r$, $L^{-1} \in \mathbb{R}^{m \times m}$, $\boldsymbol{d} \in \mathbb{R}^m$, $i \in \{1, \ldots, r\}$

Output: $\check{\boldsymbol{z}}^{\mathrm{alt}} \in \mathbb{Z}^r$, $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}^{\mathrm{alt}} \in \mathbb{R}^r$, $\boldsymbol{\lambda}^{\mathrm{alt}} \in \mathbb{R}^r$

$r = \mathrm{length}(\check{\boldsymbol{\epsilon}}_{\mathrm{c}})$

$\check{\boldsymbol{z}}^{\mathrm{alt}} = \check{\boldsymbol{z}}$

$\check{z}_i^{\mathrm{alt}} \mathrel{+}= \mathrm{sgn}\left(\check{\epsilon}_{\mathrm{c}i}\right)$

$\check{\boldsymbol{\epsilon}}_{\mathrm{c}}^{\mathrm{alt}} = \check{\boldsymbol{\epsilon}}_{\mathrm{c}} + L_{1:r,i}^{-1}\left(\check{z}_i - \check{z}_i^{\mathrm{alt}}\right)$

$\boldsymbol{\lambda}^{\mathrm{alt}} = \boldsymbol{\lambda}$

if $i == 1$ **then**

  $\lambda_i^{\mathrm{alt}} = \exp(-\frac{\check{\epsilon}_{\mathrm{c}i}^2}{2d_i})$

else

  $\lambda_i^{\mathrm{alt}} = \lambda_{i-1}^{\mathrm{alt}} \exp(-\frac{\check{\epsilon}_{\mathrm{c}i}^2}{2d_i})$

end

for $j = i + 1{:}r$ **do**

  $\check{z}_j^{\mathrm{alt}} \mathrel{+}= \lfloor \check{\epsilon}_{\mathrm{c}j}^{\mathrm{alt}} \rceil$

  $\check{\boldsymbol{\epsilon}}_{\mathrm{c}}^{\mathrm{alt}} = \check{\boldsymbol{\epsilon}}_{\mathrm{c}}^{\mathrm{alt}} + L_{1:r,j}^{-1}\left(\check{z}_j - \check{z}_j^{\mathrm{alt}}\right)$

  $\lambda_j^{\mathrm{alt}} = \lambda_{j-1}^{\mathrm{alt}} \exp(-\frac{\check{\epsilon}_{\mathrm{c}j}^2}{2d_j})$

end

Algorithm D.4: getSolution(&node)

Input : node $\in$ BinaryTree$\langle(\mathbb{Z}, \mathbb{Z}, \mathbb{R}, \mathbb{R})\rangle$
Output: $\check{\boldsymbol{z}} \in \mathbb{Z}^r$, $\check{\boldsymbol{\epsilon}}_{\mathrm{c}} \in \mathbb{R}^r$, $\boldsymbol{\lambda} \in \mathbb{R}^r$

```
temp = node
while temp.left ≠ NULL do
    temp = temp.left
end
ž = zeros(temp.data(1),1)
ε̌_c = zeros(temp.data(1),1)
λ = zeros(temp.data(1),1)
while temp.parent ≠ NULL do
    i = temp.data(1)
    ž_i = temp.data(2)
    ε̌_ci = temp.data(3)
    λ_i = temp.data(4)
    temp = temp.parent
end
```

Each node is processed according to Algorithm D.5 when visited. When a node is visited, its corresponding solution is obtained via Algorithm D.4. If the node is a leaf (i.e. its left-child is NULL), then its likelihood is included in the running total of $\lambda_{\mathcal{Z}}$. The fix and its likelihood are stored in a list of considered fixes, sorted by λ_j.

If the node is not a leaf, then an alternative solution must be considered. The partial likelihood of the right-child node, $\lambda_{i+1}^{\mathrm{right}}$, is computed assuming that $\check{z}_{i+1}^{\mathrm{right}} = \check{z}_{i+1}^{\mathrm{left}} + \mathrm{sgn}(\check{\epsilon}_{\mathrm{c}i}^{\mathrm{left}})$ and that $\check{\epsilon}_{\mathrm{c}(i+1)}^{\mathrm{right}} = \check{\epsilon}_{\mathrm{c}(i+1)}^{\mathrm{left}} - \mathrm{sgn}(\check{\epsilon}_{\mathrm{c}(\mathrm{i}+1)}^{\mathrm{left}})$ according to (D.1). The right-child node at depth $i+1$ will have at most $2^{r-(i+1)}$ descendant leaves, each with a likelihood less than or equal to $\lambda_{i+1}^{\mathrm{right}}$; e.g. in Fig. D.1, node 12 at depth 1 has $2^{3-1} = 4$ leaves in its branch. Thus the total likelihood of the right-branch is bounded above as $\lambda_{\mathrm{branch}} \leq 2^{r-(i+1)}\lambda_{i+1}^{\mathrm{right}}$. Given a running total the considered and

