

# **The E and B EXperiment: A balloon-borne cosmic microwave background anisotropy probe**

**Seth Hillbrand**

Submitted in partial fulfillment of the  
requirements for the degree  
of Doctor of Philosophy  
in the Graduate School of Arts and Sciences

**COLUMBIA UNIVERSITY**

2014

©2014

Seth Hillbrand

All Rights Reserved

# ABSTRACT

## **The E and B EXperiment: A balloon-borne cosmic microwave background anisotropy probe**

**Seth Hillbrand**

The E and B Experiment (EBEX), is a balloon-borne sub-orbital cosmic microwave background polarimeter, designed to measure polarization levels in the microwave spectrum. EBEX recently completed an 11-day Antarctic long duration balloon (LDB) science flight in January, 2013.  $\sim 1000$  transition edge sensor bolometric detectors in three frequency bands centered at 150, 250 and 410 GHz sampled a large segment of the southern sky.

Over 1.5TB of data were collected during the LDB flight. In this thesis, we describe the design and performance of the EBEX software components monitoring and controlling the system during the flight, including automation, telemetry, data storage and readout array management.

We also describe the design and development of a novel attitude reconstruction system for a balloon-borne pointed observation platform based on a daytime star camera and 3-axis gyroscopes. The data gathered during the LDB flight are analyzed and the results presented showing attitude reconstruction error at less than  $20''$  RMS for an 80 second interval.

# Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>EBEX Science</b>	<b>4</b>
2.1	Concordance Cosmology . . . . .	4
2.1.1	Expanding Universe . . . . .	6
2.1.2	Flat Universe . . . . .	9
2.1.3	Hot, Dense Early Universe . . . . .	9
2.1.4	Dark Matter Universe . . . . .	10
2.1.5	Dark Energy Dominated Universe . . . . .	12
2.1.6	Concordance Constraints . . . . .	12
2.2	Inflation . . . . .	14
2.3	Cosmic Microwave Background Radiation . . . . .	15
2.3.1	CMB Polarization . . . . .	17
2.4	EBEX B-Mode Science . . . . .	19
2.4.1	Scan Strategy . . . . .	20
2.5	Summary . . . . .	23
<b>3</b>	<b>The EBEX Instrument</b>	<b>24</b>
3.1	Overview . . . . .	24
3.2	Cryostat . . . . .	25
3.2.1	Half-Wave Plate . . . . .	26



3.2.2	Focal Planes . . . . .	27
3.2.3	Readout . . . . .	28
3.3	Gondola . . . . .	31
3.3.1	Attitude Control System . . . . .	33
3.3.2	Attitude Sensors . . . . .	33
3.4	Communications . . . . .	40
3.4.1	BLASTBus . . . . .	40
3.4.2	RS-232 . . . . .	42
3.4.3	CANBus . . . . .	42
3.4.4	TCP/IP Network . . . . .	44
3.5	Software . . . . .	45
3.5.1	Threading . . . . .	47
3.5.2	Disk Management . . . . .	53
3.5.3	BLASTBus Interface . . . . .	61
3.5.4	Automation . . . . .	62
3.5.5	Telemetry . . . . .	66
3.6	Summary . . . . .	71
<b>4</b>	<b>Attitude Reconstruction</b>	<b>72</b>
4.1	Overview . . . . .	72
4.2	Theory . . . . .	75
4.2.1	Wahba’s Problem . . . . .	75
4.2.2	Quaternion Formalism . . . . .	76
4.2.3	Creating a Rotation Matrix . . . . .	78
4.2.4	Quaternion Rotation . . . . .	80
4.2.5	Quaternion Kinematics . . . . .	80
4.2.6	Kalman Filtering . . . . .	82
4.2.7	Particle Filters . . . . .	83
4.2.8	Sequential Importance Resampling . . . . .	86

4.3	EBEX Sensor Parameterization . . . . .	95
4.3.1	Gyroscope Readouts . . . . .	95
4.3.2	Star Camera Parameterization . . . . .	98
4.3.3	Field Matching . . . . .	104
4.3.4	Filter Fitness Functions . . . . .	110
4.3.5	Time Dependence . . . . .	112
4.4	Verification Metrics . . . . .	114
4.4.1	Simulated Data . . . . .	114
4.4.2	Results . . . . .	118
<b>5</b>	<b>LDB Flight Attitude Reconstruction</b>	<b>125</b>
5.1	Summary of the EBEX Long Duration Flight . . . . .	125
5.2	Star Camera Solutions . . . . .	129
5.2.1	Generalized Symmetric Weighting Analysis . . . . .	130
5.2.2	Scan Length Distribution . . . . .	132
5.3	Gyroscope Readings . . . . .	133
5.3.1	Spike Identification . . . . .	134
5.3.2	Spike Verification . . . . .	139
5.3.3	Spike Removal . . . . .	140
5.4	Particle Filter Application . . . . .	143
5.4.1	Parameter Verification . . . . .	144
5.4.2	Attitude Reconstruction . . . . .	146
5.5	Summary . . . . .	149
	<b>Bibliography</b>	<b>150</b>
<b>A</b>	<b>Algorithm Details</b>	<b>168</b>
A.1	FIFO Algorithms . . . . .	168
A.2	Kernel Queuing . . . . .	169
A.3	SIR Particle Filter . . . . .	171

A.3.1 Sequential Importance Resampling . . . . .	171
A.4 Adaptive Boosting . . . . .	172
<b>B Automation Files</b>	<b>173</b>

# List of Figures

1.1	Atmospheric transmission at microwave frequencies for balloon-borne and ground-based experiments . . . . .	2
2.1	Concordance Cosmology . . . . .	5
2.2	Weyl's Postulate . . . . .	7
2.3	Planck + BAO Constraints on $\Omega_\Lambda$ - $\Omega_m$ . . . . .	10
2.4	Planck Constraints on $r$ - $n_s$ with Inflationary Models . . . . .	15
2.5	CMB Temperature Power Spectrum Anisotropies . . . . .	16
2.6	Polarization from Density Fluctuations and Thompson Scattering . . . . .	17
2.7	Predicted Anisotropy Power Spectra for Best-Fit $\Lambda$ CDM Values . . . . .	19
2.8	EE and BB Power Spectrum Measurements . . . . .	20
2.9	EBEX Expected Constraints on Power Spectra . . . . .	21
2.10	EBEX Scan Strategy . . . . .	22
3.1	EBEX Internal Optics . . . . .	25
3.2	HWP Schematic . . . . .	26
3.3	EBEX Focal Plane . . . . .	27
3.4	SQUID Board . . . . .	29
3.5	The EBEX Cryostat with Associated Readout Electronics . . . . .	30
3.6	EBEX Readout Schematic . . . . .	30
3.7	Gondola CAD Drawing . . . . .	32
3.8	Fiber-Optic Gyro . . . . .	36

3.9	Allan Variance Measurement . . . . .	37
3.10	BLASTBus Word . . . . .	41
3.11	EBEX Ethernet network diagram . . . . .	44
3.12	Flight Control Program Block Diagram . . . . .	46
3.13	Memory Layout of the FIFO Queue . . . . .	48
3.14	Performance of the EBEX FIFO queue . . . . .	52
3.15	EBEX Disk Pressure Vessels . . . . .	54
3.16	Typical EBEX File Lifecycle . . . . .	58
3.17	Performance of the AoE Disk Subsystem . . . . .	60
3.18	Signal Handling in the EBEX Automator . . . . .	65
3.19	EBEX Communication Paths . . . . .	67
3.20	Packetslinger Header Definition. . . . .	69
4.1	Rotation Schematic . . . . .	74
4.2	Sequential Importance Resampling . . . . .	87
4.3	Particle Regularization . . . . .	90
4.4	Image of the 3-Axis Gyroscope Box . . . . .	96
4.5	Symmetric Weighting . . . . .	102
4.6	Star Pattern Matching . . . . .	104
4.7	Star Field Hash Word . . . . .	105
4.8	Camera Platescale Variation . . . . .	109
4.9	Simulated Scans . . . . .	115
4.10	Roll Uncertainty between Two Stars . . . . .	117
4.11	Gyro Parameter Optimization . . . . .	118
4.12	Rotation Parameter Convergence . . . . .	120
4.13	Misalignment Parameter Convergence . . . . .	121
4.14	Bias Parameter Tracking . . . . .	121
4.15	Reconstruction Histograms . . . . .	122
4.16	Power Spectrum Contamination . . . . .	124

5.1	EBEX Flight Trajectory . . . . .	126
5.2	EBEX Flight Altitude . . . . .	127
5.3	EBEX LDB Azimuth . . . . .	128
5.4	Star Camera Image Noise Reduction . . . . .	130
5.5	Comparison of Centroiding Errors between Weighted and Unweighted Star Camera Images . . . . .	131
5.6	Star Camera Solution Time Distribution . . . . .	132
5.7	Comparison of Paired Gyroscope Timestreams . . . . .	133
5.8	Gyroscope Time Series Spikes . . . . .	135
5.9	Gyroscope Poincaré Map Filter . . . . .	139
5.10	Example Spike Removal . . . . .	142
5.11	Distribution of Star Camera Rotation Offsets . . . . .	145
5.12	Star Camera-Gyroscope Uncertainty Pairing as a Function of Scan Length	148
5.13	Combined Gyroscope Uncertainties as a Function of Scan Length . .	149
B.1	The Document Type Definition file that details the full structure of EBEX automation files. After creation or modification, the automation XML file is verified for grammar against this definition file. . . . .	174
B.2	The cryogenic cooler cycle initiation procedure and subprocedure as used in the EBEX LDB flight . . . . .	175
B.3	Cryogenic Cooler Cycle Step 1 . . . . .	176

# List of Tables

2.1	<a href="#"><math>\Lambda</math>CDM Constraints from Planck and WMAP . . . . .</a>	13
3.1	<a href="#">EBEX Attitude Sensors and Their Associated Uncertainties . . . . .</a>	34
3.2	<a href="#">EBEX Telemetry Links and Associated Characteristics . . . . .</a>	68
4.1	<a href="#">Particle Filter Parameter Space . . . . .</a>	113
4.2	<a href="#">Differential Evolution Parameter Selection . . . . .</a>	119
5.1	<a href="#">Spike Replacement Metrics . . . . .</a>	143

# Acknowledgments

Experimental science rarely occurs in a vacuum and the work described in this thesis is influenced and supported by many people. First and most importantly, I would like to thank my advisor, Amber Miller. She has been my strongest advocate, a tireless troubleshooter and role model of the highest calibre. Her approach to experimental science, reflective management and leadership under pressure have been shining examples from which I have been fortunate to learn.

I also cannot thank Michele Limon enough for his continual insight, hard won in past projects. Despite his own work and deadlines, Michele has always been available to review, rethink and improve upon any aspect of the experiment. More often than not, I have come away from my conversations with him knowing more, not just about my problem, but about its context in the larger community of experimental cosmology.

Other group members have similarly challenged and helped me in many ways. I am enormously grateful to Britt Reichborn-Kjennerud, Daniel Chapman and Joy Didier for helping to make EBEX successful and my own part of it better than I could have done without their help. My early exposure to experimental cosmology would not have nearly as fruitful or interesting without the excellent assistance and friendship of Will Grainger and Ross Williamson who brought a certain *joie de vivre* to the lab, despite their British roots.

Work on EBEX was always collaborative, with multiple institutions contributing their expertise. I was fortunate to be able to work side-by-side with some truly remarkable researchers from the University of Minnesota, McGill University, Brown University and UC Berkeley. Kyle Zilic was always an excellent man to have around



when things malfunctioned. His encyclopedic knowledge of our cryostat saved the day more than once during our integrations. Jeff Klein and Asad Aboobaker were inspirational in their patience and willingness to help, no matter the job. It says volumes about their competency that they had time to help me fix an issue in my systems because theirs were already working.

The work of our bolometer team cannot be overstated. As one of the first kilo-pixel microwave telescopes, EBEX’s “bolo monkeys” were breaking new ground in every aspect of their work. I was fortunate to overlap with Kate Raach, Kevin MacDermid and François Aubin in many aspects of my work. From tackling structure with Kevin to troubleshooting downlinks with Kate and tracing timing glitches with François, my work would be substantially less without their continued and abiding assistance, good humor and support.

Additionally, EBEX has the good fortune to be headed by an outstanding group of Co-Is. In addition to Amber Miller, Shaul Hanany, Matt Dobbs and Gregory Tucker have been patient, insightful and engaged not just in the experimental outcomes but in the development of their students as scientists and individuals. I am deeply indebted to each of them.

Finally, my deepest gratitude to my family, Beth and Sefira, whose love, forbearance and support are, in measure, my greatest assets. My life would be less without them in it.

# Chapter 1

## Introduction

“The first step is undertaken lightly, pleasantly,  
and with your soul in the sky; it is the  
five-hundredth that counts”

---

— Hilaire Belloc, *The Path to Rome*

The cosmic microwave background (CMB) was first observed in 1964 by Penzias and Wilson[90]. Since then, cosmologists have recovered a wealth of information encoded in its signal, allowing us to draw conclusions about the shape, content and history of the Universe. In 2002, the Degree Angular Scale Interferometer (DASI) provided the first evidence of polarization in the CMB[64], opening a new avenue of investigation for probing the early universe.

The E and B Experiment (EBEX) is a balloon-borne polarimeter, sensitive to millimeter wavelengths. Its primary science goal is to detect evidence of a stochastic, gravitational wave background field predicted by inflationary models of the universe. EBEX is unique among balloon-borne observations in that it has an angular sensitivity that will allow it to map sub-degree variations in the polarization[112]. It is competitive with measurements made by the *Planck* satellite and will provide a unique characterization of the polarized foreground dust.

Compared with a satellite experiment, ballooning offers a lower-cost, faster devel-

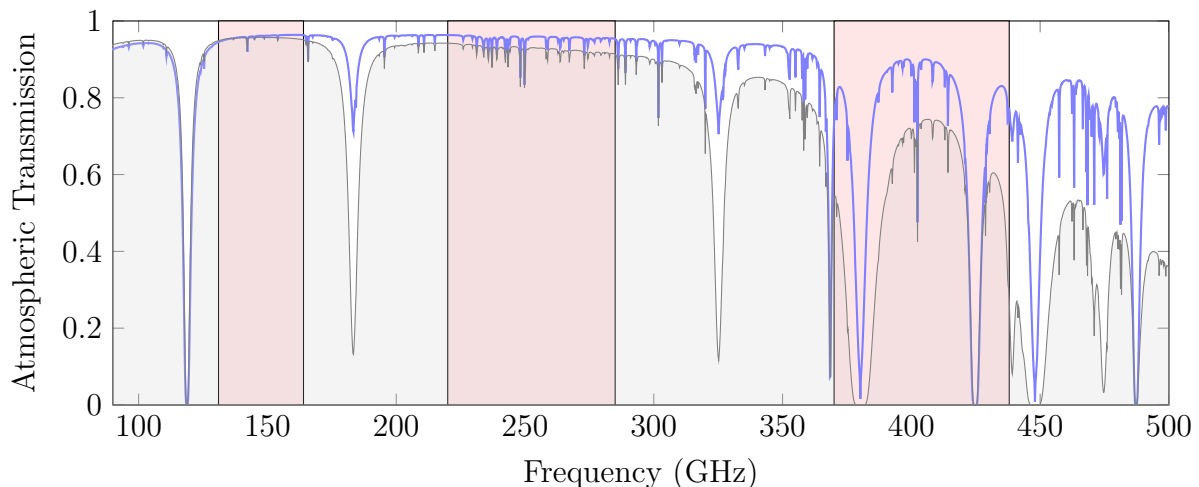


Figure 1.1: Atmospheric transmission prediction for EBEX at 35,000m (—) and observations from the Chajnantor Plateau in Chile(—), a common site for ground-based CMB experiments. Predictions use precipitable water vapor measurements from the Atacama Large Millimeter Array[98] and are calculated using Moliere forward modeling[115]. Transmission in higher frequencies is substantially suppressed for ground-based observatories relative to balloon-borne platforms.

opment platform from which to observe. As it flies in a virtual vacuum, it provides a suitable testbed for technologies prior to their incorporation in future satellite missions.

By observing from 35,000m, EBEX avoids the majority of the Earth’s atmosphere as shown in Fig. (1.1), decreasing overall loading on the detectors and thereby increasing instantaneous sensitivity. This allows EBEX to reach frequencies required to characterize foreground emissions from Galactic dust. The higher instantaneous sensitivity enabled by stratospheric observations allows EBEX to achieve in an 11-day flight a map-level sensitivity that would take an equivalent ground-based experiment multiple years to produce.

The higher payoff carries with it higher risks as well. Every hour at float is a substantial fraction of the observation time and so actions must be optimized to minimize time spent in housekeeping and calibration. Additionally, the possibility of

single-point failure is ever present. Systems must be robust against physical, electrical and cosmic interference. To the extent possible, they must be designed to reduce the risk of human error.

This thesis presents work performed on the EBEX experiment to create a robust, autonomous platform from which to observe the CMB. We will review the hardware platform but focus on the details of software implementations in the system design. Additionally, we present new methods of attitude calibration and reconstruction for balloon platforms and apply these methods to the EBEX long duration flight data.

Chapter 2 will provide an overview of the science motivating the EBEX experiment and the main goals of the long duration flight. Chapter 3 details the EBEX instrument with particular focus on the software systems and their performance. Chapter 4 introduces the attitude reconstruction system and details test metrics and its performance in a simulated environment. Chapter 5 constructs a preliminary reconstruction using data from the EBEX long duration balloon (LDB) flight. Additionally, sensor data stream cleaning methods are detailed and new performance metrics are derived and applied.

# Chapter 2

## EBEX Science

“Nothing preserved its form  
Each thing opposed the rest,  
Since in one frame,  
The cold with hot things fought”

---

— Ovid, *Metamorphoses* 1

### 2.1 Concordance Cosmology

In the past 30 years, the field of cosmology has rapidly advanced from initial observations to precision measurements sensitive enough to constrain the basic framework of how we view our Universe and its history. We have moved into an era of ‘Concordance Cosmology’ where experiments probing entirely separate metrics yield answers in profound agreement with each other.

In the late 1990s, this idea coalesced with two types of landmark measurements made by supernova observation projects[105, 92] and CMB experiments[26, 80, 105, 8, 125, 108]. As shown in Fig. (2.1), the two types of experiments made highly complimentary constraints on the relative abundance of matter and energy in the Universe. The overlap between the two brought to the forefront the idea that the

Universe is composed of roughly 70% dark energy, 25% dark matter and only 5% observable, baryonic matter.

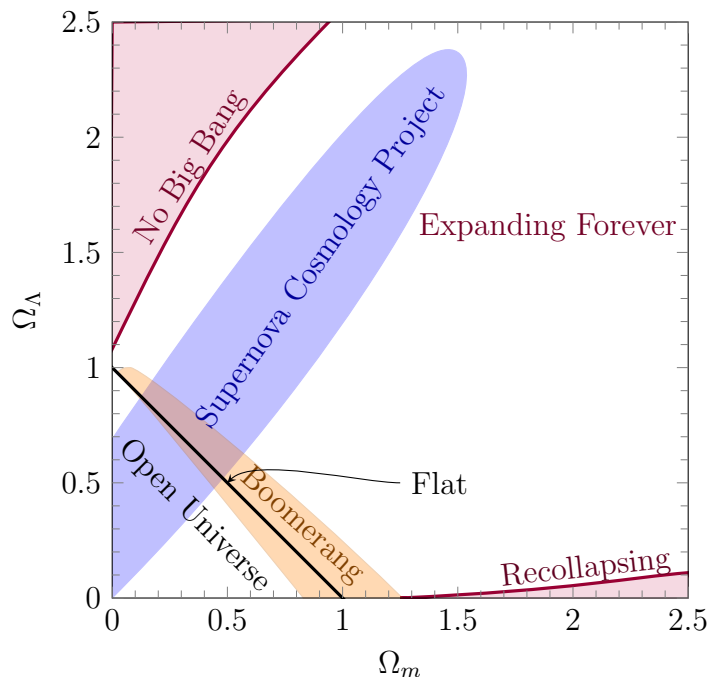


Figure 2.1: The overlap of  $2\text{-}\sigma$  constraints in the  $\Omega_\Lambda - \Omega_m$  field made by the Supernova Cosmology Project and the Boomerang CMB Experiment.  $\Omega_\Lambda$  is the fractional contribution of Dark Energy to the total energy density in the Universe.  $\Omega_m$  is the contribution of matter (both baryonic and dark). Along the  $\Omega_m + \Omega_\Lambda = 1$  line, we must have a spatially flat universe. Above and below this limit are closed and open universe geometries, respectively. Data from [92, 80].

We have since refined these values with larger and deeper supernovae surveys[137, 9, 6], measurement of the baryonic acoustic oscillation peak in large sky surveys[32, 91, 11] and ever finer surveys of the cosmic microwave background radiation. The state of experimental cosmology today is one of profound agreement that allows us to probe the nature of the very beginning of the Universe.

This concordance is based on four main pillars: 1) the Universe is expanding; 2) the Universe is largely flat; 3) matter in the Universe is predominately dark and 4) over 72% of the content of the Universe is an energy of unknown source. As agreements go, these statements are startling in their specificity while at the same time profoundly

ignorant. We know where to look but we do not yet know what we are seeking.

### 2.1.1 Expanding Universe

In 1929, Edwin Hubble published a seminal paper describing a linear relationship between the recessional velocity of extragalactic galaxies (at the time, referred to as nebulae) and their distances[55]<sup>1</sup>. This uniform expansion matches the predictions of a isotropic, homogeneous, simply-connected universe assumed by the Friedmann-Lemaître-Robertson-Walker metric[39, 70, 107, 132]:

$$ds^2 = -dt^2 + a^2(t) \left[ \frac{dx^2}{1 - Kx^2} + x^2 (d\theta^2 + \sin^2 \theta d\phi^2) \right] \quad (2.1)$$

Here,  $s$  is the co-moving spacetime distance of between world lines shown in Fig. (2.2). The variable  $t$  represents a proper time, corresponding to the time component of Einstein's general relativistic field equations<sup>2</sup>  $g_{00} = -1$ . The scale factor  $a(t)$  encodes time-evolution of the metric. We normalize this such that at the present time  $a_0 = 1$ .

Note that the spherical line element term on the right in parentheses provides an isotropic symmetry about the spacelike coordinate  $x = 0$ . We may also calculate the Ricci scalar as  $R^{(3)} = 2K/a^2(t)$ , independent of geometrical position, ensuring homogeneity. This homogeneity admits three classes of solutions for our constant of curvature  $K$ .  $K < 0$  describes a closed universe,  $K > 0$  describes an open universe and for a flat universe, the curvature and, by extension, the Ricci scalar will be exactly zero.

The last assumption of simple connectedness, however, depends on the uniqueness

---

<sup>1</sup>This relationship and a rougher calculation based on a smaller set of galaxies had been previously proposed by Georges Lemaître in 1927[68]. Unfortunately, the publication in a small, Belgian journal was not widely read and Lemaître himself removed the calculation of the relationship from the 1931 english translation for MNRAS[75], citing newer observations – presumably Hubble's.

<sup>2</sup>We will use here the notation  $R_{\mu\nu} - \frac{1}{2}g_{\mu\nu}R = 8\pi GT_{\mu\nu}$  but without justification. Interested readers are directed to the excellent and comprehensive book by Misner et al. [83] or briefer and more cosmologically-focused work by Liddle and Lyth [73].

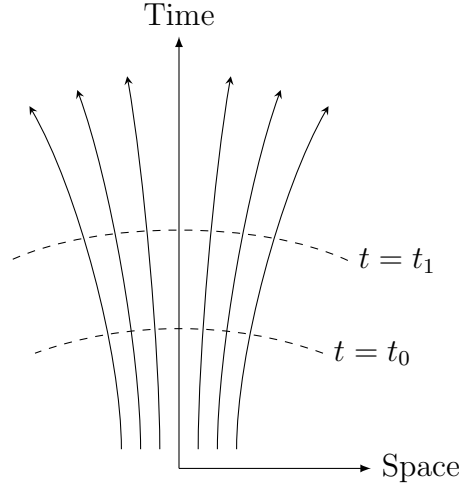


Figure 2.2: Spacelike surfaces propagate outward in time, while the comoving distances between fundamental observers’ world lines’ always increase. The world lines are then said to “carry” the comoving spatial coordinates. Figure adapted from [84].

of space-time points. This is referred to as the Weyl Postulate, which states that the world lines of fundamental observers in the Universe form non-intersecting 3-bundle geodesics orthogonal to a series of spacelike surfaces. Each spacelike surface is then uniquely described by a proper time as shown in Fig. (2.2).

Allowing a time evolution of the geometry (encoded by  $a(t)$ ), we must alter the standard continuity equation of thermodynamics<sup>3</sup>. Taking the pressure  $P$  and density  $\rho$  of a spacetime position, measured by a comoving observer, we find this expressed in terms of the scale factor as:

$$a \frac{d\rho}{da} = -3(\rho + P) \quad (2.2)$$

In a universe with multiple constituent components, we can quickly deduce limits in Eq. (2.2). In the case of matter domination, then the mean-square velocity will be  $v^2 \ll 1$  and therefore  $P \ll \rho$ , which gives  $\rho_m \propto a^{-3}$ . Alternatively, for radiation

---

<sup>3</sup>In a comoving volume  $V$ , this can be expressed as  $\dot{\rho} = -\frac{\dot{V}}{V}(\rho + P)$  where  $\rho$  is the local density and  $P$  the local pressure. Absent a heat flow, this can also be expressed in the more familiar  $dE = -PdV$  with  $E = \rho V$ .



domination, the mean-square velocity will approach 1 and  $\rho \rightarrow P/3$ . This gives  $\rho_r \propto a^{-4}$ . Taking this variability of the exponent into a new parameter  $w \equiv P/\rho$ , we can write

$$\rho_i \propto a^{-3(w_i+1)} \quad (2.3)$$

Now, taking only the time-time component of Einstein's field equations, we arrive at the first Friedmann Equation

$$\left(\frac{\dot{a}}{a}\right)^2 = H^2 = \sum_i \frac{\rho_i}{3M_{\text{Pl}}^2} - \frac{K}{a^2} \quad (2.4)$$

where  $M_{\text{Pl}}^2 \equiv 1/8\pi$  is the Planck mass<sup>4</sup> and  $H$  is the Hubble parameter that encodes the rate of change of the scale factor. Taking the current scale factor to be 1, we define the Hubble constant  $H_0 = \dot{a}(0)$ .

The best constraints on the Hubble constant come, appropriately enough, from the Hubble Space Telescope (HST). It measures the redshift of near-by (in cosmological terms) type Ia supernovae (SNe Ia) and compares their recessional velocity with their distance from the Earth. To obtain the distance, Riess et al. [106] follow much the same procedure as Hubble did for his original measurements. The distances are calibrated by using a period-luminosity relationship of Cepheid variable stars known as the Leavitt Law [67]. Using these distances, the redshifts of nearby SNe Ia are plotted against distance.

The current best-fit value for  $H_0$  using only this direct measurement is  $73.8 \pm 2.4 \text{ km s}^{-1} \text{ Mpc}^{-1}$ , an uncertainty of 3.3% [106]. An independent prediction may also be made using fits to the CMB data, with the best constraint currently from the Planck experiment, which reported  $67.3 \pm 1.2 \text{ km s}^{-1} \text{ Mpc}^{-1}$  [95].

Although the discrepancy between the HST direct measurements and the CMB model fit is significant at the  $2.5\sigma$  level, it remains to be seen whether this tension indicates potential deviations from the Concordance Model or whether it will be

---

<sup>4</sup>Assuming, for convenience, throughout this section that  $c = \hbar = G = 1$

resolved by revising assumptions made in the Planck dataset.

### 2.1.2 Flat Universe

Returning to Eq. (2.4), we can see that the Hubble constant will be dependent on the curvature of the universe as well as the constituents' pressure. The curvature is exceptionally well-constrained by the power spectrum of the CMB while statistical constraints on baryonic oscillation or lensing of the CMB itself can constrain the pressure term.

In each case, if we take the value of  $H_0$  as given, we can determine the best fit of the relative distributions of  $\Omega_\Lambda$  and  $\Omega_m$ , the dark energy and matter fractional densities to our model of the Universe. One of the first demonstrations of the power of this combined constraint was shown in Fig. (2.1). The most recent measurements are incorporated in Fig. (2.3). The emerging picture is one that strongly favors a spatially-flat universe.

### 2.1.3 Hot, Dense Early Universe

Once you observe an expanding universe, it is only natural to run this observation backwards in time to arrive at a prediction of a singular point of origination or “unique quantum” as predicted by Lemaître [69]. While observations have not yet penetrated to earliest times, work by Gamow (later revised and corrected by Alpher and Herman)[4, 41] expanded this idea to predict a relic light that should exist from the point at which the universe was sufficiently cooled to condense matter and thereby release much of the radiation pressure as photons.

This light was first detected by Penzias and Wilson in 1958. Its existence provided the strongest evidence yet for the model of a “Hot Big Bang.” The results from the COBE FIRAS satellite in the early 1990s measured this radiation temperature to be  $2.728 \pm 0.004\text{K}$ , famously showing the blackbody spectrum without visible error bars as they were smaller than the width of the fitted line[36].

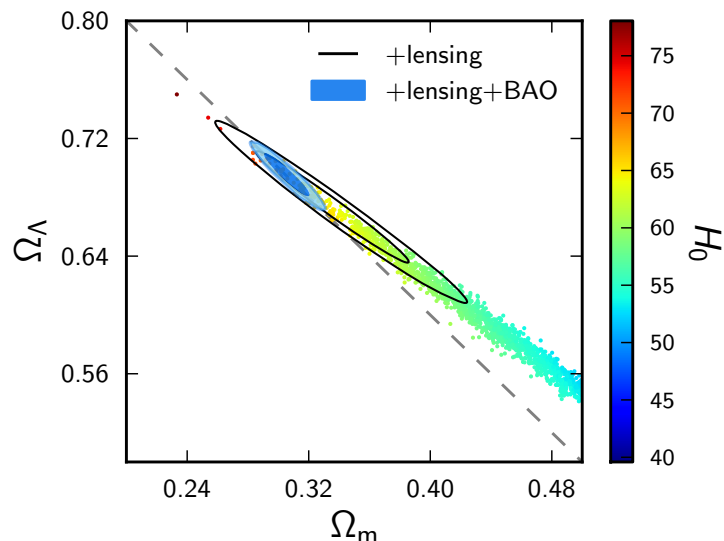


Figure 2.3: The dots represent realizations of the Planck satellite first-year temperature data set, color-coded by value of  $H_0$ . The solid contours represent confidence intervals from applying the lensing power spectrum simultaneously observed by Planck while the solid blue contours are the result of combining the lensing information with observations from Baryonic Acoustic Oscillation (BAO) surveys. The dashed line represents a spatially flat universe. Figure from [95].

This relic light is referred to as the Cosmic Microwave Background radiation (CMB). The uniformity of its spectrum shows a homogeneity of the Universe on the level of 1 part in  $10^6$  at the surface of last scattering, approximately 400,000 years after the Big Bang. The deviations from uniformity encode the matter density fluctuations that would eventually condense to form structure in the Universe.

### 2.1.4 Dark Matter Universe

The first evidence for dark matter was observed in 1932 by Zwicky during his survey of galaxy clusters. He found that the measured rotational speed could not be explained by the amount of matter that was observed[140], leading to the proposal of some form of “dark” matter. While many models for this particle or particles were proposed, observations at the time were unable to characterize their properties or determine the prevalence in the Universe.

Observations of the CMB opened a new avenue of approach. In 1970, [Sunyaev and Zeldovich](#) as well as [Peebles and Yu](#) independently predicted the existence of fluctuations in the temperature of the CMB on the order of 1 part in  $10^4$ . When observations did not find this magnitude of deviation, Peebles revised his calculations and suggested that the effect of weakly-interacting, or cold, dark matter could explain the discrepancy[89]. The revised temperature fluctuations were finally observed in the COBE FIRAS CMB data, providing strong support to the theory of a “Cold Dark Matter” (CDM)[36].

By observing the shape the CMB power spectrum anisotropies (cf. Fig. (2.5)), we constrain the relative fractions of matter allowed in the model. While the location of the first peak in the temperature power spectrum is coupled strongly to curvature, the physical density of matter in the universe controls the relative sizes of odd versus even peaks. This odd-even dichotomy is an artifact of the power spectrum which measures the absolute value of temperature deviation, thus the peaks will represent half-cycles in an oscillation.

The initial fluctuations from quantum uncertainty cause overdense regions to form out of an otherwise homogeneous plasma. Under gravitation, this region will become denser until the photon pressure is sufficiently large so as to press matter back out of the gravitational well, rarifying it. Thus the first peak represents the scale that had just become compressed at the moment of decoupling. Conversely, the second peak represents 1/2 cycle further and thus the first rarified scale.

Since dark matter particles are, by definition, weakly interacting with radiation, they do not experience substantial photon pressure and therefore only become more dense with time, suppressing the height of the second and subsequent peaks. Thus by measuring the relative heights of the first and second peaks in the temperature power spectrum, we may place constraints on the relative abundances of light and dark matter.

### 2.1.5 Dark Energy Dominated Universe

Perhaps the most enigmatic pillar of the concordance model is the existence of a dark<sup>5</sup> energy that accounts for more than 68% of the total effective energy density of the Universe.

In an analogous method to that described for the determination of  $H_0$ , SNe Ia were used to characterize the recessional velocity of different points in space time. While  $H_0$  relies on the use of local SNe, constraining dark energy requires surveying much more distant (and by extension older) SNe. By mapping the distribution of velocities relative to distance, Perlmutter et al. [92], Riess et al. [105] showed that the Universe has entered a period of accelerating expansion. The best measurements currently of the dark energy equation of state  $w$  come from combined analyses of the SNe Ia data, CMB observations and full-sky statistical analyses to constrain  $w = -1.010^{+0.046}_{-0.045}$  [18].

### 2.1.6 Concordance Constraints

Given the four pillars above, cosmologists have come into broad agreement on the primary parameterization of the Universe. This set of 6 parameters and values is generally referred to as the  $\Lambda$ CDM model, where  $\Lambda$  represents the dominance of dark energy at current time while CDM refers to cold dark matter dominated universe at early times.

Measurements of the CMB have provided the tightest constraints on many of these parameters to date, testing the limits of the model and giving clues as to where deviations might be found and their potential impact. Tab. (2.1) gives the current best estimate for these parameters from the *Planck* satellite experiment.

---

<sup>5</sup>While the term “dark” was originally applied to dark matter as it did not interact strongly with radiation, its application to dark energy merely signifies that little is known about it and we only observe its effects through secondary measurements

Parameter	68% limits	Description
$\Omega_b h^2$	$0.02205 \pm 0.00028$	Baryon density today
$\Omega_c h^2$	$0.1199 \pm 0.0027$	Cold dark matter density today
$100\theta_{\text{MC}}$	$1.04131 \pm 0.00063$	$100\times$ angular size of the sound horizon
$\tau$	$0.089^{+0.012}_{-0.014}$	Thomson scattering optical depth from reionization
$n_s$	$0.9603 \pm 0.0073$	Scalar spectrum power-law index
$\ln(10^{10} A_s)$	$3.089^{+0.024}_{-0.027}$	Log power of the primordial curvature perturbations
$\Omega_\Lambda$	$0.685^{+0.018}_{-0.016}$	Dark energy fraction of the critical density today
$\Omega_m$	$0.315^{+0.016}_{-0.018}$	Total matter fraction of the critical density today
$\sigma_8$	$0.829 \pm 0.012$	RMS matter uctuations today in linear theory
$z_{\text{re}}$	$11.1 \pm 1.1$	Redshift at which Universe is half reionized
$H_0$	$67.3 \pm 1.2$	Current expansion rate in $\text{km s}^{-1}\text{Mpc}^{-1}$
$10^9 A_s$	$2.196^{+0.051}_{-0.060}$	
$Y_p$	$0.24770 \pm 0.00012$	Fraction of baryonic mass in helium
Age/Gyr	$13.817 \pm 0.048$	Age of the Universe
$z_*$	$1090.43 \pm 0.54$	Redshift at which optical depth=1
$100\theta_*$	$1.04147 \pm 0.00062$	$100\times$ angular size of the sound horizon at $z = z_*$
$z_{\text{eq}}$	$3391 \pm 60$	Redshift of matter-radiation equality

Table 2.1: Constraints on parameters of the  $\Lambda$ CDM model using *Planck* and WMAP 9-year polarization data. The top section contains the basic model parameters while the bottom section contains the derived parameters. Data from [95].

## 2.2 Inflation

While the data from CMB and other experiments agree on virtually every aspect of the  $\Lambda$ CDM model, the implications are far from what might have been anticipated, and lay out the description of a finely-tuned universe. This tuning suggests (but certainly does not require) a more fundamental, governing model out of which our universe is either at a highly probable state or merely one arbitrarily many possible universes, special only because of chance[135, 31, 42].

The Universe we observe is flat to better than 1 part in 1000 and homogeneous to an order of magnitude greater. Homogeneity on this scale should be phenomenally unlikely, given that, at the point of recombination, the Universe was not in causal contact. One favored group of theories to explain this seeming discrepancy are termed “inflation.” While there are many varieties of inflationary theories, they share the core idea that in the first fractions of a second after the big bang, quantum fluctuations that were initially in causal contact with each other were expanded to cosmic scales, seeding the structure of the Universe. These theories also provide for the curvature to be flattened by inflationary pressure.

While appealing for its ability to resolve known observational conflicts, the simplest theories of inflation also provide a theoretically observable outcome in the form of tensor perturbations to the space time metric. Unlike scalar perturbations, which can be created by the presence of density fluctuations, tensor perturbations in the early universe should only occur as the result of the gravitational waves that would be generated in inflationary theories. The level of tensor perturbation is generally quoted as a ratio  $r$  between the amplitude of the tensor and scalar perturbations.

An inflationary gravity wave (IGW) would imprint on both CMB temperature and polarization anisotropies, increasing the temperature anisotropy at low- $\ell$ . By combining CMB data with BAO observations, Planck has limited  $r < 0.15$  at 95% confidence. However, cosmic variance—the fact that we have only one sky to observe—places a fundamental lower limit on the the constraint that may be placed on  $r$  from

this effect on large angular scales. Further limiting this constraint is the fact that any scale-dependent effect will be degenerate with the primordial tilt  $n_s$ , the scale dependence of the temperature spectrum. This degeneracy is depicted in Fig. (2.4). Note that many models for inflation are excluded based solely on the upper bound of  $r$ , irrespective of a detection.

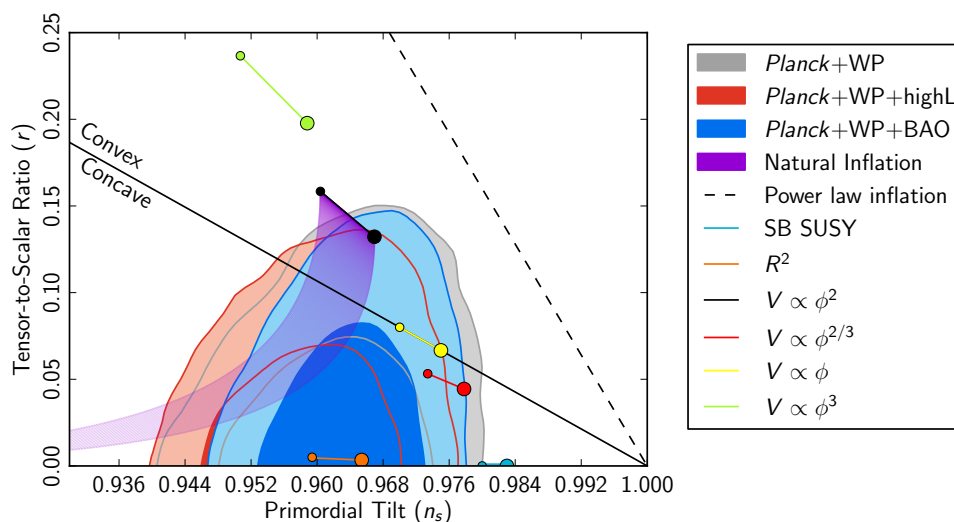


Figure 2.4: Marginalized 1- and 2- $\sigma$  confidence levels for  $r$  and  $n_s$  from combining *Planck* temperature data with BAO and WMAP polarization data. Selected inflationary models are overplotted for comparison. Figure from [21].

To further limit  $r$ , it is advantageous to probe using polarization data as a tensor perturbation will leave a unique signature on the polarized spectrum that scalar perturbations do not. EBEX is designed to probe these perturbations with the goal of measuring or limiting  $r < 0.1$ .

## 2.3 Cosmic Microwave Background Radiation

The CMB represents one of the cleanest probes of the early universe available. The photons observed today were last scattered by matter a mere  $\sim 350,000$  years after the big bang. While virtually an ideal blackbody, the CMB contains anisotropies on the level of 1 part in  $10^5$ . They reveal information about the state of the Universe, its



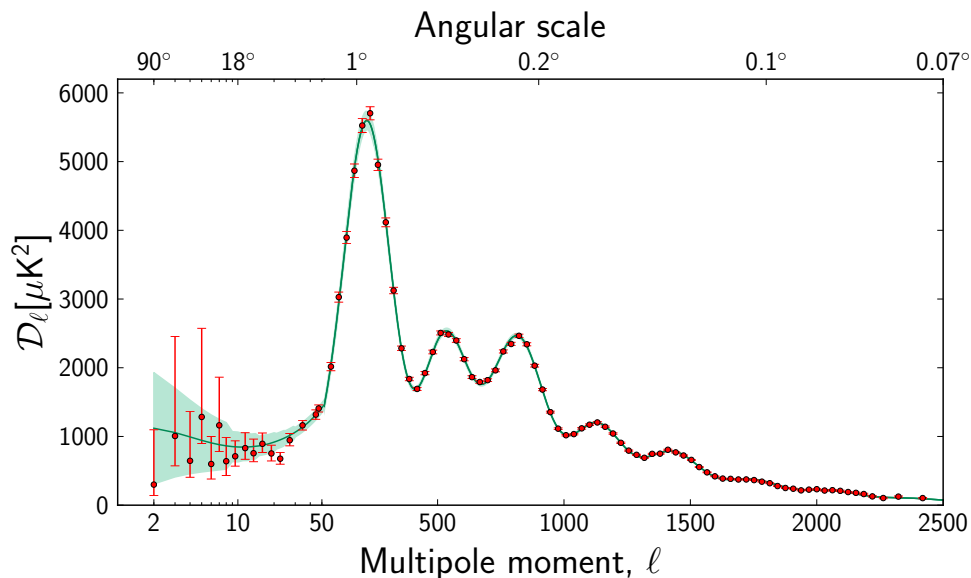


Figure 2.5: Measured temperature fluctuation power spectrum of the CMB by Planck with a best-fit  $\Lambda$ CDM model overplotted. Cosmic variance is shown as a shaded green area about the best-fit line. Figure from [94].

constituent components and relative densities that would evolve into the Universe we observe today.

Individual anisotropies do not, however, carry useful cosmological information. Only by averaging over a statistically significant sample of the CMB do we currently derive useful data that can inform our understanding of the Universe. By decomposing the fluctuation statistics on the sky into spherical harmonics as

$$\frac{\Delta X}{X} = \sum_{\ell m} a_{\ell m} Y_{\ell m}(\theta, \phi) \quad (2.5)$$

we find a convenient expression of the amount of anisotropy at a given multipole moment  $\ell$  as  $C_\ell = \langle |a_{\ell m}|^2 \rangle$ . In this formulation, which we may apply to any two-point spherical anisotropy, higher multipoles correspond to smaller angular separations on the sky.

Each of the six parameters of the  $\Lambda$ CDM model is encoded in the structure of the power spectrum created by mapping the  $C_\ell$  statistic across  $\ell$ -space. By measuring the

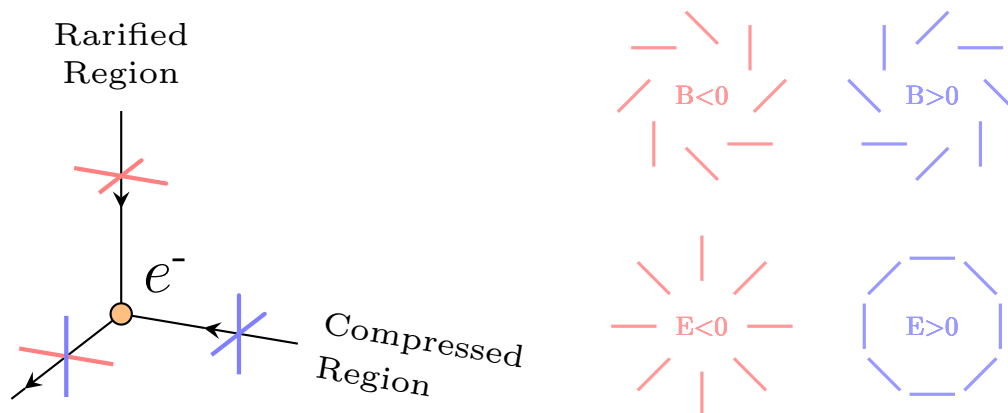


Figure 2.6: *Left:* Quadrupolar density distributions in the early universe give rise to linear polarizations in the CMB. As radiation from the compressed and rarified regions approach an electron, their light is respectively blue- and red-shifted by traveling into or out of a gravity well. When they scatter off the electron, only the component orthogonal to the emission direction is scattered, resulting in a partial polarization. *Right:* examples of pure E-mode and pure B-mode polarization. Note that when reflected about a line passing through their centers, the E-mode pattern remains unchanged while the B-mode is reflected. Figures adapted from [139, 53].

two-point temperature fluctuations, shown in Fig. (2.5), CMB experiments constrain the values of all six parameters. However, there are degeneracies in parameter expression when considering only temperature anisotropies. Breaking the degeneracy is often accomplished by combining CMB analysis with other cosmological probes such as direct measurements of the Hubble Constant, Lyman  $\alpha$ -Forest observations, SNe Ia observations or large-sky galaxy surveys. Degeneracies can also be broken in the CMB alone by expanding observations to include polarization anisotropies.

### 2.3.1 CMB Polarization

Polarization of the is generated by the combination of quadrupolar density perturbations and Thompson scattering between photons and electrons, shown diagrammatically in Fig. (2.6). This means that polarization cannot be generated after

recombination<sup>6</sup> as the electrons become bound to neutral atoms. Additionally, it cannot be sourced by perturbations prior to recombination as the photon-baryon fluid was so dense that the multiple scatterings erased any prior anisotropies that may have existed. Thus the requirement is for density perturbations *and* Thompson scattering at the moment of recombination. The relative unlikelihood of these simultaneously occurring implies—and has been borne out by experiment—that the polarization amplitude should be relatively small.

Both scalar and tensor perturbations will generate polarization by anisotropic Thompson scattering. In the case of scalar perturbations, the resultant polarization will be solely *E*-mode, or even-parity. For tensor perturbations, both *E*-mode and *B*-mode polarization will be generated. Because a stochastic gravitational wave background is predicted by the theory of inflation and gravitational waves are the only known source of tensor perturbations in the early universe, we can look for the presence of gravitational waves by decomposing the CMB polarization into *E* and *B* modes to test inflationary theories.

To date, while multiple experiments have measured the *EE* correlation power spectrum, the presence of *B* mode polarization induced by inflationary gravitational waves has not been observed. Recently, a second source of *B* mode polarization was detected by the South Pole Telescope[52], a ground-based CMB experiment. As the CMB photons stream through space, they will be gravitationally lensed at late times by the presence of structure. This lensing rotates the primordially-generated *E*-mode polarization into a *B*-mode signal. Fortunately, because this occurs at late times, the Universe has expanded causing the characteristic length scale of lensing *B*-modes to be distinct from that of inflationary gravity wave-induced *B*-modes as shown in Fig. (2.7).

The eventual detection of gravity wave-induced *B*-modes is not guaranteed but even

---

<sup>6</sup>The exception to this statement is during the period of re-ionization, which will be addressed later

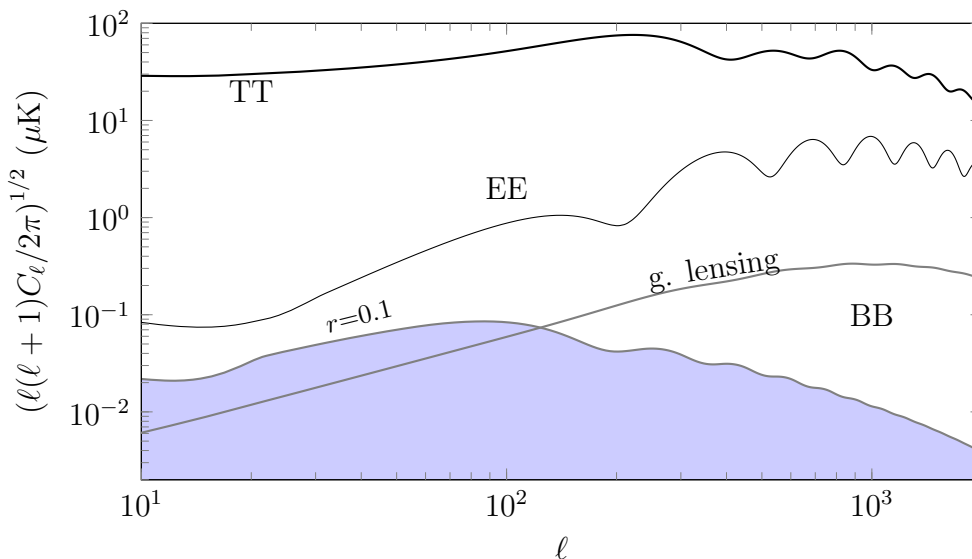


Figure 2.7: Anisotropy power spectra for temperature, E-mode and B-mode polarizations. The curves are generated using CAMB[17] and the best-fit values for the 6-parameter  $\Lambda$ CDM model from *Planck*[95]. The BB curves are separated into the contribution from gravitational lensing (labeled g. lensing) and the range of possible tensor-to-scalar ratios.

the lack of a detection is useful as shown in Fig. (2.4). By decreasing the acceptable values for  $r$ , we simultaneously decrease the energy scale of inflation, placing strong constraints on models for the early universe. The current state of power spectrum constraints from CMB polarization experiments is shown in Fig. (2.8).

## 2.4 EBEX B-Mode Science

EBEX is designed to detect not just the low- $\ell$  gravity wave B-mode signal, but also the high- $\ell$  lensing signal. In this manner, it is unique among sub-orbital balloon CMB experiments. By measuring both signals, EBEX is capable both of performing systematic cross-checks as well as delensing the CMB to recover lower limits on the

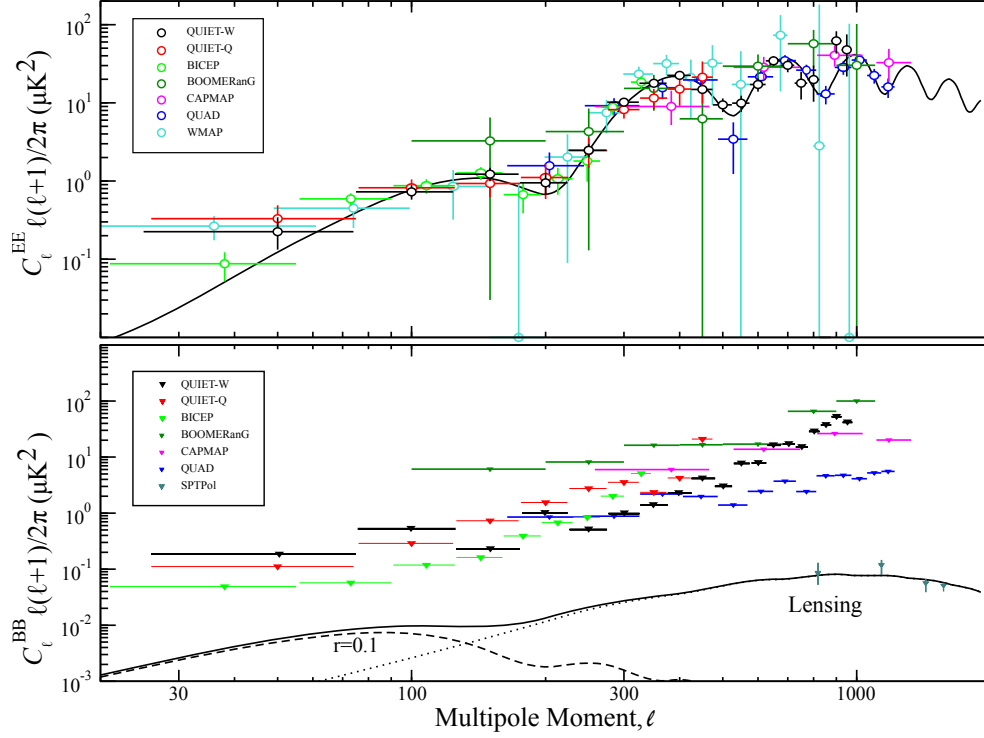


Figure 2.8: EE and BB power spectra compiled for CMB polarimetry experiments to date. Adapted from [22, 52]

IGW signal.

Measuring low- $\ell$  modes requires a relatively large scan area as  $\ell$  is inversely proportional to angular separation on the sky. To accomplish this, EBEX’s scan patch, shown in Fig. (2.10) is  $\sim 350^2$  degrees. At high- $\ell$ , however, we are limited by angular resolution. EBEX is designed with  $8'$  resolution across all frequency bands, allowing it to probe angular scales up to  $\ell \simeq 1000$ .

### 2.4.1 Scan Strategy

EBEX was designed to utilize a raster pattern in azimuth and elevation, with overlapping segments covering the same patch of sky during geodetic ascent and descent shown in figure 2.10a. At our flight latitude, there is a non-zero angular offset between right ascension and azimuth. This allows us to interlock scan patterns and

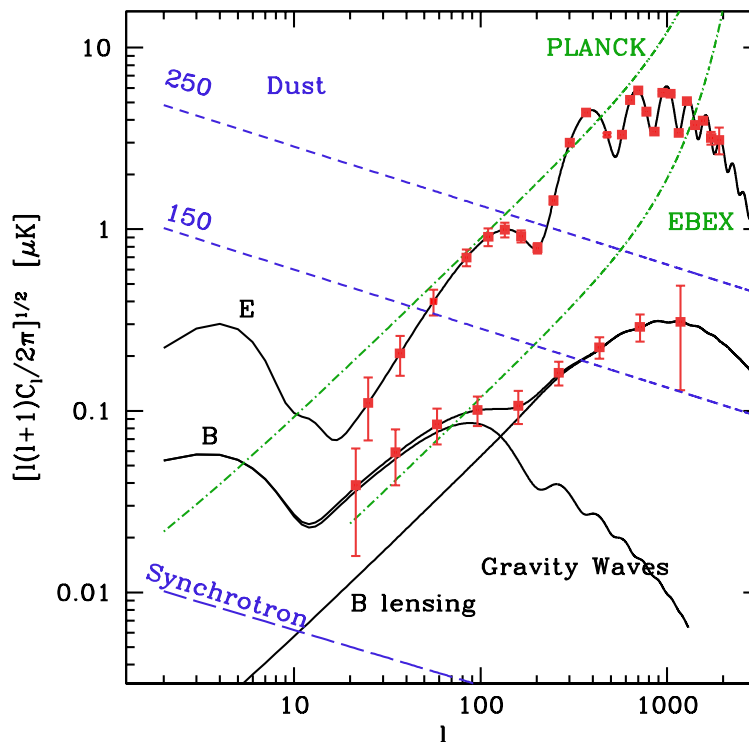
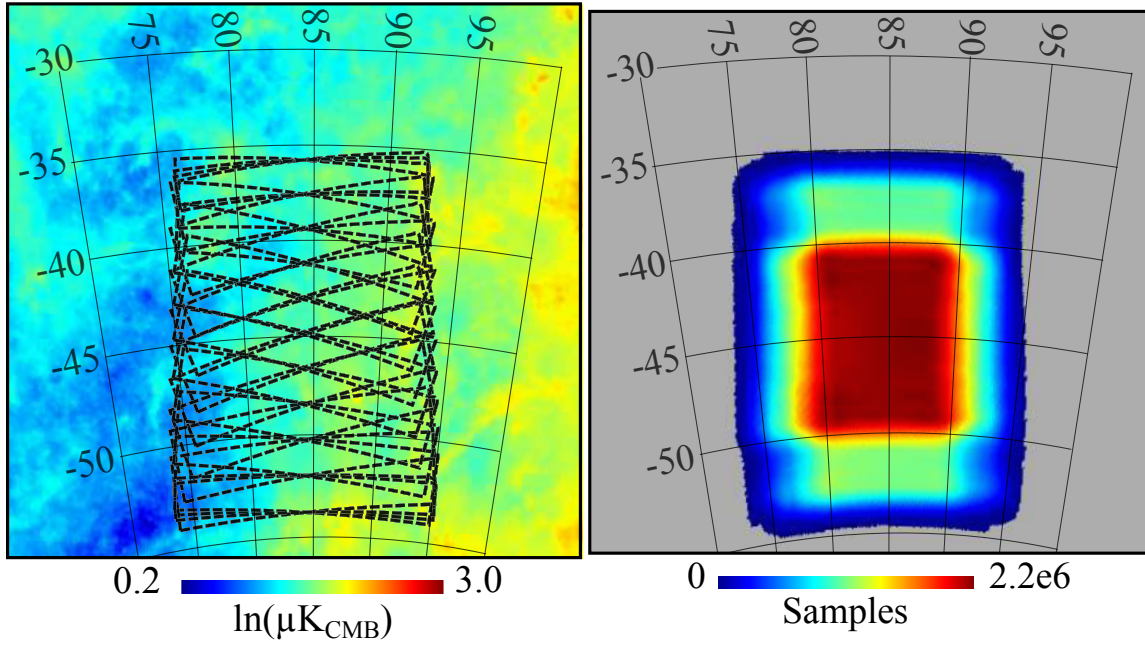


Figure 2.9: The expected  $1\text{-}\sigma$  determination on the EE and BB correlation power spectra for EBEX after a successful, 11-day long-duration balloon flight (red points). The measurement points of the EE spectrum are cosmic variance limited between  $20 \leq \ell \leq 1500$ . The dashed purple lines show projected B-mode polarization from Galactic dust emission (top) and synchrotron emission at 150GHz (bottom). While synchrotron will be negligible, dust emission must be removed prior to analysis. Beam-size pixel sensitivities for both *Planck* and EBEX are shown as dashed blue. Figure from EBEX science proposal.

thereby revisit the same patch at different relative camera orientations, providing useful cross-checks for polarization measurements.

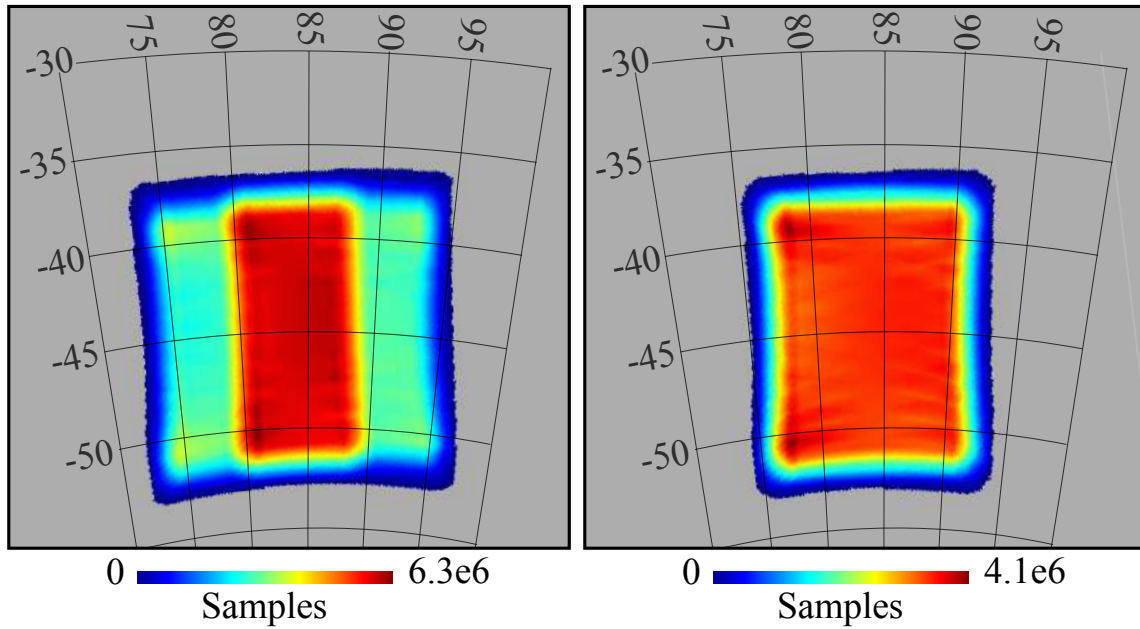
The spatial offsets of the multiple wafers over the focal planes results in slightly different hitmaps for each of the three frequency bands as shown in figures 2.10b - 2.10d.

The raster pattern is produced by scanning over azimuth in positive and negative directions multiple times before adjusting elevation and repeating. The length of an azimuthal throw is approximately  $17^\circ$ , while each elevation step is chosen to be  $1/3$  FWHM (full width at half-maximum) of the full EBEX beam. This process traces



(a) Raster Segments projected onto FDS8 dustmap at 150GHz

(b) 150 GHz Hitmap



(c) 250 GHz Hitmap

(d) 410 GHz Hitmap

Figure 2.10: Anticipated scan strategy for EBEX. The overlapping raster segments in (2.10a) maximizes cross-linking and generates uniform hitmaps shown in (2.10b)-(2.10d). Each detector makes 22 maps of the entire area during an 11 day flight.

out a rough rectangle in azimuth and elevation  $17^\circ \times 2^\circ$ .

The scan speed is chosen to minimize the time between absolute vector star camera attitude readings (taken at the end of each throw) while at the same time constraining the beam bandwidth to less than one harmonic of the half-wave plate rotation. These combine to optimize our azimuthal scan speed at  $0.4^\circ$  per second.

After the rectangular segment is covered by a single pass, the system moves  $0.5^\circ$  in RA to scan a different segment of the full patch, spatially separating the sequential scans so as to minimize systematic contamination. Every 4 hours, this pattern is repeated, resulting in the even cross-linking shown in figure [2.10a](#).

## 2.5 Summary

The detection of primordial gravity waves would provide strong evidence for inflationary models of the big bang. The CMB offers a unique method of probing their existence by measuring the B-mode polarization. By simultaneously measuring the B-modes induced by gravitational lensing at late times, EBEX can separate and remove an important foreground contaminant. Additionally, a high signal-to-noise detection of the lensing B-mode enables additional constraints on the dark energy equation of state as well as the sum of neutrino masses[\[2\]](#).

The anisotropies present in the CMB have been well-characterized by previous experiments for both temperature and E-mode polarization. EBEX builds on these results and will provide a unique dataset to advance the field of CMB observations.



## Chapter 3

# The EBEX Instrument

“It was maintained for commonality reasons, presumably based on the view that, unless proven necessary, it was not wise to make changes in software which worked well on Ariane 4.”

---

— Ariane 5 Disaster Inquiry Board Report

### 3.1 Overview

The EBEX instrument consists of two primary components: the receiver being a cryostat with associated readout electronics; and the gondola, a mounting platform for the receiver as well as electronics that provide attitude control, telemetry, data management and other support functions. The software that manages the EBEX instrument oversees the physical system, controls pointing, cryogenic cooler temperatures and scientific data collection as well as marshals the various sub-systems into a cohesive whole.

This section begins with an overview of the physical sub-systems of EBEX before proceeding to describe the software components developed to facilitate a successful experiment.

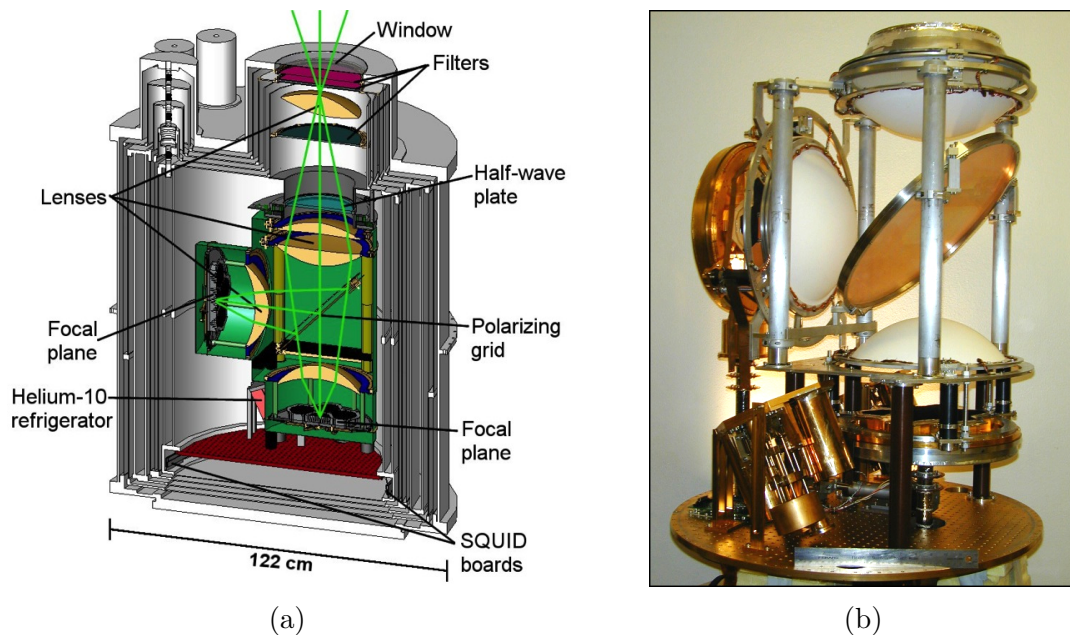


Figure 3.1: Light enters the EBEX cryostat from the top in these figures. It passes through multiple IR filters before being rotated by the half-wave plate, split into orthogonal polarizations and imaged by detectors on the focal planes. Additionally visible at the bottom in (3.1b) is the 3-stage helium dilution refrigerator.

## 3.2 Cryostat

The EBEX cryostat consists of five nested, thermally-isolated layers shielding two micro-kelvin stages. While the external shell remains at ambient ( $\sim 300\text{K}$ ) temperature, the internal shields are maintained at  $77\text{K}$  and  $4\text{K}$  by torroidal liquid nitrogen and liquid helium tanks, respectively. Between these three layers are two vapor-cooled shells at  $\sim 185\text{K}$  and  $\sim 30\text{K}$ , which increase the cryogenic efficiency of the receiver[113].

A schematic of the EBEX cold-stages is shown in Fig. (3.1). At the top, exposed to atmosphere, is the vacuum window. Then a series of low-emissivity thermal filters with decreasing cut-off frequencies are alternated with Low Pass Edge (LPE) filters and mounted on the  $300\text{K}$ ,  $77\text{K}$  and  $4\text{K}$  stages. These absorb and reflect high-frequency IR radiation, preventing unnecessary loading on the coldest stages.

At the Lyot stop, an achromatic half-wave plate (HWP) is mounted, rotating the light projected into the optics box beneath it. The lenses, shown in the middle of

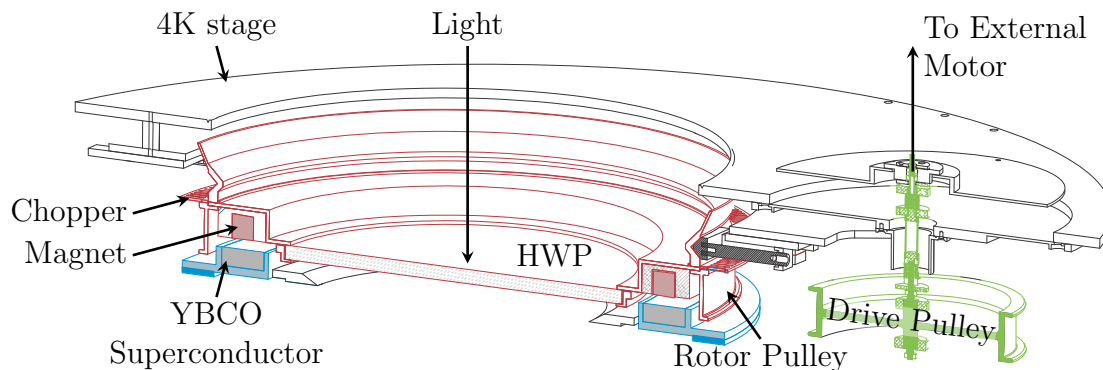


Figure 3.2: Schematic of the half-wave plate mechanism with color coding of the functional domains. Shown in red are the rotating components: a circular magnet, the half-wave plate and associated holder and the optical chopper. Shown underneath in blue are the stator components: the YBCO superconductor and its housing. On the right in green are the drive pulley and coupling shaft. A kevlar belt (not shown) connects the drive and rotor pulleys. Figure adapted from [61].

Fig. (3.1), re-image the light onto the two, complementary focal planes. Each focal plane receives a single polarization of the incident light reflected off of or transmitted through the polarizing grid mounted at  $45^\circ$  in the center of the figure.

### 3.2.1 Half-Wave Plate

The HWP consists of five birefringent sapphire plates glued together in fixed relative rotations of  $(0, 25, 85, 25 \text{ and } 0)^\circ$ . The combination of 5 layers increases the modulation efficiency over broad bandwidths[77] and, in the case of EBEX, achieves greater than 98% total efficiency over the full frequency range.

The HWP is mounted on a superconducting magnetic bearing as shown in Fig. (3.2). The bearing levitates above an Yttrium barium copper oxide (YBCO) superconductor during operation thereby minimizing friction and dissipative power in the cryostat. This allows EBEX to maintain its HWP at approximately 4K during flight, reducing the loading power generated by thermal emission to sub-dominant levels relative to the CMB[51].

The HWP is connected to an externally-mounted motor via a driveshaft and

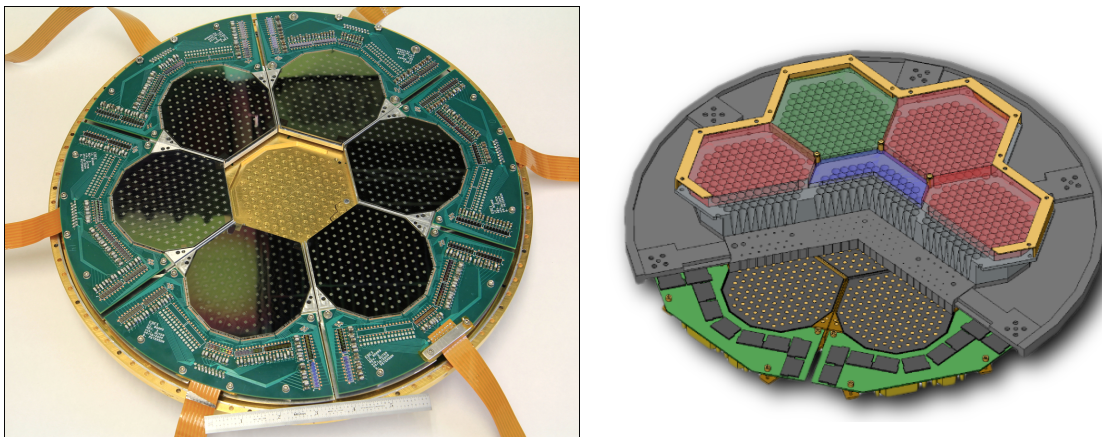


Figure 3.3: EBEX focal plane. *Left:* The bolometer wafers and associated LC circuit boards are shown in place but flipped on the focal plane to expose the wafers for the picture. *Right:* A cutaway diagram showing the wafers in their true orientation relative to the band-defining filters and metal feedhorns.

flexible coupling. The drive pulley and rotor pulley are then coupled by a tensioned kevlar belt. The rotor and HWP are rotated at a frequency of  $f_0 = 1.8\text{Hz}$ , modulating the sky signal observed at the focal plane into the sidebands of  $4f_0$ , well beneath the  $1/f$  'knee'. Additionally, internal systematics, which might otherwise contaminate our science signal, are either not modulated or modulated at smaller multiples of  $f_0$  and are therefore easily removed in analysis[61].

### 3.2.2 Focal Planes

As shown in Fig. (3.1), EBEX has two, complementary focal planes that sample light at orthogonal polarizations to each other. Each focal plane contains seven individual wafers. Four are tuned for a 150GHz signal, two are tuned for 250GHz and the center wafer is tuned for 410GHz. Frequencies are selected by a low-pass, metal-mesh filter in front of a smooth-walled, conical feedhorn. The feedhorn sizes are frequency dependent to create a uniform beam size across detector frequencies. EBEX operates 872 150GHz bolometers, 436 250GHz bolometers and 256 410GHz bolometers across both focal planes with a Strehl ratio of better than 0.9, ensuring that our optical path

is diffraction limited.

Each wafer contains 140 individual bolometric transition edge sensor (TES) bolometers coupled to spider-web geometry absorbers. The wafers are manufactured by EBEX collaborators at the University of California, Berkeley using a thin-film deposition and optical lithographic process and then tested and characterized at the University of Minnesota and McGill University. Wiring constraints to the LC circuit boards limit us to using maximum of 128 of the bolometers per wafer and hardware considerations, to be detailed in a future work, further limited the total number in use to between 121 and 124 bolometers connected per wafer.

The bolometer fabrication process was inherited work on from previous CMB experiments including the South Pole Telescope and APEX-SZ. However, as ground-based experiments, the previous bolometer designs were optimized for higher atmospheric loading than EBEX experiences observing from the upper atmosphere. EBEX developed a unique bolometer design that facilitated, for the first time, operation of TES bolometers in a space-like environment[7].

To enhance physical stability, finished wafers are mounted on metal plates made of Invar, a nickel-iron alloy notable for its low coefficient of thermal expansion, similar to that of the silicon wafer. The TES leads are wire-bonded to boards containing the frequency-defining LC resonators depicted in Fig. (3.6). These, in turn, are connected to superconducting quantum interference devices (SQUIDs) shown in Fig. (3.4) by means of a low-inductance microstrip cable developed at the University of Minnesota[96].

### 3.2.3 Readout

The digital frequency multiplexing (DfMux) boards both control and read the data from the SQUIDs. They are housed in 4 Faraday enclosures known as the bolometer readout (BRO) crates. The BROs are mounted to the inner frame of the gondola as shown in Fig. (3.5). The SQUID controller boards are located at the base of the

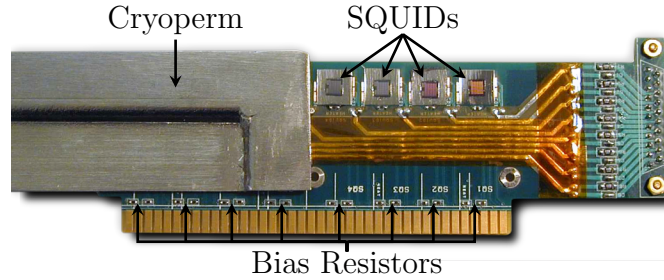


Figure 3.4: An EBEX SQUID board with its associated Cryoperm shield and bias resistors. The SQUIDs are manufactured by NIST for the EBEX collaboration. Figure adapted from [7].

gondola, in an RFI-sealed 'can' and are connected to the DfMux BROs via cabling run through shielded aluminum hoses. At the RFI can interface, SQUID controller communication lines and bolometer biasing lines are conditioned by  $\pi$ -filters to prevent signal leakage between domains.

A schematic of the EBEX readout electronics is shown in Fig. (3.6). The color-coding indicates temperature stages at which various processing is performed. The DfMux boards generate AC carrier signal from which each inductor/capacitor pair selects an unique frequency at which to voltage bias the bolometer.

The nuller port generates a  $180^\circ$  phase-shifted copy of the carrier frequency. This removes the carrier level from the measured output current prior to being amplified by the SQUID. As the SQUIDs do not have the dynamic range necessary to amplify the carrier signal, proper nulling is critical to maximizing the readout system's dynamic range. The signal is then low-pass filtered, digitized and demodulated by the DfMux board.

The DfMux boards each contain a Xilinx<sup>1</sup> field-programmable gate array (FPGA) integrated circuit that is configured for both signal processing and general-purpose computing to manage the bolometer array. The FPGA and associated hardware generate substantial waste heat that must be transferred away to prevent damage.

EBEX transfers heat away from DfMux boards by implementation of a liquid-

---

<sup>1</sup><http://www.xilinx.com>



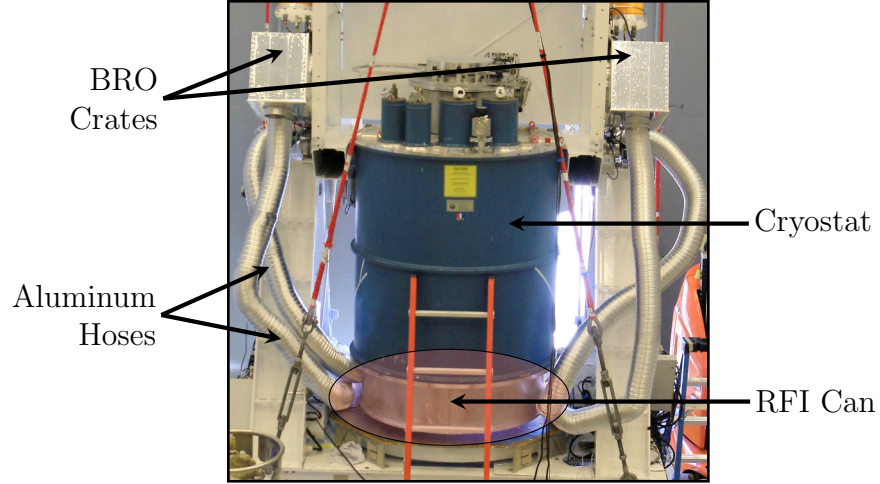


Figure 3.5: The EBEX cryostat is shown in the center mounted on the inner frame of the gondola. The bare aluminum base of the cryostat is an RFI-blocking 'can' that houses SQUID controller boards. Readout crates coated in silver teflon are mounted to the sides of the inner frame and connected to the RFI can by aluminum hosing. The hoses are secured to the cryostat prior to flight.

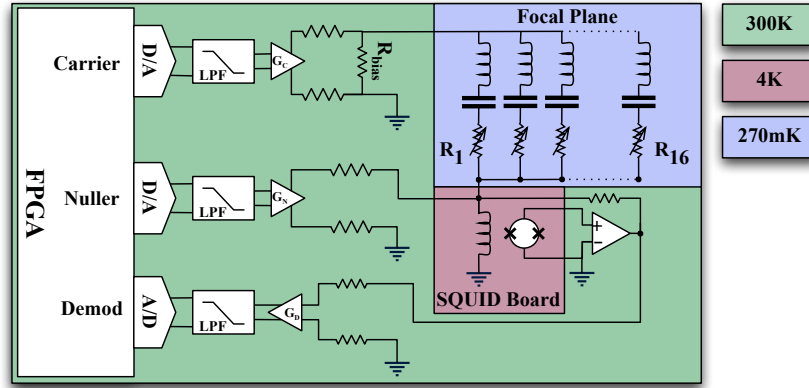


Figure 3.6: The EBEX readout schematic is shown color-coded by temperature region. Inside the cryostat, the bolometers and frequency-selecting LC resonators are at 270mK and the SQUID boards are at 4K. To the right of the SQUID board are the controlling electronics that live in the RFI-shielded can at 300K. The remaining 300K components are internal to the DfMux boards. Figure adapted from [103, 119].

cooling system. Dynalene<sup>2</sup> HC-40 is continually circulated through pipes that are heat-sunk to the backplane of the BROs that in turn receive heat conducted from the DfMux cards via NanoSpreader<sup>3</sup> heat pipes. The heated Dynalene then flows out to silver-teflon coated radiator panels mounted on the inner frame of the gondola as shown in Fig. (3.7). Temperature measurements taken during the long-duration balloon flight show this configuration is effective in maintaining consistent operational temperatures for the DfMux boards.

### 3.3 Gondola

The superstructure of EBEX consists of four main components: an outer frame that is suspended beneath the balloon flight train; an inner frame that attaches to the outer frame by two journals set along the horizontal axis; a set of sun shield baffles that connect to the outer frame; and solar panels that mount behind them. This configuration is shown in Fig. (3.7).

The inner frame contains the entire receiver and is mounted by a trunnion<sup>4</sup> with a pivot point that roughly passes through its center of gravity. The outer frame supports the trunnion and sun shield baffling and is in turn suspended from its four corners by Honeywell Spectra®1000 fiber<sup>5</sup> ropes that are spliced by Helinets<sup>6</sup>. The ropes, which are made of ultra-high molecular weight polyethylene, are wrapped in three layers of aluminized Mylar to prevent UV degradation during flight. The four ropes connect to a triangular spreader located above the inner frame. Three turnbuckles connect the spreader to a universal joint on the azimuthal pivot motor. The pivot motor connects the EBEX payload to the balloon flight train.

---

<sup>2</sup><http://www.dynalene-de.com>

<sup>3</sup><http://celsiatechnologies.com>

<sup>4</sup>The bearings in which the journals are mounted

<sup>5</sup><http://www51.honeywell.com/sm/afc/products-details/fiber.html>

<sup>6</sup><http://www.helinets.com>



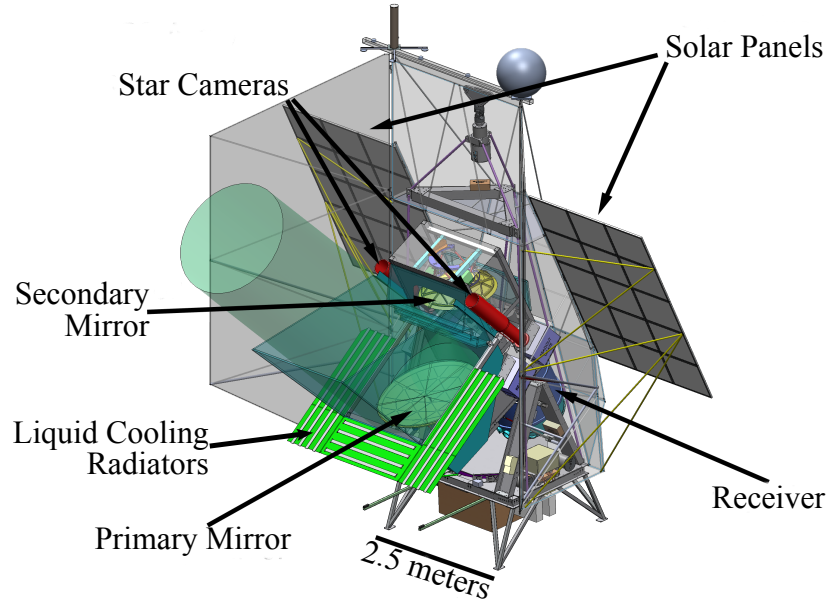


Figure 3.7: The EBEX gondola. Also depicted in this drawing is the nominal beam (light green cylinder) as it emerges from the reflection off of the primary mirror.

At the base of the outer frame, a weighted reaction wheel and associated motor is mounted that provides fine-grain azimuthal control. Also mounted on the outer frame are the flight computers, data storage pressure vessels, magnetometers, clinometers, two pinhole sun sensors and the attitude control system (ACS) electronics and associated analog to digital conversion hardware. The outer frame also provides mounting points for the sun shield baffling and two triangular legs that connect to the inner frame trunnion.

The inner frame houses the three-axis gyroscope boxes, star cameras, the cryostat and four bolometer readout (BRO) crates. Both primary and secondary mirrors are also mounted on the inner frame. At the trunnion pivot point, data cables are slack-routed to allow for differential motion in elevation between the inner and outer frames. The elevation is controlled by a single worm gear actuator mounted between inner and

outer frames on the port side. Because launch and termination shocks may exceed the load limits of the actuator, an additional locking pin was added between frames following the North America Engineering flight in 2009.

EBEX is designed to operate in the stratosphere at ~35,000 meters. At this level, the atmospheric density is less than 1% of that at sea level and does not support convective cooling thus all waste heat must be radiated away. The inner and outer frames are both constructed of 6061 aluminum and powder coated white to increase infrared emissivity while decreasing higher wavelength emissivity, allowing it to efficiently reflect sunlight while also radiating away waste heat generated by the ACS electronics.

### 3.3.1 Attitude Control System

Performing precisely directed measurements from a balloon-borne platform is a challenging task. The platform is not inherently stable and is subject to both internal and external forces that adversely affect both the actual instrument pointing as well as the instrumental measurement of that pointing. The EBEX Attitude Control System (ACS) serves a dual purpose of providing in-flight coarse attitude control as well as high-precision measurements that allow accurate post-flight attitude reconstruction.

To accomplish the first task, the ACS filters and combines attitude readouts from the star cameras, sun sensors, magnetometers, clinometers and encoder with velocity measurements from the gyroscopes. This filtering process produces real-time attitude solutions that are more coarse than those calculated in post-flight analysis. The accuracy requirement for real-time pointing is set primarily by the ability to efficiently sample in-flight calibration sources by our full focal plane.

### 3.3.2 Attitude Sensors

EBEX utilizes multiple, redundant sensors for its attitude control system. These are summarized with the associated uncertainties on their readings in Tab. (3.1). The

Sensor	Accuracy			Rate
	$\alpha$	$\beta$	$\gamma$	
Star Camera	5"	5"	120"	$\sim 0.025$ Hz
Gyroscope	40"/s	40"/s	40"/s	100 Hz
DGPS	5"	5"	120"	5 Hz
Sun Sensor	60"	60"	—	5 Hz
Magnetometer	2°	2°	—	5 Hz
Clinometer	—	72"	72"	5 Hz
Rotary Encoder	—	15"	—	5 Hz

Table 3.1: Attitude sensors for the EBEX Long Duration Balloon Flight, their associated uncertainties and update rates. The three angles listed are Euler angles for an arbitrary ZYZ rotation (see Fig. (4.1) for ZYZ rotation details). Thus the star camera will return  $\alpha$  and  $\beta$  as Right Ascension and Declination in an equatorial coordinate system while the DGPS will return the same rotations in the Azimuth and Elevation in a horizon coordinate system. The gyroscope sensor measures speed and therefore its uncertainties are velocities measured in an idealized, orthogonal coordinate system that is referenced to the gondola body.

sensors provide independent measurements of various vector quantities throughout the flight that are synthesized into a coherent estimation of the gondola attitude.

The measurement of a single vector can constrain two axes of rotation, while two vectors are required for a three-axis constraint. As more independent vector measurements are added, the uncertainty of the combined attitude estimation will decrease. While the DGPS and the Star Cameras can each measure 3 or more vectors independently, most of our sensors will only measure a single vector and thus must be utilized in combination with each other to constrain a full attitude solution.

### 3.3.2.1 Star Cameras

Two star cameras were used for the EBEX long duration flight. The first, denoted *XSC0* was built at Brown University and flew in 2009 for the EBEX engineering flight[130]. The second, *XSC1*, was built at Columbia University.

Each star camera system consists of a Kodak KAF-1602E CCD, controlled by a

MegaPlus model ES1602 camera and focused through a Canon EF f/1.8 lens. The system is attached to an autonomous PC104 computer. XSC0 uses a Digital Logic MSM855-C373 computer while XSC1 uses an Advanced Digital Logic ADL855-PC-373 computer.

Each star camera is housed in a separate pressure vessel, which are roughly co-aligned with the microwave camera beam, on opposite sides of the cryostat. The cameras communicate with the primary flight computer via the ethernet network as shown in Fig. (3.11).

Images are acquired by the star cameras' computers and stored to disk for post-flight analysis. Additionally, an estimate of the camera's attitude is calculated based on a comparison of identified stars in the captured image with a pre-generated reference catalog. This estimate is transmitted to the flight computer for use in its real-time attitude solution.

Exposure times for each star camera image are on the order of 50ms. In order to prevent spreading of the star's photons across multiple pixels, the gondola's total velocity needs to be relatively low. Each pixel is calculated to have a viewing angle (or platescale) of  $\sim 9.5$  arcseconds per pixel. Thus, the gondola will traverse a full pixel in the 50ms exposure at a speed of roughly  $5e-2^\circ/s$ . This low velocity limit means that star camera images can only be acquired at points of azimuthal reversal in our scans.

### 3.3.2.2 Gyroscopes

Between the low-velocity points at which star camera readings are acquired, EBEX integrates velocity measurements from two sets of fiber-optic gyroscopes (FOGs) mounted in independent boxes. Each box holds 3, roughly-orthogonal FOGs. Each FOG provides a measurement of the gondola rotation speed projected onto its axis. This measurement of rotation speed is independent of the platform translation.

Basic FOG operation is shown graphically in Fig. (3.8). A single source beam is

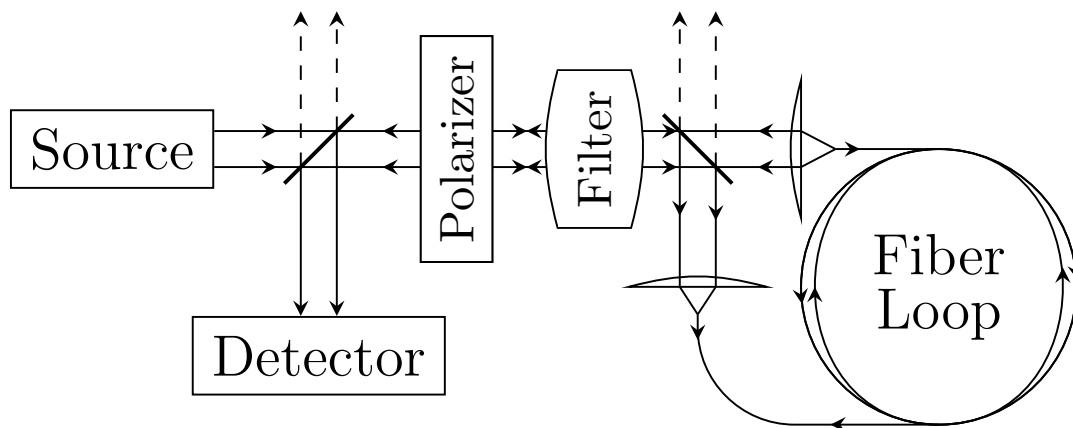


Figure 3.8: Fiber-Optic gyroscope operation. A source beam is split, sent in opposite directions around a shared coil and an interferometer measures the resulting phase shift. The polarizer and filters condition the signals to normalize their relative strengths. Figure adapted from [136].

split and injected into two, counter-rotating fiber-optic cables that encircle a shared cylinder. The detector measures the relative phase shift induced by the special relativistic Sagnac Effect[97] to determine rotation speed. This measurement is then read out at 1000 Hz over a digital interface by the attitude control system<sup>7</sup>.

At low integration times, the noise in a FOG is dominated by the angular random walk, as shown in Fig. (3.9). At 40 seconds (roughly the EBEX integration time between star camera readings), the incremental contribution to error from the bias instability begins to affect the total error, however the integrated error remains dominated by angular random walk<sup>8</sup>.

As will be detailed in §4, care is taken to properly characterize the parameters of each gyroscope using flight data. Noise on the gyroscope constrains both the accuracy with which we can determine the governing parameters as well as the accuracy of the eventual attitude solution.

---

<sup>7</sup>The gyroscope signals are filtered to the 100 Hz rate by use of a 4-pole, finite impulse response filter, developed by the BLAST experiment and detailed in [136].

<sup>8</sup>The method used to determine these values is referred to as the Allan variance method and relates frequency instabilities to characteristic time lengths[71].

### 3.3.2.3 Differential GPS

The differential Global Position System readout (DGPS) system used by EBEX is a Thales Navigation ADU<sup>9</sup>. The DGPS control box is connected to 4 omnidirectional Ashtech 103062 L1 Aircraft<sup>10</sup> antennae arranged in a cross pattern with 1.1m separation along the diagonal.

The DGPS calculates an attitude solution by measuring the phase difference in satellite signals received by multiple antennae. The phase difference yields a directional unit vector for the position of a satellite, relative to the antenna array. By comparing two of these computed vectors with calculated information about the satellite ephemeris, the DGPS constrains its position to an arc, projected on the earth.

<sup>9</sup><http://www.thalesnavigation.com>

<sup>10</sup><http://www.ashtech.com>

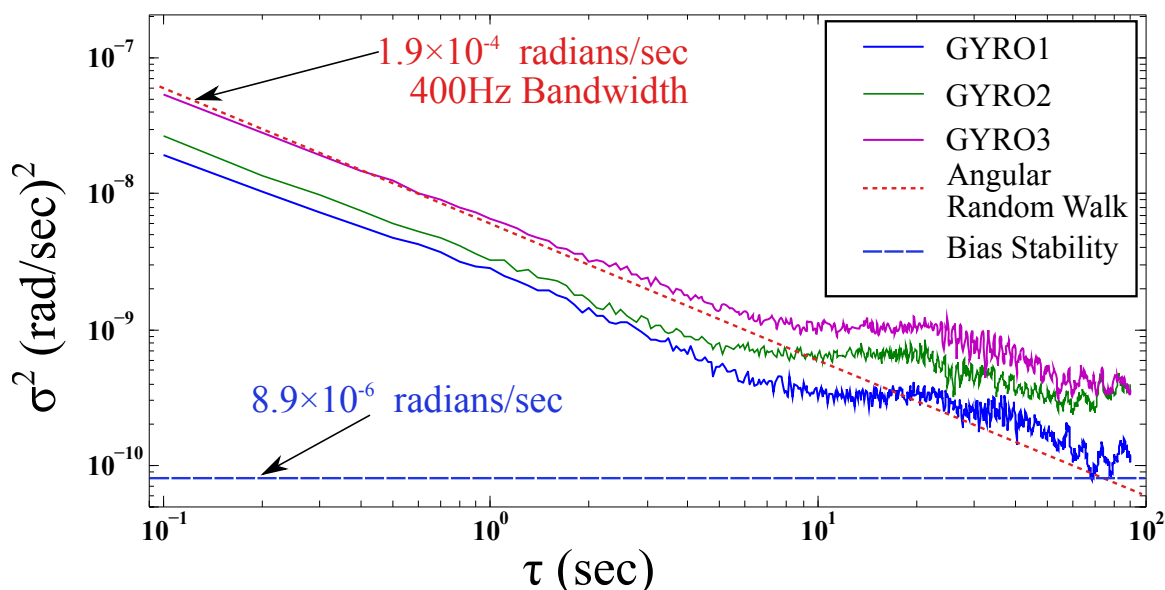


Figure 3.9: Measured Allan variance is shown for three of the six EBEX gyroscopes. The specification levels for angular random walk and bias stability for our 400Hz bandwidth are overplotted in dashed red and dashed blue, respectively.

The addition of a third satellite measurement constrains the position to a latitude and longitude but without knowledge of altitude. Once four measurements are available, the system can compute a full position and attitude solution.

#### 3.3.2.4 Magnetometers

EBEX employs two, three-axis flux gate magnetometers that measure the attitude of the local magnetic field. The field is dominated by the Earth’s magnetic field, although we observe both hard- and soft-magnetic field effects from gondola components. These may be modeled and subtracted by observing that the magnetic field strength (and thus the measured vector magnitude) is constant under gondola rotation.

The corrected vector components are then compared against a World Magnetic Model produced by the National Oceanic and Atmospheric Administration in collaboration with the British Geological Survey[78]. Utilizing the latitude, longitude and altitude information from the DGPS, we calculate the local Earth magnetic field and align it to our corrected measurement to get a measurement of gondola azimuth and elevation. Note that as this represents a single vector measurement, we cannot use this to constrain the gondola’s rotation about that vector.