Algorithm D.5: visitNode(&node, $\&\lambda_{\mathcal{Z}}$, $\&\lambda_{\text{neg}}$, &fixes, $P_{\text{neg}}, L^{-1}, \boldsymbol{d}$)

Input : node $\in$ BinaryTree$\langle(\mathbb{Z}, \mathbb{Z}, \mathbb{R}, \mathbb{R})\rangle$, $\lambda_{\mathcal{Z}} \in \mathbb{R}$, $\lambda_{\text{neg}} \in \mathbb{R}$, fixes $\in$ SortedList, $P_{\text{neg}} \in \mathbb{R}$, $L^{-1} \in \mathbb{R}^{m \times m}$, $\boldsymbol{d} \in \mathbb{R}^m$

$\left[\check{\boldsymbol{z}}^{\text{left}}, \check{\boldsymbol{\epsilon}}_{\text{c}}^{\text{left}}, \boldsymbol{\lambda}^{\text{left}}\right] = \text{getSolution}(\text{node})$
i = node.data(1)
$r = \text{length}(\check{\boldsymbol{z}}^{\text{left}})$
if node.left == NULL **then**
    $\lambda_{\mathcal{Z}}$ += λ_i^{left}
    fixes.insert$\left(\left\{\lambda_i^{\text{left}}, \check{\boldsymbol{z}}^{\text{left}}\right\}\right)$
else
    $\lambda_{\text{temp}} = \lambda_i^{\text{left}} \exp(-\frac{\left(1-\left|\check{\epsilon}_{\text{c}(i+1)}^{\text{left}}\right|\right)^2}{2d_{i+1}})$
    if $2^{r-(i+1)}\lambda_{\text{temp}} \leq P_{\text{neg}}\lambda_{\mathcal{Z}} - \lambda_{\text{neg}}$ **then**
        $[\check{\boldsymbol{z}}^{\text{right}}, \check{\boldsymbol{\epsilon}}_{\text{c}}^{\text{right}}, \boldsymbol{\lambda}^{\text{right}}] = \text{altIB}(\check{\boldsymbol{z}}^{\text{left}}, \check{\boldsymbol{\epsilon}}_{\text{c}}^{\text{left}}, \boldsymbol{\lambda}^{\text{left}}, L^{-1}, \boldsymbol{d}, i+1)$
        buildBranch(node, $\check{\boldsymbol{z}}^{\text{right}}, \check{\boldsymbol{\epsilon}}_{\text{c}}^{\text{right}}, \boldsymbol{\lambda}^{\text{right}}, i+1$)
    else
        λ_{neg} += $2^{r-(i+1)}\lambda_{\text{temp}}$
    end
end

neglected likelihoods, $\lambda_{\mathcal{Z}}$ and λ_{neg} respectively, a branch can be neglected if

$$\lambda_{\text{neg}} + 2^{r-(i+1)}\lambda_{i+1}^{\text{right}} < \frac{p_{\text{neg}}}{1-p_{\text{neg}}}\lambda_{\mathcal{Z}} \tag{D.15}$$

λ_{neg} is then updated as $\lambda_{\text{neg}} = \lambda_{\text{neg}} + 2^{r-(i+1)}\lambda_{i+1}^{\text{right}}$.

If (D.15) does not hold, then the branch cannot be assumed negligible, so an alternative IB solution is computed with Algorithm D.3 by assuming that $z_{i+1} = \check{z}_{i+1}^{\text{left}} + \text{sgn}(\check{\epsilon}_{\text{c(i+1)}}^{\text{left}})$. Given this assumption, the subsequent ambiguities are constrained and IB is performed on them. This alternative solution is then stored as the right-child by Algorithm D.1. E.g. when node 4 is visited, the likelihood of node 6 is non-negligible, so the IB solution terminating at node 5 is computed and added to the tree. The tree traversal then continues around the modified tree.