The accuracy of the magnetometer’s measurement of azimuth is additionally limited by the relatively high angle of the of the Earth’s magnetic field at extreme latitudes such as those experienced in Antarctica. In theory, one might discard the measurement of azimuth, opting instead for a measurement of roll that would be better constrained near the magnetic south pole. This is disfavored in EBEX as our scanning strategy provides for azimuthal control but not roll. The absolute magnitude of roll is also guaranteed to be smaller than the azimuth as we will rotate as part of our scanning strategy. Thus, in the absence of other sensors, taking the roll projection to be zero will induce smaller errors than an equivalent assumption about azimuth.

### 3.3.2.5 Sun Sensors

The EBEX sun sensors are each composed of a Hamamatsu<sup>11</sup> S5991-01 2-dimensional position sensitive diode (PSD). The PSD is mounted at a fixed distance behind a precision pinhole that focuses light onto the diode surface. By measuring the X-Y positions of the focused sunlight, we obtain an azimuth and elevation of the Sun.

We compute the expected sun location using the latitude, longitude and altitude from the DGPS and then align the sun's vector with our computed vector in the same manner as with the magnetometer reading. Similar to the magnetometer, this reading only provides a single vector and thus we must choose the two angular projections we wish to constrain. During the EBEX flight, the sun's elevation ranged between 15-40° above the horizon. The sun sensors were aligned such that a 25° elevation would illuminate the center of the PSD, providing good azimuth and elevation constraints, assuming again a low-roll condition.

### 3.3.2.6 Clinometers

EBEX uses 2-axis clinometers to measure the angle (but not magnitude) of the local acceleration vector. One clinometer is mounted on the outer frame and one on the inner frame. The clinometers are Applied Geomechanics Model 904-T<sup>12</sup> that measure the position of a liquid-filled electrolytic transducer as it moves along a curved track. The clinometers cannot decouple the effects of gravity from normal pendulation, however as the gondola freely oscillates about minimum energy point, the clinometer reading will represent the mean attitude of the gravity vector.

### 3.3.2.7 Rotary Encoder

A single rotary encoder measures absolute rotation of the inner frame journals relative to the outer frame trunnion in which they are mounted. As multiple sensors, e.g.

---

<sup>11</sup><http://www.hamamatsu.com/>

<sup>12</sup><http://www.jewellinstruments.com>



DGPS, Magnetometer and Sun Sensor, are mounted on the outer frame, we require a high-precision, relative encoder to map the measurements of outer frame elevation onto the inner frame where the receiver is mounted. EBEX uses a Gurley Model A25S<sup>13</sup> absolute rotary encoder to measure this angle. It couples to the port journal by means of a Gurley Model SCE flexible shaft coupling.

## 3.4 Communications

The EBEX system incorporates data transmitted via 4 distinct communication paths: the BLASTBus, CANBus, TCP/IP network and serial ports. It is the responsibility of the flight computers to synthesize these streams into a coherent picture of time-aligned data.

### 3.4.1 BLASTBus

The primary communications path used by EBEX for non-bolometric data is the BLASTBus. This protocol and communications system was developed by the University of Toronto for use on the BLAST experiment[86] and was adapted by EBEX for its attitude control system and gondola housekeeping. It consists of an RS-485 (clocked differential voltage) synchronous bus architecture with multiple connection points (drops) for BLASTBus clients.

The bus consists of 3 differential pairs (data, strobe and clock) as well as an isolated +5V supply and reference ground. Each client reads and writes data to the bus synchronously. This is ordered by the use of a word-based, push/pull architecture. Master clients, which in the EBEX case are PCI cards mounted to the flight computers, either push data to the bus, writing a word to the clients or pull data from the bus by polling each client individually and waiting for a response.

---

<sup>13</sup><http://www.gurley.com>

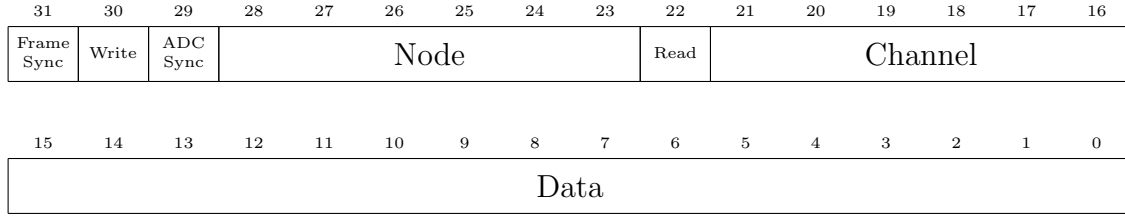


Figure 3.10: The BLASTBus 32-bit word. The top 16-bits are control information, governing the data address. Frame Sync is set high at the start of each new frame, triggering sampling on the ADC cards. The Write bit is set high to denote that the data should overwrite previous data values in the frame. The ADC Sync is set high by the flight computer only during initialization and causes the ADC cards to synchronize their output. The Node and Channel numbers are addressing for up to 64 cards with up to 64 channels each. The Read bit is set high by the flight computer to request new data from the ADC card or signal a response to the request. The bottom 16-bits are the resultant data.

The top 16 bits of the word are reserved for addressing while the bottom 16 bits are data as shown in Figure 3.10. The flight computer sends data to ADC card clients (push) and requests data from other clients (pull). As the bus is shared, each client also reads and retains a local copy of frame words and may use them for internal calculations, e.g. PID control loops.

The bus operates on a 4MHz bit clock, resulting in an effective data bandwidth of 1megabit/second (Mbps). In practice, firmware design on the flight computer’s BLASTBus PCI card limited our ability to write data to the BLASTBus. This placed an effective constraint of 40kilobits/second (Kbps) bandwidth from the flight computer to the BLASTBus, although it did not similarly limit the bandwidth of data received by the flight computer.

EBEX uses both the digital I/O as well as the analog to digital conversion (ADC) of the BLASTBus clients extensively. Magnetometers’, clinometers’ and sun sensors’ voltage-based outputs are sampled by ADC cards while gyroscopes and the encoder are sampled via the digital inputs. A trigger line controlling the star camera’s shutter is also controlled via a digital output on the BLASTBus client. In addition, thermometry and various motor controls are digitized by the ADC inputs and sent to the flight

computers via the BLASTBus.

### 3.4.2 RS-232

RS-232 is a communications standard serial lines commonly used by commercial equipment for lightweight point-to-point communications[116]. EBEX uses RS-232 connections for bi-directional communication with multiple systems that relied on commercial, off-the-shelf (COTS) equipment. These include the DGPS, battery status monitoring, motor actuator control and all communications with NASA equipment.

Initial system designs also called for interfacing with our battery charge controllers via serial communications. As the flight computers are only equipped with 4 on-board serial ports, we chose to add RS-232 ports using a commercial universal serial bus (USB) to RS-232 adapter. During the North American flight, we successfully used single-port adapters from Targus<sup>14</sup> and Keyspan<sup>15</sup>. To interface with all five flight batteries, we needed an additional seven RS-232 ports. We elected to install two, four-port industrial expansion cards manufactured by Quatech<sup>16</sup>.

Although advertised as ruggedized, the Quatech adapters proved to be unreliable in the dry, high-static environment of the Antarctic campaign, three of the four Quatech cards ceased to function during integration and we ultimately decided to replace them with single-port Keyspan adapters. This choice meant that we could only monitor a single battery and DGPS during flight and ultimately limited our ability to react to battery voltage levels during the Antarctic flight.

### 3.4.3 CANBus

EBEX uses custom-designed electronics for cryostat temperature monitors, heat switch controls and monitoring housekeeping data. These electronics boards are designed

---

<sup>14</sup><http://www.targus.com/us>

<sup>15</sup><http://www.tripplite.com/en/products/keysan-products.cfm>

<sup>16</sup><http://www.quatech.com>

at the Weizmann Institute and utilize Embedded Local Monitor Boards (ELMBs), originally created for the ATLAS control systems operating at CERN[49], as their general purpose monitors. The ELMBs communicated with the flight computers using the CANBus protocol, a broadcast-based 2-wire serial bus standard.

Built on top of the CANBus standard is a higher-level protocol called CANOpen[35, 14] This protocol allows data exchange with and control of the ELMB nodes by the EBEX flight computers. The CANOpen firmware is modified by EBEX collaborators at the Weizmann Institute to utilize our custom timestamping scheme.

The EBEX flight computers connected to the CANBus via a Kvaser Leaf<sup>17</sup> USB to CANBus converter. Initial work to interface the CANBus clients with the EBEX flight control software was developed prior to the North America flight. While successful in many respects, critical issues were identified in post-flight analysis that required a re-evaluation of the interface software[111].

For the long-duration flight, this software has been updated to work robustly in a multi-threaded environment. The design accommodates asynchronous input commands from either the ground control or on-board systems while monitoring the USB channel for both housekeeping data input and command acknowledgement from individual CANBus nodes. This is accomplished by utilizing a series of commanding queues, implementations of the the lock-free queue design described in §3.5.1.

Incoming CANBus messages are duplicated into two queues for system processing. One queue handled only data input, parsing values and writing them to the BLASTBus for recording. The queue handled only commands and the commanding response. This division ensured that multiple commands could be correctly spaced in between CANBus read events, preventing the loss of housekeeping data or queue overflows resulting from delays in handling command response packets.

---

<sup>17</sup><http://www.kvaser.com>

### 3.4.4 TCP/IP Network

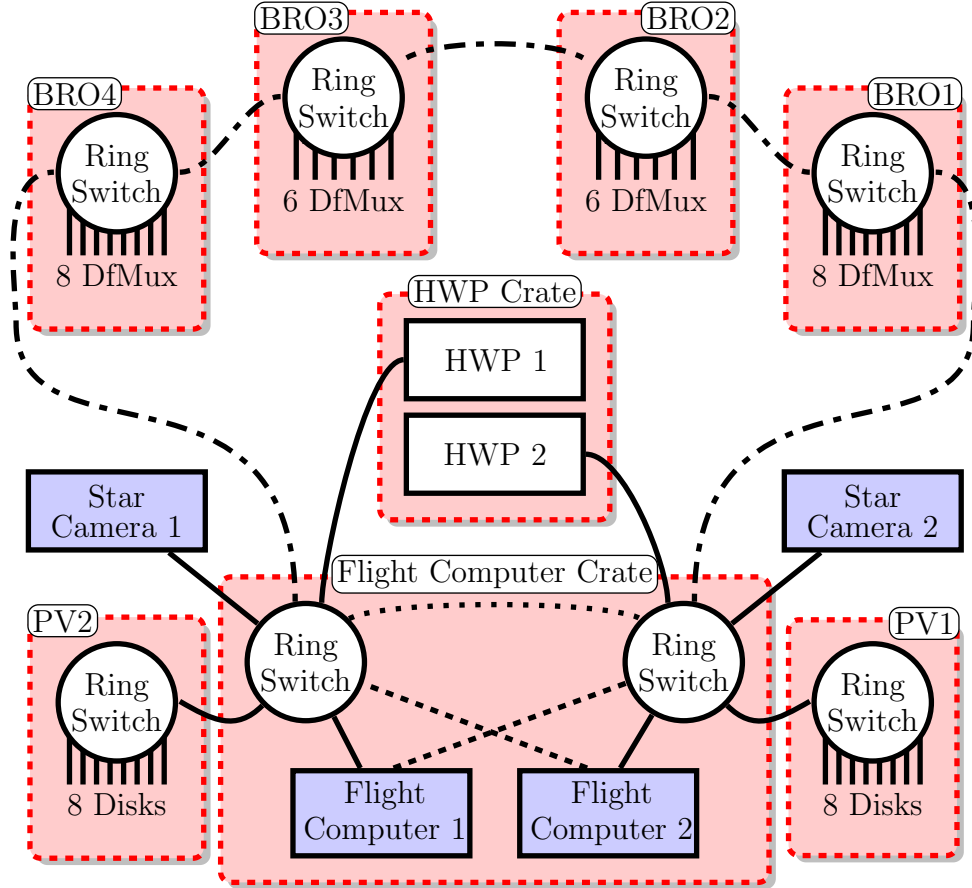


Figure 3.11: EBEX Ethernet network diagram. The dash-dotted lines (— · —) show the fiber-optic ring connections between switches in the flight computer crates and bolometer readout crates. The dotted line (· · ·) between switches in the flight computer crate denotes the fail-over link of the ring that only activates should the ring be damaged. Dashed lines (— · ·) from the ring switches to the flight computers show redundant copper ethernet connections that the flight computers activate, should their primary connections fail.

Multiple systems communicate with the EBEX flight computers via the TCP/IP network including the star cameras, half-wave plate readout systems and bolometer DfMux boards. Each of the 28 DfMux boards is connected via copper, CAT5e cable to a single 'ring switch' in the enclosing readout crate as shown in Fig. (3.11). The individual ring switches are linked together via fiber optic lines in a redundant ring

that encompasses the four readout crates and two ring switches in the flight computer crate.

Primary consideration in the design of the network is redundancy of communication paths to tolerate single-point failure. Severing a communication link to any of the four BRO crates causes the ring switch network to engage its backup link between the two ring switches in the flight computer crate. Systems that are connected to the overall network by non-redundant links are, themselves, redundant with other systems. As such, flight critical disruptions to the data network require at least two concurrent, critical errors.

## 3.5 Software

Software development in EBEX was a multi-step process as the software both facilitated and was dependent the hardware development. We began with a highly-specialized code base, developed for the BLAST experiment[136]. This code, while functional for the experiment for which it was written, had grown organically over many years and is not readily adaptable to the EBEX use case.

The first iteration of development to adapt the BLAST code base happened prior to the North America flight and focussed on adapting segments to the needs of the project. Following the North American flight, I took over development of a consolidated system that would address the shortcomings listed below that were identified during the North America engineering flight as well as allow our system to expand to its full capacity.

The major software components developed specifically for EBEX are: threading, disk management, BLASTBus interface, automation and telemetry.

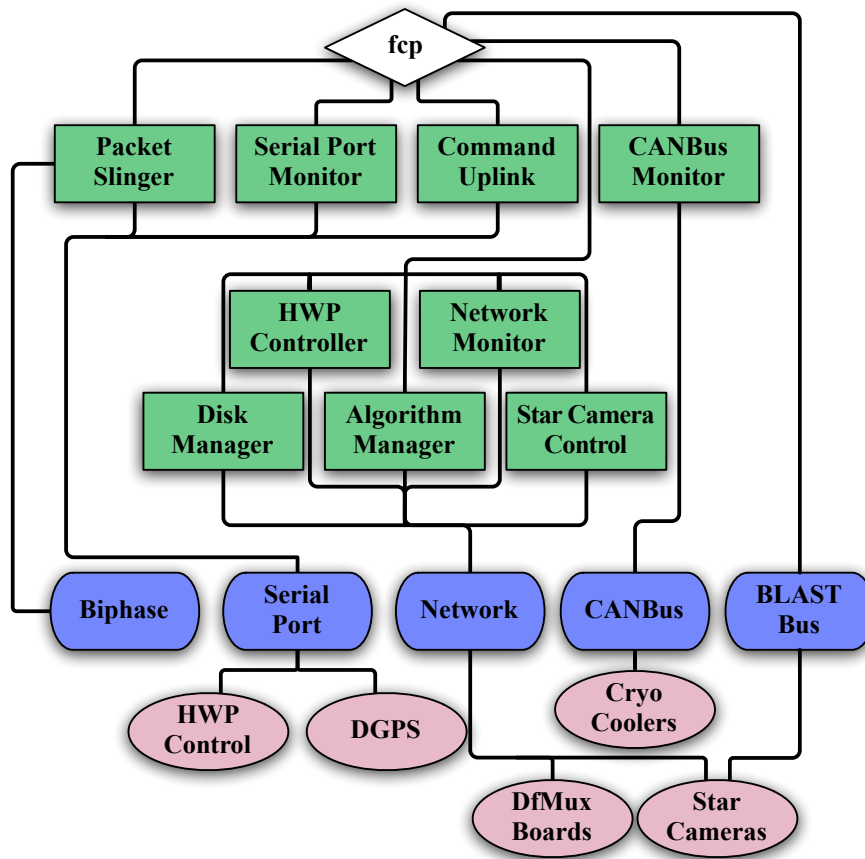


Figure 3.12: Block diagram of the major EBEX software subsystems. The primary thread for the flight control program (*fcp*) is shown at the top and controls the child threads listed as green rectangles. The communication links are shown beneath in blue and a selection of the underlying hardware is shown at the bottom in red ovals.

### 3.5.1 Threading

Many aspects of the flight control system happen asynchronously. Data arrive independently over the BLASTBus, CANBus, network and serial ports while simultaneously commands must be sent over each of these links. The requirement to handle all of these communication links on an as-needed basis means that the Flight Control Program (FCP) must be highly threaded<sup>18</sup>.

This is a substantial departure from the design of the inherited BLAST system, which ran the majority of its systems inside of a single thread. While the original BLAST system gathered its science data, housekeeping and attitude control information over the BLASTBus, EBEX's data-taking, housekeeping and attitude control are each processed over a separate communication link. Responding to each in a timely fashion requires interrupting idle states for a single thread to perform waiting functions in another.

Without careful design, the addition of multiple threads can easily cause hidden failure cases as was the case with the CANBus architecture described in Section 3.4.3. Thus, following the North America test flight, a set of lock-free, thread-safe data structures were designed and implemented. These data structures specifically addressed the two predominant use cases in EBEX.

#### 3.5.1.1 FIFO

The primary data structure used by EBEX is a multi-reader, multi-writer, First-In-First-Out (FIFO) queue. The FIFO structure is used extensively in fcp for queuing data writes, commanding, network calls and all other events that can have arbitrary readers and writers, though in practice, we almost exclusively used multiple writer threads but a single reader thread.

---

<sup>18</sup>Alternative designs such as co-processing are also feasible. However, these generally require greater complexity in the individual code modules, increasing the difficulty of identifying and fixing errors. By implementing thread-safe data structures, we displace the complexity of the threaded implementation to a centralized module.



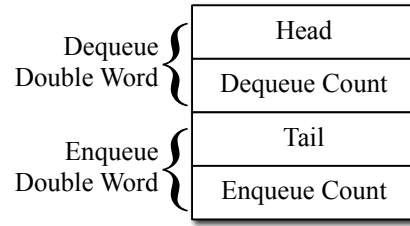


Figure 3.13: Memory layout of the FIFO queue framing structure. The two pointers into the node chain refer to the head, used by the dequeuing routine, and the tail, used by the enqueueing routine, and their respective counts.

The queue consists of two components, a framing structure that provides a common reference point for accessing the queue and individual nodes that contain stored data. The framing structure maintains two pointers into the queue, one for the head, from which data are read, and one for the tail to which data are written. It additionally maintains a numerical count of the number of nodes queued and dequeued. These values are only ever incremented (until they overflow, resetting to 0) and are stored as a word-sized integer adjacent in memory to the associated pointer as shown diagrammatically in Fig. (3.13).

Nodes are two word structures that contain a storage pointer to the referenced data and a forward pointer to the next node in the queue. If the node is last in the queue, the forward pointer is 0, a reserved value. In this manner, by knowing the beginning of the queue from the framing structure, we can sequentially access each element in the order in which they were added.

The queuing algorithm, detailed in Alg. (A.1), takes a pointer to an arbitrary memory location and stores it in the queue as follows.

1. Upon receiving a data pointer, a new *node* is allocated, its forward pointer is set to 0 and its storage pointer assigned to the input data.
2. The values of the current Enqueue Double Word are then stored in a temporary structure as a reference to the current state of the queue.

3. The temporary structure's node's forward pointer is then compared to zero and if the comparison is true, the pointer is re-directed to the new *node* allocated in step 1. This operation is atomically executed using a compare-and-swap instruction such that no other process may alter the value between when it is compared and when it is swapped. If the comparison fails, it indicates that another process has inserted a new node between steps 2 and 3. In this case, we repeat step 2 until step 3 succeeds.
4. After we successfully exchange our new node, check that the location it was inserted remains the tail of the queue. Without this step, it possible that tail's next pointer was used and then removed from the queue between steps 2 and 3. If this check fails, we return to step 1 and try again.
5. At this point, no other processes may insert another node until we complete our final step as the tail's node's forward pointer is non-zero. Therefore, we increment the enqueue count (this is not required but useful to keep track of the total number of queued items) and set the tail node to our newly allocated *node*. Setting a pointer value is an atomic operation, so we are guaranteed to leave the queue in a valid state.

This enqueueing routine is notable for its simplicity and speed. Step 4 utilizes a single word compare-and-swap (CAS) instruction. In the case of a multiprocessor machine, this generates a single `LOCK` instruction (an expensive step on the x86 micro-processor, requiring approximately 100 cycles in testing on the EBEX system) instead of the two `LOCK` instructions required for the double word equivalent (CAS2)[56] or multiple CAS steps. We thereby improve performance over previously published algorithms[128, 82, 99] that are used by standard code libraries.

However, EBEX's flight computers are single-processor machines, a fact that we may exploit to improve queue performance further. Single processor machines only execute a single processing thread at a time. Thus, if we wish to ensure atomic access

to a memory location, we must only ensure that an alternate thread does not interrupt our routine between the comparison and swap operations. To do this, we wrap the CAS and CAS2 calls in a lighter-weight interrupt-disabling mechanism. Thus the CAS and CAS2 calls are as follows:

1. Save the current state of interrupts. This is a PUSH instruction on the x86 platform.
2. Disable interrupts, ensuring our current thread can not be interrupted by other processes. Between steps 1 and 2, we do not care if we are interrupted as there are two cases: either interrupts are enabled or disabled. In the case of disabled interrupts prior to step 1, then we are guaranteed not to be interrupted. In the case of enabled interrupts, other processes may disable interrupts but this will proceeding to step 2 until the interrupts are re-enabled, ensuring that our saved interrupt state remains valid.
3. Compare the value (in the case of CAS) or values (in the case of CAS2) at the destination with those expected. If they are the same, exchange the old and new values. If they are different, we do not alter either.
4. Restore the previous state of interrupts. This is a POPF instruction on the x86 platform. Other processes may once again interrupt us, if the previous state allowed it.

Our implementation of this single-processor variant of a CAS/CAS2 operation uses 4 and 6 operations respectively for a successful swap. We thereby gain substantial performance increases over comparable algorithms generally available.

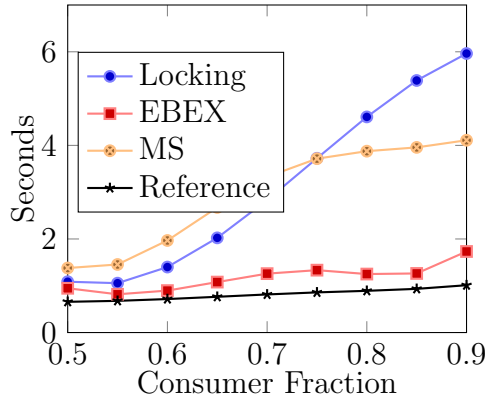
The complimentary FIFO dequeuing algorithm, detailed in Alg. (A.2), takes no arguments apart from the queue on which to operate and returns the oldest item in the queue. This works as follows:

1. We store the current dequeue double word and create a new dequeue double word that will represent the state of the queue after dequeuing. This has the new queue head pointer assigned to the old queue head pointer's next node and the dequeue count incremented by 1.
2. If the head pointer is the same as the tail pointer, then we have an empty queue and return a signifier to calling routine.
3. Otherwise, we store the data value from the head pointer's next node and attempt a CAS2 operation to exchange the queue's dequeue double word with our newly-created value from step 1. The comparison step is made with the stored double word from step 1.
4. If the CAS2 operation succeeds, the data value is returned to the calling routine. If we fail, it implies that another routine has dequeued the head pointer between steps 1 and 3. In this case, we return to step 1 to try again.

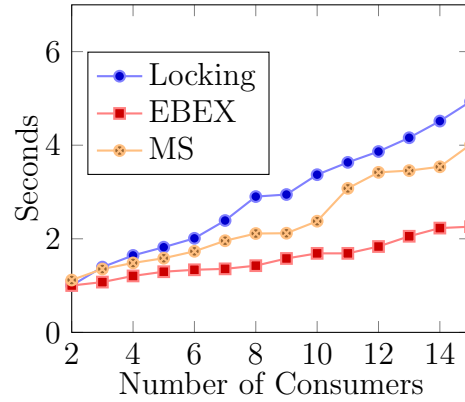
These algorithms represent improvements on previously published reference algorithms[128, 82, 65] for the most common EBEX use cases. They take advantage of systems knowledge that more generalized implementations cannot assume. In this use, these algorithms outperform the state-of-the-art reference implementations as shown in Fig. (3.14)

### 3.5.1.2 ABA Problem

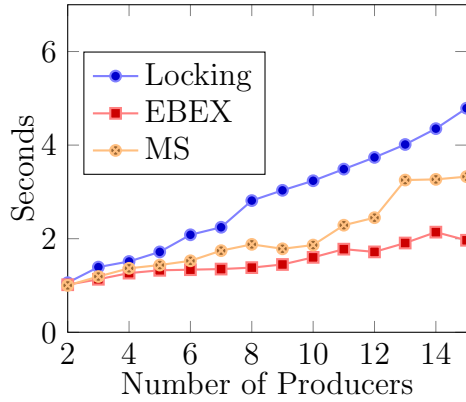
In the design of a shared memory, multi-process system, special care must be taken to avoid the ABA problem, a fundamental issue in compare-and-swap systems[81]. This refers to the situation where one process thread (A) may be interrupted by another thread (B) that performs operations on the queue before yielding back to the first thread (A), whose following actions leave the queue in a corrupted state[126]. To show that this implementation is immune to ABA, it is sufficient to show that all CAS



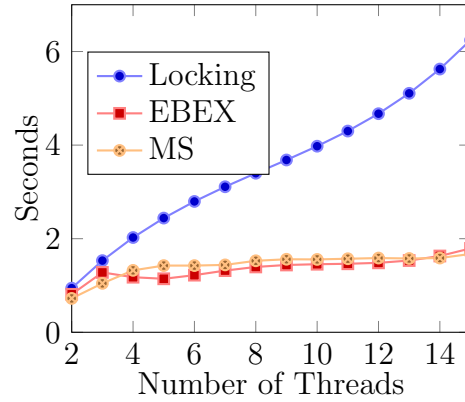
(a) One Producer, One Consumer



(b) One Producer, Multiple Consumers



(c) Multiple Producers, One Consumer



(d) Multiple Producers and Consumers

Figure 3.14: Performance of the FIFO queue under common utilization scenarios. In the legend, “Locking” refers to a single queue lock implemented using the standard `pthread` library, “MS” is the lock-free algorithm of Michael and Scott [82], as implemented by a popular data structures library<sup>19</sup>. In each case, 20 million records were enqueued and dequeued. (a) shows the single-producer, single-consumer case. Here, the x-axis shows the fraction of total calls by the consumer thread and illustrates the performance overhead of each algorithm, relative to a single thread reference. (c) shows the standard EBEX use case in which many producers are funneled into a single consumer such as the downlink case, described in Section 3.5.5 and the disk manager case, described in Section 3.5.2. (b) and (d) show the improved performance in contentious consumption cases. While not used by EBEX, cases (b) and (d) have many applications, particularly in multiprocessing pipelines.

operations either fail or succeed without leaving the queue in an inconsistent state. That is to say that the compare operand represents a unique state of the queue.

We verify consistency in the queue by examining the points at which loop memory

may be altered. In the enqueueing operation, a CAS is performed in step 3, comparing the tail's next node pointer with a 0 value. As the 0 value is invalid as a memory location, it can never represent a true value. By replacing this with a non-zero value, we block further processes from proceeding past step 3 until we have updated the tail's next node pointer at the end of step 5 as long as it was truly the tail's next node pointer we updated. This check is performed by step 4, thereby ensuring ABA safety.

Dequeuing is similarly verified. The single CAS operation occurs in step 3 and operates directly on the queue's dequeue double word. The inclusion of a reference counter here ensures uniqueness not just of the location in memory (which may be re-used) but also the word-sized reference count associated with that location, effectively creating a unique value for the queue state that is not replicated<sup>20</sup>.

### 3.5.2 Disk Management

EBEX generates more data than any previous balloon-borne experiment. EBEX store 1792 bolometer channels<sup>21</sup> sampled at 190.73 Hz. Each bolometer is sampled with 16 bits of resolution that, together with half-wave plate encoding and attitude control data, generates over 700 GB of data over the course of our 11 day flight. At the time EBEX was designed, this amount of data could not fit on a single hard disk, so we elected to create a multi-disk storage system.

Rotating magnetic platter storage is not designed to operate in a vacuum. Both the spindle lubricant and proper spacing between the head and media depend on air pressure for their performance. Atmospheric pressure at our float altitude of 35,000m is less than 0.1torr, a level that would have prevented proper disk operation. This constraint led us to build and use two pressure vessels for our disk arrays, with all power and signal lines passing through vacuum connectors.

---

<sup>20</sup>The absolute statistical chance of any value is 1 in  $2^{64}$  but the chance of 2 double words sharing the same values is 1 in  $\binom{2^{64}}{2}$ , an astronomically large number.

<sup>21</sup>While some of these channels are not connected to physical hardware, they remain present in framing structure received by the flight computer.

EBEX uses, as an interface to this disk array, Advanced Technology Attachment (ATA) over Ethernet (AoE). This protocol, developed by Coraid, Inc<sup>22</sup>, wraps ATA commands in a lightweight ethernet frame for transmission over a local area network[20]. This design allows us to connect two independent arrays of disks to both flight computers via the ethernet network.

### 3.5.2.1 Multiple disks

Each pressure vessel contains 8 independent disks mounted on individual AoE hardware translation boards called EtherDrive Mark II 'blades'. The blades are designed by Coraid to provide a modular interface between an Integrated Drive Electronics (IDE) - formatted connector and an ethernet network.

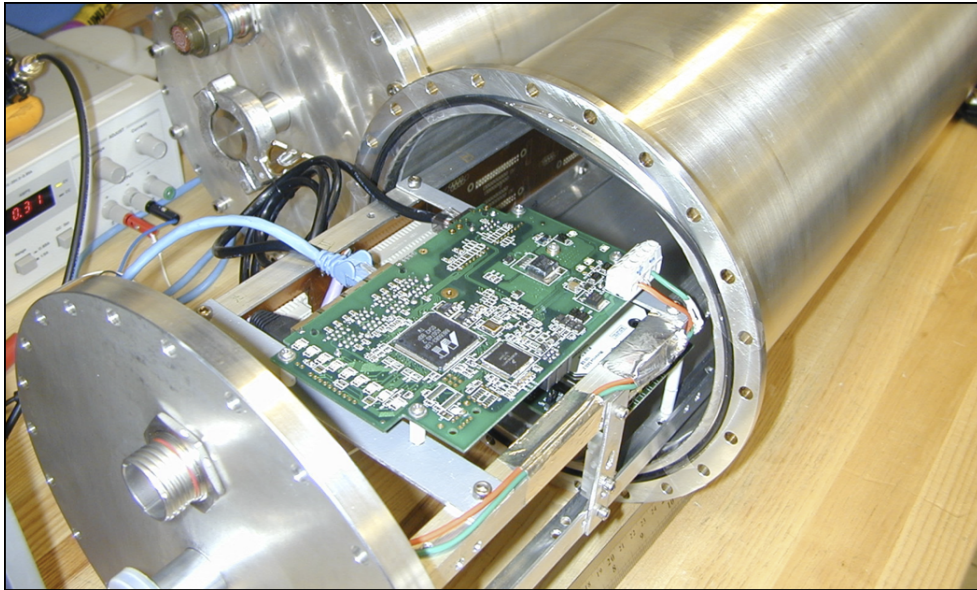


Figure 3.15: The EBEX disk pressure vessels are shown with one opened. Visible in the foreground is the network switch while the backplane is mounted on the long edge of the internal frame. The hermetic connector on the front of the pressure vessel provides signal and power connectivity. Image from [111].

The blades are, in turn, mounted on a backplane that provides both power and network connections. The network cables are connected to an ethernet switch co-

<sup>22</sup><http://www.coraid.com>

located in the pressure vessel. Additionally, the backplane integrated a pressure sensor and 2 fans used to ensure convective distribution of the heat generated by the disk array. The pressure vessels are shown in Fig. (3.15).

### 3.5.2.2 POSIX-level abstraction

EBEX must ensure that it can write two full copies of the data to the array safely, even if an individual disk should fail during the flight. EBEX utilizes each of the pressure vessel disk arrays as linear volumes, filling one disk in each array at a time with data until it is either full or experiences an error condition that causes it to malfunction. We then proceed to do the same with the next disk in the array.

From the point of view of our existing codebase, this background processing needed to be completely interchangeable with standard file system reference calls. EBEX implements a Portable Operating System Interface (POSIX) compatible layer of system calls for opening, closing, reading and writing files to the disks. This wraps standard system calls, facilitating a smooth transition of the legacy BLAST codebase and allowing non-expert developers to continue using familiar paradigms.

The disk management system consists of layered threads handling the file-level interface that is exposed to the larger system and the disk management thread that controls where files are written in the larger array.

### 3.5.2.3 Disk level management

Before writing a data file to disk, the system must first decide which disk to use and verify that the network connection, AoE blade and disk all remain functional. In a directly connected disk array, this level of processing happens transparently in the background. The addition of a network connection and AoE blade hardware between the disk and flight computer introduces latencies in the case of hardware failure that order on minutes. EBEX is unable to buffer more than 30 seconds worth of accumulated data without writing it to disk or discarding it. Therefore, we implement



a poll-based system check prior to issuing write.

When the system starts, it calls a polling function, asking each disk to respond. This populates a list of disks, from which the disk management interface can select one for use. The disk management interface tracks available free space on each disk, failure counts and responsivity to query requests. These data are periodically written to a section of EEPROM<sup>23</sup> flash memory on each blade. By storing the data with the disk unit, the disk management interface does not need to mount a remote disk in order to determine either its viability or available free space.

The disk management interface is responsible for transparently shifting the disk in use between system file writes. To do this, file writes are consolidated in a pool with disk verification happening between each batch of writes. When the disk management detects that a disk is not longer responding or has run out of space on the disk for the next batch of file writes, it mounts the next viable disk in its array and creates an identical set of files on the new disk.

Handling of the previous disk is moved to a low-priority, background process. The disk is marked as either full or degraded, depending on the reason for our ceasing use of it. Each file on the disk is closed (if possible) and the file descriptors returned to the operating system for re-use. In the case of disk or network failure that prevents the disk management system from correctly closing the file, a system-level timeout of 2 minutes takes effect, after which the files are forcibly closed.

In principle, this action may result in the loss of data whose writes are either pending in a system-level cache or have not been verified by the underlying hardware as written to disk. In practice, we instruct the operating system to avoid caching files as we perform caching at the disk management level prior to issuing disk writes. This leaves only the potential for losing writes that have been issued but not verified.

The ATA over Ethernet framing structure limits the size of an atomic write to a single ethernet frame maximum transmission unit (MTU), a configurable parameter.

---

<sup>23</sup>Electrically Erasable Programmable Read-Only Memory

The default MTU size is 1500 bytes, an acceptably small size for loss in an error case that we expect to be extremely rare. Indeed, during the Antarctic science flight, we experienced no events where disk failure caused the system to shift its writing target in this fashion.

#### 3.5.2.4 File level management

The file level management maps open files to their respective interfaces in the disk management level. The typical life-cycle of a file is show in Fig. (3.16). At the open stage, a new file is created in the disk management level and memory is allocated for a sequential write queue. The application then writes data to the file reference, which is stored in an expandable queue. Finally, the file is marked for closing when the application has completed its use of the file.

At no point in the file level management is there any interaction with the underlying disk media. The file's sequential write queue is indefinitely expandable, allowing us to accept data until we have expended all available system memory. This ensures that file actions, which may occur inside of a critical loop, are not delayed by media errors or network latency. All file system interface calls that can be delayed or blocked are handled asynchronously by the Disk Manager thread as shown in Fig. (3.16). In the case where no disks are available for extended periods of time, we are forced to discard data until and unless the disks resume communications.

#### 3.5.2.5 Problems

While our implementation of multi-disk management successfully operated during the long duration flight, it is important to note a number of the limitations we encountered, how we worked around them and what effects they might have on similar experiments in the future.

**Network compatibility** We initially intended to utilize a redundant ring network configuration between disk pressure vessels. This would have provided an extra layer

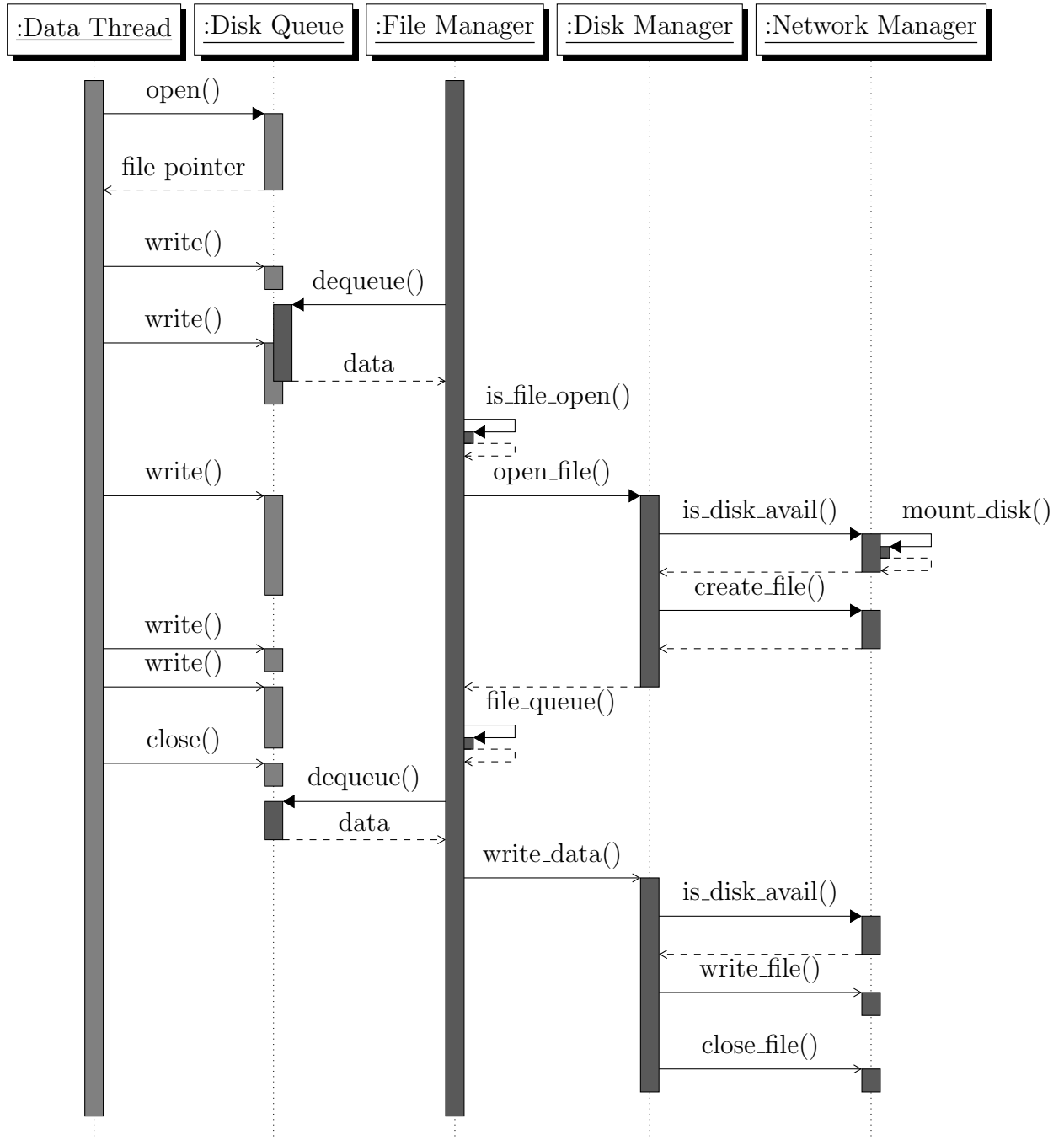


Figure 3.16: A typical file lifecycle. All file actions are serialized by the Disk Queue FIFO. Data are queued per file by the File Manager in order to amortize the network access costs of probing the AoE disks prior to writes.

of redundancy against a network failure by providing a secondary path over which data could be written to the pressure vessels.

EBEX's real-time ring switches are manufactured by Sixnet<sup>24</sup> and designed to provide high-reliability networking. Unfortunately, we discovered that the Sixnet ring switches did not handle high-volume ethernet frames reliably when operating in a ring configuration due to conflict between the ring and AoE protocols. As the AoE protocol relies exclusively on ethernet frames, this limitation caused periodic packet loss while writing data to disk, across an ethernet ring.

We attempted to resolve the issue with Sixnet after it became apparent that this issue was exclusively caused by their hardware. However, as they reported that we were the only customer attempting to use their hardware in a configuration that relied heavily on ethernet frames, they declined to provide a firmware update to fix the problem. We eventually disabled the ring switching functionality over the segment of our network from the flight computers to the disk pressure vessels that carried AoE data. Shown in Fig. (3.11), each disk pressure vessel (denoted PV1 and PV2) are connected by a single ethernet cable to the flight computer crate. Data redundancy remains ensured as PV1 and PV2 contain identical copies of the flight data.

**Coraid AoE Blades** We chose the AoE protocol for our data storage because it offered the ability to design a redundant connection (see above) and mount an array of disks on a convenient backplane. The AoE blades were advertised as supporting 5,000 Input/Output Operations per Second (IOPS)[127] and 5 megabytes per second (MBps) sustained throughput.

In our testing, while our flight hardware is able to sustain network-limited IOPS and MBps using the AoE protocol, the AoE blade is limited to 3.5 MBps and fewer than 2,500 IOPS. In the use case of writing to a single file, each I/O operation pair consisted of a 1,480 byte write (MTU limited) and the return confirmation of the operation. However, when writing to a filesystem with multiple open files, the first

---

<sup>24</sup>[http://www.sixnet.com/dist/ET-9RS\\_datasheet.pdf](http://www.sixnet.com/dist/ET-9RS_datasheet.pdf)

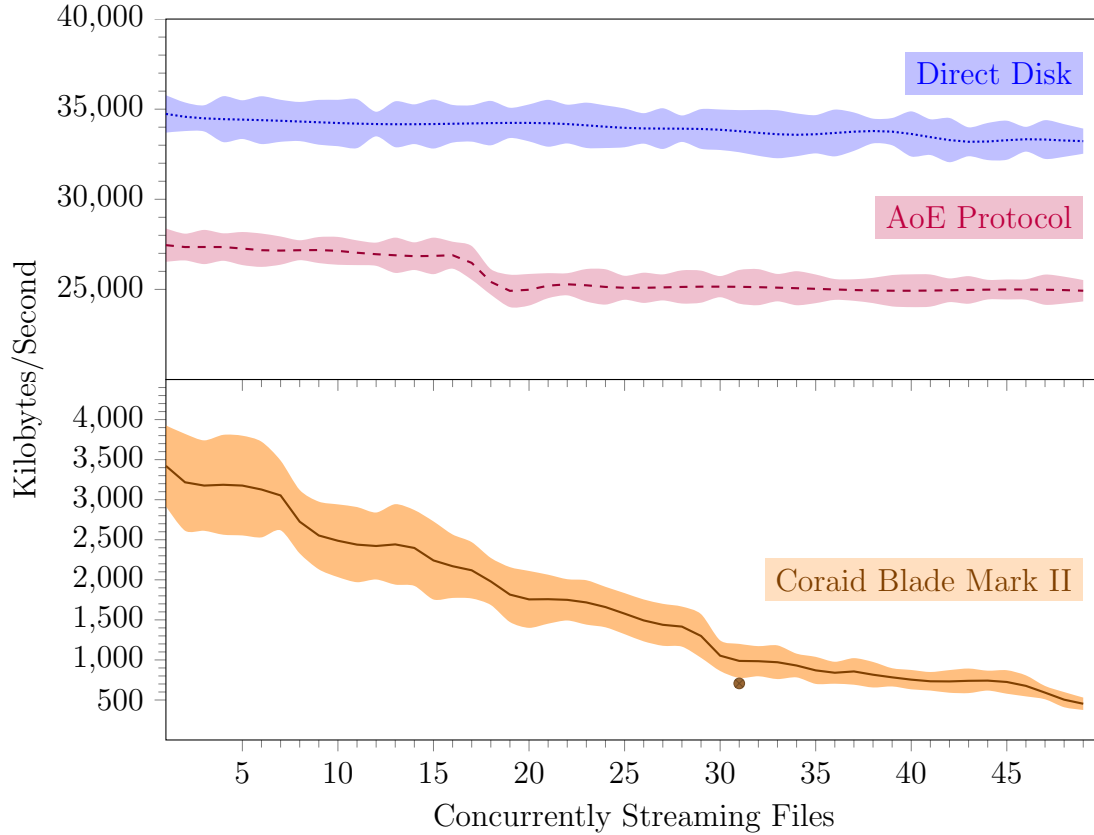


Figure 3.17: Cumulative performance of the disk subsystem when streaming to multiple open files. The top two plots show the performance of writes performed directly to the disks (.....) and writing via the AoE network protocol to the disks hosted on a separate computer (---). The bottom plot (different scale) shows the performance when writing to the disks mounted on the Coraid Blade targets (—). The shaded regions denote the  $2\sigma$  confidence interval for the sampling done over 18 of the flight disks. The EBEX data rate (•) lies just under our limitation.

$\sim 50$  I/O operations are consumed by directory and file look-up operations, after which data can be written at the sustained rate until a new filesystem block needs to be allocated.

The EBEX system maintains between 35-50 open files at a given time. 31 files receive streaming data that require rapid, continuous writes to disk. Accounting for overhead, the AoE subsystem can sustain  $< 1$  MBps of actual data writing bandwidth as shown in Fig. (3.17). This speed is sufficient for EBEX as we require a minimum

of 707KBps for recording basic bolometric and attitude control data.

Additional improvements to the EBEX including recording higher precision bolometer samples, additional characterization information and multiple star camera images, are implemented in EBEX software but were ultimately disable for the LDB flight due to the bandwidth limitation imposed by the AoE blades.

### 3.5.3 BLASTBus Interface

The BLASTBus PCI card provided the flight computers with their connection to the BLASTBus itself (see Section 3.4.1) as well as providing the biphas-encoded line of sight communications link (see §3.5.5).

The flight interface to this card is handled by a PCI kernel driver which exposes a block-level file interface to the user-mode flight software on one end and the PCI card firmware buffer on the other end. The original driver was written for the BLAST experiment and was utilized by EBEX during the 2009 North America engineering flight.

The original driver suffered from a number of limitations, identified during this test flight, that made it unsuitable for use for the EBEX LDB science flight. Primary among these were a fixed buffer size that frequently overflowed in the EBEX use case and utilization of interface calls that limited use to a specific, outdated Linux kernel version. The kernel driver has been largely re-written to follow modern driver design guidelines. This allows EBEX to utilize error fixes and performance improvements implemented in newer Linux kernels. In addition, the buffer size limitation was overcome implementation of a novel, single-reader, single-writer, lock-free expandable buffer.

The previous FIFO algorithm, described in §3.5.1.1 is unsuitable to this use case as write to the BLASTBus and biphas interface occur in 32-bit increments. Thus a node assignment for each write would create an overhead cost of 100%, rendering such a system impractical. Instead, wrap a queues structure around a multi-word buffer.

The underlying buffer is a standard linear buffer with head and tail markers allowing single-reader, single-writer concurrent access. This type of structure is well described in literature[62]. However, to allow expandability, this buffer is wrapped by a queue structure whose access is ordered to synchronize concurrent reading and writing. The enqueue and dequeue operations are detailed in Algorithms A.3 and A.4 respectively.