Using this traversal-construction method, a list of all non-negligible alternative fix candidates can be constructed without evaluating any of the neglected alternatives explicitly. The list is then used to compute the probabilities (D.12), which are in turn needed to compute (3.48). The summation in (3.48) can be truncated as soon as R_{GIAB} is less than a specified bound, reducing the number of required computations of R_k, which requires the computationally expensive Gaussian CDF.

D.2 Distribution of Truncated Gaussian

Consider a known conditional distribution $f_{\boldsymbol{X}|A}(\boldsymbol{x})$ of a random vector $\boldsymbol{X} \in \mathbb{R}^n$ given some event A. Also consider event $B : \{\boldsymbol{X} \in \mathcal{B} \subset \mathbb{R}^n\}$. Define the

indicator function for the region $\mathcal{B}$ as

$$\mathbf{1}_{\mathcal{B}}(\boldsymbol{x}) \triangleq \begin{cases} 1 & \boldsymbol{x} \in \mathcal{B} \\ 0 \, \text{otherwise} \end{cases} \tag{D.16}$$

The distribution of $\boldsymbol{X}$ when further conditioned on event B can be expressed via the definition of a multivariate PDF.

$$f_{\boldsymbol{X}|A,B}\left(\boldsymbol{x}\right) = \frac{\partial^n}{\partial x_1 \ldots \partial x_n} F_{\boldsymbol{X}|A,B}\left(\boldsymbol{x}\right) \tag{D.17}$$

The conditional cumulative density function (CDF) can be determined by conditional probability as

$$F_{\boldsymbol{X}|A,B}\left(\boldsymbol{x}\right) = \frac{P(\boldsymbol{X} \leq \boldsymbol{x}, B|A)}{P(B|A)} \tag{D.18}$$

Multiplication of $f_{\boldsymbol{X}|A}\left(\boldsymbol{\xi}\right)$ by the indicator function produces the joint PDF which must be integrated to compute the numerator and denominator above:

$$P(\boldsymbol{X} = \boldsymbol{\xi}, B|A) = f_{\boldsymbol{X}|A}\left(\boldsymbol{\xi}\right) \mathbf{1}_{\mathcal{B}}(\boldsymbol{\xi}) \tag{D.19}$$

The numerator is computed as the integral over the region $\boldsymbol{X} \leq \boldsymbol{x}$ while the denominator is the integral over the region $\mathcal{B}$. In $\mathcal{B}$, the in indicator function is identically equal to 1, and so is elided.

$$F_{\boldsymbol{X}|A,B}\left(\boldsymbol{x}\right) = \frac{\int_{-\infty}^{\boldsymbol{x}} f_{\boldsymbol{X}|A}\left(\boldsymbol{\xi}\right) \mathbf{1}_{\mathcal{B}}(\boldsymbol{\xi}) d\boldsymbol{\xi}}{\int_{\mathcal{B}} f_{\boldsymbol{X}|A}\left(\boldsymbol{\zeta}\right) d\boldsymbol{\zeta}} \tag{D.20}$$

Substituting (D.20) into (D.17) and application of the second fundamental theorem of calculus yields:

$$\begin{aligned} f_{\boldsymbol{X}|A,B}\left(\boldsymbol{x}\right) &= \frac{\partial^n}{\partial x_1 \ldots \partial x_n} \frac{\int_{-\infty}^{\boldsymbol{x}} f_{\boldsymbol{X}|A}\left(\boldsymbol{\xi}\right) \mathbf{1}_{\mathcal{B}}(\boldsymbol{\xi}) d\boldsymbol{\xi}}{\int_{\mathcal{B}} f_{\boldsymbol{X}|A}\left(\boldsymbol{\zeta}\right) d\boldsymbol{\zeta}} \\ &= \frac{f_{\boldsymbol{X}|A}\left(\boldsymbol{x}\right) \mathbf{1}_{\mathcal{B}}(\boldsymbol{x})}{\int_{\mathcal{B}} f_{\boldsymbol{X}|A}\left(\boldsymbol{\zeta}\right) d\boldsymbol{\zeta}} \end{aligned} \tag{D.21}$$

Appendix E

Proof of Correlation Agnostic Bound

To prove the result in (4.15), begin with the expression

$$R_{\mathrm{MLV}} \leq 1 - \int_{-AL}^{AL} \int_{-AL}^{AL} f_{\epsilon_j,\epsilon_k}(\xi,\zeta)\, d\zeta d\xi \tag{E.1}$$

Next, use conditional probability to separate the joint distribution into the product of a marginal and a conditional distribution. Bring the marginal distribution out of the inner integral.

$$R_{\mathrm{MLV}} \leq 1 - \int_{-AL}^{AL} f_{\epsilon_j}(\xi) \int_{-AL}^{AL} f_{\epsilon_k|\epsilon_j}(\zeta|\epsilon_j = \xi)\, d\zeta d\xi \tag{E.2}$$

The law of total probability shows that

$$\int_{-AL}^{AL} f_{\epsilon_k|\epsilon_j}(\zeta|\epsilon_j = \xi)\, d\zeta = 1 - P(|\epsilon_k| > AL|\epsilon_j = \xi) \tag{E.3}$$

Substituting (E.3) into (E.2), distributing $f_{\epsilon_j}(\xi)$, and separating the two parts of the integral yields

$$R_{\mathrm{MLV}} \leq 1 - \int_{-AL}^{AL} f_{\epsilon_j}(\xi)\, d\xi + \int_{-AL}^{AL} f_{\epsilon_j}(\xi) \cdot P(|\epsilon_k|| > AL|\epsilon_j = \xi)\, d\xi \tag{E.4}$$

Defining $R_j \triangleq P(|\epsilon_j| \geq AL) = 1 - \int_{-AL}^{AL} f_{\epsilon_j}(\xi)\, d\xi$, simplify

$$R_{\mathrm{MLV}} \leq R_j + \int_{-AL}^{AL} f_{\epsilon_j}(\xi)\, P(|\epsilon_k|| > AL|\epsilon_j = \xi)\, d\xi \tag{E.5}$$

Expand $P\left(|\epsilon_k|| > AL|\epsilon_j = \xi\right)$ into integrals over $(-\infty, -AL]$ and $[AL, \infty)$:

$$R_{\text{MLV}} \leq R_j + \int_{-AL}^{AL} \left\{ f_{\epsilon_j}(\xi) \cdot \left[\begin{array}{l} \displaystyle\int_{-\infty}^{-AL} f_{\epsilon_k|\epsilon_j}(\zeta|\epsilon_j = \xi) d\zeta \\ \qquad + \displaystyle\int_{AL}^{\infty} f_{\epsilon_k|\epsilon_j}(\zeta|\epsilon_j = \xi)\, d\zeta \end{array} \right] \right\} d\xi \quad \text{(E.6)}$$

Multiply the conditional and marginal PDFs to form the joint PDF and reverse the order of integration.

$$R_{\text{MLV}} \leq R_j + \int_{-\infty}^{-AL} \int_{-AL}^{AL} f_{\epsilon_j,\epsilon_k}(\xi, \zeta)\, d\xi d\zeta + \int_{AL}^{\infty} \int_{-AL}^{AL} f_{\epsilon_j,\epsilon_k}(\xi, \zeta)\, d\xi d\zeta \quad \text{(E.7)}$$

Because PDFs are non-negative, expanding the limits of the inner integral yields another upper bound.

$$R_{\text{MLV}} \leq R_j + \int_{-\infty}^{-AL} \int_{-\infty}^{\infty} f_{\epsilon_j,\epsilon_k}(\xi, \zeta)\, d\xi d\zeta + \int_{AL}^{\infty} \int_{-\infty}^{\infty} f_{\epsilon_j,\epsilon_k}(\xi, \zeta)\, d\xi d\zeta \quad \text{(E.8)}$$

The inner integrals marginalize out ϵ_j.

$$R_{\text{MLV}} \leq R_j + \int_{-\infty}^{-AL} f_{\epsilon_k}(\zeta) d\zeta + \int_{AL}^{\infty} f_{\epsilon_k}(\zeta) d\zeta \quad \text{(E.9)}$$

The sum of the integrals may be recognized as the probability that the k^{th} solution error exceeds the AL.

$$R_{\text{MLV}} \leq R_j + R_k \quad \text{(E.10)}$$

Bibliography

[1] Fueled in flight: X-47B first to complete autonomous aerial refueling. NAVAL AIR SYSTEMS COMMAND, April 2015.