These algorithms are appealing particularly for their simplicity. They place no system requirements on the implementation and are limited in their size only by available memory. The underlying buffer size can be tuned to balance the allocation costs against memory usage constraints. In our implementation, the allocator is handled by the Linux kernel `kmem_cache` cache chain functions. These are low CPU-overhead allocation and deallocation operations for fixed-size chunks[13].

### 3.5.4 Automation

The EBEX telescope is designed to operate autonomously during its science flight. Telemetry is not always available, depending on satellite visibility, scheduling and potential equipment malfunction. Even when available, commanding response times are sufficiently long that ground-based operators cannot respond to events with sufficient rapidity to meaningfully effect short-timescale events. These limitations necessitated the use of an automated scheduling system to handle both science and housekeeping operations during flight.

#### 3.5.4.1 Historical Scheduler

EBEX inherited a telescope scheduling software from the BLAST experiment that provided some of the desired features including Local Sidereal Time (LST) based scheduling and configurable commanding files. This code provided telescope pointing control and could switch between pre-defined schedules based on hard-coded, discrete events such as telescope latitude[136].

The requirements for EBEX automation are more expansive, in large part due to the complexity of cycling our cryogenic fridge and tuning a substantially larger array of bolometers. As such, we require the additional ability to time our events based off of Coordinated Universal Time (UTC) rather than solely LST. Additionally, we need the ability to conditionally execute various actions, based on the results of previous actions.

These requirements necessitated the development of a new model of telescope automation, designed to not just control the pointing but also perform housekeeping functions, monitor and control cryogenic cooler cycles and ensure proper attitude control system operation.

#### 3.5.4.2 Event-Based Automation

The automation design for EBEX is event-based, where each event is represented by a unique signal name. Signals are either emitted or caught by various components in the automator. This allows for a modularity of design, where multiple individuals or groups can work on and design their own scheduled components without requiring continual integration of the automation files.

In EBEX, the schedule files controlling the DfMux readout system actions including tuning the full bolometer array are designed and maintained by collaborators at McGill University. Schedule files controlling the half-wave plate and performing cryogenic cycling are written and maintained by collaborators at the University of Minnesota. The schedule files governing pointing that control scanning strategy implementation are maintained at Columbia University. These are all brought together under a master rubric that allows synchronization between each subsystem.

The structure of the EBEX automator is broken down into groupings of commands, procedures, signals and monitors with each object operating in a specific bailiwick, while controlling or communicating with other objects through passed signals.



### 3.5.4.3 Commands

Commands are discrete function calls from the automator. Functions are defined for the program at compile-time and are available to both the automator system as well as the human interface for command and control.

In the context of the automator, a command will execute a single function call with parameters defined in one of three ways: set statically in the configuration file, as variables that are set internally by the automator, or as the values of channels stored in the data frame.

The automator can also parse the return value of a command and emit signals based on the value. Distinct signals may also be emitted in the case of the command being interrupted by another action of the automator.

### 3.5.4.4 Procedures

Many commands naturally occur in groups such as the series of commands for cycling the cryogenic coolers show in Figs. (B.2 and B.3). For these cases, EBEX allows multiple commands to be grouped together as procedures. The grouping allows multiple steps to be executed based on a single event. These events may be other procedures, commands, wait statements, signal emissions, variable setting or flow control of the procedure.

Similar to commands, procedures may conditionally emit signals on the occurrence of their events such as beginning, completion or interruption. The sub procedures of the cryogenic cycling are split such that unintended interruption would restart at an appropriate and safe position in cycle. Similar procedures exist to coordinate HWP activity, heat switch activation, attitude control and bolometer tuning.

### 3.5.4.5 Signals

Signals are the base element of the EBEX automator and may be used to stop or start a procedure. Signals are generic in their design, in that all signals look alike,

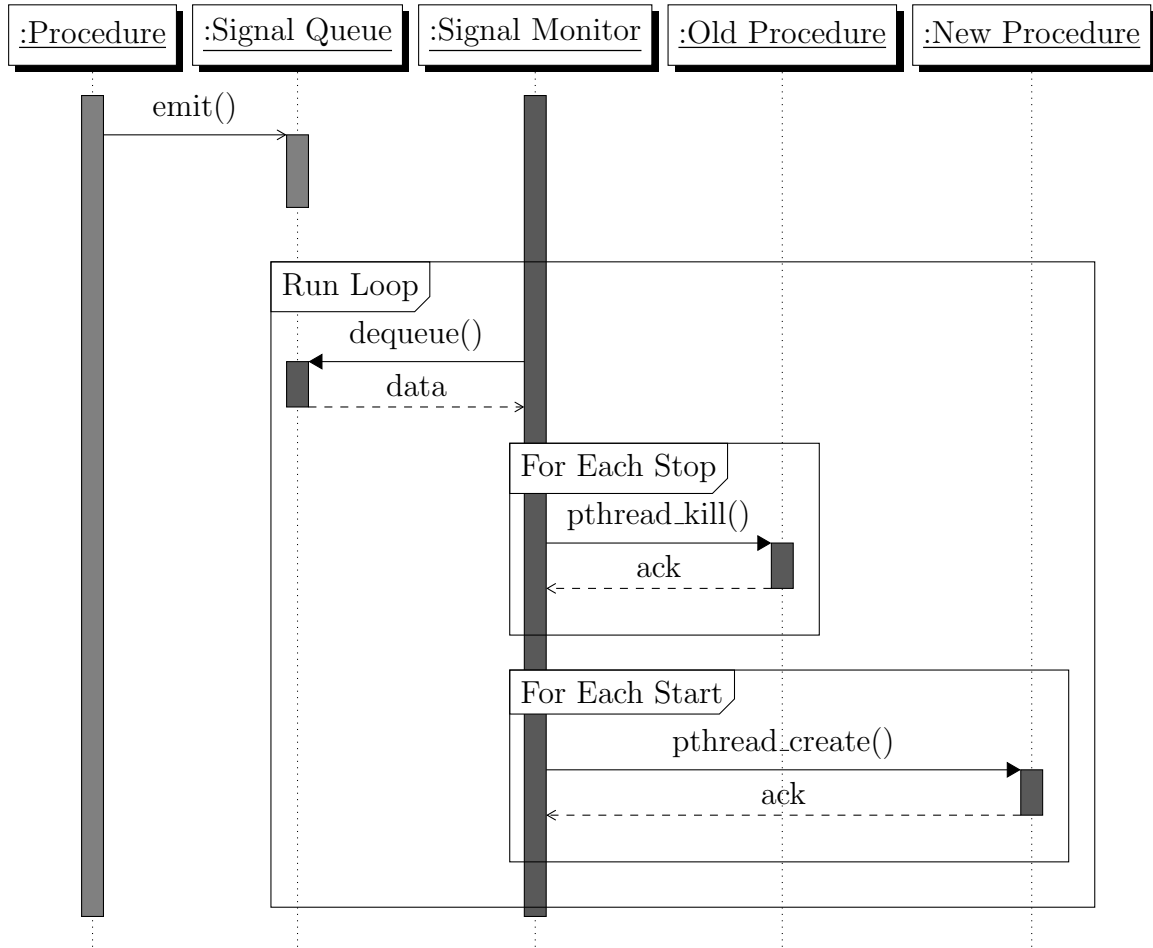


Figure 3.18: Signal handling in the EBEX automator. The signal monitor thread dequeues all waiting signals every 100ms.

regardless of their function. Signals may be emitted by monitor or procedure elements in the automation file.

Each signal is represented by a 32-bit hash of the signal name. When a procedure or monitor emits a signal, the hash of the signal is pushed into a shared signal queue using the FIFO structure described in §3.5.1. A single thread handles signal events every 100ms. Its loop routine is shown in Fig. (3.18).

The automator stores signal events in a table of lists, indexed by the signal hash. For each hash, there are two lists for the two actions that a signal may generate: starting and stopping procedures. When a signal is processed, the handler loop

first spawns a thread for each procedure in the starting stack. It then sends a `pthread_kill()` signal to each running thread in the stopping list.

#### 3.5.4.6 Monitors

Monitors exist to emit specific signals in response to a change in the system state. Monitors can track both data elements that are recorded in the BLASTBus frames as well as variables internal to the automator itself.

Monitors are designed to allow complex combinations of state variables, including boolean and algorithmic combinations. This formalism therefore provides an excellent base for triggering reactions to multi-faceted environmental conditions

#### 3.5.5 Telemetry

EBEX is designed to operate reliably, independent of operator interactions, for many potential flight scenarios. However, continuous monitoring of the status of the system and reliable commanding from ground stations can provide needed corrections to unanticipated events. EBEX uses three downlink channels and three uplink channels for monitoring and commanding. Two channels – TDRSS and Iridium – are satellite-based and available for the majority of the flight. The remaining line-of-sight link is only available until the payload passes over the horizon, typically 12-14 hours after launch.

Telemetry between EBEX ground station equipment and the balloon payload passed over CSBF-provided links. The primary Operations Control Center (OCC) for CSBF is located in Palestine, TX, while the remote OCC (ROCC) is located at the launch site at Williams Field, Antarctica. EBEX ground equipment is co-located with CSBF at both locations. The ROCC provides line-of-sight and Iridium communication links with the payload while the OCC provides an Iridium link as well as connectivity to the White Sands TDRSS ground station as shown in Fig. (3.19).

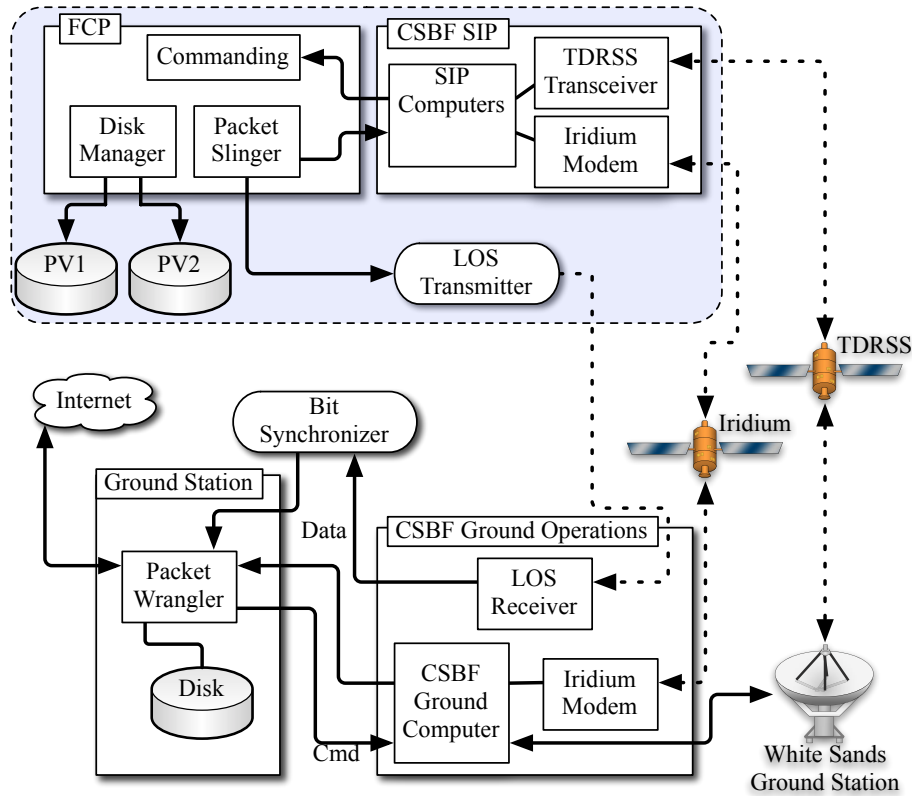


Figure 3.19: EBEX Communication Paths. The blue, enclosed section in the top left represents the gondola at altitude. Links shown in dashed lines are wireless, while solid lines represent physical connections. On the ground, our commanding and downlink work is handled by the “Packet Wrangler” program that writes data out to disk and provides internet connection for further downstream clients requiring commanding and data.

### 3.5.5.1 Telemetry Software

EBEX generates over 6Mbits of data per second during normal operations. This exceeds even our widest bandwidth channel as listed in Tab. (3.2). Packet Slinger, our telemetry software that runs in FCP provides two means of winnowing the rate down sufficiently to fit in the bandwidth provided. First, it integrates with a lossless compression library specifically designed for EBEX data. And, second, it creates a prioritized downlink structure that selects the most important data first for downlink and fills the bandwidth channel in order of importance.

Telemetry Link	Path	Downlink Rate	Uplink Rate	Limitations
Biphase	LOS	1,000kbps	1,920bps	Limited Time
High-Gain PCM	TDRSS	92kbps	1,920bps	Slew Rate $\leq 0.7^\circ/\text{sec}$
High-Gain Std	TDRSS	60kbps	1,920bps	Slew Rate $\leq 0.7^\circ/\text{sec}$
Omnidirectional	TDRSS	$\leq 20\text{kbps}$	1,920bps	Rate depends on signal strength
Dial-up	Iridium	6kbps	3 commands per 2 minutes	

Table 3.2: EBEX Telemetry Links and Associated Characteristics

Compression of the downlink data works by predicting the next byte in a stream, based on the contents of previously observed bytes. While a lossy-type compression will also choose select data to discard based on some importance metric, lossless data compression works by representing predictable sequences by smaller markers. EBEX approaches generates predefined statistic predictions of byte sequences based on pre-flight calibration.

These statistics provide a starting distribution that is assumed at the beginning of each new packet. Compression ratios of the first bytes in a packet are likely to be worse than compression ratios of the last bytes in a packet as the prediction statistics build over the full packet length. Because of this, data that are similar will have a higher compression ratio when grouped together in the same packet than when compressed separately. Packet slinger accommodates this disparity by grouping data streams into bundles, which are then handled as a single unit.

A bundle is one of three types of data handled by Packet slinger. The remaining two are known as *periodic* and *file* types. A periodic data type is used for data that change infrequently over the course of the flight. These data often encode state values, error flags or command variables. Although passing a stream of identical values through a compression routine would yield excellent compression, this creates both unnecessary system loading as well as consuming, at a minimum, the space required

for a packet header in the downlink bandwidth. Instead, periodic types are sent over our downlink only when they change.

The final type of data that is sent over the downlink is the file type. This type indicates to the ground station that it should create a new file with the enclosed data as it does not represent a continuous stream. This type of data is used to downlink tuning algorithm returns from the DfMux cards, logging file data and command results returned from serial interfaces such as the half wave plate controller. These files are multiplexed into the downlink stream as they are generated.

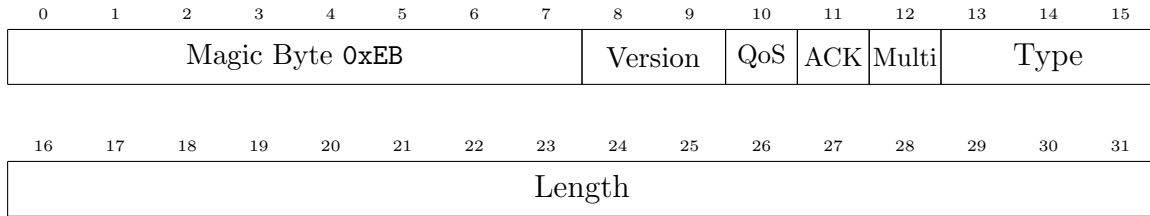


Figure 3.20: Packetslinger Header Definition.

To differentiate between data types as well as provide a consistent interface for telemetry links, EBEX communications are prefixed by a common header shown in Fig. (3.20). The header always begins with the same byte 0xEB, allowing for a rapid synchronization of an unknown data stream. Following the magic byte, is the version number differentiating between different formats of data. In practice, EBEX used only a single version number, effectively extending the magic byte match by an additional 2 bits, which are set to 0b01.

Following the version number, the *Quality of Service* (QoS) bit is conditionally set when the sender requests an acknowledgement of successful transmission from the sender. As EBEX utilizes 60-100 packets per second during steady-state operation, and greater than 400 packets per second during the line of sight period, uplink volume of acknowledgements for each packet received is infeasible.

Instead, EBEX implements a selective acknowledgement scheme whereby certain packets, deemed high priority, set the QoS bit. Packets having this bit set are retained by the flight computer until their receipt is acknowledged by the ground

station, whereupon they would be discarded. If an acknowledgment value is received out of order or no acknowledgement is received for 3 minutes, the flight computer will re-send the unacknowledged packets, ensuring their receipt.

Acknowledgements consist of a 32-bit Cyclic Redundancy Check (CRC) value calculated on the full content of the packet, including its header. Acknowledgements are appended to an extant data packet when available. This allows the relatively small acknowledgement value to avoid its own header overhead when being transmitted concurrently with other data. When a data packet is followed by an acknowledgement, the ACK flag (bit 11) is set in the header to signal to the recipient that an acknowledgement follows.

The *Multi* flag (bit 12) in the header is another means of amortizing header cost over multiple, similar packets. Packets smaller than 256 bytes, that share the same QoS flag, are concatenated together, with a single byte prefix marking the sub-packet length. This process saves 24 bits of overhead per subpacket. Subpackets are gathered and concatenated until the full packet length ( $2^{16} - 1 = 65535$  bytes) is filled or the gathering routine reaches a configurable timeout period, whichever occurs first.

As a substantial percentage of the EBEX data are either the intrinsically small *periodic* data types or are highly compressed, this process saves an average of 13.4% of the 20kbps standard TDRSS downlink.

### 3.5.5.2 Line-of-Sight

During the first 12 hours following launch, the primary communications path is over a 1Mbit per second, biphasic-modulated data link, broadcasting on either L- or S-band. On the payload, the signal passes to the LOS transmitter, which is connected to an omni-directional antenna suspended beneath the gondola.

This broadcast is received by CSBF and forwarded over coaxial cable to a bit synchronizer that locks onto the biphasic signal and demodulates it into a clock stream and a non-return-to-zero (NRZ) data stream. These two lines are read in by a PCI

card designed by the BLAST experiment that translates the NRZ data into a digital stream for reading.

Although the default mode of operation for packetwrangler is to utilize compressed packets, EBEX uses the higher-rate LOS datastream in a hybrid manner, with part of the bandwidth allocated for uncompressed, frame-based data and part of the bandwidth utilizing compressed packets.

The frame-based data are those data written over the BLASTBus architecture into 100Hz frames and include ACS data and housekeeping information. The split provides high-rate updates of critical control information during the initial tuning and testing of the system upon reach float altitudes. It leaves approximately 700kbps of bandwidth for bolometer readout, dfMux housekeeping data and algorithm returns encoded in the packet-based format.

## 3.6 Summary

The EBEX system provides a robust, highly configurable platform from which to conduct balloon-borne CMB measurements. It builds on the substantial work of BLAST and previous experiments. The software built by EBEX addresses many problems faced by the next generation of kilo-pixel based arrays and provides a useful baseline for future work.



## Chapter 4

# Attitude Reconstruction

“The general ORDER, since the whole began,  
Is kept in Nature, and is kept in man.”

---

— Alexander Pope, *Essay on Man*

### 4.1 Overview

The process of attitude (pointing) reconstruction for balloon-borne observation platforms is crucial to successful mapmaking and data reduction. Bolometric measurements must each be correctly associated with their position on the celestial sphere. Additionally, as EBEX is a polarized receiver, the rotation of each bolometer relative to a fixed reference vector must also be determined.

EBEX uses three distinct reference frames to refer to its attitude. The first is a standard, equatorial coordinate system on the sky. We will use decimal radians to mark right ascension and declination. Attitude additionally requires a third rotation, called roll, that will be taken to be a right-handed rotation with the north celestial pole corresponding to zero roll. This coordinate system is used by our star cameras as well as being our reference coordinate system for science observations.

The second coordinate system used is a horizontal coordinate system. This is

marked in azimuth, a rotation about the gravitational vector, and elevation, a rotation orthogonal to the plane of the horizon. Similar to the equatorial coordinate system, there is a roll component as well with the gravity vectors complement corresponding to zero roll. This coordinate system is used to describe gondola scanning patterns as well as coordinates returned by the coarse attitude sensors.

The final coordinate system is the reference or gondola coordinate system. This is a right-handed system with rotations relative to the current attitude of the gondola. In the case of zero roll, a gondola coordinate system would be equivalent to the horizontal coordinate system. This coordinate system is used to propagate attitude solutions using velocity measurements from our three-axis gyroscopes, which measure motion in a frame fixed to the gondola orientation.

We represent the gondola payload as a rigid body with coordinate frame  $[\mathbf{X}_g, \mathbf{Y}_g, \mathbf{Z}_g]$ . Translating between coordinate systems then requires rotating by angles  $[\alpha, \beta, \gamma]$  that are, in general, dependent on other variables of the system for their values. The order in which the rotations are applied is important to note as they do not commute.

For the purposes of this work, we will use **YYZ** ordering, where  $\alpha$  is the yaw motion about  $\mathbf{Z}_g$ ,  $\beta$  is the pitch motion about  $\mathbf{Y}_g$  and  $\gamma$  is the roll of the platform about  $\mathbf{Z}_g$ . This schema is shown in detail in Fig. (4.1).

The constraints EBEX faces with regards attitude reconstruction are substantially different than those encountered by previous balloon-borne experiments. In order to cover the requisite patch size a sufficient number of times to achieve our low- $\ell$  science goals of measuring the imprint of inflationary gravitational waves, EBEX utilizes a scan speed of 42'/second. This speed limits star camera image acquisition to the scan edges. The patch size also means that there are approximately 40 seconds of motion between each star camera images.

During the time between star camera images, EBEX integrates the rotational speed measurements from 2 sets of 3-axis fiber optic gyroscopes. Gyroscopes have an intrinsic noise level in their readout (detailed in §4.3.1) that accumulates in the

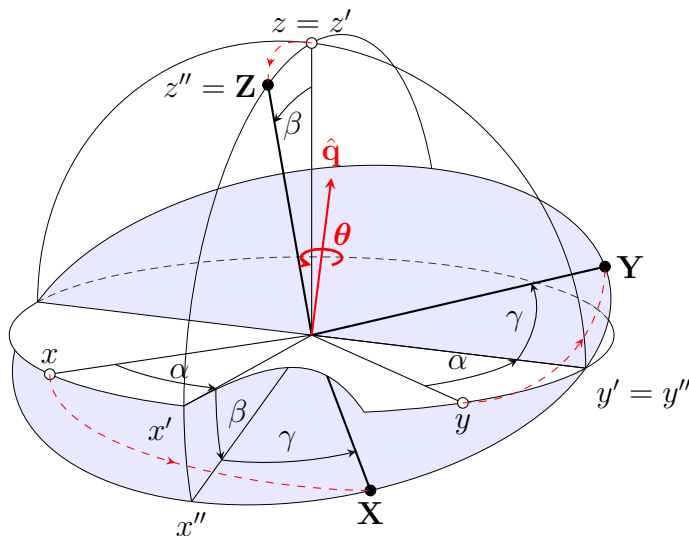


Figure 4.1: We show the effect of a **ZYZ** rotation, taking the (0,0,0) coordinate to lie along the **Z** vector. In addition to inertial coordinates, we use this system to represent celestial coordinates as well. In the case of equatorial coordinates, the line from the Earth's center to the North Celestial Pole will represent the  $z$ ,  $\alpha \equiv$  right ascension,  $\beta \equiv \pi/2$ -declination and  $\gamma \equiv$  roll about **Z**.

attitude, leading to a random walk in a naively integrated solution.

While this uncertainty would not substantively affect science at low- $\ell$ , EBEX's second science goal of measuring the B-mode spectrum induced by gravitational lensing is negatively impacted by even tens of arcseconds of error. The combination of these constraints means that previous solutions used by experiments such as BOOMERanG[87], MAXIMA/MAXIPOL[100], HEFT[48], BLAST[86] and HiPEG[19] are insufficient for EBEX's use.

This section presents one of the two methods EBEX uses to reconstruct a finalized attitude stream from its various sensors. An alternate method is currently being developed and tested and may be used for the finalized EBEX data set.

In this section, we begin by laying out the theoretical underpinnings of the attitude estimation problem and, as it relates directly to our ability to solve the attitude estimation problem, a brief overview of the quaternion formalism used. We then review some of the various solutions used by previous experiments to estimate attitude

information and use this to motivate our filtering solution. Finally, we derive the verification metrics we will use to measure performance of our filter.

## 4.2 Theory

### 4.2.1 Wahba's Problem

In 1965, Grace Wahba formulated the generalized problem for free platform attitude estimation from multiple sensors as finding a rotation matrix  $M$  that brings a set of  $n$  vector measurements  $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$  into least squares coincidence with a set of reference vectors  $\{\mathbf{v}_1^*, \mathbf{v}_2^*, \dots, \mathbf{v}_n^*\}$  [131]. This can be written as minimizing a cost function  $J$

$$J(M) = \frac{1}{2} \sum_{j=1}^n a_j \|\mathbf{v}_j^* - M\mathbf{v}_j\|^2 \quad (4.1)$$

where  $a_j$  represents the positive weighting of the  $j^{\text{th}}$  vector measurement.

The Wahba problem was solved analytically the following year by Wahba herself as well as a number of other researchers [34]. All proposed solutions used a similar approach of re-writing the cost function as

$$J(M) = \text{tr} \left[ (V^* - MV)^T (V^* - MV) \right] \quad (4.2a)$$

$$= \text{tr} (V^{*T} - V^T M^T) (V^* - MV) \quad (4.2b)$$

$$= \text{tr} V^{*T} V^* + \text{tr} V^T V - 2\text{tr} V^T M^T V^* \quad (4.2c)$$

where  $V \equiv \{a_1 \mathbf{v}_1, a_2 \mathbf{v}_2, \dots, a_n \mathbf{v}_n\}$  and  $V^* \equiv \{a_1 \mathbf{v}_1^*, a_2 \mathbf{v}_2^*, \dots, a_n \mathbf{v}_n^*\}$ . Noting that the first two terms are independent of  $M$ , it is clear that the minimum of  $J(M)$  is achieved by maximizing  $K(M) = \text{tr} V^T M^T V^*$ .

In 1968, Davenport showed that this function can be re-expressed in terms of quaternion variables [25], a formalism that will be discussed in more detail in the following section. For the current purposes, it will suffice to recognize that a quaternion

represents a three-dimensional, rotational transformation by use of a unit vector and a rotation about that vector. As such, it is a 4-vector with a single degeneracy.

Following Davenport's derivation, note that  $K(M)$  may be re-written as the sum of entry-wise products as

$$K(M) = \text{tr} [V^T M^T V^*] \quad (4.3a)$$

$$= \text{tr} \left[ \left( \sum_{j=1}^n a_j \mathbf{v}_j \mathbf{v}_j^{*T} \right) M \right] \quad (4.3b)$$

$$= \text{tr} [B^T M] \quad (4.3c)$$

where  $B$  is the  $3 \times 3$  attitude profile matrix. I should be noted here that the solution to Eq. 4.2c, in this formulation is linear in  $M$ , the attitude matrix. Through algebraic manipulation ([25], Eqs. 12-24), Davenport derived a quaternion transformation function  $Y(B)$

$$Y(B) = \begin{bmatrix} 2K_{11} - \text{tr}B & B_{12} + B_{21} & B_{13} + B_{31} & B_{23} - B_{32} \\ B_{12} + B_{21} & 2B_{22} - \text{tr}B & B_{23} + B_{32} & B_{31} - B_{13} \\ B_{13} + B_{31} & B_{23} + B_{32} & 2B_{33} - \text{tr}B & B_{12} - B_{21} \\ B_{23} - B_{32} & B_{31} - B_{13} & B_{12} - B_{21} & \text{tr}B \end{bmatrix} \quad (4.4)$$

that allows a re-expression of Eq 4.3c in terms of a quaternion rotation as  $K(\bar{\mathbf{q}}) = \bar{\mathbf{q}}^T Y(B) \bar{\mathbf{q}}$ . The quaternion that then maximizes this expression is the eigenvector  $\bar{\mathbf{q}}'$  that corresponds to the maximum eigenvalue.

### 4.2.2 Quaternion Formalism

In the simplest representation, an attitude state vector need only contain three distinct parameters. Conventionally, in flight applications, these are defined relative to the gondola body as *yaw*, *pitch* and *roll*. Parameterized as shown in Fig. (4.1), we rotate first about the  $z$  axis by  $\alpha$  for the yaw, then about the  $y'$  axis for the pitch and finally

about the  $z''$  for the roll. This is referred to as the **ZYZ** convention.

There exists a degeneracy in this and indeed any minimal attitude representation, as rotating the state by  $\beta = \{\pi, 0\}$  results in the inability to distinguish between rotations  $\alpha$  and  $\gamma$ . This phenomenon is referred to as 'gimbal lock'.

Gimbal lock is analogous to the two-dimensional parameterization of a sphere. Although we can specify any point on the Earth (or any smooth sphere) by latitude and longitude, at either the North or South Pole, the idea of longitude becomes meaningless. To solve this problem on a sphere, we can also specify each point as a three-element vector  $\mathbf{r} = \{x, y, z\}$  subject to the constraint  $x^2 + y^2 + z^2 = |\mathbf{r}|^2$ .

Extending this idea to three parameters on a unit hypersphere, quaternions parameterize a rotation as a unit vector  $\hat{\mathbf{q}}$  on a sphere, scaled by the rotation. Thus  $\hat{\mathbf{q}} = q_x \hat{\mathbf{i}} + q_y \hat{\mathbf{j}} + q_z \hat{\mathbf{k}}$ . Rotating about this vector in three-space by angle  $\theta$  gives

$$\bar{q} = e^{\frac{1}{2}\theta(q_x \hat{\mathbf{i}} + q_y \hat{\mathbf{j}} + q_z \hat{\mathbf{k}})} \quad (4.5a)$$

$$= \sin \frac{\theta}{2} (q_x \hat{\mathbf{i}} + q_y \hat{\mathbf{j}} + q_z \hat{\mathbf{k}}) + \cos \frac{\theta}{2} \quad (4.5b)$$

The choice of quaternions for parameterization is driven not just by the desire to avoid gimbal lock as this might also be achieved by utilizing a full rotation matrix, but also by the simplicity of visualizing rotations. Additionally, from an implementation stand-point, following a single rotation, both an attitude matrix and quaternion must be re-normalized to remain valid. Renormalizing an attitude matrix requires either approximation or matrix inversion, both of which are disadvantageous with regards either accuracy or speed. Alternatively, as a four-vector, quaternion normalization is trivial to implement, dividing each element by the vector norm.

### 4.2.3 Creating a Rotation Matrix

To motivate the use of quaternions to represent an arbitrary rotation, let us first imagine a rotation about the  $\hat{\mathbf{z}}$  vector of magnitude  $\theta$ . Then in rotated coordinates:

$$\hat{\mathbf{x}}' = \cos \theta \hat{\mathbf{x}} + \sin \theta \hat{\mathbf{y}} \quad (4.6)$$

$$\hat{\mathbf{y}}' = -\sin \theta \hat{\mathbf{x}} + \cos \theta \hat{\mathbf{y}} \quad (4.7)$$

$$\hat{\mathbf{z}}' = \hat{\mathbf{z}} \quad (4.8)$$

Alternatively, written in matrix form, we isolate a rotation matrix  $\mathbf{R}$ :

$$\begin{bmatrix} \hat{\mathbf{x}}' \\ \hat{\mathbf{y}}' \\ \hat{\mathbf{z}}' \end{bmatrix} = \underbrace{\begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{R}(\hat{\mathbf{z}}, \theta)} \cdot \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{y}} \\ \hat{\mathbf{z}} \end{bmatrix} \quad (4.9)$$

By cyclic permutation, we can also find

$$\mathbf{R}(\hat{\mathbf{x}}, \theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix} \quad (4.10)$$

$$\mathbf{R}(\hat{\mathbf{y}}, \theta) = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \quad (4.11)$$

Now, noting that

$$\mathbf{R}(\hat{\mathbf{z}}, \theta) \cdot \hat{\mathbf{x}} = \cos \theta \hat{\mathbf{x}} - \sin \theta \hat{\mathbf{y}} \quad (4.12)$$

$$= \cos \theta \hat{\mathbf{x}} - \sin \theta \hat{\mathbf{z}} \times \hat{\mathbf{x}} \quad (4.13)$$

With similar expressions for the permutations of other vectors, we can take  $\hat{\mathbf{n}}$  as an

arbitrary unit vector and  $\mathbf{v}_\perp$  as the projection of a column vector  $\mathbf{v}$  onto the plane orthogonal to  $\hat{\mathbf{n}}$  to write

$$\mathbf{R}(\hat{\mathbf{n}}, \theta) \mathbf{v}_\perp = \cos \theta \mathbf{v}_\perp - \sin \theta \hat{\mathbf{n}} \times \mathbf{v}_\perp \quad (4.14)$$

The component of  $\mathbf{v}$  parallel to  $\hat{\mathbf{n}}$  is unchanged by the rotation about  $\hat{\mathbf{n}}$ . Thus, in general

$$\mathbf{R}(\hat{\mathbf{n}}, \theta) \mathbf{v} = \mathbf{v}_\parallel + \cos \theta \mathbf{v}_\perp - \sin \theta \hat{\mathbf{n}} \times \mathbf{v}_\perp \quad (4.15)$$

Evaluating the cross product in this expression, we note that

$$(\mathbf{x} \times \mathbf{y})_i = \sum_{k=1}^3 \epsilon_{ijk} x_k y_j \quad (4.16a)$$

$$\mathbf{x} \times \mathbf{y} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \cdot \mathbf{y} \quad (4.16b)$$

$$\mathbf{x} \times \mathbf{y} = -[[\mathbf{x}]] \cdot \mathbf{y} \quad (4.16c)$$

Where we have defined a *cross-product operator* by  $[[\ ]]$ <sup>1</sup>. Therefore, from Eq. (4.15)

$$\mathbf{R}(\hat{\mathbf{n}}, \theta) \mathbf{v} = \mathbf{v}_\parallel + \cos \theta \mathbf{v}_\perp - \sin \theta [[\hat{\mathbf{n}}]] \cdot \mathbf{v}_\perp \quad (4.17)$$

Decomposing  $\mathbf{v}$  into parallel and orthogonal components, we see that

$$\mathbf{v} = \mathbf{v}_\parallel + \mathbf{v}_\perp \quad (4.18a)$$

$$= (\hat{\mathbf{n}} \cdot \mathbf{v}) \hat{\mathbf{n}} - \hat{\mathbf{n}} \times (\hat{\mathbf{n}} \times \mathbf{v}) \quad (4.18b)$$

$$= \hat{\mathbf{n}} \hat{\mathbf{n}}^T \mathbf{v} - [[\hat{\mathbf{n}}]]^2 \mathbf{v} \quad (4.18c)$$

---

<sup>1</sup>This operator has many useful properties which will not be re-derived here but can be found in [121]



Substituting into Eq. (4.17), we have that

$$\mathbf{R}(\hat{\mathbf{n}}, \theta) \mathbf{v} = \hat{\mathbf{n}} \hat{\mathbf{n}}^T \mathbf{v} - \cos \theta [[\hat{\mathbf{n}}]]^2 \mathbf{v} + \sin \theta [[\hat{\mathbf{n}}]] \mathbf{v} \quad (4.19)$$

Which allows us to express the rotation matrix as

$$\mathbf{R}(\hat{\mathbf{n}}, \theta) = \cos \theta \mathbf{I} + (1 - \cos \theta) \hat{\mathbf{n}} \hat{\mathbf{n}}^T + \sin \theta [[\hat{\mathbf{n}}]] \quad (4.20)$$

By using the half-angle formulas, we can then re-write this as

$$\begin{aligned} \mathbf{R}(\hat{\mathbf{n}}, \theta) = & \left( \cos^2 \frac{\theta}{2} - \sin^2 \frac{\theta}{2} \right) \mathbf{I} + \\ & 2 \sin^2 \frac{\theta}{2} \hat{\mathbf{n}} \hat{\mathbf{n}}^T + 2 \cos \frac{\theta}{2} \sin \frac{\theta}{2} [[\hat{\mathbf{n}}]] \end{aligned} \quad (4.21)$$

#### 4.2.4 Quaternion Rotation

Returning to our quaternion form from Eq. (4.5b), we use the so-called *Euler-Rodrigues Symmetric Parameters* to represent our rotation in quaternion form as

$$\bar{\mathbf{q}} = \begin{bmatrix} \sin \frac{\theta}{2} \hat{\mathbf{n}} \\ \cos \frac{\theta}{2} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\varrho} \\ q_4 \end{bmatrix} \quad (4.22)$$

Noting that  $\boldsymbol{\varrho}^2 + q_4^2 = 1$ , we see that this parameterization is properly normalized and allows for simple substitution into Eq. (4.21) to parameterize the rotation matrix in terms of quaternions

$$\mathbf{R}(\bar{\mathbf{q}}) = (q_4^2 - |\boldsymbol{\varrho}|^2) \mathbf{I} + 2 \boldsymbol{\varrho} \boldsymbol{\varrho}^T + 2 q_4 [[\boldsymbol{\varrho}]] \quad (4.23)$$

#### 4.2.5 Quaternion Kinematics

To apply the quaternion formalism to the kinematic equations representing the EBEX gondola, we need to propagate the quaternion through time. To do this, we need to

calculate the time derivative of the quaternion. Consider the rotation of a vector  $\mathbf{v}$  by quaternion  $\bar{\mathbf{q}}$ . We write  $\mathbf{v}$  as a quaternion by expanding the dimension and placing a zero in the scalar component. Then the time derivative is

$$\bar{\mathbf{v}}(t) = \bar{\mathbf{q}}(t) \otimes \bar{\mathbf{v}}(0) \otimes \bar{\mathbf{q}}(t)^{-1} \quad (4.24a)$$

$$\Rightarrow \frac{d\bar{\mathbf{v}}}{dt} = \frac{d\bar{\mathbf{q}}}{dt} \otimes \bar{\mathbf{v}}(0) \otimes \bar{\mathbf{q}}^{-1} + \bar{\mathbf{q}} \otimes \bar{\mathbf{v}}(0) \otimes \frac{d\bar{\mathbf{q}}^{-1}}{dt} \quad (4.24b)$$

Then, since  $\bar{\mathbf{q}} \otimes \bar{\mathbf{q}}^{-1} = 1$ , we have

$$\frac{d\bar{\mathbf{v}}}{dt} = \frac{d\bar{\mathbf{q}}}{dt} \otimes \bar{\mathbf{q}}^{-1} \otimes \bar{\mathbf{v}}(t) + \bar{\mathbf{v}}(t) \otimes \bar{\mathbf{q}} \otimes \frac{d\bar{\mathbf{q}}^{-1}}{dt} \quad (4.25a)$$

$$\Rightarrow = \frac{d\bar{\mathbf{q}}}{dt} \otimes \bar{\mathbf{q}}^{-1} \otimes \bar{\mathbf{v}}(t) - \bar{\mathbf{v}}(t) \otimes \bar{\mathbf{q}}^{-1} \otimes \frac{d\bar{\mathbf{q}}}{dt} \quad (4.25b)$$

Note that  $\frac{d\bar{\mathbf{q}}}{dt} \otimes \bar{\mathbf{q}}^{-1}$  is a vector (i.e. the scalar component is 0) by Eq. (4.22). Thus, defining this vector as  $\mathbf{p}(t) \equiv \frac{d\bar{\mathbf{q}}}{dt} \otimes \bar{\mathbf{q}}^{-1}$ , we have

$$\frac{d\mathbf{v}(t)}{dt} = 2\mathbf{p}(t) \times \mathbf{v}(t) \quad (4.26)$$

This gives us a clear representation of our rotation vector  $\boldsymbol{\omega}(t) = 2\mathbf{p}(t)$ .

$$\Rightarrow \bar{\boldsymbol{\omega}} = 2 \frac{d\bar{\mathbf{q}}}{dt} \otimes \bar{\mathbf{q}}^{-1} \quad (4.27)$$

$$\Rightarrow \frac{d\bar{\mathbf{q}}}{dt} = \frac{1}{2} \bar{\boldsymbol{\omega}} \otimes \bar{\mathbf{q}} \quad (4.28)$$

Then, applying the rules of quaternion multiplication, we have

$$\dot{\bar{\mathbf{q}}} = \frac{1}{2} \begin{bmatrix} -[[\boldsymbol{\omega}]] & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{\rho} \\ q_4 \end{bmatrix} \quad (4.29)$$

Taking the time step  $\Delta t$  as constant, we thus can solve for the quaternion propa-

gation as

$$\bar{\mathbf{q}}_{k+1} = \exp \left( \frac{1}{2} \begin{bmatrix} -[[\boldsymbol{\omega}_k]] & \boldsymbol{\omega}_k \\ -\boldsymbol{\omega}_k^T & 0 \end{bmatrix} \Delta t \right) \bar{\mathbf{q}}_k \quad (4.30)$$

Note that here, we have an explicit notation for propagation from one star camera solution  $(\bar{\mathbf{q}})_k$  to the subsequent star camera solution  $(\bar{\mathbf{q}})_{k+1}$ , given a time-interval  $\Delta t$  and 3-axis gyroscopic velocity measurements  $\boldsymbol{\omega}$ . The devil is in the details but we now have a clear formalism with which to begin our attitude reconstruction.

### 4.2.6 Kalman Filtering

Utilizing the rotation, propagation and optimal quaternion derivations above, we are now able to apply these to the problem of determining a time-ordered stream of attitude information. The canonical solution to this problem of linear is the *Kalman filter*[\[59\]](#).

The Kalman filter estimates a generalized state vector, given a time-ordered series of observations. These observations are used to correct state predictions made by the filter. In this manner, the filter creates a classic, predictor-corrector feedback.

While the Kalman filter has achieved substantial success in the field of attitude estimation, its shortcomings are readily apparent as the required system model becomes non-linear[\[79\]](#). For these cases, which in general are more common than the linear ones, the covariance matrices of both the prediction and correction cease to represent the true system noise. Without an accurate measurement of the covariances, the Kalman filter will converge to the solution which is best approximated by a gaussian distribution rather than the optimal solution.

In general, non-linear, non-Gaussian models present no closed-form solution to the estimation problem. Many extensions to the Kalman filter have been proposed to address this problem, including the Extended Kalman Filter, which models the first order correction to the non-linearities and the Unscented Kalman Filter, which models

probability distribution isocurvatures as they are transformed by the state model. We present an implementation of a more powerful methodology called a *Particle Filter* or *Sequential Monte Carlo Filter*.

### 4.2.7 Particle Filters

Particle filters implement numerical simulations to sequentially approximate an optimal solution to a state approximation problem. By multiply sampling the state space<sup>2</sup>, a particle filter obtains a finite number of potential solutions—particles—that may be processed through Bayesian inference to refine their approximations.

Bayesian inference is a statistical method of evaluating time-ordered data (evidence) that affect a hidden process (hypothesis) [40]. Particle filters are an application of Bayes' Theorem to sequential sampling, which in general states

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} \quad (4.31)$$

Where  $P(H|E)$ , the *posterior probability*, is the probability of the hypothesis (H), given all evidence (E) to this point.  $P(E|H)$  is the *conditional probability* of observing the evidence, given a specific hypothesis.  $P(H)$  is the *prior probability* of the hypothesis, before the more recent evidence was considered and  $P(E)$  is the *marginal probability*, that is the probability of observing the evidence under all hypotheses.

If we consider an unobservable, true system, called the process and denoted  $x_k$ , it will have an initial probability density function (PDF)  $p_{x_k=0}$ . This process PDF may then be time-evolved by a kernel  $p_{x_k|x_{k-1}}$ . This should be read as “the probability of finding the state  $x_k$  at time step  $k$ , given the state  $x_{k-1}$ , at time step  $k - 1$ .”

Similarly, we consider an observation of the true system denoted  $y_k$  with a conditional PDF  $p_{y_k|x_k}$ . In this formalism, the process denotes a hypothesis of the

---

<sup>2</sup>State space may be generalized to any number of variables. For our purposes, we limit state space to just the attitude of the gondola, a choice we will justify later in §4.2.8.1.

unobservable, true system and the inaccurate observation represents the evidence for that hypothesis.

We can then apply the Chapman-Kolmogorov equation to Eq. (4.31), eliminating our nuisance variable to obtain a recursive filter

$$p_{x_k|y_{1:k-1}}(\tilde{x}_k|\tilde{y}_{1:k-1}) = \int_{-\infty}^{\infty} p_{x_k|x_{k-1}}(\tilde{x}_k|\tilde{x}_{k-1}) \times p_{x_{k-1}|y_{1:k-1}}(\tilde{x}_{k-1}|\tilde{y}_{1:k-1}) d\tilde{x}_{k-1} \quad (4.32a)$$

$$p_{x_k|y_{1:k}}(\tilde{x}_k|\tilde{y}_{1:k}) = \frac{p_{y_k|x_k}(\tilde{y}_k|\tilde{x}_k) \cdot p_{x_k|y_{1:k-1}}(\tilde{x}_k|\tilde{y}_{1:k-1})}{\int_{-\infty}^{\infty} p_{y_k|x_k}(\tilde{y}_k|\tilde{x}_k) p_{x_k|y_{1:k-1}}(\tilde{x}_k|\tilde{y}_{1:k-1}) d\tilde{x}_k} \quad (4.32b)$$

Here, realizations of the process and observation models (as opposed to the models themselves) have been denoted by the  $\sim$  diacritic mark. While complete, this notation is tedious and leads to unnecessary confusion. For the remainder of this section, we will take the probabilities to be implied by the PDF space unless specifically noted. Thus  $p_{x_k|y_{1:k-1}}(\tilde{x}_k|\tilde{y}_{1:k-1}) \equiv p(x_k|y_{1:k-1})$ .

To connect this formalism to the problem we are addressing, the reader should consider the straightforward case where  $y_k$  represents a star camera vector measurement of the state (attitude) at time step  $k$ . Gyroscopic data, on the other hand, are not measurements for the purposes of the attitude state because they measure velocity, which is not a component of our state. Instead, they represent a transition kernel between time steps and the noise on the readout governs probability density. Thus, we might read  $p(x_k|y_{k-1})$  as “the probability of the gondola being in the true state  $x_k$  at timestep  $k$ , given a star camera measurement  $y_{k-1}$  at timestep  $k-1$  and a gyroscopic measurement from  $k-1$  to  $k$ .”

In all but the most trivial cases, these equations have no closed-form solution. Thus the marginal likelihoods given in Eq. (4.32) must be approximated. This is the heart of the sequential Monte Carlo method. We will sample from the probability

distribution at each step  $k$ , replacing the integrals with summations over the particles (samples) that most closely approximate the distribution.

In the limit where the number of particles becomes very large, the summation approximation approaches the the integral and we achieve the ideal filter. Clearly an infinite number of particles is infeasible and thus choosing which particles to propagate will form the heart of our approximation function.

To begin this, we take a function  $f$  which is integrable with respect to  $p(x_{0:k}|y_{1:k})$ . Then, assume that we are able to sample from that function a number  $N$  of independent, equally weighted particles. By the strong law of large numbers[57], we have that

$$\frac{1}{N} \sum_{i=1}^N f(\tilde{x}_k^{(i)}) \rightarrow E[f(x_{0:k}) | \tilde{y}_{1:k}] \quad (4.33)$$

Where  $E[ ]$  denotes the expectation value and  $\rightarrow$  should be read as “almost certainly converges to”. Note that we include the tilde distinction of observation and process realization on left hand side of this equation for the process and on the right hand side for observation. So this states that the convergence of the realized state will be to the expectation of true state, given the realized observations. We will use parenthetical superscripts to denote particle number where needed while subscripts will continue to represent the fixed time step.

Unfortunately, efficiently sampling directly from the posterior distribution is, in almost all cases, impossible. However, given that our state’s posterior distribution is well-localized, the vast majority of particles in an even sampling would carry no information to characterize the distribution. Thus, as an optimization, we may adopt a measure of importance to guide which points of the posterior distribution are most important to approximate the full distribution.

This corresponds to creating an arbitrary “importance distribution”, that we denote  $\pi$ , that supports the PDF. In order to account for the discrepancy between

$p(x)$  and  $\pi(x)$ , we weight each sample by a set of normalized importance weights  $w$  that satisfy  $w^{(i)} \propto p(x^{(i)}) / \pi(x^{(i)})$  and  $\sum_i^N w^{(i)} = 1$ .