[2] Yaakov Bar-Shalom, X. Rong Li, and Thiagalingam Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley and Sons, New York, 2001.

[3] Andreas Brack. On reliable data-driven partial GNSS ambiguity resolution. *GPS Solutions*, 19(3):411–422, 2015.

[4] Andreas Brack and Christoph Gunther. Generalized integer aperture estimation for partial GNSS ambiguity fixing. *Journal of Geodesy*, 88(5):479–490, 2014.

[5] Andreas Brack, Patrick Henkel, and Christoph Günther. Sequential best integer-equivariant estimation for GNSS. *Navigation*, 61(2):149–158, 2014.

[6] J. M. Davis and Robert J. Kelly. RNP tunnel concept for precision approach with GNSS application. In *Proceedings of the 49th Annual Meeting of The Institute of Navigation*, pages 135–154, Cambridge, MA, June 1993.

[7] B. Decleene. Defining pseudorange integrity - overbounding. In *Proceedings of the ION GPS Meeting*, pages 1916–1924, September 2000.

[8] W. A. Feess and S. G. Stephens. Evaluation of GPS ionospheric time-delay model. *IEEE Transactions on Aerospace and Electronic Systems*, AES-23(3):332–338, May 1987.

[9] G. Nathan Green. *Advanced Techniques for Safety-of-Life Carrier Phase Differential GNSSPositioning with Applications to Triplex Architectures*. PhD thesis, The University of Texas at Austin, Dec. 2017.

[10] G. Nathan Green and Todd E. Humphreys. World-wide triplex CDGNSS performance. In *"The Proceedings of the Royal Institute of Navigation"*, Glasgow, Scotland, UK, 2016.

[11] G. Nathan Green and Todd E. Humphreys. Data-driven generalized integer aperture bootstrapping for high-integrity positioning. *IEEE Transactions on Aerospace and Electronic Systems*, 2017. Submitted for review.

[12] G. Nathan Green and Todd E. Humphreys. Position domain integrity analysis for generalized integer aperture bootstrapping. *IEEE Transactions on Aerospace and Electronic Systems*, 2017. Submitted for review.

[13] G. Nathan Green and Todd E. Humphreys. Correlation-aware integrity protection for fault-free federated triplex CDGNSS solutions. *Navigation, Journal of the Institute of Navigation*, 2018. In preparation.

[14] G. Nathan Green, Martin King, and Todd E. Humphreys. Fault free integrity of mid-level voting for triplex differential GPS solutions. In *Proceedings of the ION GNSS+ Meeting*, Tampa, FL, 2015.

[15] G. Nathan Green, Martin King, and Todd E. Humphreys. Data-driven generalized integer aperture bootstrapping for real-time high integrity applications. In *Proceedings of the IEEE/ION PLANS Meeting*, Savannah, GA, 2016.

[16] Yanqing Hou, Sandra Verhagen, and Jie Wu. A data driven partial ambiguity resolution: Two step success rate criterion, and its simulation demonstration. *Advances in Space Research*, 58(11):2435 – 2452, 2016.

[17] S. Khanafseh, M. Joerger, and B. Pervan. Integrity risk of cycle resolution in the presence of bounded faults. In *Proceedings of the IEEE/ION PLANS Meeting*, pages 664–672, April 2012.

[18] S. Khanafseh and B. Pervan. A new approach for calculating position domain integrity risk for cycle resolution in carrier phase navigation systems. In *Proceedings of the IEEE/ION PLANS Meeting*, pages 583–591, May 2008.

[19] S. Khanafseh and B. Pervan. Detection and mitigation of reference receiver faults in differential carrier phase navigation systems. *IEEE Transactions on Aerospace and Electronic Systems*, 47(4):2391–2404, Oct. 2011.

[20] M.D. Krstic, M.K. Stojcev, G. Lj. Djordjevic, and I.D. Andrejic. A mid-value select voter. *Microelectronics Reliability*, 45(34):733 – 738, 2005.

[21] G. A. McGraw. Tropospheric error modeling for high integrity airborne gnss navigation. In *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pages 158–166, April 2012.

[22] Gary A. McGraw. Generalized divergence-free carrier smoothing with applications to dual frequency differential GPS. *NAVIGATION, Journal of The Institute of Navigation*, 56(2):115–122, 2009.

[23] Pratap Misra and Per Enge. *Global Positioning System: Signals, Measurements, and Performance*. Ganga-Jumana Press, Lincoln, Massachusetts, revised second edition, 2012.