Then, we can approximate the expectation value of Eq. (4.33) using the normalized weights  $\tilde{w}$  as

$$\sum_{i=1}^N \tilde{w}_k^{(i)} f(\tilde{x}_k^{(i)}) \rightarrow E[f(x_{0:k}) | \tilde{y}_{1:k}] \quad (4.34)$$

This leaves only determining the appropriate representation of our importance distribution. While many alternative representations for the importance distribution have been proposed [30, 40, 58], most focus (justifiably) on specific system characteristics that do not apply generally to our model. Instead, we will assume a simpler, recursive form

$$\pi(x_{0:k+1} | y_{1:k+1}) = \pi(x_{0:k} | y_{0:k}) p(x_{k+1} | y_k) \quad (4.35)$$

In other words, the importance function at time step  $k + 1$  is the previous step's importance function, updated by the state transition kernel, which we recall as the quaternion propagation of Eq. (4.30) utilizing our three-axis gyroscopes and assuming a noise on the readout.

We will now proceed to describe our method of filtering, utilizing this Bayesian framework.

### 4.2.8 Sequential Importance Resampling

Our method of filtering is termed in the literature *Sequential Importance Resampling* (SIR)[44] where it is an extension to previous work on *Sequential Importance Sampling*[50]. The principle contrast to SIS is the attempt to limit degeneracies inherent in a self-contained weighting. By degeneracies, we refer to higher weight particles

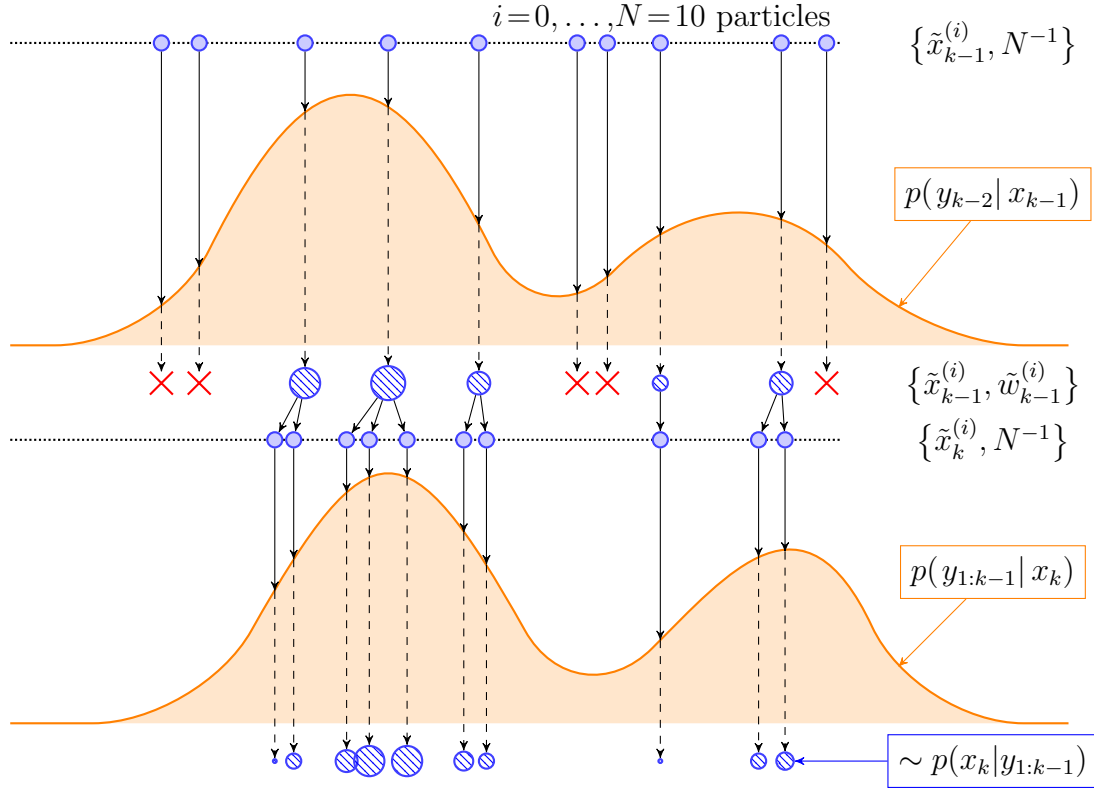


Figure 4.2: Sequential Importance Resampling. Here the filter starts at timestep  $k-1$  with all 10 particles equally weighted. Each particle's weight is computed by the information available at the timestep in the posterior probability. The weights in this diagram are denoted by the size of the blue dots. Next, resampling is performed by multiplying high weight particles and distributing according to their kernel and bandwidth. The resulting particles have uniform weight and approximate  $p(x_k | y_{1:k-1})$ .

propagating to the posterior distribution of the proceeding step with higher weighting, leading to the distribution collapsing to a single, most-likely value in absence of additional measurements.

To alleviate this problem, SIR introduces an additional step into our particle filter algorithm called resampling. Practically, this means that we eliminate particles with weights below a certain threshold and multiply particles with higher weights. After resampling, each particle in  $\{\tilde{x}\}$  has equal weight  $N^{-1}$ . This is schematically shown in Fig. (4.2).

This requires a delicate balance, however. If we resample every step, the filter



converges slowly as the particles lose part of the historical path, namely the weighting accumulated from multiple steps. If, however, we resample too slowly, the filter diverges as all of the weighting is consolidated in few particles and we do not accurately map the posterior PDF.

Kong et al. [63] devised a criterion for balancing these extremes by utilizing an “effective” number of particles  $N_{\text{eff}}$  based on the variance of the particle weights as

$$N_{\text{eff}} = \left( \sum_{i=1}^N w_k^{(i)2} \right)^{-1} \quad (4.36)$$

Then, a threshold value  $N_{\text{th}}$  is used for comparison, where resampling occurs only when  $N_{\text{eff}} < N_{\text{th}}$ . Determining the appropriate value for  $N_{\text{th}}$  should be done with the model distribution in mind. Below this value, the particle cloud will no longer represent the distribution accurately. Thus it is appropriate to consider  $N_{\text{th}}$  as the minimum representation number, given both expected distributions and the state space over which sampling occurs. This is distinct from the optimal value for  $N$ , the total number of particles, which should be chosen to be as large as tolerable as can be seen from Eq. (4.34).

The basic implementation of EBEX’s particle filter that incorporates sequential importance resampling is detailed in Alg. (A.5) and we present the overview as follows:

1. Draw a population of  $N$  particles from the initial state probability density function  $p(x_0|y_0)$ .
2. Increment the time step  $k \rightarrow k + 1$ .
3. For each particle, sample one outcome from that particle’s prior  $p(\tilde{x}_{0:k-1}^{(i)})$  distribution, time-evolved by the process kernel  $p(x_k|\tilde{x}_{0:k-1}^{(i)})$
4. Concatenate the realization for step 3 with the prior step realizations to create a full and unique state realization history to timestep  $k$  for each particle.

5. Calculate the posterior probability for each particle, utilizing Eq. (4.32), Eq. (4.34) and Eq. (4.35).
6. Determine the particle weight by multiplying the previous particle weight by the posterior probability.
7. Output the optimal state solution by convolving the weighted particle cloud with an Epanechnikov kernel (detailed in Eq. (4.37)) and extracting the mode probability.
8. Calculate the effective number of particles  $N_{\text{eff}}$  as given by Eq. (4.36). If this number is less than our threshold number, resample the particle cloud.
9. Iterate back to step 2.

The critical point in this process is the implementation of the resampling step. Some previous work [30] has found improvement in the resampling problem by merely injecting a constant level of noise into the new particles. While appealing in its simplicity, this approach suffers from the problem of discretization of the distribution, losing information about the past trajectories, which contaminates the posterior distribution and ultimately degrades the filter performance.

It is preferable to regularize the distribution of particles as this maintains as much information as is available from the particle trajectories to the current point. We do this first by approximating the continuous distribution over the full variate space. To transition from discrete to continuous distribution, we convolve each particle and its associated weight with a kernel to generate a distribution over which to sample. Epanechnikov [33] derived an optimal<sup>3</sup> kernel  $\mathcal{K}_{\text{opt}}$  for multivariate distributions and

---

<sup>3</sup>In the sense of minimizing the relative global error of the kernel

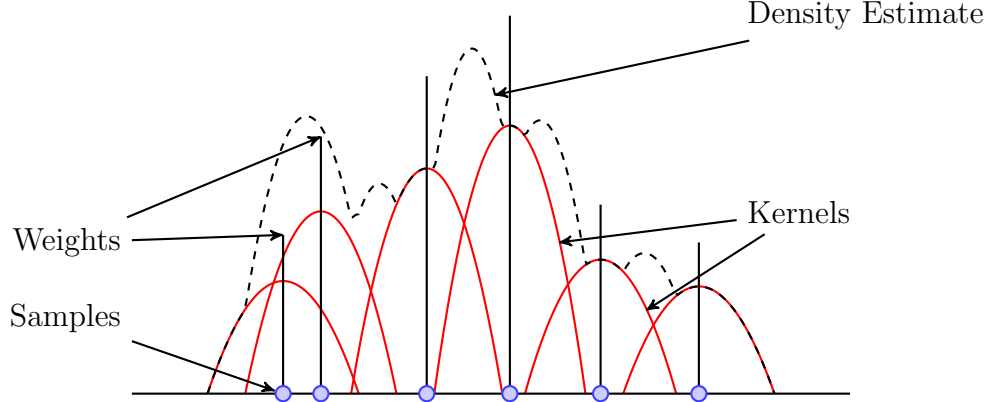


Figure 4.3: Particle regularization schema. On the x-axis, we show an arbitrary 1-dimensional cut on the particle cloud values. On the y-axis is the weighting of each particle. The Density Estimate is a convolution of the kernels and is used to re-distribute the particle values such that their weights are equal and the Density Estimate remains invariant.

this is the one we use for EBEX.

$$\mathcal{K}_{\text{opt}}(\mathbf{x}) = \begin{cases} \frac{n_x+2}{2c_{n_x}}(1 - |\mathbf{x}|^2) & \text{if } |\mathbf{x}|^2 < 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.37)$$

Here  $n_x$  is the dimensionality of sample (4 in our case using quaternions) and  $c_{n_x}$  is the volume of a unit  $n_x$ -sphere. The result of convolving the Epanechnikov kernel with our discrete distribution is a probability density function (PDF) from which we can sample to re-populate our particle cloud during regularization. The result of this sampling process is given schematically in Fig. (4.3).

Note that, despite beginning with a discrete sampling, we represent our state space continuously for the purposes of resampling. By drawing our samples from this continuous distribution, we effectively dither the probability steps between adjacent particles to re-align the importance function  $\pi(\mathbf{x})$  with our continuous distribution. Algorithmically, this is detailed in Alg. (A.6). We provide an overview of the steps here:

1. Calculate the probability distribution function for the Epanechnikov kernel  $\mathcal{K}$ .

2. Sample a value from the Epanechnikov PDF. This represents a new particle state realization  $x$ .
3. Calculate the posterior probability of the new value, using the Epanechnikov PDF as the marginal probability. This value is assigned as the weight  $w^{(i)}$  of the new particle.
4. Iterate to step 2 until the sum of the weights of the newly generated particles is equal to or greater than unity.
5. Normalize the weights such that they sum to 1.

#### 4.2.8.1 Joint Filtering

Correctly reconstructing the state requires an assumption of correctness in the measurements. This assumption is limited by our parameterization of the sensors used to make the measurement. In their most basic form for attitude estimation, the parameters control how we translate a sensor measurement into a prediction of gondola motion or attitude. Our parameterization will be discussed in detail in §4.3 but as a useful example, the reader may envision a fixed rotation as a parameter of the system. This rotation  $\mathbf{R}$ , when applied to a vector measurement returned by the star camera, will rotate that measurement so that its coordinate frame is co-aligned with the coordinate frame of a 3-axis gyroscope box. We could then apply the gyroscope rotational velocity measurements to propagate the rotated vector forward in time.

The process of optimizing parameters proceeds in parallel with the attitude estimation as the attitude calculation will inform our judgement of the validity of the parameters while the parameters themselves control the expression of  $p(x_{k+1}|x_k)$ , our state transition kernel. This recursive property is a common feature in many modern filtering applications and our solution, while specific to the use case of EBEX, nevertheless is broadly applicable.

Approaches documented in the literature broadly break into two categories: augmented state estimation (cf. Pitt and Shephard [93], Andrieu et al. [5], and Campillo and Rossi [16]) and dual-track feedback (cf. Mariani et al. [76] and Rodriguez-Losada et al. [109]). In the case of augmentation, the state vector is expanded to include the parameter estimation as well. With dual-track filters, parameter estimation lags the filter by one or more timesteps as the measurement of the state at time  $k$  uses parameter estimation from  $k - 1$ .

The augmented state estimation approach requires efficiently sampling the  $(4 + n)$  dimensional space of our quaternion and  $n$  parameters. The computational difficulty of this approach rises as  $\mathcal{O}(N^{4+n})$  where  $N$  is the number of sample points per dimension. Even in the constrained EBEX optimization, we have 12 parameters (given in Tab. (4.1)) in addition to our state quaternion. Practical considerations of computer memory and disk space would limit our ability to sample this space to fewer than 8 particles per dimension<sup>4</sup>.

We therefore concentrate on the dual estimation approach. In this approach, the complexity of our particle filter is  $\mathcal{O}(N^4)$  with an additive contribution from the complexity of our parameter estimation routines. This effectively replaces  $p(\bar{\mathbf{q}}_k, \boldsymbol{\eta}_k | y_k)$  with  $p(\bar{\mathbf{q}}_k | \boldsymbol{\eta}_k, y_k)$ , breaking the single, intractable problem into two smaller, soluble ones.

The particle filter processes the low-dimensional problem of state-estimation, while a second filter processes the higher-dimensional problem of parameter estimation. Previous dual-track estimation filters have required the parameter estimation to lag the state estimation by one or more measurements[27, 47, 76, 109, 133]. This has primarily been driven by the need to optimize both parameters and state using the same measurement, leading to systemic feedback if not staggered[23].

This work avoids the feedback problem by devising a new metric for the parameter

---

<sup>4</sup>Assume that we can represent a parameter as an 8-byte floating point value, then using 2TB ( $2E12$ ) for storage would yield  $2.75E11$  values. Thus particles per dimension is  $e^{\ln(2.75E11)/(4+3)} \approx 7.6$ .

optimization that is independent of the state measurements. This approach leverages the system constraints to make predictions about the noise residual properties. In this way, we effectively generate a basis, in which the state transition function is sparse, allowing us to optimize parameterizations independently of absolute measurements. That is to say, we know what  $p(x_k|x_{k-1})$  should look like over multiple measurements. If we improperly characterize either the gyroscope parameters or the state itself, given some non-zero motion, it skews the distribution function as will be shown for the EBEX gyroscopes in Fig. (4.11a).

#### 4.2.8.2 Parameter Estimation

The properties required of the parameter estimation filter are the following:

1. Generic in the sense of allowing discontinuous cost gradients. Costs are calculated by integration of the attitude solution, creating feedback with the parameter space and introducing potentially discontinuous functional forms. This ruled out the use of a large class of filters that depend for their function on computing the Jacobian of explored space.
2. Expandable optimization goal counts. This requirement is set as EBEX has between 6 (modeling only star camera rotation and gyroscope bias) and 38 (including input from coarse sensors detailed in §3.3.2) individual parameters that may be required for optimization, depending on the system characteristics and number of sensors used in the estimation.
3. Robust against local error minima. Given the feedback between the state and parameter estimation, local minima will be present at multiple points, thus the filter should be strongly biased against local minima.

The algorithm used for EBEX parameter estimation is a modified version of Differential Evolution[123], a heuristic-based optimization method that satisfies all three requirements. It utilizes genetic algorithm concepts of a population of multiple

solutions, each with a determined fitness metric whose parameters may be combined and altered to improve the global solution set.

The basic algorithm begins by assuming a population of size  $NP$  each with a potential parameter set of length  $D$ . The optimization then proceeds as follows:

1. Evaluate all members of the population for their fitness.
2. Create a new member of the population by combining 2 or more members of the original population such that  $v_{i+NP} = v_j + F(v_k - v_l)$ . Here the sub-indices  $j, k$  and  $l$  may represent any (not necessarily unique) members of the original population while  $i \in \{1 \dots x\}$  with  $x$  representing the number of new trial vectors to create.
3. Salt the diversity of the resulting vector pool by exchanging individual parameters between members of the population according to a probabilistic selection criterion.
4. Evaluate the fitness of the newly created population members and cull the full population back to a steady-state population size ( $NP$ ). **Storn and Price** achieve this by comparing the fitness of their newly created  $v_{i+NP}$  with that of  $v_j$  and keeping the most fit.
5. Iterate to step 2 unless the global stopping condition has been met.

The EBEX implementation follows much of the basic outline above with notable departures. First, in step 3, EBEX utilizes groupings to exchange parameters. We find improvement in our convergence speed by limiting parameter exchanges whose effects are interrelated. Next, in step 4, we evaluate the fitness of an solution relative to the full population, maintaining only the globally fittest  $NP$  solutions.

Finally, we introduce an additional step of *oppositional evolution*. This is based on work by Rahnamayan et al. [101] and Wang et al. [134]. In this step, we compute the *opposite number* ( $x_{\text{opp}}$ ) for each parameter, where  $x_{\text{opp}} = x_{\text{min}} + x_{\text{max}} - x$  for scalar

valued parameters. Here,  $x_{\min}$  and  $x_{\max}$  are the minimum and maximum values for the parameter  $x$  over the full population. For quaternion parameters, we choose a similar method, selecting  $\langle \bar{\mathbf{q}} \rangle$ , the mean-valued quaternion as the approximate center of the distribution. Then  $\delta \bar{\mathbf{q}} = \langle \bar{\mathbf{q}} \rangle \otimes \bar{\mathbf{q}}^{-1}$  is the error quaternion that rotates  $\bar{\mathbf{q}}$  into  $\langle \bar{\mathbf{q}} \rangle$ . Thus  $\bar{\mathbf{q}}_{opp} = (\bar{\mathbf{q}} \otimes \delta \bar{\mathbf{q}}) \otimes \delta \bar{\mathbf{q}}$ . The set of opposite parameters are then evaluated for fitness against the global population, again with only the  $NP$  fittest members remaining.

Critical to the success of any optimization method is the selection of the fitness evaluation. In the next section, we will describe the parameterization of the primary EBEX pointing sensors and demonstrate the performance of our fitness function.

### 4.3 EBEX Sensor Parameterization

In this work, we focus on parameterizing two types of sensors: star cameras and gyroscopes. We will show in §4.3.2 that the star camera internal parameters such as platescale and lens deformation are constrained by reference to a catalog of known star locations. The remaining external parameter is a coordinate frame rotation  $T_{ref,sc}$  that rotates a given star camera solution into the shared reference frame.

Similarly, in 4.3.1, we separate the gyroscope parameters into an external rotation from the gyroscope box frame to the reference frame ( $T_{ref,gy}$ ) and internal parameters characterizing deviations from an idealized readout.

#### 4.3.1 Gyroscope Readouts

Parameterizing the gyroscopes' readouts requires constraining two types of effects that alter the measurement returned by the sensor: physical and electrical. In the case of an electrical parameter, this will refer to the non-uniform response of the gyroscope to physical stimuli. These can be modeled to first order by use of a scale factor ( $\mathbf{\Lambda}$ ) and offset ( $\mathbf{\beta}$ ) between gyroscope measurement and true rotational speed.



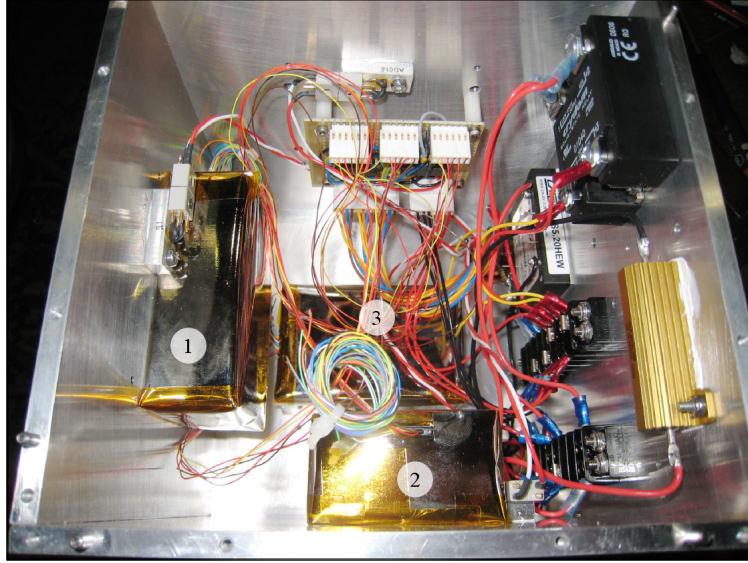


Figure 4.4: The internals of the 3-axis gyroscope box are shown. Gyroscopes (labeled 1, 2 and 3) are mounted on box sides that are roughly orthogonal to each other. Non-orthogonality enters due to uncertainty in the machining of the box walls and thickness variation in the magnetic shielding (visible as the gray metallic wrap) and electrical isolation (visible as the orange Kapton tape) that separate the gyroscope from the box walls. Figure from [103].

The physical parameters specify the orientation of each fiber optic gyroscope, relative to the gondola reference frame. Nominally, this corresponds to 2 rotation parameters per gyroscope<sup>5</sup>. The gyroscopes are mounted together in a shared box as shown in 4.4.

Due to the shared alignment of a roughly orthogonal coordinate frame in the form of the gyroscope box's internal walls, we find it computationally efficient to split the 6 gyroscope rotation parameters into two classes of rotations: a large coordinate frame rotation ( $T_{ref,g}$ ) representing the box and small misalignments ( $\Delta$ ) between that coordinate frame and the true orientation of the three individual gyroscopes.

Additionally, each gyroscope readout has an additive noise contribution ( $\nu$ ) that

---

<sup>5</sup>Gyroscopes measure rotation about their  $\hat{z}$  axis and so are not affected by a rotation about it

is roughly gaussian. The final measurement model used by EBEX is

$$\boldsymbol{\omega}_{meas} - \boldsymbol{\beta} - \boldsymbol{\nu} = (I - \boldsymbol{\Lambda}) \underbrace{(I - \boldsymbol{\Delta}) T_{g,ref}}_{\boldsymbol{M}} \boldsymbol{\omega}_{true} \quad (4.38)$$

Here  $\boldsymbol{\beta}$  is a vector of measurement offsets,  $\boldsymbol{\nu}$  is the gaussian noise contribution,  $\boldsymbol{\Lambda} = \text{diag}[\lambda_x, \lambda_y, \lambda_z]$  is a matrix of linear scale factors and  $\boldsymbol{\Delta}$  is the matrix of small misalignments.  $\boldsymbol{M}$  is the combination of all physical parameters into a single transformation matrix that we will reference later. To calculate  $\boldsymbol{\omega}_{true}$ , our rotation speed in the reference frame, we gather terms to the measurement side of the equation as

$$\boldsymbol{\omega}_{true} = \underbrace{T_{ref,gy} (I - \boldsymbol{\Delta})^{-1}}_{\boldsymbol{M}^{-1}} (I - \boldsymbol{\Lambda})^{-1} (\boldsymbol{\omega}_{meas} - \boldsymbol{\beta} - \boldsymbol{\nu}) \quad (4.39)$$

We now note that  $\boldsymbol{\Delta}$  is a *misalignment* matrix. That is, it consists only of terms that deviate from a coordinate frame rotation. We know from linear algebra that we may separate any matrix into a unitary component and an upper-triangular component by QR decomposition[122]. In our case,  $\boldsymbol{M}$  is a real, square matrix and thus  $Q$ , in the QR decomposition, will be an orthogonal matrix. Orthogonal matrices represent a coordinate frame, which is what we call  $T_{ref,gy}$ . We may therefore represent  $\boldsymbol{\Delta}$  as an upper triangular matrix.

$$\boldsymbol{\Delta} \equiv \begin{bmatrix} 0 & \delta_{yz} & \delta_{zy} \\ 0 & 0 & \delta_{zx} \\ 0 & 0 & 0 \end{bmatrix} \quad (4.40)$$

Now, we can expand Eq. (4.39), by calculating the matrix inverses as

$$(I - \mathbf{\Delta})^{-1} = \begin{bmatrix} 1 & -\delta_{yx} & -\delta_{zx} \\ 0 & 1 & -\delta_{zy} \\ 0 & 0 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & \delta_{yx} & \delta_{zx} + \delta_{yx}\delta_{zy} \\ 0 & 1 & \delta_{zy} \\ 0 & 0 & 1 \end{bmatrix} \quad (4.41)$$

and

$$\begin{aligned} (I - \mathbf{\Lambda})^{-1} &= \begin{bmatrix} 1 - \lambda_x & 0 & 0 \\ 0 & 1 - \lambda_y & 0 \\ 0 & 0 & 1 - \lambda_z \end{bmatrix}^{-1} \\ &= \frac{1}{\det(I - \mathbf{\Lambda})} \begin{bmatrix} 1 - \lambda_y - \lambda_z + \lambda_y\lambda_z & 0 & 0 \\ 0 & 1 - \lambda_x - \lambda_z + \lambda_x\lambda_z & 0 \\ 0 & 0 & 1 - \lambda_x - \lambda_y + \lambda_x\lambda_y \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{1 - \lambda_x} & 0 & 0 \\ 0 & \frac{1}{1 - \lambda_y} & 0 \\ 0 & 0 & \frac{1}{1 - \lambda_z} \end{bmatrix} \end{aligned} \quad (4.42)$$

Here, making only the non-singularity assumption of  $\delta_{yx}\delta_{zy} + \delta_{zx} \neq 0$ , our equation for the full expression of Eq. (4.39) is

$$\boldsymbol{\omega}_{true} = T_{ref,g} \begin{bmatrix} 1 & \delta_{yx} & \delta_{zx} + \delta_{yx}\delta_{zy} \\ 0 & 1 & \delta_{zy} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{1 - \lambda_x} & 0 & 0 \\ 0 & \frac{1}{1 - \lambda_y} & 0 \\ 0 & 0 & \frac{1}{1 - \lambda_z} \end{bmatrix} (\boldsymbol{\omega}_{meas} - \boldsymbol{\beta} - \boldsymbol{\nu}) \quad (4.43)$$

### 4.3.2 Star Camera Parameterization

The EBEX Star Cameras' external parameters in our attitude solution are represented by a single rotation into the reference frame. However, we must first constrain the

internal parameters to determine the attitude of the star field imaged by the star camera. This process is performed by EBEX in many different ways. This work will deal solely with a single method, developed after the North American flight, for dealing with low signal-to-noise images.

It is important to note that the level of processing described in this thesis may not be used in the final EBEX analysis of LDB flight data. Star camera images taken during the long duration flight show a robust performance and high signal to noise due in large measure to the comprehensive work and extensive improvements made to the star camera acquisition system that will be described in a future thesis<sup>6</sup>. This performance may render the additional processing work described below unnecessary for the current EBEX science data. Nevertheless, as a novel approach to improving attitude solutions generated from astronomical images, this work was an important piece of the attitude solution from EBEX's North American flight and may find future application in similar, low signal to noise environments.

To determine the attitude of a star camera image, the problem is divided into two steps: source extraction and field solving.

#### 4.3.2.1 Source Extraction

The problem of source extraction is to determine the potential astrophysical light sources recorded in an image and accurately locate their centers in pixel coordinates. The standard, coarse formula for this is well known[3]

$$\begin{bmatrix} C_x \\ C_y \end{bmatrix} = \begin{bmatrix} \sum_i \frac{i \cdot P_{i,y}}{\sum_{i,j} P_{i,j}} \\ \sum_j \frac{j \cdot P_{x,j}}{\sum_{i,j} P_{i,j}} \end{bmatrix} \quad (4.44)$$

where  $P_{i,j}$  is the pixel value of the  $i, j$ th pixel and the summation occurs over a fixed window. While accurate, this formulation allows for error accumulation resulting

---

<sup>6</sup>D. Chapman, Personal communication, 2013

from lower signal to noise as well as signal truncation. We chose to address the issues sequentially by applying a noise-knowledge transformation to the full image and then re-evaluating the centroid window based on a gaussian point source distribution.

The noise-knowledge transformation exploits the fact that for a given pixel  $p_{x,y}$ , absent signal, the adjacent pixels  $p_{i,j} | \{i,j\} \in [\{x-1, y-1\} \dots \{x+1, y+1\}]$  are equally likely to encode higher values as lower values. Thus averaging over the gradient vectors relative to a central pixel, we expect to encode 0. Contrariwise, given a centralized source, the gradient vectors will average to a coherent value under a symmetric transformation [104].

To define more clearly the action of this filter, consider a point  $p_{i,j}$  and define the gradient of pixel values as

$$\nabla_{p_{i,j}} = \frac{\partial}{\partial x} \frac{\partial p_{i,j}}{\partial y} \quad (4.45)$$

decomposing the resultant vector into radial coordinates, we have

$$r_{i,j} = \ln (1 + \|\nabla_{p_{i,j}}\|) \quad (4.46)$$

$$\theta_{i,j} = \arctan \left( \frac{\partial p_{i,j}}{\partial x} / \frac{\partial p_{i,j}}{\partial y} \right) \quad (4.47)$$

where the radius  $r_{i,j}$  denotes the strength of the gradient while the phase  $\theta_{i,j}$  is the counter-clockwise angle, relative to the x-axis.

Next, we note that symmetry dictates we choose a pair of pixels  $p_a$  and  $p_b$  that are both equidistant from the pixel of interest and arranged such that the line between them bisects the pixel of interest. This line makes an angle with the x-axis, denoted  $\psi_{ab}$ . Our symmetric weighting function considers two primary factors: phase and distance.

As  $p_a$  and  $p_b$  bracket the pixel of interest, the phase weight of the pair should be highest when the two are  $180^\circ$  out of phase. Thus, the primary phase weight should be  $[1 - \cos(\theta_a - \theta_b)]/2$ . However, this neglects the relative phase of the line

connecting the pair. Here, we wish to weight the vectors highest when the sum of their angles, relative to the connecting line is also  $180^\circ$  out of phase. Thus, the second component in the weighting should be  $[1 - \cos((\theta_a - \psi_{ab}) + (\theta_b - \psi_{ab}))]/2$  and the full phase weighting is thus

$$W_\phi(a, b) = \frac{1}{4} [1 - \cos(\theta_a - \theta_b)] \cdot [1 - \cos((\theta_a - \psi_{ab}) + (\theta_b - \psi_{ab}))] \quad (4.48)$$

$$= \frac{1}{4} [\cos(\theta_a - \psi_{ab}) - \cos(\theta_b - \psi_{ab})]^2 \quad (4.49)$$

Next, we consider the distance weighting  $W_d$ . We assume that our lens' point spread function is roughly gaussian, which is justified to first order, and use a simple gaussian distribution to weight the symmetric vector contribution as

$$W_d(a, b, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{|p_a - p_b|}{2\sigma}\right) \quad (4.50)$$

Finally, we wish to weight highly those points that are symmetric in many axes, rather than merely at a single angle. To do this, we first recognize that star images may be blurred across multiple pixels as the gondola will exhibit non-zero motion during image acquisition. To the distance weighting, this will have only a marginal effect as the distribution will still exhibit gaussianity but with larger  $\sigma$  along the axis of movement.

For the purposes of weighting multi-axis symmetry, however, we must consider that the symmetric orientation is no longer  $(\theta_a + \theta_b)/2$ . Rather, this elongation of the distribution represents an affine transformation, resulting in a skewed symmetry[138]. Under affine transformation,  $p_{i,j}$ , the middle pixel, will remain invariant while the axis of symmetry becomes

$$\vartheta_{ab} = \pi - \arctan \frac{2 \cos(\theta_a - \psi_{ab}) \cos(\theta_b - \psi_{ab})}{\sin(\theta_a + \theta_b - \psi_{ab})} \quad (4.51)$$

Defining  $\Theta \equiv \vartheta_{ab}$  for the  $ab$  pair that maximizes  $W_d(a, b, \sigma)W_\phi(a, b)r_ar_b$ , we can

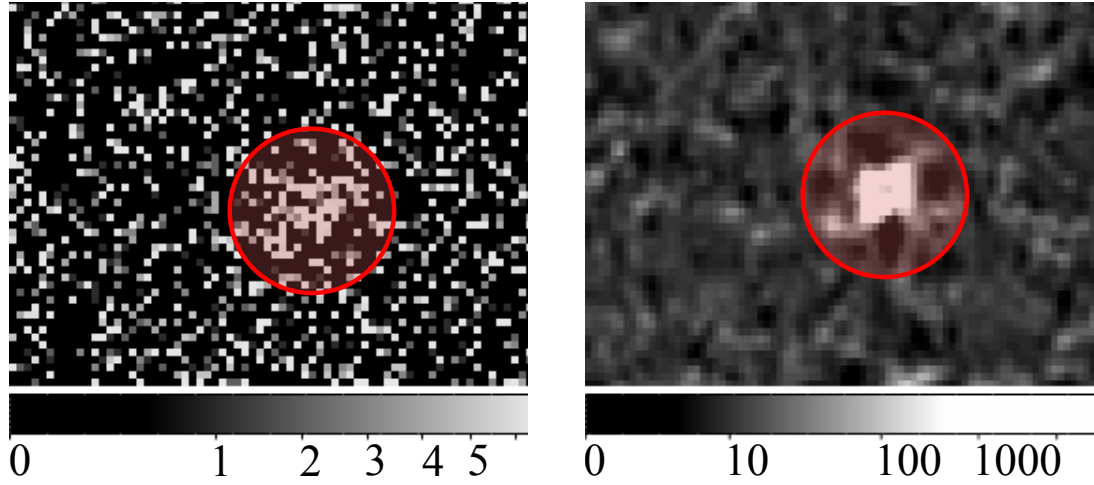


Figure 4.5: The application of symmetric weighting to a sub-region of a full star camera image, taken while partially defocussed in flight, shows the successful distinction of a low-signal star from background noise of the same order. The recovered image shows high signal-to-noise with  $\sigma = 4\text{px}$ .

express the radial symmetry weight as maximizing the contribution from off-axis components as

$$W_r(a, b) = \sin^2(\vartheta_{ab} - \Theta) \quad (4.52)$$

The final symmetric transformation function for each pixel is therefore

$$P(\sigma) = \sum_{a,b} W_d(a, b, \sigma) W_\phi(a, b) W_r(a, b) r_a r_b \quad (4.53)$$

where  $a, b$  may be allowed to vary over the full image but in practice will be limited by the choice of  $\sigma$ , which governs our distance weighting.

Note that this transform will produce values that are both positive and negative, corresponding to symmetries that point away from a central point and those that point toward a central point respectively. Since we are only interested in those symmetries that point inward (corresponding to brighter central regions), we simply truncate the negative values at 0. The result of this process is shown in Fig. (4.5) for the region about a single star.

With the application of our symmetry weighting, we then attempt to locate the

center of the star's image within the pixel grouping. Our first pass locates all pixels  $2\sigma$  above the local noise floor, where the local noise is calculated in  $6 \times 6$  pixel blocks about the pixel of interest.

Once candidate pixels have been identified, we utilize Eq. (4.44) to generate an initial guess of the star centroid. We then generate an idealized response to a star at the initial guess location. This is calculated using a circularly symmetric Gaussian normalized to unity

$$I(x, y) = \exp \left[ -\frac{(x - x_0)^2 (y - y_0)}{2\sigma^2} \right] \quad (4.54)$$

We integrate this function over the area of each pixel, thus the expected value is

$$D_a = \int_{x_{min}}^{x_{max}} \int_{y_{min}}^{y_{max}} I(x, y) \, dx dy \quad (4.55)$$

$$= C \cdot \left[ \operatorname{erf} \left( \frac{x_{max} - x_0}{\sqrt{2}\sigma_x} \right) - \operatorname{erf} \left( \frac{x_{min} - x_0}{\sqrt{2}\sigma_x} \right) \right] \cdot \left[ \operatorname{erf} \left( \frac{y_{max} - y_0}{\sqrt{2}\sigma_y} \right) - \operatorname{erf} \left( \frac{y_{min} - y_0}{\sqrt{2}\sigma_y} \right) \right] \quad (4.56)$$

$D_a$  is calculated for for each pixel and the field is then convolved with Eq. (4.53) to determine the idealized response  $D'_a$  to this realization under symmetry transformation. The goodness of fit estimator is given by

$$\chi^2 = \sum_a W_a \cdot (D'_a - P_a)^2 \quad (4.57)$$

where the weighting function  $W_a$  may encode information such as flat fielding, degraded pixels or photon statistics. The positioning is iterated to maximize the goodness of fit using the Nedler-Mead Simplex algorithm[85].



### 4.3.3 Field Matching

Following the determination of pixel centroid locations, we must now determine which, if any, of the pixel centroids correspond to actual stars as opposed to noise, cosmic ray hits or satellite reflections. There are many existing methods for determining star field correlations[120].

For this work, we have adapted work by Samaan et al. [114] and Lang et al. [66] to create a field matching algorithm that is both computationally efficient and robust against image noise, star occlusion and non-astronomical light sources.

**Field Code Creation:** To align the identified point sources with a reference catalog, we need a uniform and unique metric that will represent relative locations of multiple point sources.

In our implementation, four point sources are selected for a given field. The line between the two most distant stars is chosen as the diameter of a circle inside of which the remaining two stars lie. Then, the angles formed by the legs from an exterior star to the two interior stars are calculated and their tangents are used as 42 bits of a 64-bit hash value. Note that our angular range is limited from 90-180 degrees. The use of the tangent ensures that the sampling remains linear in cartesian coordinates and we select the smaller of the two possible angles formed to ensure the range limitation.

The quadrilateral described by the four stars requires one additional value to constrain the relative positions up to a symmetry. For this we select the distance

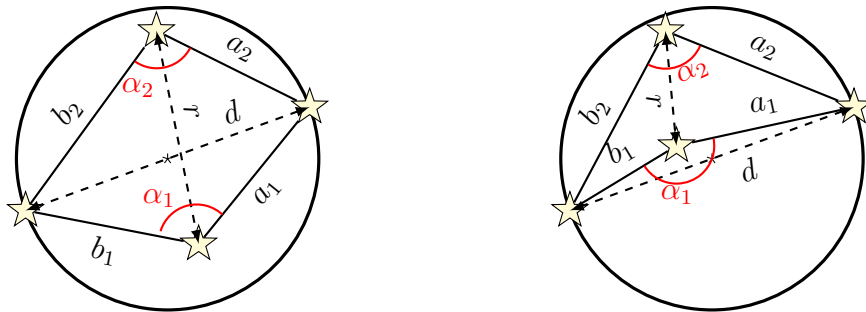


Figure 4.6: Relevant data for matching two examples of a 4-star pattern. The distance  $r$  is normalized to the circle's diameter  $d$  before being stored.

63	62	61	60	59	58	57	56	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	32
p	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r	1	2	r

Figure 4.7: The distribution of data values inside of a 64-bit star field code hash. The most significant bit (p) is the parity of the two triangles inside of the circle. Then, alternating bits from the larger-valued angle (1), the smaller-valued angle (2) and the normalized distance (r) between the two interior stars.

between the two interior stars, normalized such that the distance between the two exterior stars is equal to one. This value is recorded in an additional 21 bits.

The three constraining values are then given by

$$\tan \alpha_1 = \frac{1}{a_1^2 + b_1^2 - d^2} \sqrt{4a_1^2 b_1^2 - (a_1^2 + b_1^2 - d^2)^2} \quad (4.58)$$

$$\tan \alpha_2 = \frac{1}{a_2^2 + b_2^2 - d^2} \sqrt{4a_2^2 b_2^2 - (a_2^2 + b_2^2 - d^2)^2} \quad (4.59)$$

$$r' = \frac{r}{d} \quad (4.60)$$

where the angles  $\alpha_1$  and  $\alpha_2$  are independent of the overall scaling factor, thus we do not show the normalization of the legs  $a$  and  $b$ . The distance between interior stars  $r$  must, however be normalized to the exterior distance in order to remain scale invariant.

This bit structure now utilizes 63 of the 64 bits in our hash. While this is sufficient, we can achieve a substantial implementation speed increase by utilizing the last bit as a gross discrimination factor. Since interior stars may lie in the same half of the the circle defined by the exterior stars or in opposite halves, we encode this parity information in the most significant bit in the hash.

Next, we combine our three data into the remaining 63 bits of the hash value. To do this, we recognize that an efficient hashing comparison function will use only single comparison operators to determine positioning and sorting. Thus we wish to have bits with similar significance co-located in the hash value as shown in Fig. (4.7).

By reducing the star-matching to a scale- and rotation-invariant set of parameters, we are rapidly able to locate candidate matches in a pre-computed hash table. Additionally, by mixing the values of the three parameters on the bit-level, we may efficiently search a sorted structure to arbitrary precision by stepping forward and backward from the closest match.

### 4.3.3.1 Star Index

We are able to gain substantial speed in this implementation by shifting the coarse alignment to a pre-use index, calculated against previously observed star positions. For this, we utilized the Sloan Digital Sky Survey, DR7's  $r$ -band data[1], which roughly corresponds to the peak efficiency of our star camera's CCD imager at 616.5 nanometers.

To create the index, we first divide the sub-divide the sky into equal-area segments using the Hierarchical Equal Area iso-Latitude Pixelization (HEALPix) scheme[46]. HEALPix subdivides a sphere into a number of equal-area, projected quadrilaterals where the resolution is determined by the number of subdivisions (called  $N_{side}$ ) of the base, 12-segment pixelization required to resolve the desired feature. As such,  $N_{side}$  is always a power of 2 and the number of pixels is  $N_{pix} = 12 \times N_{side}^2$ .

Inside of each HEALPix pixel, we select the brightest star as measured by SDSS's  $r$ -band intensity. Then, for each pixel, we select the 14 pair stars that are 2 HEALPix pixels away and calculate any quadrilaterals formed within the circle diameter.

We form one index for each pixelization size. The minimum pixelization is determined by the star camera's field of view ( $4^\circ \times 2.7^\circ$ ). For  $N_{side} = 64$ , the HEALPix pixel area is  $4\pi / (12 \times 64^2) \approx 0.84\text{deg}^2$ , which allows for roughly 12 pixels per star camera field of view. The maximum pixelization will be physically constrained by the pixel size of the star camera but in practice, we find that 99.99% of the stars brighter than an apparent magnitude<sup>7</sup> of 8 are contained in a pixelization with  $N_{side} = 256$ .

---

<sup>7</sup>Apparent magnitude is a measure of luminosity on a logarithmic scale given by roughly  $l_1/l_2 =$

Our search routine begins with the coarsest index ( $N_{side} = 64$ ). If a solution is not found using the coarse index, we proceed to the next finest index and search again. In this manner, we roughly match against the brightest stars, and thus the most likely to be observed, first before proceeding to the larger — slower — search of the less luminous stars.

#### 4.3.3.2 Index Image Matching

Once a candidate field has been identified in an index, a zeroth-order Simple Imaging Polynomial (SIP)[117] distortion field is calculated. This is represented as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \underbrace{\begin{bmatrix} \text{CD1\_1} & \text{CD1\_2} \\ \text{CD2\_1} & \text{CD2\_2} \end{bmatrix}}_{\mathbf{M}} \cdot \begin{bmatrix} u \\ v \end{bmatrix} \quad (4.61)$$

where  $x$  and  $y$  are “intermediate world coordinates” in radians while  $u$  and  $v$  are the pixel positions.

The  $\mathbf{M}$  matrix encodes the field rotation, scale factor and skew (if any). By inverting  $\mathbf{M}$ , we project the remaining stars detected in the image onto RA/Dec space and compare them against RA/Dec values in the index. Potential correlations between image and index are then added to our comparison list as long as their addition increases the overall likelihood of a match (computed by Gaussian likelihood distribution as in Eq. (4.67)).

Once all potential correlations are identified, we expand the SIP to 2nd order such that

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{M} \cdot \begin{bmatrix} u + f(u, v) \\ v + g(u, v) \end{bmatrix} \quad (4.62)$$

---

$10^{-0.4(m_1 - m_2)}$ [118]. The brightest stars (apart from the Sun) are magnitude 1. Magnitude 6 stars are only just visible to the naked eye.

where

$$f(u, v) = \mathbf{A\_2\_0}u^2 + \mathbf{A\_0\_2}v^2 + \mathbf{A\_1\_1}uv \quad (4.63)$$

$$g(u, v) = \mathbf{B\_2\_0}u^2 + \mathbf{B\_0\_2}v^2 + \mathbf{B\_1\_1}uv \quad (4.64)$$

This  $\mathbf{A\_i\_j}$ ,  $\mathbf{B\_i\_j}$  space is explored about zero while the  $\mathbf{CDi\_j}$  space is explored about the coarse solution. The optimization procedure used begins with the Differential Evolution procedure as described in §4.2.8.2. Once the trial solution is identified, we refine the estimate by application of an Augmented Lagrangian method as described by Birgin and Martínez [12].

To explore the validity and robustness of the full star camera solutions pipeline, we introduce an additional, degenerate variable into our search and optimization problem. For this, we use the platescale  $\Omega$  of the star camera's CCD, defined as

$$\Omega_{x,y} = 2 \cdot \arctan \frac{0.5 \cdot d_{x,y}}{f} \quad (4.65)$$

where  $d_{x,y}$  is the  $\hat{x}$  or  $\hat{y}$  dimension of a single pixel of the CCD and  $f$  is the focal length of the camera lens. While the focal length may change during the flight as the star camera performs its autofocus routine, this change will affect both  $d_x$  and  $d_y$  equally.

Thus, by allowing both  $\Omega_x$  and  $\Omega_y$  to be fit in our algorithm, we introduce a non-trivial parameter, whose incorrect value could substantially distort the returned solution. Solution sets that are marginally or under-constrained or which exist at non-global minima are known to be susceptible to the addition of a nuisance variable[37]. If our solution set is under-constrained or does not consistently represent the true attitude solutions of the star camera, the effect of this variable would be to increase the variance between  $\Omega_x$  and  $\Omega_y$  for a distribution of images.

As shown in Fig. (4.8), the solution pipeline returns consistent  $x$  and  $y$  platescales over the majority of images, despite the two dimensions being allowed to vary inde-

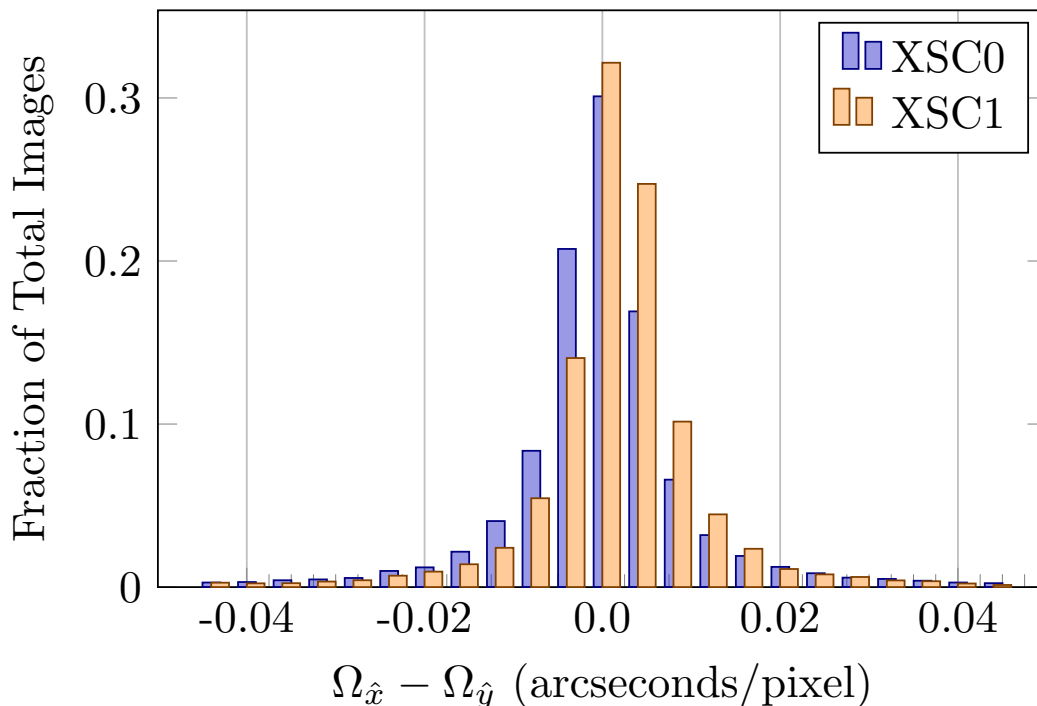


Figure 4.8: Histogram of the platescale reconstruction deviation  $\Omega_{\hat{x}} - \Omega_{\hat{y}}$  between the  $\hat{x}$  and  $\hat{y}$  CCD coordinates for the EBEX star cameras over 39,547 images acquired during the long-duration balloon flight and successfully solved in post-flight reconstruction. The histograms are computed for each camera independently. The mean for XSC0 is  $-6.5 \times 10^{-5}$  with standard deviation of  $5.4 \times 10^{-3}$ . The mean for XSC1 is  $1.55 \times 10^{-3}$  with a standard deviation of  $6.25 \times 10^{-3}$ .

pendently.

The variance between the two platescale dimensions translates to an error on sky in one dimension proportional to the platescale. Using Eq. (4.73), and applying the  $3\sigma$  uncertainty level from XSC1, we can calculate that the induced uncertainty in the attitude solution from differential platescale variation would be

$$\sigma = 3 \cdot 6.25 \times 10^{-3} \frac{0.5 \text{ pixel}}{\sqrt{4}} = 4.7 \times 10^{-3} \text{ arcsecond} \quad (4.66)$$

In the analysis of LDB flight data presented in §5, this additional degree of freedom is removed, but we see that the star camera parameterization is stable to milliarcseconds. We will discuss absolute accuracy of the solution set in detail in §5.2.

These results give us greater confidence that our accuracy measurement represents the true accuracy and not an artifact of our parameterization.

### 4.3.4 Filter Fitness Functions

Having defined the optimization procedure for the star camera internal parameters, we now have a set of attitude solutions corresponding to discrete points in time. To determine the attitude of the system between those points, we implement our particle filter as described in §4.2.7 using Alg. (A.5) and Alg. (A.6).

At each star camera reading, the dual estimation described in §4.2.8.1-§4.2.8.2 is executed. For both the particle filter and the interlaced parameter optimization routine, we now define our fitness metrics.

This work uses two metrics for its fitness calculations. The first is an attitude error between the predicted attitude and the measured attitude while the second is a measure of the noise distribution in velocity readouts. By leveraging two distinct, complementary fitness functions, we are able to converge our solution more quickly than with alternative algorithms as well as partially separate true gondola motion from readout noise in our propagating velocity measurements.

The partial separation will be critical and bears a point of emphasis here. By asserting knowledge about the level of signal independence in our gyroscope readout, we are able to reconstruct an accurate attitude solution to levels beneath the Cramér-Rao lower bound that exists for systems without this knowledge. We are not aware of previous literature that describe a similar approach to the attitude reconstruction problem.

#### 4.3.4.1 State Fitness

The metric, against which the fitness of a given particle's state (attitude) estimation is measured, is the proceeding vector measurement of attitude returned by a star camera solution. The particle cloud of trial attitude realizations  $\mathbf{y}_k^{(1...N)}$  is propagated

from the previous star camera reading  $(\mathbf{y}_{sc,(k)})$  using Eq. (4.30) up to the subsequent star camera reading  $(\mathbf{y}_{sc,(k+1)})$ .

Defining covariance matrices  $\mathbf{C}$  and  $\mathbf{N}$  as the covariances of the star camera attitude reading  $\mathbf{y}$  and the propagated gyroscope measurement  $\mathbf{x}$  respectively, the likelihood of this measurement is

$$\mathcal{L}(\mathbf{y}_{k+1}|\mathbf{y}_k, \mathbf{x}_k) \propto \frac{\exp \frac{1}{2} \left[ (\mathbf{y}_{k+1}^T \mathbf{C}^{-1} \mathbf{y}_{k+1}) \otimes (\mathbf{x}_k^T \mathbf{N}^{-1} \mathbf{x}_k)^{-1} \right]}{\sqrt{\det(\mathbf{C}) \det(\mathbf{N})}} \quad (4.67)$$

Here it should be noted that we are justified in the assumption of Gaussianity, only in the case of a properly parameterized gyroscope system from Eq. (4.39). While not accurate immediately following the system initialization, as the representation parameter errors decrease, their non-Gaussian contribution decreases and the residuals approach a normal distribution. The validity of this assumption will be shown in the following section on verification metrics.

Recalling also that attitude readings are represented as quaternions and that the error quaternion  $\delta \bar{\mathbf{q}}$  is calculated as

$$\delta \bar{\mathbf{q}} = \bar{\mathbf{q}} \mathbf{1} \otimes \bar{\mathbf{q}} \mathbf{2}^{-1} \quad (4.68)$$

For the likelihood calculation in Eq. (4.67), we use the absolute rotational magnitude of the error quaternion. Recall from Eq. (4.22) that a quaternion is represented by a vector component and a scalar component that is  $\cos(\theta/2)$ . The minimum displacement angular rotation  $\theta$  is therefore  $\theta = 2 \arccos(q_4)$ .



#### 4.3.4.2 Parameterization Fitness

The second metric, which is used by the parameter optimization routine, uses the the unrotated gyro noise process  $\boldsymbol{\nu}$  from Eq. (4.38)

$$\boldsymbol{\nu} = (I - \boldsymbol{\Lambda})(I - \boldsymbol{\Delta})T_{g,ref}\boldsymbol{\omega}_{est} - \boldsymbol{\omega}_{meas} + \boldsymbol{\beta} \quad (4.69)$$

where  $\boldsymbol{\omega}_{est}$  is the optimal particle trail calculated from the particle filter. As the three gyroscopes are independent, their noise vectors are, in principle, uncorrelated. It should be noted that there exist a number of potentially correlated noise sources (e.g. temperature dependence, physical shock, electrical noise) that are not modeled here but could contribute correlated readings. While acknowledging this possibility, in practice we do not observe distinguishable effects from secondary sources and so disregard them for the present effort.

Thus we take, as an error measure for this step, the sum of the absolute values of Pearson's correlation coefficients between the noise vectors  $\boldsymbol{\nu}$  for each pair of the three gyroscopes:

$$r_{ij} = \frac{\sum_{k=1}^N \nu_i^{(k)} \nu_j^{(k)}}{\sqrt{\sum_{k=1}^N \left(\nu_i^{(k)}\right)^2} \sqrt{\sum_{k=1}^N \left(\nu_j^{(k)}\right)^2}} \quad (4.70)$$

#### 4.3.5 Time Dependence

Up to this point, we have considered all parameters as equivalently represented in the system. This is convenient while determining effective metrics but, in practice, we wish to optimize the functional relations between parameter and state in order to provide the lowest error estimate of state vector. As the system moves from initialization to parameter convergence, the error on the full state decreases. This decreased error may ideally be applied back to the time-ordered beginning for additional accuracy of the overall attitude.

Parameter					
	Name	Description	Range (radians)	Process Noise (rad/interval)	Time Dependent
Rotation, Star Camera to Reference	$T_{ref,sc} \begin{Bmatrix} \psi \\ \theta \\ \phi \end{Bmatrix}$	Yaw	$[0, 2\pi]$	0.01	No
		Pitch	$[0, \pi]$	0.01	No
		Roll	$[-\pi/2, \pi/2]$	0.01	No
Misalignment, Gyroscopes to Reference	$\Delta \begin{Bmatrix} \delta_{12} \\ \delta_{13} \\ \delta_{23} \end{Bmatrix}$	Gyro 2 $\rightarrow$ 1	$[-0.035, 0.035]$	0.001	No
		Gyro 3 $\rightarrow$ 1	$[-0.035, 0.035]$	0.001	No
		Gyro 3 $\rightarrow$ 2	$[-0.035, 0.035]$	0.001	No
	Name	Description	Range	Process Noise	Time Dependent
Gyroscope Scale Factors	$\Lambda \begin{Bmatrix} \lambda_{11} \\ \lambda_{22} \\ \lambda_{33} \end{Bmatrix}$	Gyro 1 Scale	$[-1 \times 10^{-4}, 1 \times 10^{-4}]$	$1 \times 10^{-6}$	Yes
		Gyro 2 Scale	$[-1 \times 10^{-4}, 1 \times 10^{-4}]$	$1 \times 10^{-6}$	Yes
		Gyro 3 Scale	$[-1 \times 10^{-4}, 1 \times 10^{-4}]$	$1 \times 10^{-6}$	Yes
	Name	Description	Range rad/sec	Process Noise (rad/sec/interval)	Time Dependent
Gyroscope Offsets	$\beta \begin{Bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{Bmatrix}$	Gyro 1 Random Walk	$[-2 \times 10^{-4}, 2 \times 10^{-4}]$	$1 \times 10^{-5}$	Yes
		Gyro 2 Random Walk	$[-2 \times 10^{-4}, 2 \times 10^{-4}]$	$1 \times 10^{-5}$	Yes
		Gyro 3 Random Walk	$[-2 \times 10^{-4}, 2 \times 10^{-4}]$	$1 \times 10^{-5}$	Yes

Table 4.1: Particle filter parameter space. Shown are the 12 parameters that are optimized by the particle filter parameter optimization routine.  $T_{sc,ref}$  is shown as 3 components but will be optimized jointly as a single quaternion. The ranges are selected to represent physical limits. In the case of the misalignment, the filter searches the space between  $\pm 2$  degrees off of orthogonal, a limit that would be visible by eye. The ranges for the scale factor and offset are chosen to be twice the limits given by the manufacturer's specification sheet. Process noise represents the allowable update rate of the estimated parameter between measurements. Larger values of process noise explore parameter space more quickly but will converge more slowly.

For the iteration to function properly, we must define two things: those parameters that are constant in time and those parameters whose values are time-dependent. In the case of EBEX, we have time-independent values composing the  $T_g$  rotation matrix and the  $\Delta$  mis-alignment matrix (see §4.3). The remaining values, including  $\Lambda$ , the matrix of linear gyroscope scale factors, and  $\beta$  the vector of gyroscope offsets, are assumed to be time-dependent<sup>8</sup>. These are summarized in Tab. (4.1)

## 4.4 Verification Metrics

In verifying filter performance, we take a multi-pronged approach. First, we apply the filter to simulated data. This provides us with a direct measurement of filter performance at each timestep and for a range of input parameters and noise levels. We then use the simulation's direct measurement to cross-check the predictive capability of our indirect or proxy metrics that will be available to use when applying the filter to real, flight data.

In this section, we will step through the process of verifying performance in the simulated data. We will use these results in the following section to verify performance on flight data from the 2012 Antarctica flight.

### 4.4.1 Simulated Data

We simulate an 11-hour section of an EBEX flight. During this time, we scan a constant patch of the sky  $15^\circ \times 18^\circ$ , with a 1-hour segment scanning a calibration source (in this case, RCW38 at RA  $134.75^\circ$ —Dec  $-31.52^\circ$ ). The scan is depicted in Fig. (4.9).

From this simulated data stream, we can generate simulated sensor measurement

---

<sup>8</sup>The incorrect classification of stationary parameters as time-dependent does not materially affect the accuracy of a final state solution as parameters that have no time-dependence will merely converge to their stationary values. The effect of incorrect classification is therefore merely an issue of processing time

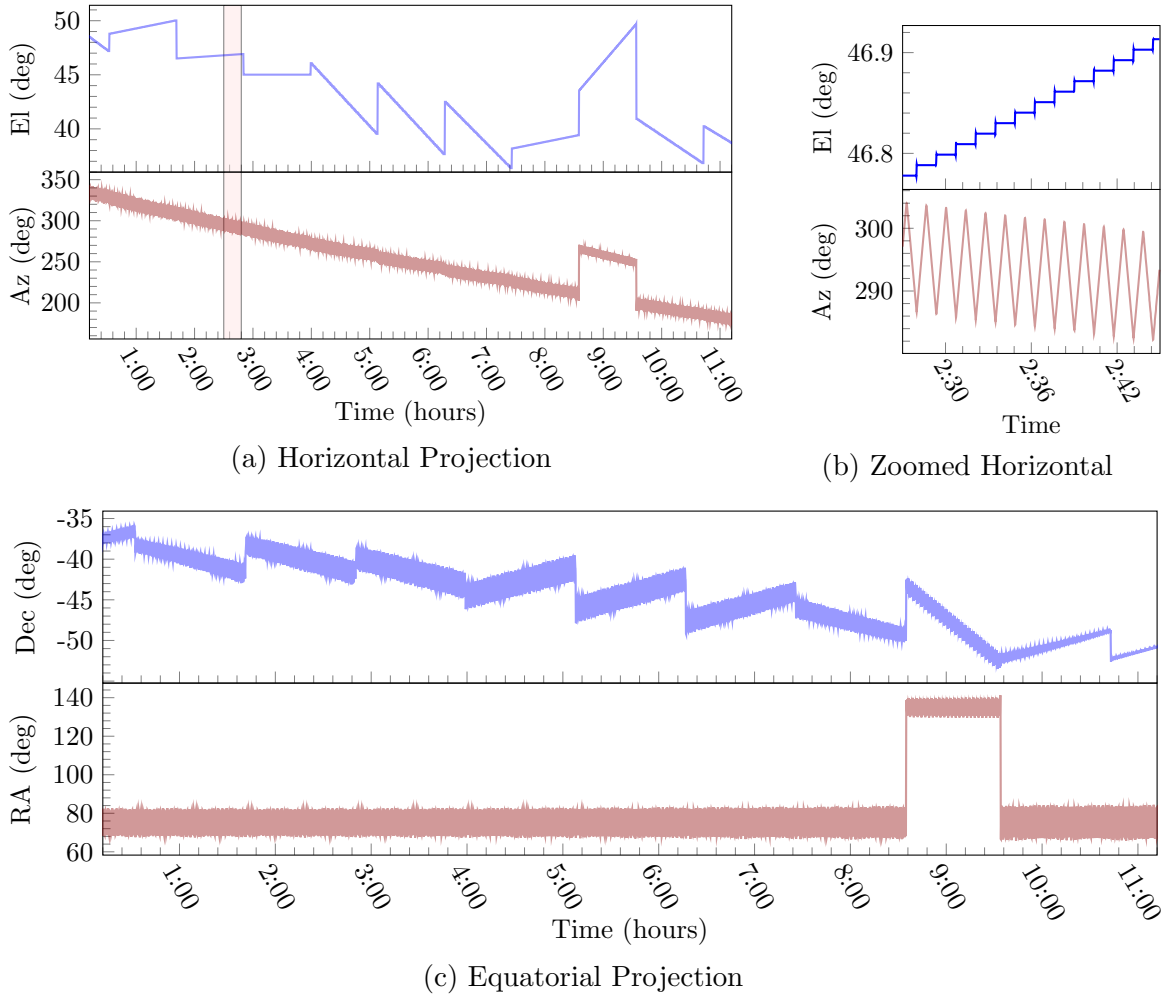


Figure 4.9: Simulated scans over  $\sim 270\text{deg}^2$  patch of the sky. The jump at  $t \simeq 8:30$  represents rotating the gondola to scan a calibration source (in this case, RCW38 at RA  $134.75^\circ$ —Dec  $-31.52^\circ$ ). Note in (b) that the scan completes both a positive and negative throw in azimuth prior to changing elevation. The projection in equatorial coordinates (c) shows the even coverage over right ascension with a stepping sweep through Declination. The horizontal coordinate projection (a) shows the effect of tracking a constant patch on the sky as the Earth and, by extension, our observation point rotates over 11 hours.

streams for our primary attitude sensors, the 3-axis gyroscopes and star cameras. The star camera measurements are spaced at low-velocity points, immediately following each motion. In the EBEX scan strategy, azimuth and elevation motions are discrete steps. Therefore the star camera measurements are acquired after each azimuthal

turn around as well as after each elevation step.

The star camera measurements represent the true, underlying attitude value, with additive gaussian noise in each of the three axes. The level of noise is chosen conservatively to be comparable to the uncertainty level expected for a star camera reading in flight. As a star camera image represents a tangent plane projection of the spherical sky, we can break the divide the 3 coordinates necessary to represent the attitude into two classes: 2 coordinates that represent motion orthogonal to the tangent plane's normal vector and 1 coordinate that represents rotation about the normal vector. The uncertainties on these coordinates will be referred to as "cross-boresight" and "roll" uncertainty, respectively.

Following a basic derivation in [74], we estimate our cross-boresight uncertainty to be

$$\sigma_{\text{cross}} = p_{\text{scale}} \frac{\sigma_{\text{centroid}}}{\sqrt{N_{\text{star}}}} \quad (4.71)$$

$$\simeq 9.54''/\text{pixel} \frac{0.5\text{pixel}}{\sqrt{4\text{stars}}} \quad (4.72)$$

$$= 2.4 \text{ arcseconds} \quad (4.73)$$

where  $p_{\text{scale}}$  is the field of view of a single pixel in the star camera imaging system,  $\sigma_{\text{centroid}}$  is the uncertainty in placement of an individual star center and  $N_{\text{star}}$  is the total number of stars expected in an image. The numbers used represent conservative estimates based on data from our engineering flight.

The remaining parameter is the boresight roll uncertainty. Estimating this number requires moderate additional detail. The roll uncertainty of the focal plane, given two stars and their associated centroid uncertainties is given by the arctangent of the ratio between the centroid uncertainty and half the distance between the stars as shown in Fig. (4.10).

We are calculating a statistical average of our expected roll uncertainty, therefore, we first calculate the average distance in pixels between two stars in the focal plane.

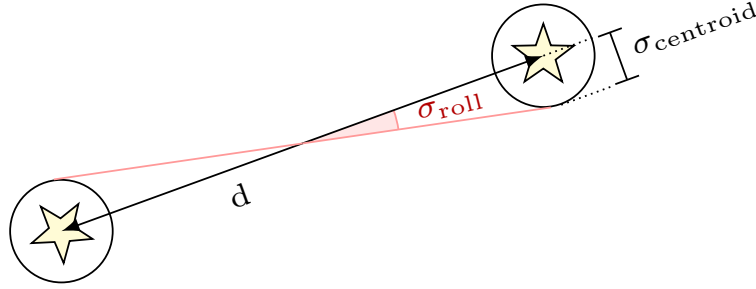


Figure 4.10: The roll uncertainty  $\sigma_{\text{roll}}$  is calculated by using the right triangle formed by the centroid uncertainty  $\sigma_{\text{centroid}}$  and the line bisecting the center of each star.

This is equivalent to the expectation value difference between two draws from a two-dimensional, uniform distribution<sup>9</sup>. In the case of the EBEX star cameras, the dimensions are given by the focal plane pixel count of 1536px×1024px.

$$\frac{1}{4 \cdot 512^2 \cdot 768^2} \cdot \int_0^{1024} \int_0^{1536} \sqrt{u^2 + v^2} (1024 - u) (1536 - v) \, du \, dv \simeq 674.3 \quad (4.74)$$

This average distance represents twice the length of the hypotenuse of a right triangle, with the star centroid uncertainty as its opposite cathetus. The estimated boresight roll uncertainty for  $N_{\text{star}} = 4$  is therefore

$$\sigma_{\text{roll}} = \frac{1}{\sqrt{N_{\text{star}}}} \arctan \left( \frac{\sigma_{\text{centroid}}}{674.4 \text{ pixels}} \right) \quad (4.75)$$

$$= \frac{1}{2} \cdot \arctan \left( \frac{0.5}{674.3} \right) \quad (4.76)$$

$$\simeq 76.5 \text{ arcseconds} \quad (4.77)$$

The gyroscope measurements are also convolved with noise meant to simulate the actual EBEX readout. This follows Eq. (4.38). The noise vector  $\boldsymbol{\nu}$  is simulated as a 40"/sec gaussian process. While actual measurements show that this value is overstated (c.f. Fig. (3.9)), we chose to keep it as a conservative value and because it

---

<sup>9</sup>Given  $X_1, X_2$  drawn from a uniform distribution  $[-a, a]$ , it can be shown that the density function of  $U = |X_1 - X_2|$  is  $f(u) = (2a - u)/2a^2$  for  $0 \leq u \leq 2a$ . The expectation value shown in Eq. (4.74) is the combined probability of two dimensions for a given distance.

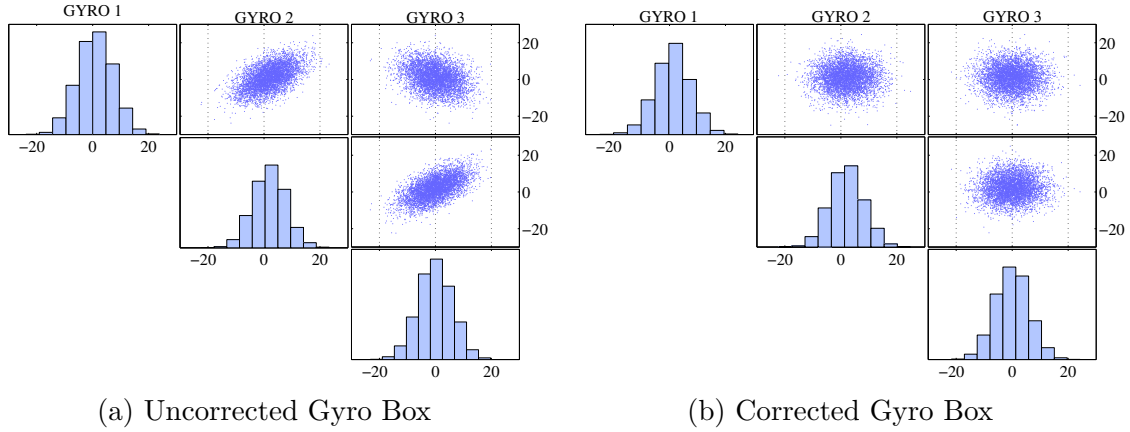


Figure 4.11: Shows the correlation matrix between 3 orthogonalized gyroscope noise streams. The histograms show distribution of the noise values while the scatter plots use the row number gyroscope for the x coordinate and the column number for the y coordinate. All axes units are in arcseconds. Shown in (a) is the noise for a single gyroscope box after particle filter optimization. The values show non-zero correlations between data streams, indicating that motional data remain encoded in the noise. In (b), after the parameter optimization has occurred and off-axis covariance values approach 0, residual data are normally distributed about the origin.

reflects the specification sheet.

The offset vector  $\beta$  is modeled as a piecewise continuous random walk function with  $\frac{d\beta}{dt} \sim \sigma_\beta$ . Integrated into the gyro velocity as an offset, this contribution should be expected to contribute as  $f^2$  or Brownian noise. Again drawing from the specification sheet, we model  $\sigma_\beta$  as a gaussian process with uncertainty of  $1^\circ/\text{hour}$ .

#### 4.4.2 Results

Using the simulated data and noise parameters as shown above, we generate 11 hours of realistic sensor readout. Utilizing only the simulated sensor readouts, we reconstruct both the simulated attitude state as well as the parameters from Tab. (4.1).

As the parameter optimization depends on our ability to remove the covariance between gyroscope noise vectors, we tuned the Differential Evolution filter parameters using the Pearson's  $r$  residual from Eq. (4.70). While we find that our naive filter

Vectors	Opposition Rate	Cross Over Rate	Selection	$p$ -Value	Convergence Time
20	0.25	0.1	Rand/Best	0.06	395ms
20	0.25	0.4	Rand/Best	0.09	714ms
30	0.15	0.3	Rand/Best	0.04	673ms
<b>40</b>	<b>0.20</b>	<b>0.1</b>	<b>Rand/Best</b>	<b>0.009</b>	<b>551ms</b>

Table 4.2: Parameter selection during the optimization stage of the parameter filter evaluation. The  $p$ -value measures the extent to which gyroscope parameters successfully reduce the residual noise vectors to uncorrelated streams (lower is better). The convergence time is an average time spend evaluating potential solutions before an accepted proposal is generated. The convergence time is measured per filter step.

parameters<sup>10</sup> chosen for the filter were sufficient to achieve acceptable results, a coarse, by-hand optimization resulted in a substantial improvement in resulting  $p$ -value while acceptably increasing the computation time.

The results of the final filter parameter selection (**bold** in Tab. (4.2)) on the gyroscope noise levels are shown in Fig. (4.11). This removal process occurs at each star camera measurement step in the filter. The removal of correlated signals should not be read as an indication of absolute parameter convergence. Rather the removal of correlated signals represents the maximum likelihood of the parameter set, given a state realization.

The improvement in gyroscope parameter selection can, in principle, be recursively applied to the same data interval. Re-evaluating the same state data with alternate parameters does provide new noise vectors whose data are typically correlated by the residual error in parameter and state estimation. However, we find that overall convergence rates are faster when new data are introduced, propagating the parameters forward in time.

Once new data are exhausted, however, there may still be benefit in recursing to

<sup>10</sup>An unfortunate collision of nomenclature occurs here where we have parameters governing our overall filter as well as the system parameters that our filter is optimizing. We will refer to the parameters governing the filter explicitly as *filter parameters*. Other uses of *parameter* should be implicitly read as referring to the parameters detailed in Tab. (4.1)



the beginning of our time stream with the improved parameter estimates. Indeed, post-flight filtering is an iterative process and we expect stationary parameters' reconstructed values to converge over time with the inclusion of improved state estimates. This process requires a valid halting condition, for which we use a windowed variance function. We consider a parameter in our filter to have converged when the windowed variance no longer decreases with additional data.

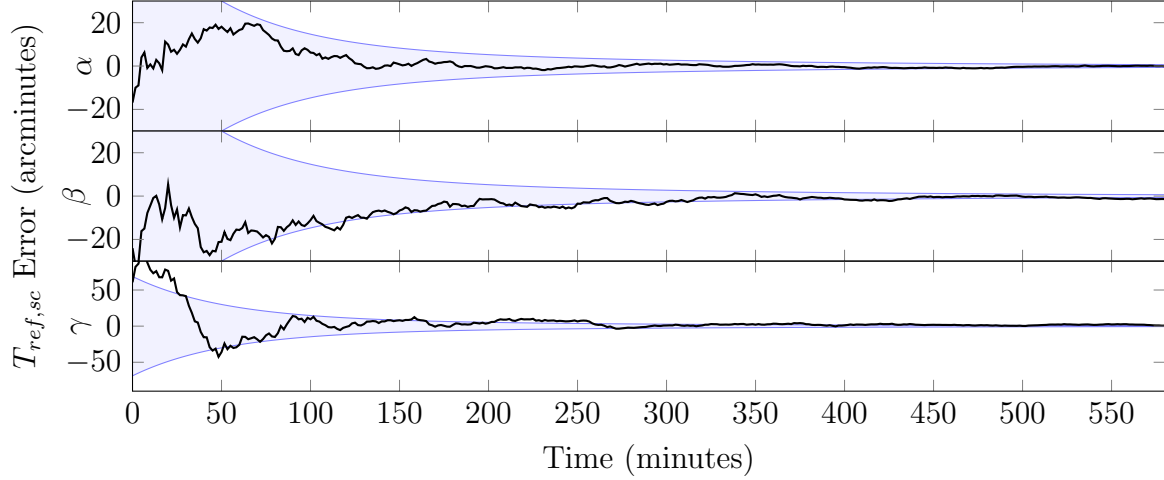


Figure 4.12: Orthogonal rotation parameter ( $T_{ref,sc}$ ) errors during an initial filter run. The parameter error (—) is the difference between the true value (a constant) and the filter's best estimate at that timestep. The envelope ( $\square$ ) is an idealized Cramer-Rao lower bound on the estimate, given known noise parameters of the filter.

The performance of the filter in parameter reconstruction for a single iteration is shown in Figs. (4.12, 4.13 and 4.14). We compare the true error against the Cramer-Rao Lower bound, a statistic measure inversely proportional to the assumed information content of the system[129]. That is, as the information available to the particle filter increases, the minimum variance of an estimated solution decreases. For the purposes of this we utilize the Posterior Cramer-Rao Bound (PCRB) as described in [10].

Additionally, because we are dealing with simulated data, we can assess the absolute error levels associated with our reconstruction. We allow the particle filter to

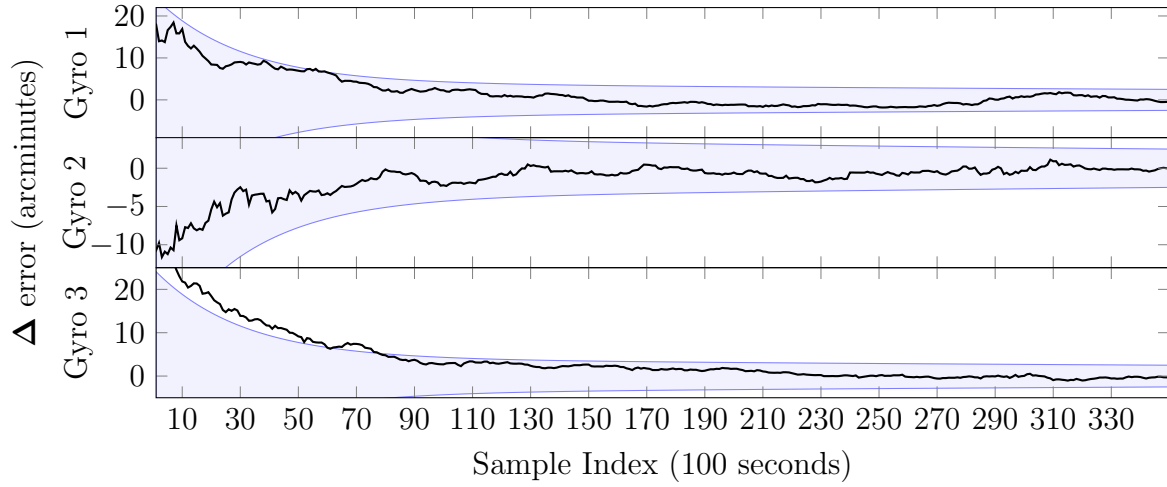


Figure 4.13: Non-orthogonal gyroscope rotation ( $\Delta$ ) errors during an initial filter run. The filter parameter error (—) is the difference between the true value (a constant) and the filter’s best estimate at that timestep. The envelope ( $\square$ ) is an idealized Cramer-R ao lower bound on the estimate, given known noise parameters of the filter.

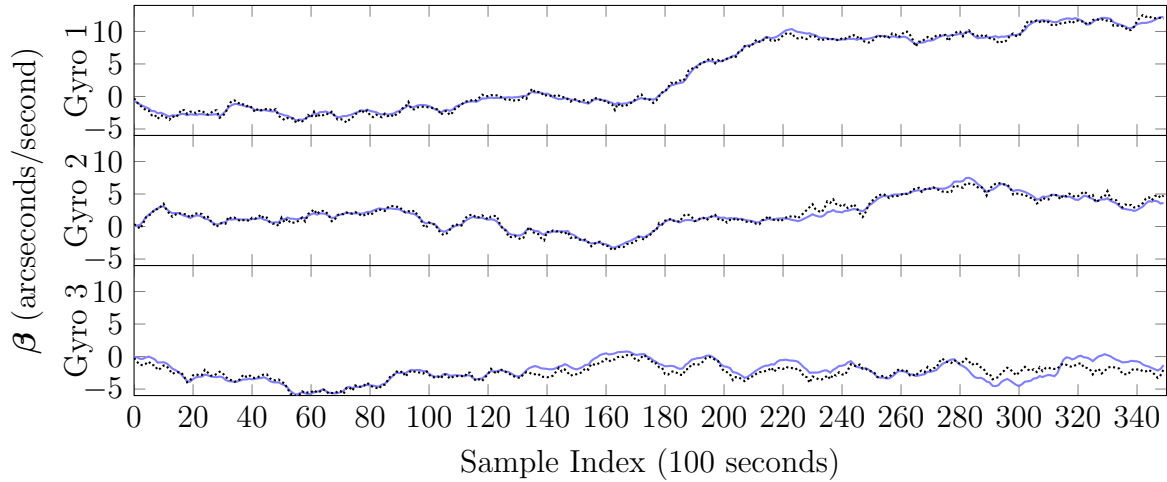


Figure 4.14: Underlying (.....) and reconstructed (—) gyroscope offset ( $\beta$ ) parameters. Gyro 1 and Gyro 2 are grossly aligned with the azimuth scan motion and thus exhibit closer tracking of their offset values. Gyro 3 tracks the elevation velocity, which is close to 0 for the majority of the scan time, and therefore is less tightly constrained by the particle filter.

converge in its parameter space before conducting a final reconstruction utilizing the

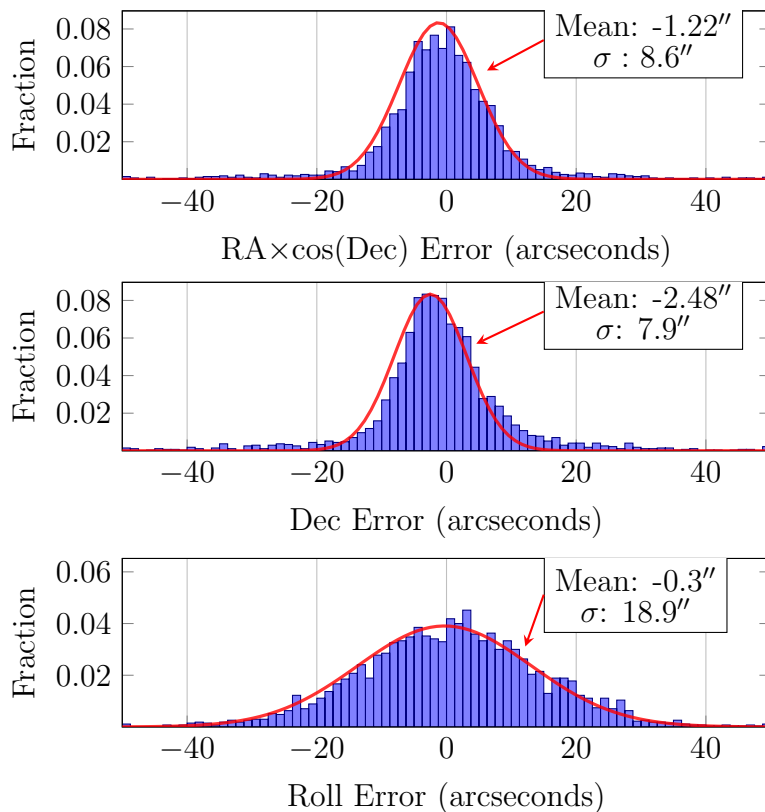


Figure 4.15: Reconstruction histograms for an equatorial projection. *Top*: Difference between simulated and reconstructed data for right ascension, projected into equidistant spacing by cosine of the declination. *Middle*: Difference between simulated and reconstructed data for declination. *Bottom*: Difference between simulated and reconstructed data for roll. Note that the roll error distribution is significantly larger than either RA or Dec as this axis is less constrained by the star camera measurements.

best-fit values for each of the system parameters. We then compare the reconstructed attitude with the true attitude used to generate the simulation. The results of this are shown in Fig. (4.15)

The levels reconstructed are roughly 4.2% of the EBEX beam size<sup>11</sup>. This is well beneath the 10% margin calculated by Hu et al. [54], as the minimum RMS required to constrain the energy scale of inflation to  $E_i < 10^{16}\text{GeV}$ , or roughly  $r < 0.05$ .

There are two caveat to this, however. The first is that it assumes a gaussian

<sup>11</sup>The EBEX beam full width at half maximum (FWHM) is  $8'$ . Thus the beam size is  $8'/\sqrt{2 \cdot \ln 2} \simeq 3.4$

distribution of error in the pointing. This is not accurate in general as our error is minimized at the edges of our scans where we have star camera readings and maximized at the middle of the scan. Thus the errors will not be gaussian over the full patch. Second, recall the imprint of gravitation lensing at later timescales will occur at much smaller angular scales (cf. Fig. (2.7)). Thus even if we are sufficiently accurate to constrain the signal at low- $\ell$ , we may not be able to separate it from the high- $\ell$  lensing signal.

Thus, we would like to quantify the calculations of [54] for the specific case of our reconstruction. The two contaminants from pointing error represent the projections of an E-mode signal onto an incorrect orthogonal frame. These are

$$\delta C_{\ell}^{BB} = \int \frac{d^2 \ell_1}{(2\pi)^2} C_{\ell_2}^{EE} C_{\ell_1}^{\theta\theta} \left[ \sigma \left( \ell_2 \times \hat{\ell}_1 \right) \cdot \hat{z} \sin [2 (\phi_{\ell_2} - \phi_{\ell})] \right]^2 \quad (4.78)$$

$$\delta C_{\ell}^{BB} = \int \frac{d^2 \ell_1}{(2\pi)^2} C_{\ell_2}^{EE} C_{\ell_1}^{\theta\theta} \left[ \sigma \left( \ell_2 \cdot \hat{\ell}_1 \right) \sin [2 (\phi_{\ell_2} - \phi_{\ell})] \right]^2 \quad (4.79)$$

Where  $\sigma$  is the beam width,  $\ell$  is the mode of interest,  $\ell_2 = \ell - \ell_1$  and  $\phi_{\ell} = \ell_x / \ell$ . We are not concerned by the individual components and so calculate them as a unit. To compute the  $C_{\ell}^{\theta\theta}$  error generated by the errors in the pointing solution, we divide the map into equal area HEALPix representation[46] and decompose into spherical harmonics to recover the amplitudes of anisotropies for the range of multipole moments available in our map.

Then, using the current best-fit values to the  $\Lambda$ CDM model, we generate an spectrum for  $C_{\ell}^{EE}$  as we did for Fig. (2.7). By application of Eq. (4.78) and Eq. (4.79), we can then generate the anticipated for power spectrum contamination arising from the pointing error in our reconstruction simulation. This is shown in Fig. (4.16).

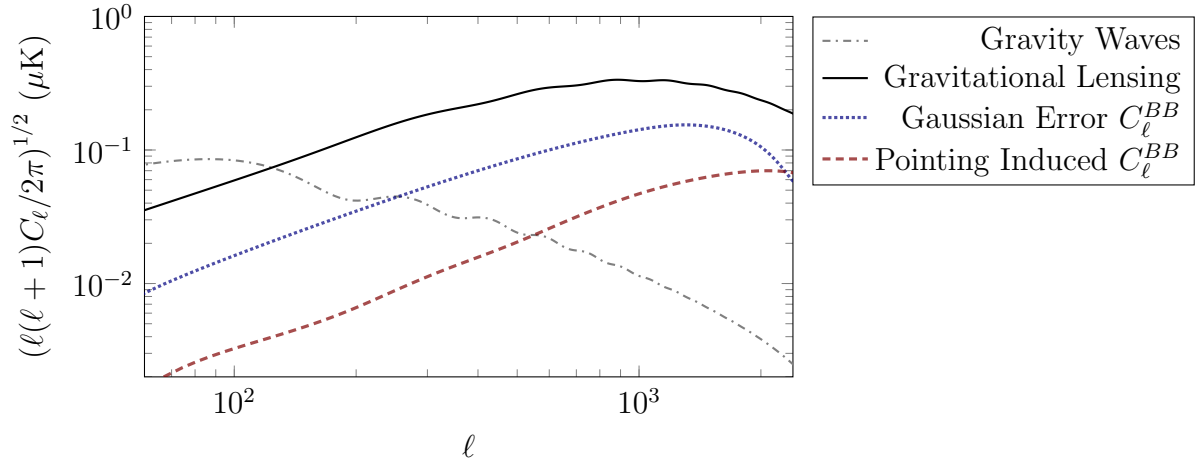


Figure 4.16: Contamination arising from our pointing reconstruction error rotating an E-mode signal into a B-mode. The red dashed line represents the effect of our pointing reconstruction error for 11 hours of data. We are sub-dominant to the lensing signal, shown in black as well as the anticipated inflationary gravity wave signal for  $r = 0.1$ . For comparison, a  $9''$  constant, gaussian error is also included as this was assumed by [54].

## Chapter 5

# LDB Flight Attitude Reconstruction

“In some calm Stream, their Oars and Helm explore,  
And learn their Art, preluding near to shore  
But well-experienc’d now, they tempt remoter Seas,  
And miss not the Land lost by swift Degrees.”

---

— Publius Papinius Statius, *Thebaid*

### 5.1 Summary of the EBEX Long Duration Flight

After 2 months of integration and pre-flight tests at Williams Field, Antarctica, EBEX was launched on December 29th, 2012. The flight path is shown in Fig. (5.1). The payload took data for 11 days prior to exhausting its cryogenics. This corresponded closely with pre-flight calculations of loading on the cryogenic stages. After the cryogenics were exhausted, the payload remained at float altitude until it drifted within range of the CSBF recovery team. The data vessels, star cameras, DfMux crates and additional small items were recovered on February 1st, 2013. The remaining, heavier items including the gondola and receiver were unable to be recovered due to weather conditions and an abbreviated science season. It is planned to recover the remaining

equipment during the upcoming austral summer.

This chapter will describe, in general terms, the EBEX flight and specifically the application of the described attitude reconstruction algorithm to the Long Duration flight data. Because the actual flight attitudes differed substantially from the pre-flight plans, the higher-accuracy solution method described herein may not be required and the final EBEX data products may be generated by a faster, lower accuracy solution currently being developed.

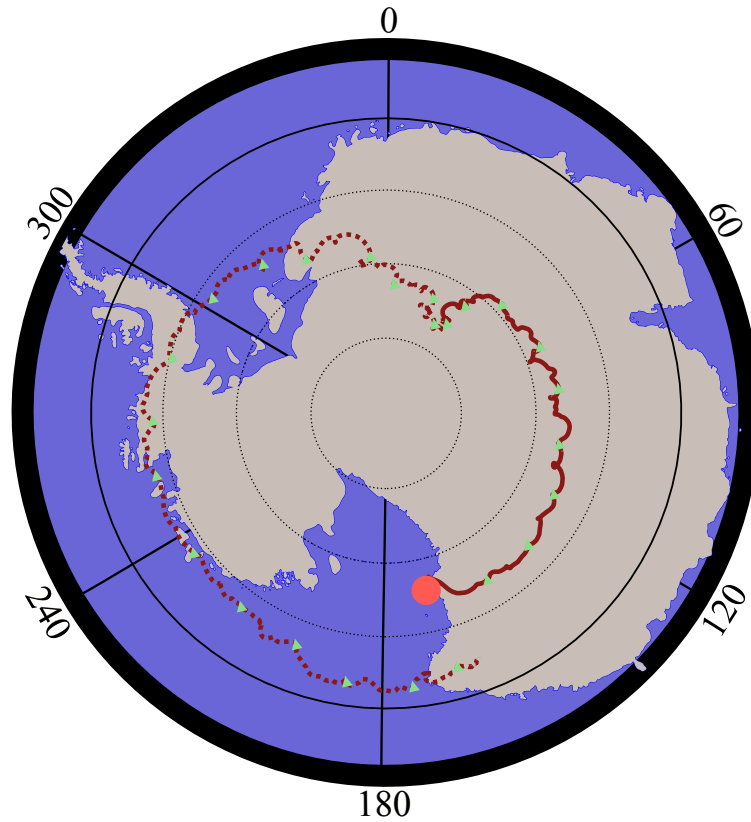


Figure 5.1: The EBEX 2013 LDB flight trajectory. The solid path denotes flight periods where data were recorded. The dashed line shows the flight path after the expiration of our cryogenics. The payload remained at float altitude but could not land due to its location relative to a potential recovery route.

Upon reaching float altitude, the temperature of the azimuthal pivot motor controller began to rise. This temperature rise was substantial enough to trip a thermal circuit, effectively preventing azimuthal control while the motor controller was exposed

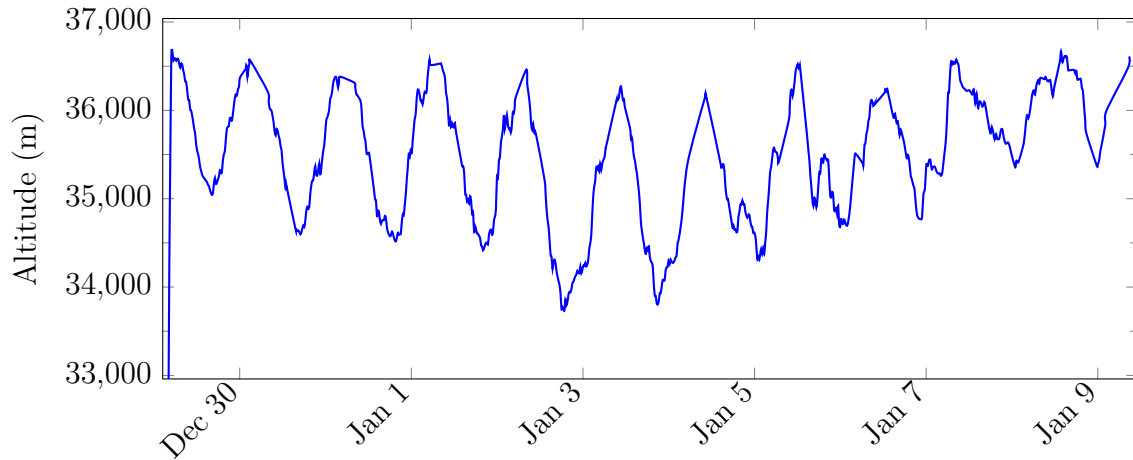


Figure 5.2: Flight altitude of the EBEX payload over the 11 days of data gathering. Clearly visible are the diurnal expansion cycles. The balloon altitude was largely stable over the course of the flight with no sustained loss of altitude.

to the sun. Due to the mount position of the motor controller, it was not feasible to prevent this exposure and we therefore decided to pursue a free-rotating azimuth scan strategy, covering a larger section of the sky,

In this mode, we set our elevation between  $57 - 59^\circ$ , so as to prevent the sun from illuminating our focal plane and allowed the payload to turn freely. As the wind currents pushed our balloon, the wind speed gradient over latitudes, although slight, was sufficient to rotate our payload on average of  $\sim 1.5$  full rotations per hour as shown in the top plot of Fig. (5.3).

Superimposed on this low-frequency rotation were lower-amplitude, higher-frequency rotations. This higher-frequency rotations provided a scan pattern that was similar, in principle, to the planned scan patterns in that they provided multi-pass coverage over the same sky patch with sufficiently low acceleration to acquire high-accuracy star camera images as shown in the bottom plot of Fig. (5.3).