[24] Kenneth M. Pesyna, Jr. *Advanced Techniques for Centimeter-Accurate GNSS Positioning on Low-Cost Mobile Platforms*. PhD thesis, The University of Texas at Austin, Dec. 2015.

[25] M.L. Psiaki and S. Mohiuddin. Global positioning system integer ambiguity resolution using factorized least-squares techniques. *Journal of Guidance, Control, and Dynamics*, 30(2):346–356, March-April 2007.

[26] P. J. G. Teunissen. The probability distribution of the ambiguity bootstrapped gnss baseline. *Journal of Geodesy*, 75(5-6):267–275, 2001.

[27] P. J. G. Teunissen. GNSS ambiguity resolution with optimally controlled failure-rate. *Artificial Satellites*, 40:219–227, 2005.

[28] Peter Teunissen. Integer aperture GNSS ambiguity resolution. *Artificial Satellites*, 38(3):79–88, 2003.

[29] P.J.G. Teunissen. Success probability of integer GPS ambiguity rounding and bootstrapping. *Journal of Geodesy*, 72(10):606–612, 1998.

[30] PJG Teunissen. Gnss ambiguity bootstrapping: theory and application. In *Proceedings of International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation*, pages 246–254, 2001.

[31] P.J.G. Teunissen. A carrier phase ambiguity estimator with easy-to-evaluate fail-rate. *Artificial Satellites*, 38(3):89–96, 2003.

[32] P.J.G. Teunissen. Theory of carrier phase ambiguity resolution. *Wuhan University Journal of Natural Sciences*, 8(2):471–484, 2003.

[33] P.J.G. Teunissen. Integer aperture bootstrapping: a new GNSS ambiguity estimator with controllable fail-rate. *Journal of Geodesy*, 79(6-7):389–397, 2005.

[34] PJG Teunissen, PJ De Jonge, and CCJM Tiberius. The least-squares ambiguity decorrelation adjustment: its performance on short GPS baselines and short observation spans. *Journal of geodesy*, 71(10):589–602, 1997.

[35] P.J.G. Teunissen and Sandra Verhagen. The GNSS ambiguity ratio-test revisited: a better way of using it. *Survey Review*, 41(312):138–151, 2009.

[36] Sandra Verhagen and Peter JG Teunissen. New global navigation satellite system ambiguity resolution method compared to existing approaches. *Journal of Guidance, Control, and Dynamics*, 29(4):981–991, 2006.

[37] Sandra Verhagen and PeterJ.G. Teunissen. The ratio test for future gnss ambiguity resolution. *GPS Solutions*, 17(4):535–548, 2013.

[38] WAAS Test Team. Wide-Area Augmentation System performance analysis report. Technical report, Federal Aviation Administration, January 2014.

[39] J. Wang, M. P. Stewart, and M. Tsakiri. A discrimination test procedure for ambiguity resolution on-the-fly. *Journal of Geodesy*, 72(11):644–653, Nov 1998.

[40] Lei Wang and Sandra Verhagen. A new ambiguity acceptance test threshold determination method with controllable failure rate. *Journal of Geodesy*, 89(4):361–375, 2015.

[41] Z. Wen, P. Henkel, A. Brack, and C. Gnther. Best integer equivariant estimation for precise point positioning. In *ELMAR, 2012 Proceedings*, pages 279–282, Sept 2012.

[42] Shuwu Wu, Steven R. Peck, Robert M. Fries, and G. A. McGraw. Geometry extra-redundant almost fixed solutions: A high integrity approach for carrier phase ambiguity resolution for high accuracy relative navigation. In *ionplans*, pages 568–582, 2008.

[43] Jingyu Zhang, Meiping Wu, Tao Li, and Kaidong Zhang. Integer aperture ambiguity resolution based on difference test. *Journal of Geodesy*, 89(7):667–683, 2015.

Vita

Gary Nathan Green has worked in the aerospace and defense industry for ten years. He is currently a senior systems engineer at Coherent Technical Services, Inc. where he consults for the United States Navy as an expert in navigation and control systems. He is also an adjunct professor of electrical engineering at LeTourneau University in Longview, TX, where he has taught Communications Engineering and led a capstone senior design project as faculty advisor.

Permanent address: gnathangreen@gmail.com

This dissertation was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.