This azimuthal pattern is critical to our ability to successfully reconstruct the attitude information during flight. As detailed in §4.3, periodic star camera measurements are required to constrain the overall uncertainty of the attitude solution. The pre-flight plan, however, called for azimuthal scans of roughly similar size, encompass-



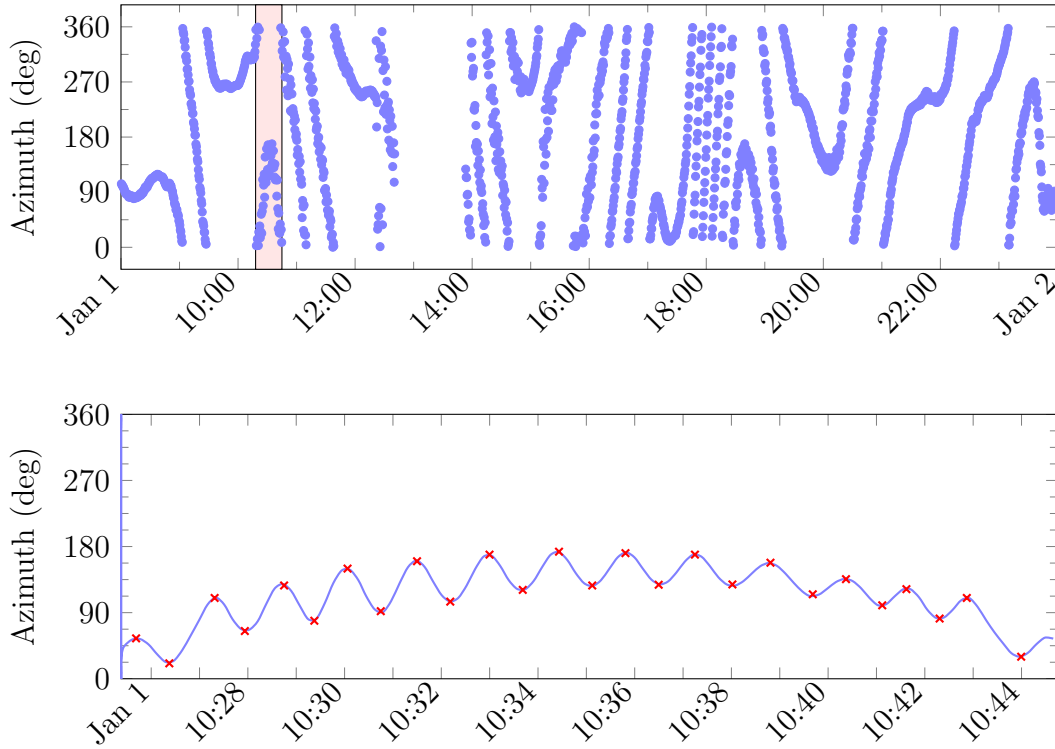


Figure 5.3: Coarse azimuth from a single day of scanning during the LDB flight. *Top:* the low-frequency oscillations driven by wind-induced balloon rotation are visible. *Bottom:* The highlighted section from 10:26-10:46 is shown. Here, (—) shows the in-flight, coarse azimuth while the (\*) marks denote where star camera images were acquired.

ing approximately 40 seconds of relative gyroscope measurements between vector star camera measurements. This would have allowed for roughly constant uncertainties based on the time from the last star camera measurement.

While that remains true in the actual data acquired during the LDB flight, by varying the time between star camera readings, as shown in Fig. (5.6), our reconstruction uncertainties span a much larger range than previously intended. The process of reconstructing, however, remains largely the same.

## 5.2 Star Camera Solutions

The initial step in the reconstruction is to pass the star camera images through the post-processing pipeline. Because of the sheer volume of images, this is an automated, iterative process. The procedure is as follows

1. **Noise Reduction** Images are processed using the symmetric weighting technique described in §4.3.2.1. The point spread function (PSF) for the associated parameters of the symmetric weighting filter is saved in the image header.
2. **Extract source candidates** The processed images are convolved with their associated PSFs to identify potential source candidates. The X-Y coordinate list and associated centroid error is saved in the image header.
3. **Solve fields - First Pass** The X-Y coordinates are passed to the field solver to attempt matches against the star catalog. Images with cross-boresight uncertainty higher than 8'' are set aside for further processing.
4. **Iterative Processing** All unsolved images and those set aside for further processing are collected and examined in a semi-automated process. Each image is normalized and false-colored to enhance the point source contrast. The identified sources are then over-plotted to show the source extraction accuracy. Examples of this are shown in Fig. (5.4).

Images with no visible features are marked for exclusion. For the remaining images, the analyst adjusts control parameters of the noise reduction and source extraction algorithms to minimize the residual uncertainty in an image. Adjusting parameters in this manner changes which point sources are recovered for the image and with what uncertainty. It does not modify their underlying positions in the image — and by extension the field solution — without concurrent uncertainty modification.

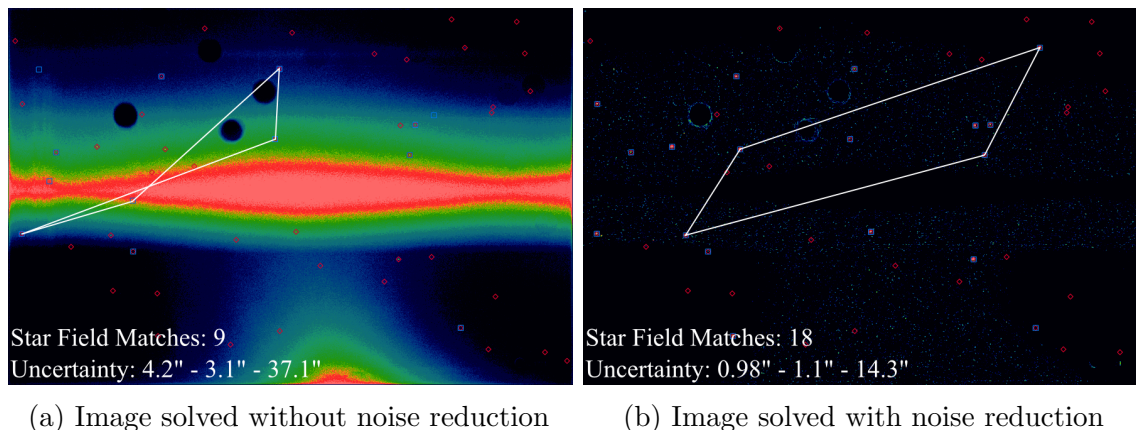


Figure 5.4: Examples of processing the same image with and without the symmetric noise reduction applied. False color images are created to assist human interpretation. Extracted sources are plotted as blue boxes( $\square$ ). Catalog sources are plotted as red diamonds( $\diamond$ ). The angular tetragram (described in Fig. (4.6)) used to identify the field is plotted in white. The number of co-located catalog and extracted points is shown in the inset text along with the resultant solution uncertainties are displayed for the current realization. (a) shows fewer matches to the catalog and exhibits an overall higher uncertainty. After noise reduction, (b) locates 9 additional stars and filters a number of false matches, resulting in a factor of 3 improvement in solution uncertainty.

### 5.2.1 Generalized Symmetric Weighting Analysis

Given the robust performance of the EBEX star cameras during the LDB flight, we have a large number of star camera images with varying noise levels corresponding to our azimuth, relative to the Sun. These allow us to analyze the performance of the symmetric noise reduction filter introduced in §4.3.2.1. To do this, we attempt to solve all images from the LDB flight using a gaussian PSF but without noise reduction. This results in a set of 27,401 images solved out of a total of 40,930.

This subset of images is then re-processed with the noise reduction filter and the new sources extracted using the convolved PSF. 99.04% of these new point fields yield matches to the star catalog. Only one set of filter parameters is used to extract star positions for this test and so it is feasible that valid solutions to the remaining 264 unmatched may be found by selecting alternate weightings.

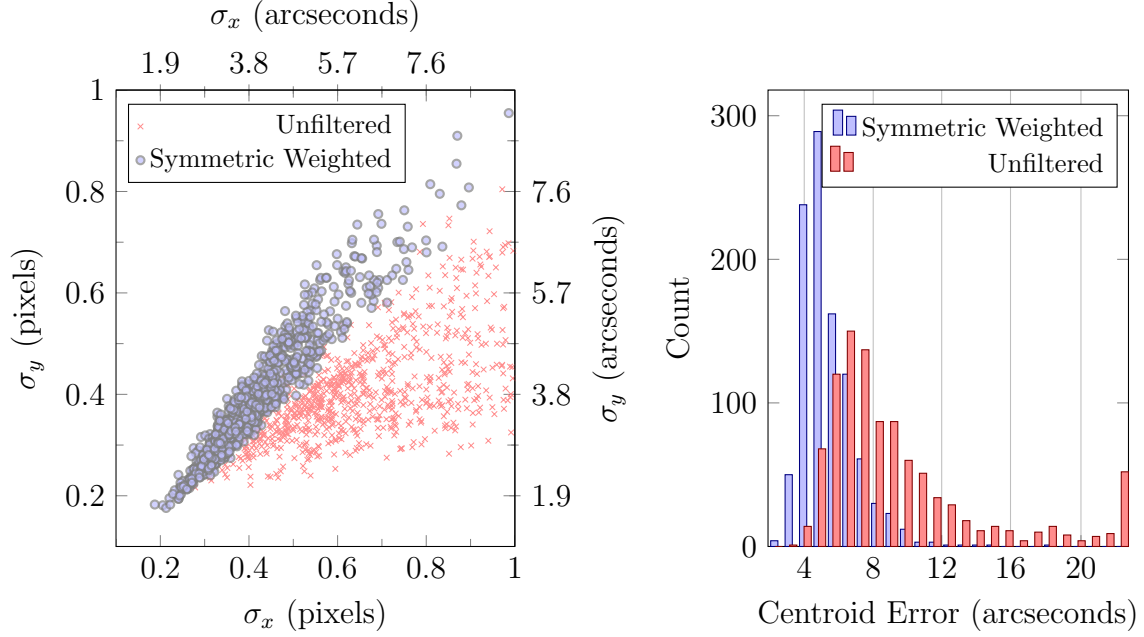


Figure 5.5: *Left:* The distribution of centroiding errors in the focal plane X-Y space is shown for unprocessed images (\*) and images processed by the generalized symmetric weighting filter (•). *Right:* The same data are projected into a histogram showing the absolute distance from the origin. Note that the symmetric weighting values

The uncertainty of each solution is calculated by a weighted matching of catalog positions to the projected X-Y positions on the star camera focal plane. We are able to invert this calculation to find the average positional uncertainty in X-Y space of a point source in the image, given the total number of stars matched ( $N_{\text{star}}$ ) as:

$$\langle \sigma_{x,y} \rangle^{(\text{star})} = \sqrt{N_{\text{star}}} \sigma_{x,y}^{(\text{image})} \quad (5.1)$$

where  $\sigma_{x,y}^{(\text{image})}$  is the cross-boresight uncertainty given in Eq. (4.73).

The result of this process is shown in Fig. (5.5). The unfiltered images show a larger uncertainty distribution in general ( $\mu_{x,y}^* = (0.87, 0.52)$  pixels vs.  $\mu_{x,y}^\circ = (0.47, 0.44)$  pixels) as well as a higher relative uncertainty in the  $\hat{x}$  coordinate compared to the  $\hat{y}$  coordinate. This may be explained by the primary gondola scanning motion being orthogonal to the  $\hat{y}$  direction for the low-roll conditions that dominate the flight.

We also observe a tighter distribution of the filtered points about the  $\sigma_x = \sigma_y$  line. This is to be expected for a symmetric point weighting and indicates that the filter is performing as expected. While in itself, the increased precision is not indicative of an improved solution set, it provides an explanation for the lower mean uncertainty of the dataset. As the symmetric filter excludes corrupting noise from the unfiltered image, it allows the centroiding algorithm to more accurately choose the point source center. This also shrinks the uncertainty distribution about the center point.

### 5.2.2 Scan Length Distribution

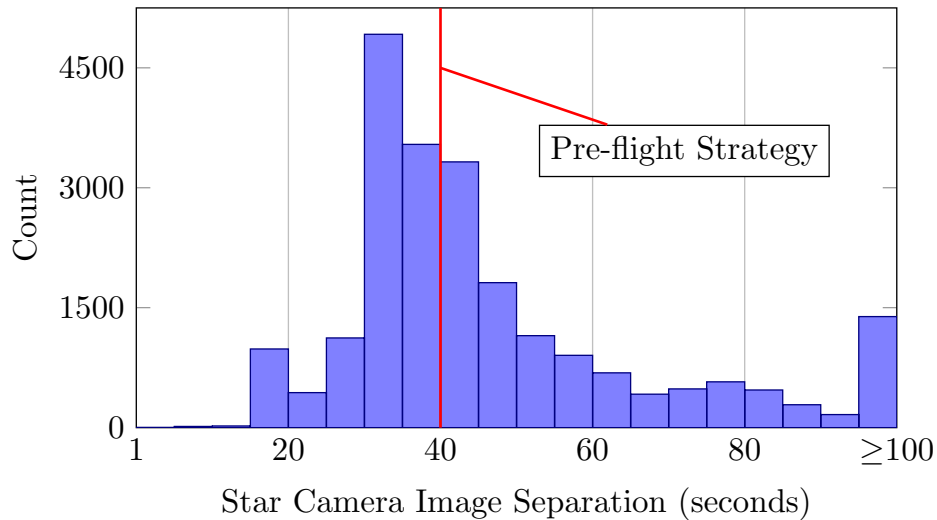


Figure 5.6: The distribution of star camera solutions as a function of distance to the subsequent solution. Shown in red (—) is the planned scan length.

By applying the full image solution pipeline described above, we arrive at a data stream that can be divided, for the purposes of attitude reconstruction, into segments of gyroscope data, bracketed by star camera solutions. As our attitude reconstruction uncertainty will be dominated by the integrated uncertainty of the gyroscope data between star camera measurements, we seek to minimize the time between star camera readings. The mode of our star camera solution time separation lies at approximately 33 seconds, just below the pre-flight planned separation of 40 seconds. This is shown

in Fig. (5.6).

We will return to this distribution in §5.4 as the broad distribution of scan lengths will prove useful in mapping out the performance of our particle filter compared to a theoretical prediction.

### 5.3 Gyroscope Readings

After solving star camera images, the next task is to clean the gyroscope time streams. We are assisted in this task by the fact that EBEX utilized two sets of independent gyroscopes. Additionally, in each set of gyroscopes, one gyroscope was roughly aligned to read out only the elevation motion while the remaining two read out orthogonal projections of the azimuth and roll motions.

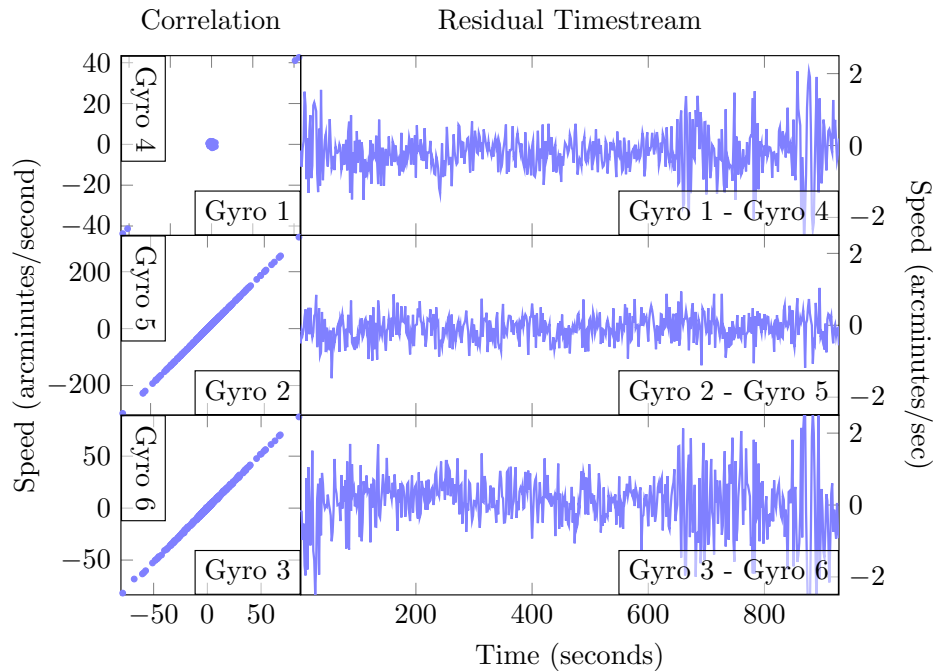


Figure 5.7: The left column in this plot shows plotting gyroscope pairs against each other. The slope of each plot is nearly identical to 1.0. The right column shows the residuals between gyroscope pairs, assuming a slope of 1.0 and offset of 0. By comparing anomalous readings between paired gyroscopes, we can distinguish between readings that are truly a response to extraordinary physical motion and readings that represent readout noise.

This is most clearly seen in the comparison plots shown in Fig. (5.7). The system shows a highly aligned response to physical stimuli allowing for a system differencing that we use to distinguish readout values that are motional in nature as opposed to those that are merely readout noise.

Despiking the gyroscope data consists for EBEX of three distinct steps. We will enumerate them first before describing each in detail.

1. **Identification** Potential spikes are first identified by one of three related metrics for determining deviation.
2. **Verification** Once identified, we compare the underlying data point against the paired gyroscope.
3. **Replacement** Once all spikes in a segment have been verified, their values are replaced by a maximum likelihood prediction, given proximate, analogous data.

### 5.3.1 Spike Identification

Since the EBEX gyroscope readouts encode not only the true angular velocity but also a number of noise sources as detailed in §4.3.1, we must distinguish those sources, which will be dealt with in the context of our filter, from the the concept of a *spike* in the data. The spikes that we wish to identify are believed to be caused by either electrical malfunction in the gyroscope itself, noise in the cabling<sup>1</sup> from the gyroscope to the readout card or temporary malfunction in the readout card. These types of events are, by definition, of a transient nature and high relative amplitude.

For the purposes of identifying the transient events, we first process our data by subtracting the signal median and linear trend to push the bias random walk response into the low-frequency spectrum. Next, we use of a high-pass filter to remove the majority of true motion and bias noise from the gyroscope readouts. Since the

---

<sup>1</sup>Because the readout is a digital signal, an electrical noise event in the cable that corrupts the signal is not distributed in frequency space

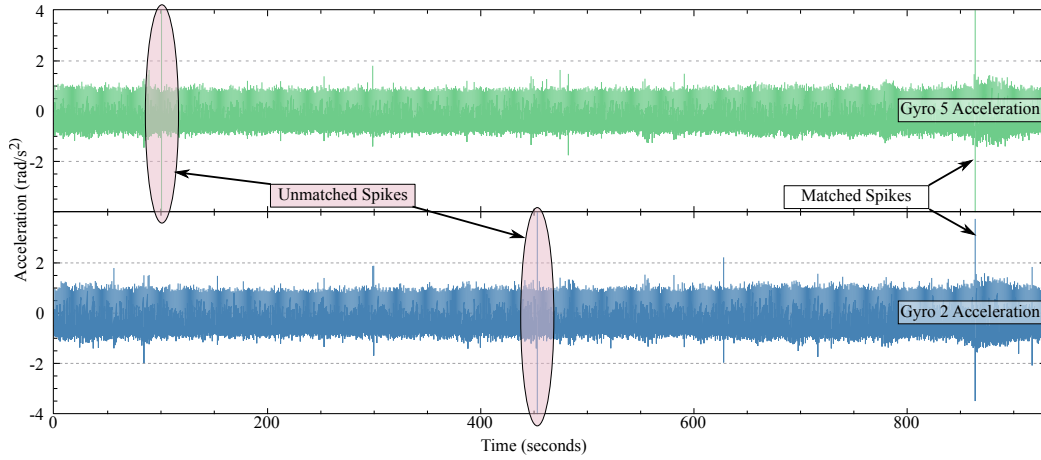


Figure 5.8: Shown here is the acceleration for the two matched gyroscopes. Spikes highlighted in this plot are those that naively fall outside of the maximum expected value for the time series. The matched spikes on the right are true motion, incorrectly identified for removal but later allowed by the verification process of §5.3.2. There are additional, smaller magnitude spikes that will be identified by expanding the analysis to additional dimensions as will be described.

majority of the gondola motion is on the order of a few Hz, we choose conservatively to high-pass at 40Hz. The remaining signal is dominated by intrinsic gyroscope noise with a largely gaussian distribution about 0.

Given  $N_x$  samples (samples denoted  $x$ ) from this normal distribution, the maximum expected value  $E(|x|_{\max})$  is

$$E(|x|_{\max}) = \sigma_x \sqrt{2 \ln N_x} \equiv \zeta_x \quad (5.2)$$

Defining  $\zeta_x$  as this limiting expectation, for a coarse spike removal we might simply look at the values in the time series that are higher than  $\zeta_x$ . This is shown in Fig. (5.8). Numerous other methods have been shown in the literature to perform adequately in spike identification including wavelet shrinkage[29], soft-threshholding[28] and Kalman Filtering[72].

While this simply analysis with remove some spikes, additional spikes remain and can corrupt our attitude reconstruction. Fortunately, this secondary metric of attitude



reconstruction provides us with a quantitative tool with which to assess the quality of our spike removal algorithm<sup>2</sup>.

For the EBEX data, we utilize the de-spiked output in our full reconstruction pipeline. This allows us to characterize the performance of the despiking algorithms on our gyroscope time streams relative to an independent measurement of the attitude, namely star camera measurements. Using the filter performance as a our characterization metric, we utilize a 3-dimensional ellipsoid thresholding model, based on work by Goring and Nikora [45].

To amplify the magnitude of the transient events relative to the background noise, we project the data onto a Poincaré map using fixed-step approximates to the first and second derivatives. The application of first and second derivatives to the problem of motion classification has been well-studied[15, 60]. In our approach, we use this projection to identify outliers relative to the ellipsoid formed by the three-dimensional version of Eq. (5.2).

First, utilizing the raw data, we generate fixed-step proxies for the first and second derivatives. Here  $x_i$  represents an angular velocity measurement at time step  $i$  from an arbitrary gyroscope. Thus  $\Delta x_i$  approximates the acceleration at that time  $i$  and  $\Delta^2 x_i$  approximates the change in acceleration.

$$\Delta x_i = \frac{1}{2} (x_{i+1} - x_{i-1}) \quad (5.3)$$

$$\begin{aligned} \Delta^2 x_i &= \frac{1}{2} (\Delta x_{i+1} - \Delta x_{i-1}) \\ &= \frac{1}{2} (x_{i+2} + x_{i-2} - 2 \cdot x_i) \end{aligned} \quad (5.4)$$

Next, we calculate  $\zeta_x$ ,  $\zeta_{\Delta x}$  and  $\zeta_{\Delta^2 x}$ , the expected maximal values for velocity, acceleration and change in acceleration, respectively, using Eq. (5.2). These will represent projections of the semi-major and semi-minor axes of our ellipses onto the

---

<sup>2</sup>Recall that the particle filter works by integrating gyroscope readouts from one star camera reading to the next. While the integrated gyroscope data will be affected by these spikes, the star camera reading will not.

Cartesian axes. Goring and Nikora found that for large  $N$ , the  $\zeta$  were roughly 10% overstated and suggested a simplistic scaling. We avoid this by replacing  $\sigma_x$  in Eq. (5.2) with a median deviation estimator denoted  $Q_x$ , and defined by Rousseeuw and Croux [110] as

$$Q_x \equiv d \cdot \{|x_i - x_j|; i < j\}_{(h)} \quad (5.5)$$

Here,  $(h)$  is the statistical order where  $h = (N_x/2) + 1$ , just over half of  $N_x$ , the total number of observations. The bias correction factor  $d$  is

$$d = 2.2219 \cdot \begin{cases} \frac{N_x}{N_x+1.4} & N_x \text{ is odd} \\ \frac{N_x}{N_x+3.8} & N_x \text{ is even} \end{cases} \quad (5.6)$$

With this estimator, our calculations of  $\zeta$  empirically do not require scaling over large datasets. While computationally more expensive than alternative robust estimators, we utilize an algorithm given by Croux and Rousseeuw [24] for computing  $Q_x$  in  $\mathcal{O}(n \log n)$  time. This computational burden is sub-dominant in our full estimation pipeline.

Our next step is to account for a linear correlation between the vectors. Physically, this could imply we are more likely to experience positive acceleration while we measure positive velocity and negative acceleration with negative velocity, etc. The magnitude and sign of the angle encode the level of correlation as well as whether the measurements are correlated or anti-correlated. By symmetry, the measurement of velocity ( $x$ ) should be entirely uncorrelated with either acceleration or the change in acceleration, providing we have removed all physical motion and long-term drift using the high-pass filter. In practice, this is not always the case, and some residual correlation exists. This does not pose a problem for spike identification, provided we allow for its existence.

We accomplish this by calculating the angles of rotation given by a simple linear

regression as

$$\tan \alpha = \frac{\sum_i x_i \Delta x_i}{\sum_i x_i^2} \quad (5.7a)$$

$$\tan \beta = \frac{\sum_i x_i \Delta^2 x_i}{\sum_i x_i^2} \quad (5.7b)$$

$$\tan \gamma = \frac{\sum_i \Delta^2 x_i \Delta x_i}{\sum_i (\Delta^2 x_i)^2} \quad (5.7c)$$

These angles are applied to the phase space map of Eqs. (5.3 & 5.4) to get the unrotated projections. We then apply Eq. (5.2) to get the semiaxis for each dimension. While these can be calculated through a more complicated trigonometric inversion of the variance, we find this method to be far clearer in the implementation and only trivially more computationally expensive.

In cartesian space, the maximal expected ellipse for each Poincaré map is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathcal{R}(\gamma)\mathcal{R}(\beta)\mathcal{R}(\alpha) \begin{bmatrix} \zeta_x \cos(\theta) \cos(\phi) \\ \zeta_{\Delta x} \cos(\theta) \sin(\phi) \\ \zeta_{\Delta^2 x} \sin(\theta) \end{bmatrix} \quad (5.8)$$

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \mathcal{R}^{-1}(\alpha)\mathcal{R}^{-1}(\beta)\mathcal{R}^{-1}(\gamma) \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (5.9)$$

where  $\mathcal{R}$  is the rotation matrix for each angle, while  $\theta$  and  $\phi$  are polar coordinates evaluated from  $[-\pi/2, \pi/2]$  and  $[-\pi, \pi]$  respectively. Similarly,  $x'$ ,  $y'$  and  $z'$  compose the rotated coordinate frame that lies along the ellipse axes.

Spikes are identified by first rotating the point under consideration into alignment with the Cartesian axes using Eq. (5.9) and then solving the characteristic equation for an ellipse<sup>3</sup> as shown in Fig. (5.9)

---

<sup>3</sup> $(x'/\zeta_x)^2 + (y'/\zeta_{\Delta x})^2 + (z'/\zeta_{\Delta^2 x})^2 > 1$  indicates a point that lies outside of the ellipsoid where primes indicate the rotated coordinate system.

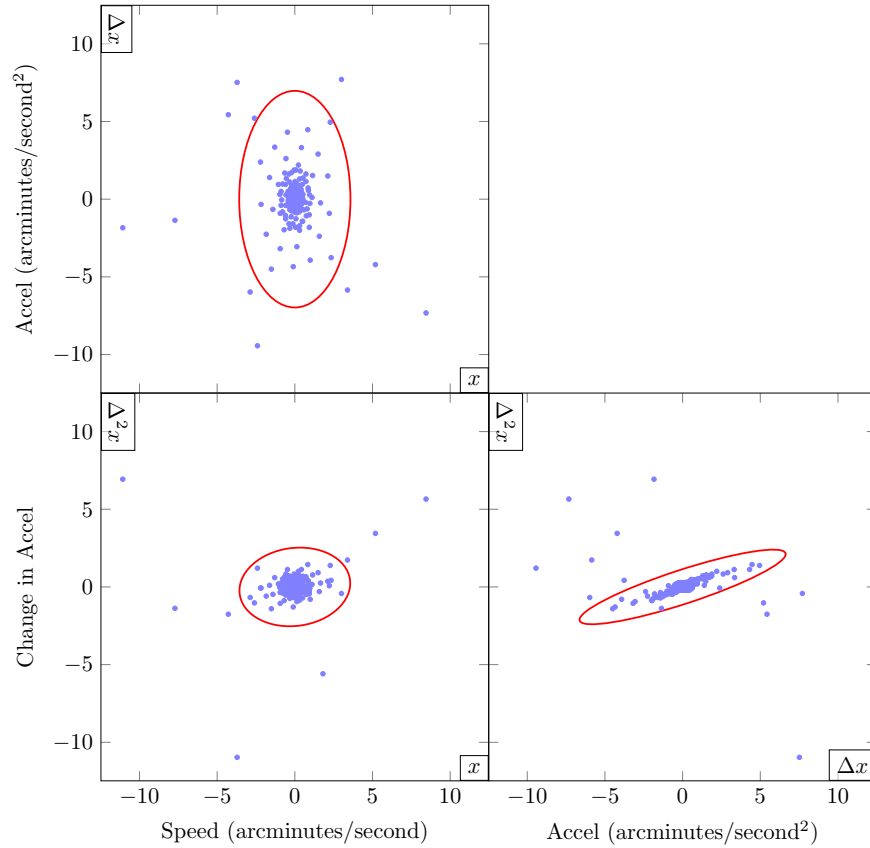


Figure 5.9: The three slices of phase-space taken at each axis show the projection of measurements from a segment of Gryoscope 2 data, as well as its first and second fixed-step derivatives (Note: axes are labeled on their interiors). The ellipses mark the maximal expected region of phase space, given the number of samples and variance of the sample set. Clearly visible are outlying spikes that will be marked for filtering. The actual filtering process occurs in three-dimensional space.

After all potential spikes have been identified, they are removed from the calculation of  $\zeta$  and the process is iterated until no further spikes are removed.

### 5.3.2 Spike Verification

Once a potential spike has been identified using this method, we verify that the anomaly is not observed by both gyroscope boxes. Since physical motion is recorded by both gyroscope boxes, this is a reliable positive test for the spike. We note that this test cannot, however, rule out that the spike is non-physical as there may be correlated

electrical events that effect both systems. While, in practice, we observe adequate discrimination measured by the particle filter reconstruction performance later in this chapter, this remains preliminary and upon further analysis, the verification step described here may be modified or eliminated for some sections of data.

The most likely method of modification would be an increased stringency in determining whether a spike exists in both timestreams. Currently, a spike is considered non-physical if the difference between the filtered data streams at the relevant time step is greater than 2 standard deviations.

### 5.3.3 Spike Removal

The removal step is the most critical of the three for the simple reason that poorly chosen replacement values can skew the attitude solution as much as, if not more than, the original spike. Previous works[102, 43] have suggested a number of alternatives such as extrapolation ( $x_i = x_{i-1}$ ), linear interpolation ( $x_i = [x_{i+1} + x_{i-1}]/2$ ), multi-point polynomial splines and Kalman filters.

In the EBEX LDB data, we observe a broad range of characteristics in its gyroscope readout data, depending on the motion of the gondola at the time and the relative orientation of the gyroscope in question. As such, we find that the performance of any single spike removal algorithm varies with application case and induces unwanted behavior in a subset of cases.

To avoid this problem while maintaining a consistent metric for removal, we apply the ensemble learning method of *Adaptive Boosting*[38] to four models for spike removal: Linear interpolation, 8-point cubic spline, 8-point B-spline and an autoregressive moving average (ARMA) model.

In the simplest sense, adaptive boosting solves the same problem with multiple algorithms and creates a weighted average of the proposed solutions. To generate the weights, we first train our adaptive boosting model on a subset of data that is proximate to the spike being replaced. In this manner, we ensure that the weights

reflect the local characteristics of the data stream.

If we denote the data vector around (but not include) the spike as  $\mathbf{x}$ , then our goal is to determine the  $\mathbf{y}$  vector of values that will replace the spike such that our loss function  $\Psi(\mathbf{y}|\tilde{F}(\mathbf{x}))$  is minimized. In our case, the loss function is modeled by a gaussian error over the length of the data masked by the spike identification. Given the 4 component filter population, our function will be

$$\tilde{F}(\mathbf{x}) = \sum_{m=1}^4 \alpha_m f_m(\mathbf{x}) \quad (5.10)$$

where  $f_m$  represents a single filter with corresponding weight  $\alpha_m$ .

The process begins with  $\alpha_m = 1/M = 1/4$  and then proceeds to iteratively update the weights until our loss function is minimized. The full number of steps is chosen such that our training set is at least 10 times the number of consecutive missing samples. These steps are distributed on each side of the gap. The weighting algorithm is detailed in Alg. (A.7) and we give a brief overview here.

1. Begin by assigning uniform weights to all of the constituent filters.
2. Select the training segment of data and set aside a segment of data, equal in length to that of the spike to be replaced.
3. Allow each filter to approximate the missing data and compare its prediction with the true values. Assign a weighted error value based on the filter's weight and average gaussian error.
4. Average all errors in quadrature and compare against a stopping condition. If total error is sufficiently small, return the weights as the solutions.
5. Calculate the log-likelihood of the error for each filter.
6. Update the linear weights  $\alpha_m$  of each filter and normalize the total weight to 1. Iterate back to step 2.

Once the weights have been assigned for the relevant section of data, the algorithms are applied to the spike and the final replacement calculated by Eq. (5.10). The effects of spike identification and removal are shown in Fig. (5.10) for three classes of spikes: a single bounce, a high-amplitude, narrow spike and a low-amplitude, broader spike. The three cases are selected to show the differences between individual replacement methods. In the case of the single bounce, the B-spline leaves much of the original spike in place due to edge constraints. In the case of the narrow peak, all methods perform comparably. In the case of the broad peak, the ARMA filter injects higher-amplitude variability at high frequencies than the combined filter.

We emphasize that these qualitative assessments do not, by themselves, indicate performance characteristics. Rather they are post-facto characterizations based on the quantitative performance of the full particle filter on the despiked gyroscope timestreams.

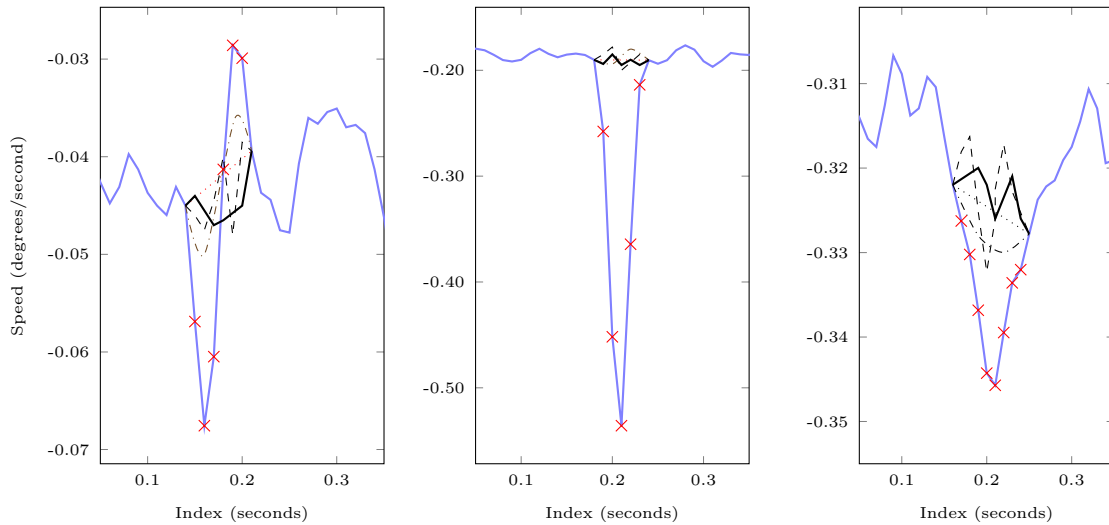


Figure 5.10: Examples of spike identification and removal. The thick, blue line ( $\bullet$ ) shows the raw gyroscope reading data. Each of the  $\times$  marks denotes a reading identified for removal by the algorithm of §5.3.1. Overplotted are four replacement algorithms: linear interpolation ( $\cdots$ ), B-spline ( $- \cdot -$ ), ARMA ( $- - -$ ) and the boosted combination used in this analysis ( $-$ ).

To assess the despiking performance in the context of the particle filter, we select

Algorithm	Particle Filter Convergence Time	Final Reconstruction Error
Linear	238 minutes	1.2
B-Spline	220 minutes	1.3
Cubic Spline	225 minutes	1.2
ARMA	215 minutes	0.95
Boosted	212 minutes	0.9

Table 5.1: Comparison of the effectiveness of 5 spike replacement methods as measured by the induced error on the propagated gyroscope measurement. The propagated measurement is compared against a star camera reading to measure accuracy. The units of reconstruction error are the average scaling from an idealized-noise case. As our scan lengths increase, our expected reconstruction errors increase. We normalize to the  $40''/\text{s}$  integrated error level, providing a consistent measure across varying scan lengths. This will be shown in Fig. (5.12) as a red line for the final reconstruction.

a segment of data, approximately 5 hours long that has 817 spike events identified by the identification routine described in §5.3.1. The four spike replacement routines are then applied individually to the data before reconstructing with the particle filter four separate times. Finally, the boosted combination is applied and the reconstruction is performed again. The results of this process are shown in Tab. (5.1).

## 5.4 Particle Filter Application

We now apply the particle filter from Alg. (A.5) to the resolved star camera solutions and de-spiked gyroscope timestreams from the LDB balloon flight. As was performed using the simulated data, this process is iterative as the filter converges on its parameters. Unlike with the simulated data, we do not have access to the underlying, true components and thus will be limited in our ability to judge the absolute accuracy of the attitude solution. However, we do have access to two independent star cameras that we can leverage for consistency checks on both parameter convergence and absolute attitude reconstruction.



### 5.4.1 Parameter Verification

Next, we verify the performance of the filter in reconstructing parameters that characterize the EBEX system. These are listed in Tab. (4.1) for a single gyroscope box and single star camera. They are duplicated for the second star camera and gyroscope box and include the addition of a  $T_{gy2,ref} = (\psi, \theta, \phi)$  rotation triplet that rotates the second gyroscope box into the reference frame.

Since both SC1 and SC2 have independent rotations to the reference frame ( $T_{sc1,ref}$  and  $T_{sc2,ref}$ ), we can combine the two rotations to generate a rotation from SC1 to SC2 as

$$T_{sc1,sc2} = T_{sc1,ref} \otimes T_{sc2,ref}^{-1} \quad (5.11)$$

This is then checked against the value calculated by comparing concurrent solutions from the two star cameras. During the LDB flight, 19,127 images were taken concurrently by both star cameras. Of those, 9,520 are currently solved for attitude solutions. Using these concurrent attitude solutions, we are able to compare the particle filter's converged values for  $T_{sc1,ref}$  and  $T_{sc2,ref}$  with the distribution of calculated rotations. This is shown in Fig. (5.11).

It should be noted here that there are multiple ways of constraining the offset of the star cameras. The first, and easiest, way is to make independent measurements with each star camera and compare them. This method requires a transformation by the roll component of each star camera solution independently. As the roll component is the least well-constrained axis of our star camera measurement (cf. Eq. (4.73) and Eq. (4.77)), the solution distribution is limited by this uncertainty. The result of this method is shown as the  $\bullet$  marks in Fig. (5.11). Note that this results in a dual-gaussian distribution as the roll uncertainties from both star cameras are represented in the final solution.

The second method is to take the rotation between star cameras as fixed and iteratively solve for the platform rotation. This is the approach taken by the particle

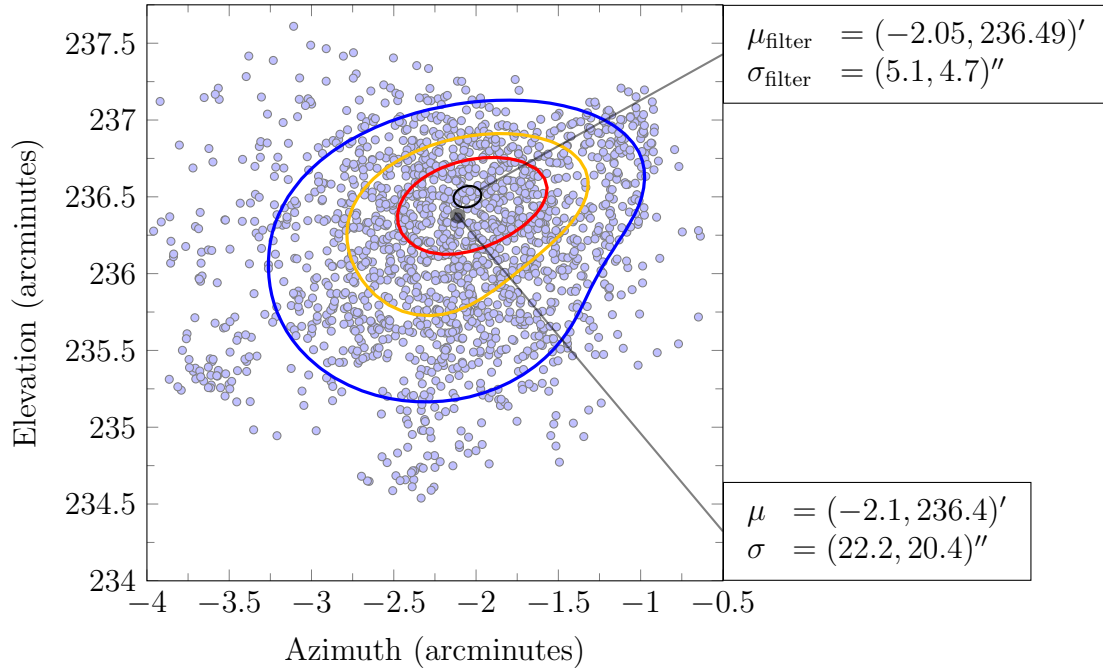


Figure 5.11: Shown are the individual rotation offsets between XSC0 and XSC1. The blue circles each represent a pair of star camera images, taken simultaneously, from which we calculate an offset in the reference frame. The contour lines represent 1, 2 and 3 $\sigma$  of the star camera offset distribution. The interior, black ellipse is 1 $\sigma$  on the particle filter solution for the same offset. The callout boxes show the mean and uncertainty for each method. The particle filter solution is shown in the top box while the image-based solution is in the bottom box.

filter. This method allows us to more tightly constrain the uncertainty on the roll axis as the star cameras jointly subtend a larger section of the sky. Taking the mean value from Fig. (5.11) of 236', we can convert it to pixels using the approximation of 9.5''/pixel. This gives us an effective focal plane center-to-center distance of 1493 pixels.

We can recalculate Eq. (4.74) and Eq. (4.77) with this new distance and twice the average number of stars matched per image (for two images that do not overlap). Additionally, we can use the average sub-pixel centroiding accuracy from Fig. (5.5)

the LDB star camera images of 0.47 pixels.

$$\begin{aligned}
 \sigma_{\text{roll}} &= \frac{1}{\sqrt{2 \cdot \langle N_{\text{star}} \rangle}} \arctan \left( \frac{\sigma_{\text{centroid}}}{1073.02 \text{pixels}} \right) \\
 &= \frac{1}{\sqrt{2 * 13}} \cdot \arctan \left( \frac{0.15}{1073.02} \right) \\
 &\simeq 17.7 \text{ arcseconds}
 \end{aligned} \tag{5.12}$$

This result is roughly consistent with our observations from Fig. (5.11). The increased accuracy observed in the actual solution is primarily attributable to two improvements on the generalization of Eq. (5.12). First, the filter utilizes not only concurrent star camera attitude solutions but also solutions propagated through the gyroscopes, providing a large effective angular distance over which to constrain rotation. Second, the filter takes image weighting into account, assigning higher weight to those images that more accurately represent the attitude.

### 5.4.2 Attitude Reconstruction

We also have absolute attitude information with an associated uncertainties at each star camera reading. This constrains the uncertainty on our measurement of accuracy to the combined uncertainty of two star camera measurements; those from the beginning and end of a data segment over which we will integrate our gyroscope velocity measurements.

Given the specification noise level of  $40''/s$  of our fiber-optic gyroscopes and our pre-flight measured noise on the gyroscope readout (cf. Fig. (3.9)), we can predict the accumulated uncertainty in a measurement segment by knowing the average uncertainty of the initial star camera reading and the integration time until the next star camera reading.

Additionally, due to the distribution of segment lengths shown in Fig. (5.6), we can map the mean error as a function of scan length as well as the uncertainty on

that mean. This is shown in Fig. (5.12) for each pairing of star camera and gyroscope box. Each combination is able to consistently outperform the theoretical minimum variance that would exist without the contribution of our noise modeling discussed in §4.3.4.

We also note the statistically consistent performance, regardless of sensor pairing, indicating that the filter performance is not sensitive to individual sensor characteristics. Additionally, by mixing combinations of sensors in this fashion, we verify a consistent level of parameter reconstruction as the combination of absolute and relative sensor is changed for each pairing.

After comparing separately, we combine both star cameras and both gyroscopes in our final solution. As the two gyroscope boxes are independent measurements of the same motion, the effect of combining them is to reduce the integrated uncertainty at all times by a factor of  $\sqrt{2}$ . Utilizing both star cameras acts to reduce the average time between star camera readings but does not substantially affect the the same of the plots in Fig. (5.12) as they are calculated against the scan length.

The improvement from incorporation of both gyroscope boxes is shown in Fig. (5.13). Here we integrate the same pair of gyroscope boxes in both examples, leading to roughly the same characteristics, differing only in which solution sets are chosen based on which star camera reading are available. We add the particle filter's posterior probability here as well, shown as a dashed green line. This represents the absolute distribution  $p(x_k|y_{1:k-1})$  as given in Eq. (4.32) for the particle filter's distribution of solutions. Note that the filter's assumed uncertainty is systematically lower than our error as measured by comparison with the subsequent star camera solution by roughly 4-5" at the 80 second interval. This indicates the presence of a small systematic effect, the nature of which we have not yet ascertained.

We can fold this additional systematic uncertainty into our full analysis, noting that the discrepancy, while consistent, exists within the  $1\sigma$  limit of our measured solution uncertainty. Because the effect appears systematic, we will add it to our full

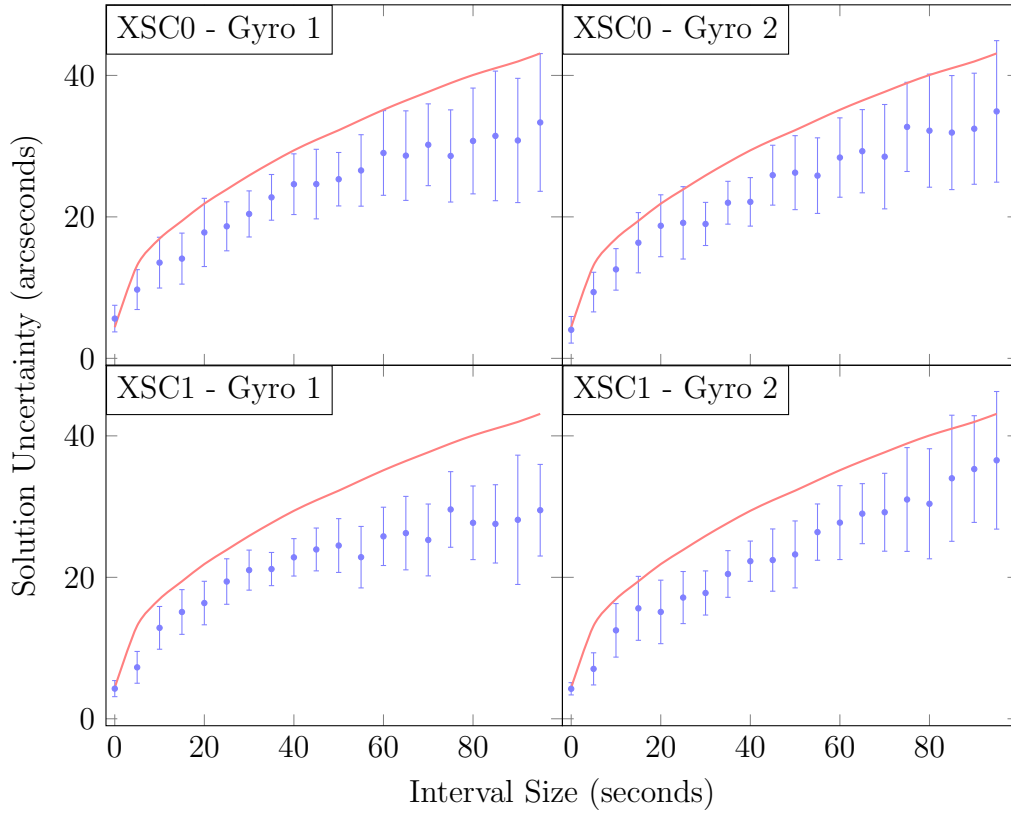


Figure 5.12: For each pairing of gyroscope box and star camera, we calculate the mean absolute attitude uncertainty as a function of the scan length as well as the uncertainty on the mean. The theoretical accumulated mean error is also plotted as the solid, red line. Note that the theoretical error assumes that the gyroscope readout noise is inseparable from true motion. Performance beneath this bound measures the extent to which we have been able to successfully separate the noise from actual motion.

calculation linearly, rather than in quadrature. We can then use the filter’s internal posterior distribution to assign uncertainty weightings to attitude readings, providing a metric by which to assess pointing uncertainty in a complete CMB data analysis pipeline.

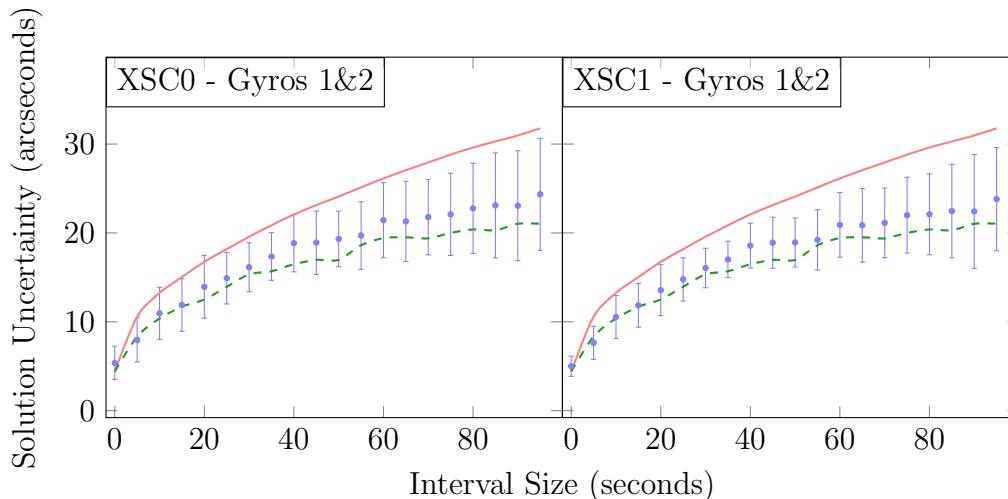


Figure 5.13: Both 3-axis gyroscope boxes are utilized to measure the gondola rotational velocity. *Left:* XSC0’s attitude solutions are used as the absolute constraint at both beginning and end of the scan. *Right:* XSC1’s attitude solutions are used as the absolute constraint at both beginning and end of the scan. Different scan segments are measured due to the star camera’s providing solutions at different times but the solution uncertainty remains roughly consistent due to the shared gyroscopes.

## 5.5 Summary

We have demonstrated the performance of the multiple components in a full attitude reconstruction pipeline against multiple sections of data from the EBEX Long Duration Balloon flight. Individual components appear robust against a variety of inputs and perform at predicted levels based on pre-flight testing.

We have defined a number of independent metrics available in flight for both cleaning and verifying the flight data. These have been used to verify both individual reconstruction components and adjudge the end-to-end performance of our pipeline.

The next steps for the EBEX data will be:

1. Verify this attitude stream against the independent reconstruction pipeline being developed by other members of the EBEX collaboration. Agreement to within the error margins will provide stronger confirmation of the accuracy of our final solution.

2. Extend the attitude solution to the full data set. Currently, 5 of the 11 days worth of data have been processed by the full filter. This is an area of active work at the moment.
3. Calibrate an offset from our gondola pointing reference frame to the microwave beam reference frame. This will be accomplished by comparing attitude solutions from our reconstruction with measurements of RCW38, a known microwave source that was scanned multiple times during the long duration flight.

# Bibliography

- [1] K. N. Abazajian, J. K. Adelman-McCarthy, M. A. Agüeros, S. S. Allam, C. A. Prieto, D. An, K. S. Anderson, S. F. Anderson, J. Annis, N. A. Bahcall, et al. The seventh data release of the sloan digital sky survey. *The Astrophysical Journal Supplement Series*, 182(2):543, 2009.
- [2] V. Acquaviva and C. Baccigalupi. Dark energy records in lensed cosmic microwave background. *Physical Review D*, 74(10):103510, 2006.
- [3] J. W. Alexander, S. Lee, and C.-C. Chen. Pointing and tracking concepts for deep-space missions. In *Optoelectronics' 99-Integrated Optoelectronic Devices*, pages 230–249. International Society for Optics and Photonics, 1999.
- [4] R. A. Alpher and R. Herman. Evolution of the universe. *Nature*, 162:774–775, 1948.
- [5] C. Andrieu, A. Doucet, and V. B. Tadic. On-line parameter estimation in general state-space models. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 332–337. IEEE, 2005.
- [6] P. Astier, J. Guy, N. Regnault, R. Pain, E. Aubourg, D. Balam, S. Basa, R. Carlberg, S. Fabbro, D. Fouchez, et al. The Supernova Legacy Survey: Measurement of  $\Omega_m$ ,  $\Omega_\Lambda$  and  $w$  from the first year data set. *Astronomy and Astrophysics*, 447:31–48, 2006.



- [7] F. Aubin. *Detector readout electronics for EBEX: a balloon-borne cosmic microwave background polarimeter*. PhD thesis, McGill University, 2012.
- [8] J. C. Baker, K. Grainge, M. Hobson, M. E. Jones, R. Kneissl, A. Lasenby, C. O’Sullivan, G. Pooley, G. Rocha, R. Saunders, et al. Detection of cosmic microwave background structure in a second field with the cosmic anisotropy telescope. *Monthly Notices of the Royal Astronomical Society*, 308(4):1173–1178, 1999.
- [9] B. J. Barris, J. L. Tonry, S. Blondin, P. Challis, R. Chornock, A. Clocchiatti, A. V. Filippenko, P. Garnavich, S. T. Holland, S. Jha, et al. Twenty-three high-redshift supernovae from the Institute for Astronomy Deep Survey: Doubling the supernova sample at  $z > 0.7$ . *The Astrophysical Journal*, 602(2):571, 2004.
- [10] N. Bergman, A. Doucet, and N. Gordon. Optimal estimation and cramér-rao bounds for partial non-gaussian state space models. *Annals of the Institute of Statistical Mathematics*, 53(1):97–112, 2001.
- [11] F. Beutler, C. Blake, M. Colless, D. H. Jones, L. Staveley-Smith, L. Campbell, Q. Parker, W. Saunders, and F. Watson. The 6df galaxy survey: baryon acoustic oscillations and the local hubble constant. *Monthly Notices of the Royal Astronomical Society*, 416(4):3017–3032, 2011.
- [12] E. Birgin and J. Martínez. Improving ultimate convergence of an augmented lagrangian method. *Optimization Methods and Software*, 23(2):177–195, 2008.
- [13] J. Bonwick et al. The slab allocator: An object-caching kernel memory allocator. In *USENIX summer*, volume 16. Boston, MA, USA, 1994.
- [14] H. Boterenbrood. CANopen high-level protocol for CAN-bus ver. 3.0, NIKEF. Technical report, NIKHEF, 2000.

- [15] R. H. Brown, S. C. Schneider, and M. G. Mulligan. Analysis of algorithms for velocity estimation from discrete position versus time data. *IEEE Transactions on Industrial Electronics*, 39(1):11–19, 1992.
- [16] F. Campillo and V. Rossi. Convolution particle filter for parameter estimation in general state-space models. *Aerospace and Electronic Systems, IEEE Transactions on*, 45(3):1063–1072, 2009.
- [17] A. Challinor and A. Lewis. The linear power spectrum of observed source number counts. *Phys.Rev.*, D84:043516, 2011. doi: 10.1103/PhysRevD.84.043516.
- [18] C.-H. Chuang, Y. Wang, and M. D. P. Hemantha. A robust distance measurement and dark energy constraints from the spherically averaged correlation function of sloan digital sky survey luminous red galaxies. *Monthly Notices of the Royal Astronomical Society*, 423(2):1474–1484, 2012.
- [19] G. D. Cocco, A. Basili, T. Franceschini, G. Landini, G. Malaguti, G. Palladino, S. Silvestri, L. Gizzi, A. Barbini, and M. Galimberti. HiPeG: A high performance balloon gondola for fine angular resolution x-ray telescopes. *Advances in Space Research*, 37(11):2103 – 2107, 2006.
- [20] B. Coile and S. Hopkins. The ATA over Ethernet Protocol. *Technical Paper from Coraid Inc*, 2005.
- [21] P. Collaboration, P. Ade, N. Aghanim, C. Armitage-Caplan, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. Banday, et al. Planck 2013 results. i. overview of products and scientific results. *arXiv preprint arXiv:1303.5062*, 2013.
- [22] Q. Collaboration, D. Araujo, C. Bischoff, A. Brizius, I. Buder, Y. Chinone, K. Cleary, R. Dumoulin, A. Kusaka, R. Monsalve, et al. Second season quiet observations: Measurements of the cosmic microwave background polarization power spectrum at 95 ghz. *The Astrophysical Journal*, 760(2):145, 2012.

- [23] C. Cotta, M. Sevaux, and K. Sörensen. *Adaptive and Multilevel Metaheuristics*. Studies in Computational Intelligence. Springer, 2008.
- [24] C. Croux and P. J. Rousseeuw. Time-efficient algorithms for two highly robust estimators of scale. *Computational Statistics*, 1:411–428, 1992.
- [25] P. B. Davenport. *A vector approach to the algebra of rotations with applications*. National Aeronautics and Space Administration, 1968.
- [26] P. de Bernardis, P. Ade, J. Bock, J. Bond, J. Borrill, A. Boscaleri, K. Coble, B. Crill, G. De Gasperis, P. Farese, et al. A flat universe from high-resolution maps of the cosmic microwave background radiation. *Nature*, 404(6781):955–959, 2000.
- [27] F. Ding and T. Chen. Combined parameter and output estimation of dual-rate systems using an auxiliary model. *Automatica*, 40(10):1739–1748, 2004.
- [28] D. L. Donoho. De-noising by soft-thresholding. *Information Theory, IEEE Transactions on*, 41(3):613–627, 1995.
- [29] D. L. Donoho and I. M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the american statistical association*, 90(432):1200–1224, 1995.
- [30] A. Doucet, S. Godsill, and C. Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and computing*, 10(3):197–208, 2000.
- [31] G. Efstathiou. An anthropic argument for a cosmological constant. In *Cosmological Constant and the Evolution of the Universe*, volume 1, page 225, 1996.
- [32] D. J. Eisenstein, I. Zehavi, D. W. Hogg, R. Scoccimarro, M. R. Blanton, R. C. Nichol, R. Scranton, H.-J. Seo, M. Tegmark, Z. Zheng, et al. Detection of the baryon acoustic peak in the large-scale correlation function of sdss luminous red galaxies. *The Astrophysical Journal*, 633(2):560, 2005.

- [33] V. Epanechnikov. Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications*, 14(1):153–158, 1969.
- [34] J. L. Farrell, J. C. Stuelpnagel, R. H. Wessner, and J. R. Velman. Problem 65-1: A least squares estimate of satellite attitude. *SIAM Review*, 8(3):pp. 384–386, 1966.
- [35] M. Farsi, K. Ratcliff, and M. Barbosa. An introduction to CANopen. *Computing & Control Engineering Journal*, 10(4):161–168, 1999.
- [36] D. Fixsen, E. Cheng, J. Gales, J. C. Mather, R. Shafer, and E. Wright. The cosmic microwave background spectrum from the full coBE FIRAS data set. *The Astrophysical Journal*, 473(2):576, 1996.
- [37] W. Forst and D. Hoffmann. *Optimization—Theory and Practice*. Springer Undergraduate Texts in Mathematics and Technology. Springer, 2010.
- [38] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.
- [39] A. Friedman. Über die Krümmung des Raumes. *Zeitschrift für Physik*, 10(1):377–386, 1922. ISSN 0044-3328. doi: 10.1007/BF01332580.
- [40] D. Gamerman and H. F. Lopes. *Markov chain Monte Carlo : stochastic simulation for Bayesian inference*. Chapman & Hall/CRC, Boca Raton, 2006.
- [41] G. Gamow. Expanding universe and the origin of elements. *Physical Review*, 70(7-8):572–573, 1946.
- [42] J. Garriga and A. Vilenkin. On likely values of the cosmological constant. *Physical Review D*, 61(8):083502, 2000.

- [43] C. Goodall. A survey of smoothing techniques. In J. Fox and J. Long, editors, *Modern methods of data analysis*, pages 126–176. Sage Publications, Newbury Park, CA, 1990.
- [44] N. J. Gordon, D. J. Salmond, and A. F. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, pages 107–113. IET, 1993.
- [45] D. G. Goring and V. I. Nikora. Despiking acoustic doppler velocimeter data. *Journal of Hydraulic Engineering*, 128(1):117–126, 2002.
- [46] K. M. Gorski, E. Hivon, A. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix: a framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759, 2005.
- [47] J. Gove and D. Hollinger. Application of a dual unscented kalman filter for simultaneous state and parameter estimation in problems of surface-atmosphere exchange. *J. Geophys. Res*, 111:1–21, 2006.
- [48] K. S. Gunderson, C. H. Chen, F. E. Christensen, W. W. Craig, T. A. Decker, C. J. Hailey, F. A. Harrison, R. McLean, R. E. Wurtz, and K. Ziock. Ground performance of the high-energy focusing telescope (heft) attitude control system. In *Optical Science and Technology, SPIE's 48th Annual Meeting*, pages 158–168. International Society for Optics and Photonics, 2004.
- [49] B. Hallgren, H. Boterenbrood, H. Burckhart, and H. Kvedalen. The embedded local monitor board (elmb) in the lhc front-end i/o control system. In *7th Workshop on Electronics for LHC Experiments, Stockholm*, 2001.
- [50] J. Hammersley and D. Handscomb. *Monte Carlo methods*. Methuen's monographs on applied probability and statistics. Methuen, 1964.

- [51] S. Hanany, T. Matsumura, B. Johnson, T. Jones, J. R. Hull, and K. B. Ma. A cosmic microwave background radiation polarimeter using superconducting bearings. *Applied Superconductivity, IEEE Transactions on*, 13(2):2128–2133, 2003.
- [52] D. Hanson, S. Hoover, A. Crites, P. A. R. Ade, K. A. Aird, J. E. Austermann, J. A. Beall, A. N. Bender, B. A. Benson, L. E. Bleem, J. J. Bock, J. E. Carlstrom, C. L. Chang, H. C. Chiang, H. Cho, A. Conley, T. M. Crawford, T. de Haan, M. A. Dobbs, W. Everett, J. Gallicchio, J. Gao, E. M. George, N. W. Halverson, N. Harrington, J. W. Henning, G. C. Hilton, G. P. Holder, W. L. Holzapfel, J. D. Hrubes, N. Huang, J. Hubmayr, K. D. Irwin, R. Keisler, L. Knox, A. T. Lee, E. Leitch, D. Li, C. Liang, D. Luong-Van, G. Marsden, J. J. McMahon, J. Mehl, S. S. Meyer, L. Mocanu, T. E. Montroy, T. Natoli, J. P. Nibarger, V. Novosad, S. Padin, C. Pryke, C. L. Reichardt, J. E. Ruhl, B. R. Saliwanchik, J. T. Sayre, K. K. Schaffer, B. Schulz, G. Smecher, A. A. Stark, K. Story, C. Tucker, K. Vanderlinde, J. D. Vieira, M. P. Viero, G. Wang, V. Yefremenko, O. Zahn, and M. Zemcov. Detection of B-mode Polarization in the Cosmic Microwave Background with Data from the South Pole Telescope. *ArXiv e-prints*, July 2013.
- [53] W. Hu and M. White. A cmb polarization primer. *New Astronomy*, 2(4):323–344, 1997.
- [54] W. Hu, M. M. Hedman, and M. Zaldarriaga. Benchmark parameters for cmb polarization experiments. *Physical Review D*, 67(4):043004, 2003.
- [55] E. Hubble. A relation between distance and radial velocity among extra-galactic nebulae. *Proceedings of the National Academy of Sciences*, 15(3):168–173, 1929.
- [56] Intel Corporation. Intel 64 and IA-32 Architectures Optimization Reference Manual. Document Number: 248966–17, 2008.

- [57] J. Jacod and A. Shiryaev. *Limit theorems for stochastic processes*. Grundlehren der mathematischen Wissenschaften. Springer-Verlag, 1987.
- [58] A. M. Johansen and A. Doucet. A note on auxiliary particle filters. *Statistics & Probability Letters*, 78(12):1498–1504, 2008.
- [59] R. E. Kalman et al. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.
- [60] E. Kennaugh and D. Moffatt. Transient and impulse response approximations. *Proceedings of the IEEE*, 53(8):893–901, 1965.
- [61] J. Klein, A. Aboobaker, P. Ade, F. Aubin, C. Baccigalupi, C. Bao, J. Borrill, D. Chapman, J. Didier, M. Dobbs, et al. A cryogenic half-wave plate polarimeter using a superconducting magnetic bearing. *Proc. SPIE Cryogenic Optical Systems and Instruments XIII*, 8150, 2011.
- [62] D. Knuth. *The Art of Computer Programming: Fundamental algorithms*. The Art of Computer Programming. Addison-Wesley, 2008.
- [63] A. Kong, J. Liu, and W. Wong. Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Association*, Jan 1994.
- [64] J. M. Kovac, E. Leitch, C. Pryke, J. Carlstrom, N. Halverson, and W. Holzapfel. Detection of polarization in the cosmic microwave background using dasi. *Nature*, 420(6917):772–787, 2002.
- [65] E. Ladan-Mozes and N. Shavit. An optimistic approach to lock-free fifo queues. *Distributed Computing*, 20(5):323–341, 2008.
- [66] D. Lang, D. W. Hogg, K. Mierle, M. Blanton, and S. Roweis. Astrometry. net: Blind astrometric calibration of arbitrary astronomical images. *The Astronomical Journal*, 139(5):1782, 2010.

- [67] H. S. Leavitt. 1777 variables in the magellanic clouds. *Annals of Harvard College Observatory*, 60:87–108, 1908.
- [68] G. Lemaître. Un univers homogène de masse constante et de rayon croissant rendant compte de la vitesse radiale des nébuleuses extra-galactiques. *Annales de la Societe Scietifique de Bruxelles*, 47:49–59, 1927.
- [69] G. Lemaître. The beginning of the world from the point of view of quantum theory. *Nature*, 127(3210):706, 1931.
- [70] G. Lemaître. L’univers en expansion. *Annales de la Societe Scietifique de Bruxelles*, 53:51, 1933.
- [71] P. Lesage and T. Ayi. Characterization of frequency stability: analysis of the modified allan variance and properties of its estimate. *Instrumentation and Measurement, IEEE Transactions on*, 33(4):332–336, 1984.
- [72] A. Leśniak, T. Danek, and M. Wojdyła. Application of kalman filter to noise reduction in multichannel data. *Schedae Informaticae*, 17(18):63–73, 2009.
- [73] A. Liddle and D. Lyth. *Cosmological Inflation and Large-Scale Structure*. Cambridge University Press, 2000.
- [74] C. C. Liebe. Accuracy performance of star trackers-a tutorial. *Aerospace and Electronic Systems, IEEE Transactions on*, 38(2):587–599, 2002.
- [75] M. Livio. Lost in translation: Mystery of the missing text solved. *Nature*, 479(7372):171–173, 2011.
- [76] S. Mariani et al. Dual estimation of partially observed nonlinear structural systems: a particle filter approach. *Mechanics Research Communications*, 2012.
- [77] T. Matsumura, S. Hanany, P. Ade, B. R. Johnson, T. J. Jones, P. Jonnalagadda, and G. Savini. Performance of three-and five-stack achromatic half-wave plates at millimeter wavelengths. *Applied optics*, 48(19):3614–3625, 2009.



- [78] S. Maus, S. Macmillan, S. McLean, B. Hamilton, A. Thomson, M. Nair, and C. Rollins. *The US/UK world magnetic model for 2010-2015*. British Geological Survey, 2010.
- [79] L. Mcgee. *Discovery of the Kalman filter as a practical tool for aerospace and industry*. National Aeronautics and Space Administration, 1985.
- [80] A. Melchiorri, P. Ade, P. De Bernardis, J. Bock, J. Borrill, A. Boscaleri, B. Crill, G. De Troia, P. Farese, P. Ferreira, et al. A measurement of  $\Omega$  from the north american test flight of BOOMERanG. *The Astrophysical Journal Letters*, 536(2):L63, 2000.
- [81] M. M. Michael. Hazard pointers: Safe memory reclamation for lock-free objects. *Parallel and Distributed Systems, IEEE Transactions on*, 15(6):491–504, 2004.
- [82] M. M. Michael and M. L. Scott. Simple, fast, and practical non-blocking and blocking concurrent queue algorithms. In *Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing*, pages 267–275. ACM, 1996.
- [83] C. Misner, K. Thorne, and J. Wheeler. *Gravitation*. Gravitation. W. H. Freeman, 1973. ISBN 9780716703440.
- [84] J. Narlikar. *An Introduction to Cosmology*. Cambridge University Press, 2002.
- [85] J. A. Nelder and R. Mead. A simplex method for function minimization. *The computer journal*, 7(4):308–313, 1965.
- [86] E. Pascale. *The Balloon-borne Large Aperture Submillimeter Telescope: BLAST*. PhD thesis, University of Toronto, 2007.
- [87] E. Pascale and A. Boscaleri. Attitude control system for balloon-borne experiments. *AIP Conference Proceedings*, 616(1):56–58, 2002. doi: 10.1063/1.1475603.

- [88] P. J. Peebles and J. Yu. Primeval adiabatic perturbation in an expanding universe. *The Astrophysical Journal*, 162:815, 1970.
- [89] P. J. E. Peebles. Primeval adiabatic perturbations-effect of massive neutrinos. *The Astrophysical Journal*, 258:415–424, 1982.
- [90] A. A. Penzias and R. W. Wilson. A Measurement of Excess Antenna Temperature at 4080 Mc/s. *The Astrophysical Journal*, 142:419–421, July 1965. doi: 10.1086/148307.
- [91] W. J. Percival, B. A. Reid, D. J. Eisenstein, N. A. Bahcall, T. Budavari, J. A. Frieman, M. Fukugita, J. E. Gunn, Ž. Ivezić, G. R. Knapp, et al. Baryon acoustic oscillations in the sloan digital sky survey data release 7 galaxy sample. *Monthly Notices of the Royal Astronomical Society*, 401(4):2148–2168, 2010.
- [92] S. Perlmutter, G. Aldering, G. Goldhaber, R. Knop, P. Nugent, P. Castro, S. Deustua, S. Fabbro, A. Goobar, D. Groom, et al. Measurements of  $\Omega$  and  $\Lambda$  from 42 high-redshift supernovae. *The Astrophysical Journal*, 517(2):565, 1999.
- [93] M. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, Dec 1999.
- [94] Planck Collaboration, P. Ade, N. Aghanim, C. Armitage-Caplan, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. Banday, R. Barreiro, et al. Planck 2013 results. XV. CMB power spectra and likelihood. *arXiv preprint arXiv:1303.5075*, 2013.
- [95] Planck Collaboration, P. Ade, N. Aghanim, C. Armitage-Caplan, M. Arnaud, M. Ashdown, F. Atrio-Barandela, J. Aumont, C. Baccigalupi, A. Banday, et al. Planck 2013 results. xvi. cosmological parameters. *arXiv preprint arXiv:1303.5076*, 2013.

- [96] D. Polsgrove. *Calibration of the E and B EXperiment, A Balloon-Borne Cosmic Microwave Background Polarimeter*. PhD thesis, University of Minnesota, 2009.
- [97] E. Post. Sagnac effect. *Reviews of Modern Physics*, 39(2):475, 1967.
- [98] D. Pozo, L. Illanes, M. Caneo, and M. Cur. Pwv forecast validation at alma site. *Revista Mexicana de Astronomía y Astrofísica*, 41:55, 2011.
- [99] S. Prakash, Y. H. Lee, and T. Johnson. A nonblocking algorithm for shared queues using compare-and-swap. *Computers, IEEE Transactions on*, 43(5):548–559, 1994.
- [100] B. Rabii, C. Winant, J. S. Collins, A. Lee, P. Richards, M. E. Abroe, S. Hanany, B. Johnson, P. Ade, A. Balbi, J. Bock, J. Borrill, R. Stompor, A. Boscaleri, E. Pascale, P. De Bernardis, P. G. Ferreira, V. V. Hristov, A. Lange, A. Jaffe, C. B. Netterfield, G. Smoot, and J. H. P. Wu. Maxima: A balloon-borne cosmic microwave background anisotropy experiment. *Review of Scientific Instruments*, 77(7):071101–071101–25, 2006. ISSN 0034-6748. doi: 10.1063/1.2219723.
- [101] S. Rahnamayan, H. R. Tizhoosh, and M. M. Salama. Opposition-based differential evolution. *Evolutionary Computation, IEEE Transactions on*, 12(1):64–79, 2008.
- [102] M. Razaz and K. Kawanisi. Signal post-processing for acoustic velocimeters: detecting and replacing spikes. *Measurement Science and Technology*, 22(12):125404, 2011.
- [103] B. Reichborn-Kjennerud. *Building and flying the E and B Experiment to measure the polarization of the cosmic microwave background*. PhD thesis, Columbia University, 2010.
- [104] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Detection of interest points us-

- ing symmetry. In *Computer Vision, 1990. Proceedings, Third International Conference on*, pages 62–65. IEEE, 1990.
- [105] A. G. Riess, A. V. Filippenko, P. Challis, A. Clocchiatti, A. Diercks, P. M. Garnavich, R. L. Gilliland, C. J. Hogan, S. Jha, R. P. Kirshner, B. Leibundgut, M. M. Phillips, D. Reiss, B. P. Schmidt, R. A. Schommer, R. C. Smith, J. Spyromilio, C. Stubbs, N. B. Suntzeff, and J. Tonry. Observational evidence from supernovae for an accelerating universe and a cosmological constant. *The Astronomical Journal*, 116(3):1009, 1998.
- [106] A. G. Riess, L. Macri, S. Casertano, H. Lampeitl, H. C. Ferguson, A. V. Filippenko, S. W. Jha, W. Li, and R. Chornock. A 3% solution: determination of the hubble constant with the hubble space telescope and wide field camera 3. *The Astrophysical Journal*, 730(2):119, 2011.
- [107] H. P. Robertson. Kinematics and world-structure. *The Astrophysical Journal*, 82:284, 1935.
- [108] G. Rocha, R. Stompor, K. Ganga, B. Ratra, S. R. Platt, N. Sugiyama, and K. M. Górski. Python i, ii, and iii cosmic microwave background anisotropy measurement constraints on open and flat- $\lambda$  cold dark matter cosmogonies. *The Astrophysical Journal*, 525(1):1, 1999.
- [109] D. Rodriguez-Losada, P. San Segundo, F. Matia, and L. Pedraza. Dual fast-slam: Dual factorization of the particle filter based solution of the simultaneous localization and mapping problem. *Journal of Intelligent and Robotic Systems*, 55(2-3):109–134, 2009.
- [110] P. J. Rousseeuw and C. Croux. Alternatives to the median absolute deviation. *Journal of the American Statistical Association*, 88(424):1273–1283, 1993.
- [111] I. Sagiv. *The EBEX Cryostat and Supporting Electronics*. PhD thesis, University of Minnesota, 2011.

- [112] I. Sagiv, A. M. Aboobaker, P. Ade, F. Aubin, C. Baccigalupi, J. Borrill, D. Chapman, J. Didier, M. Dobbs, W. Grainger, S. Hanany, S. Hilbrand, C. Hogen-Chin, H. Hubmayr, B. Johnson, A. Jaffe, T. Jones, J. Klein, A. Korotkov, S. Leach, A. Lee, L. Levinson, M. Limon, J. Macaluso, K. MacDermid, T. Matsuura, X. Meng, A. Miller, M. Milligan, E. Pascale, D. Polsgrove, N. Pontieu, B. Reichborn-Kjennerud, T. Renbarger, F. Stivoli, R. Stompor, H. Tran, G. Tucker, J. Vinokurov, M. Zaldarriaga, and K. Zilic. The E and B EXperiment (EBEX); Progress and Status. In *American Astronomical Society Meeting Abstracts #213*, volume 41 of *Bulletin of the American Astronomical Society*, pages 357–8, Jan. 2009.
- [113] I. Sagiv, A. M. Aboobaker, C. Bao, S. Hanany, T. Jones, J. Klein, M. Milligan, D. E. Polsgrove, K. Raach, K. Zilic, et al. The ebex cryostat and supporting electronics. *arXiv preprint arXiv:1005.3339*, 2010.
- [114] M. A. Samaan, D. Mortari, and J. L. Junkins. Nondimensional star identification for uncalibrated star cameras. *The Journal of the Astronautical Sciences*, 54(1): 95–111, 2006.
- [115] N. Schneider, J. Urban, and P. Baron. Potential of radiotelescopes for atmospheric line observations: I. observation principles and transmission curves for selected sites. *Planetary and Space Science*, 57(12):1419–1433, 2009.
- [116] M. Seyer. *Complete guide to RS232 and parallel connections*. Prentice Hall, 1988.
- [117] D. L. Shupe, M. Moshir, J. Li, D. Makovoz, R. Narron, and R. N. Hook. The sip convention for representing distortion in fits image headers. In *Astronomical Data Analysis Software and Systems XIV*, volume 347, page 491, 2005.
- [118] W. Smart and R. Green. *Textbook on Spherical Astronomy*. Cambridge University Press, 1977.

- [119] G. Smecher, F. Aubin, E. Bissonnette, M. Dobbs, P. Hyland, and K. MacDermid. A biasing and demodulation system for kilopixel tes bolometer arrays. *Instrumentation and Measurement, IEEE Transactions on*, 61(1):251–260, 2012.
- [120] B. B. Spratling and D. Mortari. A survey on star identification algorithms. *Algorithms*, 2(1):93–107, 2009.
- [121] K. W. Spring. Euler parameters and the use of quaternion algebra in the manipulation of finite rotations: a review. *Mechanism and machine theory*, 21(5):365–373, 1986.
- [122] G. Stewart. *Matrix Algorithms: Volume 1, Basic Decompositions*. Matrix Algorithms. Society for Industrial and Applied Mathematics, 1998.
- [123] R. Storn and K. Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.
- [124] R. A. Sunyaev and Y. B. Zeldovich. Small-scale fluctuations of relic radiation. *Astrophysics and Space Science*, 7(1):3–19, 1970.
- [125] E. Torbet, M. Devlin, W. Dorwart, T. Herbig, A. Miller, M. Nolta, L. Page, J. Puchalla, and H. Tran. A measurement of the angular power spectrum of the microwave background made from the high chilean andes. *The Astrophysical Journal Letters*, 521(2):L79, 1999.
- [126] R. K. Treiber. *Systems programming: Coping with parallelism*. International Business Machines Incorporated, Thomas J. Watson Research Center, 1986.
- [127] Unattributed. ATA-over-Ethernet enables low-cost Linux-oriented SAN. *Linux for Devices*, June 2004.

- [128] J. D. Valois. Implementing lock-free queues. In *Proceedings of the seventh international conference on Parallel and Distributed Computing Systems*, pages 64–69, 1994.
- [129] A. van den Bos. *Parameter Estimation for Scientists and Engineers*. Wiley, 2007.
- [130] Y. Vinokurov. *EBEX, a balloon-borne telescope for observing the polarization of the cosmic microwave background*. PhD thesis, Brown University, 2010.
- [131] G. Wahba. A least squares estimate of satellite attitude. *Siam Review*, 7(3): 409–409, 1965.
- [132] A. G. Walker. On milne’s theory of world-structure. *Proceedings of the London Mathematical Society*, s2-42(1):90–127, 1937. doi: 10.1112/plms/s2-42.1.90.
- [133] E. Wan, R. V. D. Merwe, and A. Nelson. Dual estimation and the unscented transformation. *Advances in Neural Information Processing Systems*, Dec 2000.
- [134] H. Wang, Z. Wu, and S. Rahnamayan. Enhanced opposition-based differential evolution for solving high-dimensional continuous optimization problems. *Soft Computing*, 15(11):2127–2140, 2011.
- [135] S. Weinberg. Anthropic bound on the cosmological constant. *Physical Review Letters*, 59(22):2607, 1987.
- [136] D. V. Wiebe. *BLAST: A Balloon-borne, Large-aperture, Submillimetre Telescope*. PhD thesis, University of Toronto, 2009.
- [137] W. M. Wood-Vasey, G. Miknaitis, C. Stubbs, S. Jha, A. Riess, P. Garnavich, R. Kirshner, C. Aguilera, A. Becker, J. Blackman, et al. Observational constraints on the nature of dark energy: first cosmological results from the essence supernova survey. *The Astrophysical Journal*, 666(2):694, 2007.

- [138] T. Yoshikawa and I. Iida. On finding the axes of skewed symmetry using the image normalization with moment. *Systems and computers in Japan*, 24(2): 45–56, 1993.
- [139] M. Zaldarriaga. The polarization of the cosmic microwave background. In W. Freedman, editor, *Measuring and Modeling the Universe: Volume 2, Carnegie Observatories Astrophysics Series*, volume 2 of *Carnegie Observatories astrophysics series / Carnegie Observatories astrophysics series*. Cambridge University Press, 2004.
- [140] F. Zwicky. Die rotverschiebung von extragalaktischen nebeln. *Helvetica Physica Acta*, 6:110–127, 1933.



# Appendix A

## Algorithm Details

Below, the details of algorithms described in the text are present. These are generic representations of the underlying code as implemented for the EBEX experiment.

### A.1 FIFO Algorithms

---

**Algorithm A.1** Enqueue Item
 

---

```

1: procedure ENQUEUE(FIFO, data)                                ▷ The enqueueing operation
2:   loop
3:     new_node.next  $\leftarrow$  0
4:     new_node.data  $\leftarrow$  data
5:     repeat
6:       sched_yield()                                             ▷ Back off if other processes are waiting
7:       tail.node  $\leftarrow$  FIFO.tail                             ▷ Temporary tail pointer
8:     until CAS(tail.next, 0, new_node)
9:     if tail.node == FIFO.tail then
10:      break                                                     ▷ Success
11:    end if
12:  end loop
13:  FIFO.tail.node  $\leftarrow$  new_node                               ▷ Set the new node as the FIFO tail
14:  FIFO.enq_count  $\leftarrow$  FIFO.enq_count + 1    ▷ Optionally increment enqueue
    counter
15: end procedure

```

---

**Algorithm A.2** Dequeue Item

---

```

1: function DEQUEUE(FIFO)                                ▷ The dequeuing operation
2:   loop
3:     head.node  $\leftarrow$  FIFO.head                        ▷ Temporary head pointer
4:     head.count  $\leftarrow$  FIFO.deq_count                 ▷ Save dequeue counter
5:     next.node  $\leftarrow$  head.next
6:     next.count  $\leftarrow$  FIFO.deq_count + 1
7:     if head.node == FIFO.tail then                      ▷ The queue is empty
8:       if next.node == 0 then
9:         return 0
10:      end if                                              ▷ A node is being inserted, yield for that process
11:    else if next.node  $\neq$  0 then
12:      data  $\leftarrow$  next.node.data
13:      if CAS2(FIFO.head, head, next) then
14:        break                                           ▷ Successfully removed head item
15:      end if
16:    end if
17:    sched_yield()
18:  end loop
19:  return data
20: end function

```

---

## A.2 Kernel Queuing

**Algorithm A.3** Enqueue Buffer

---

```

1: function BUFFER_ENQUEUE(queue)                        ▷ The buffer enqueueing operation
2:   buf  $\leftarrow$  queue.tail                                ▷ Return value
3:   if buf.produced == queue.node_length then           ▷ Buffer chunk is full
4:     if queue.chunks  $\geq$  MAX_CHUNKS then                 ▷ Optional upper limit
5:       buf.produced  $\leftarrow$  buf.consumed
6:       return buf
7:     end if
8:     buf  $\leftarrow$  newnode
9:     queue.tail.next  $\leftarrow$  buf
10:    queue.tail  $\leftarrow$  buf
11:    queue.chunks  $\leftarrow$  queue.chunks + 1
12:  end if
13:  return buf
14: end function

```

---

---

**Algorithm A.4** Dequeue Buffer

---

```

1: function BUFFER_DEQUEUE(queue)           ▷ The buffer dequeuing operation
2:   loop
3:     if queue.head.next == NULL then
4:       return NULL
5:     end if
6:     buf ← queue.head.next                 ▷ Return value
7:     if buf.consumed == queue.node_length then
8:       free(queue.head)
9:       queue.head ← buf
10:    else
11:      if buf.consumed == buf.produced then
12:        return NULL                       ▷ No new data available
13:      end if
14:      return buf
15:    end if
16:  end loop
17: end function

```

---

## A.3 SIR Particle Filter

---

**Algorithm A.5** Sequential Importance Resampling Particle Filter

---

```

1: procedure PARTICLEFILTER
2:    $k \leftarrow 0$ 
3:   for  $i \leftarrow 1, N$  do ▷ Initialize
4:      $x_0^{(i)} \sim p_0(x_0)$ 
5:   end for
6:   repeat
7:      $k \leftarrow k + 1$ 
8:     for  $i \leftarrow 1, N$  do
9:        $x_k^{(i)} \sim p(x_k | \tilde{x}_{0:k-1}^{(i)})$  ▷ Sample from prior distribution
10:       $x_{0:k}^{(i)} \leftarrow (\tilde{x}_{0:k-1}^{(i)}, x_k^{(i)})$ 
11:       $\gamma_k(x_{0:k}^{(i)}) \leftarrow p(x_{0:k} | y_{1:k})$ 
12:       $\alpha_k(x_{0:k}^{(i)}) \leftarrow \frac{\gamma_k(x_{0:k}^{(i)})}{\gamma_{k-1}(x_{0:k-1}^{(i)}) \pi_k(x_k | x_{0:k-1}^{(i)})}$ 
13:       $w_k^{(i)} \leftarrow \tilde{w}_{k-1}^{(i)} \alpha_k(x_{0:k}^{(i)})$  ▷ Compute importance weights
14:    end for
15:     $N_{\text{eff}} \leftarrow \left( \sum_{i=1}^N w_k^{(i)^2} \right)^{-1}$ 
16:    if  $N_{\text{eff}} < N_{\text{th}}$  then
17:      resample
18:    end if
19:  until quit
20: end procedure

```

---

### A.3.1 Sequential Importance Resampling

---

**Algorithm A.6** Sequential Importance Resampling

---

```

1: procedure RESAMPLE
2:    $i \leftarrow 0$ 
3:   repeat
4:      $\mathbf{x}_k^{(i)} \sim p_{\mathcal{K}}(\mathbf{x}_k | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})$  ▷ Sample importance distribution
5:      $w_k^{(i)} \leftarrow w_{k-1}^{(i)} \frac{p(\mathbf{y}_k | \mathbf{x}_k^{(i)}) p(\mathbf{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)})}{p_{\mathcal{K}}(\mathbf{x}_k^{(i)} | \mathbf{x}_{0:k-1}^{(i)}, \mathbf{y}_{1:k})}$ 
6:      $i \leftarrow i + 1$ 
7:   until  $\sum_i w^{(i)} \geq 1$ 
8:   for  $j \leftarrow 1, i$  do
9:      $w^{(j)} \leftarrow \frac{w^{(j)}}{\sum_i w^{(i)}}$  ▷ Normalize weights to unity
10:  end for
11: end procedure

```

---

## A.4 Adaptive Boosting

---

**Algorithm A.7** Boosted Functional Algorithm Weighting
 

---

```

1: procedure BOOST WEIGHTING( $\mathbf{x}$ )
2:   weights  $\alpha_m \leftarrow 1/M$ 
3:   loop
4:     for  $f_m \in F$  do ▷ Evaluate each function over the domain
5:        $\epsilon_m \leftarrow \alpha_m \frac{1}{N} \sum_{n=1}^N 1 - \exp \{ - [y - f_m(x_n)]^2 / Q_x \}$ 
6:     end for
7:      $\epsilon_{\text{tot}} \leftarrow \frac{1}{M-1} \sum_m \epsilon_m^2$ 
8:     if  $|0.5 - \epsilon_{\text{tot}}| \leq E_{\text{lim}}$  then ▷ Where  $E_{\text{lim}}$  is a pre-defined stopping criterion
9:       break
10:    end if
11:     $\beta_m \leftarrow \frac{1}{2} \ln (1 - \epsilon_m) / \epsilon_m$  ▷ Update the log-likelihood based on our error
12:     $\alpha_m \leftarrow \alpha_m \exp (\beta_m \epsilon_m / \alpha_m)$  ▷ Update the linear weights
13:     $\alpha_m \leftarrow \alpha_m / \sum \alpha_m$ 
14:  end loop
15: end procedure

```

---

# Appendix B

## Automation Files

Below we detail some of the telescope automation files used by EBEX during the long-duration balloon flight. A full listing would extend hundreds of pages but the details for general application are contained in samples below.

The first listing is the Document Type Definition file that defines the grammar for all automator files. The following two listings are examples from the cryogenic cooler cycling routine.

---

```

<!ELEMENT schedule (procedures/commands/monitors/conditions/
                    set_variables/variables)*>
<!--ATTLIST schedule version CDATA #FIXED "2"-->
<!ELEMENT procedures (procedure+)>
<!--ELEMENT procedure (emit*,(step/wait/procedure/command/condition/
                    set_variable)+)>
<!--ATTLIST procedure name ID #REQUIRED-->
<!--ATTLIST procedure trigger_on CDATA #IMPLIED-->
<!--ATTLIST procedure trigger_off CDATA #IMPLIED-->
<!--ATTLIST procedure timeout CDATA "-1"-->
<!--ATTLIST procedure loop_limit CDATA "1"-->
<!--ELEMENT wait EMPTY-->
<!--ATTLIST wait seconds CDATA "0"-->
<!--ATTLIST wait minutes CDATA "0"-->
<!--ATTLIST wait hours CDATA "0"-->
<!--ELEMENT step EMPTY-->
<!--ATTLIST step name IDREF #REQUIRED-->
<!--ELEMENT commands (command+)>
<!--ELEMENT command (emit*,function)-->
<!--ATTLIST command name ID #REQUIRED-->
<!--ATTLIST command timeout CDATA "-1"-->
<!--ATTLIST command async (true/false) "false"-->
<!--ELEMENT function (parameter*)-->
<!--ATTLIST function name CDATA #REQUIRED-->
<!--ELEMENT parameter (value/var)-->
<!--ATTLIST parameter name CDATA #IMPLIED-->
<!--ELEMENT conditions (condition+)>
<!--ELEMENT condition (emit*,(boolean/compare))-->
<!--ATTLIST condition name ID #REQUIRED-->
<!--ATTLIST condition timeout CDATA #REQUIRED-->
<!--ATTLIST condition onTimeout (continue/terminate/skip/restart)
                    "continue"-->
<!--ELEMENT variables (variable+)>
<!--ELEMENT variable EMPTY-->
<!--ATTLIST variable name ID #REQUIRED-->
<!--ATTLIST variable initialize CDATA #IMPLIED-->
<!--ELEMENT set_variables (set_variable+)>
<!--ELEMENT set_variable ((var/channel),
                    (arithmetic/value/var/channel))-->
<!--ATTLIST set_variable name ID #REQUIRED-->
<!--ELEMENT emit EMPTY-->
<!--ATTLIST emit signal CDATA #REQUIRED-->
<!--ATTLIST emit emit_condition (onTrue/onFalse/onStart/
                    onComplete/onInterrupt) #IMPLIED-->
<!--ATTLIST emit type (once/continuous/reset) "reset"-->
<!--ELEMENT monitors (monitor+)>
<!--ELEMENT monitor (emit*,(boolean/compare))-->
<!--ATTLIST monitor name ID #REQUIRED-->
<!--ELEMENT boolean ((boolean/compare), (boolean/compare))-->
<!--ATTLIST boolean op (AND/OR/NOT) #REQUIRED-->
<!--ELEMENT compare ((arithmetic/value/var/channel),
                    (arithmetic/value/var/channel))-->
<!--ATTLIST compare op (lt/gt/eq/neq/leq/geq) #REQUIRED-->
<!--ELEMENT arithmetic ((arithmetic/value/var/channel),
                    (arithmetic/value/var/channel))-->
<!--ATTLIST arithmetic op (add/sub/mul/div) #REQUIRED-->
<!--ELEMENT value (#PCDATA)-->
<!--ELEMENT var EMPTY-->
<!--ATTLIST var name IDREF #REQUIRED-->
<!--ELEMENT channel EMPTY-->
<!--ATTLIST channel name CDATA #REQUIRED-->

```

---

Figure B.1: The Document Type Definition file that details the full structure of EBEX automation files. After creation or modification, the automation XML file is verified for grammar against this definition file.

---

```

<procedure name="proc_fridge"
  trigger_on="sig_fridge_on"
  trigger_off="sig_fridge_off">
  <emit signal="sig_fridge_cycle_He8_stop" emit_condition="onInterrupt" />
  <emit signal="sig_fridge_cycle_He10_stop" emit_condition="onInterrupt" />
  <emit signal="sig_end_fridge" emit_condition="onInterrupt" />
  <emit signal="sig_end_fridge" emit_condition="onComplete" />
  <set_variable name="setvar_current_mode_to_fridge">
    <channel name="sched_current_mode" /> <var name="var_mode_fridge" />
  </set_variable>
  <step name="proc_begin_fridge" />
  <step name="cond_wait_for_cycle_procedures_to_start" />
  <step name="cond_wait_for_cycle_procedures_to_finish" />
</procedure>
<procedure name="proc_begin_fridge">
  <emit signal="sig_fridge_cycle_He8_start" emit_condition="onComplete" />
  <emit signal="sig_fridge_cycle_He10_start" emit_condition="onComplete" />
  <step name="cmd_el_pointing_mode_enc_untrimmed" />
  <step name="cmd_antisun_el_goto_43" />
  <step name="cond_wait_for_el_fridge" />
  <step name="cmd_el_super_lock" />
  <condition name="cond_he8_not_running" timeout="5" onTimeout="skip">
    <compare op="eq">
      <channel name="he8_cycle_step"/> <value>0</value>
    </compare>
  </condition>
  <step name="setvar_he8_running" />
  <condition name="cond_he10_not_running" timeout="5" onTimeout="skip">
    <compare op="eq">
      <channel name="he10_cycle_step"/> <value>0</value>
    </compare>
  </condition>
  <step name="setvar_he10_running" />
</procedure>

```

---

Figure B.2: The cryogenic cooler cycle initiation procedure and subprocedure as used in the EBEX LDB flight



---

```

<!-- He8 Cycle Step 1:
  Power cycle the BRUs, open the heat switches,
  run bro_startup and reset dalgman -->
<procedure name="proc_he8_cycle_step1">
  <!-- If the he8_cycle_step channel is not exactly 1, skip this step -->
  <condition name="cond_he8_step1" timeout="5" onTimeout="terminate">
    <compare op="eq">
      <channel name="he8_cycle_step"/><value>1</value>
    </compare>
  </condition>
  <step name="cmd_bro_crate_power_off"/> <wait seconds="5"/>
  <step name="cmd_bro_crate_power_on"/> <wait seconds="5"/>
  <step name="cmd_secondary_hsw_open"/> <wait seconds="5"/>
  <step name="cmd_primary_hsw_open"/> <wait seconds="5"/>
  <step name="cmd_I4_hsw_open"/> <wait seconds="5"/>
  <step name="cmd_I3_hsw_open"/> <wait seconds="5"/>
  <step name="cmd_ultra_hsw_open"/> <wait seconds="60"/>
  <step name="cmd_bro_startup"/> <wait seconds="5"/>
  <step name="cmd_bro_stop_streamers" /> <wait seconds="5"/>
  <step name="cmd_bro_algman_rst"/>
  <!-- Set the he8_cycle_step channel to our next step -->
  <set_variable name="setvar_he8_step_2">
    <channel name="he8_cycle_step"/><value>2</value>
  </set_variable>
</procedure>

```

---

Figure B.3: The first step in the He8 refridgerator cycle as used in the EBEX LDB flight