ONLINE CALIBRATION FOR STAR TRACKERS

by

Brendon Vaz, B.Eng.

Aerospace Engineering Ryerson University, 2011

A thesis presented to Ryerson University

in partial fulfillment of the requirements for the degree of Master of Applied Science in the program of Aerospace Engineering

Toronto, Ontario, Canada. © 2013 by Brendon Vaz

Author's Declaration

I hereby declare that I am the sole author of this thesis.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other intitutions or individuals for the purpose of scholarly reserach.

Abstract

Star trackers are perhaps the most accurate means of measuring a spacecraft's orientation in space and are becoming a popular sensing instrument for attitude determination systems amongst conventional larger satellites as well as micro satellites. In order to produce and maintain high fidelity measurements, the systematic effects of lens distortion and possible sensor alterations due to environmental changes and instrument aging must all be accounted for through calibration, both on the ground and on orbit. In this study, a calibration method is presented to account for errors in star camera parameters, namely the focal length, bore sight offset, higher order radial distortion terms and the tip and tilt of the detector array in relation to the lens arrangement. This method does not depend on a costly high-precision lab setup; instead it simply employs the star camera images and a star catalogue to calibrate the instrument given reasonable initial estimates. This allows for a reduction in pre-mission calibration requirements and is feasible for an online implementation, allowing the star tracker to calibrate itself through out its life-cycle.

Acknowledgements

I would like to express my sincere gratitude to my supervisor Dr. John Enright for putting up with me for the past two years. Without his guidance, enthusiasm and input, (not to mention the stare down after every unproductive week) completing this work would have never been possible.

I would also like to thank the guys in the lab for all their assistance and for reminding me that windows are not everything. Sometimes all you need is a coffee, a pack of cards and a few Nerf guns to make your thesis work.

I would especially like to thank Marce Soto for her assistance with my courses, my work and for being a close and reliable friend. Without your companionship and your help, this thesis would certainly have been written in Notepad (if written at all).

Lastly, I cannot fully express my gratitude to my family. My parents and brother have always been pillars of support in my life. Thank you for everything.

Table of Contents

Author's Declaration I			ſ
Abstract II			I
Ack	Acknowledgements II		
Tab	Table of Contents		
List	List of Tables		
List	of Figur	es VI	I
1 I	Introduct	ion 1	L
1	LI Proble	em Statement	2
1	L2 Prior	WORK	5
1	1.3 Contr	Compared Deviat)
	1.0.1	Outline	{ 7
1	1.0.2		1
1	1.4 Dackg	Principles of Star Tracking	2
	1.4.1 1/1.9	Basic Components))
	1.4.2 1 4 3	Principle of Operation 10)]
	1.4.0 $1 \ 4 \ 4$	Sources of Error 11)
	1.4.4 1 4 5	Typical Calibration Approaches	2
1	L5 Onlin	e Calibration Overview 15	3
-	1.5.1	Attitude Dependent vs Attitude Independent Calibration 14	1
	1.5.2	Unconstrained Nonlinear Optimization	5
	1.5.3	Batch Estimation & Recursive Estimation	7

2	Mat	hematical Framework	18
	2.1	Basic Sensor Geometry	18
		2.1.1 Frames of Reference	18
		2.1.2 Star Camera Inverse Model	20
		2.1.3 Systematic Errors	22
	2.2	Problem Formulation	24
		2.2.1 Optimization Parameters	24
		2.2.2 Cost Function	25
		2.2.3 Forming the Jacobian	26
	2.3	Unconstrained Optimization	31
		2.3.1 Gradient Descent Method:	31
		2.3.2 Gauss Newton Method:	32
		2.3.3 Levenberg Marquardt Method:	33
		2.3.4 BFGS Method:	34
		2.3.5 Recursive Least Squares:	35
3	Test	ting and Results	39
	3.1	Simulated Data	39
	3.2	Ground Test Data	46
		3.2.1 Data Analysis	46
		3.2.2 Algorithm Selection	49
	3.3	Batch Results	51
	3.4	Batch Selection	58
	3.5	Recursive Estimation	63
	3.6	On - Orbit Data	67
4	Con	clusion	79
	4.1	Summary	79
		4.1.1 Simulated Data	80
		4.1.2 Ground Data Tests	80
		4.1.3 Orbital Data Tests	82
	4.2	Future Work	83
	4.3	Concluding Remarks	84
Bi	bliog	raphy	86

List of Tables

2.1	OPTIMIZATION PARAMETERS	25
3.1	SIMULATION RESULTS	43
3.2	NOISE SIMULATION RESULTS	44
3.3	ESTIMATED VARIANCE FROM SUBSETS OF DATA	62
3.4	BATCH VS RECURSIVE RESULTS	65
3.5	SENSOR A CALIBRATION RESULTS	72
3.6	SENSOR B CALIBRATION RESULTS	72

List of Figures

ST-16 star tracker (used in this project) [Enright, 2010].	6
star tracker Schematic [Liebe, 1995].	9
Inertial and star tracker Frame	19
Star Camera Model.	20
Radial Errors	22
Translation Errors.	23
Translation Errors.	24
Co-variance Propagation Through Camera Geometry	36
Measurement Variance vs Centroid Spacing	38
Synthetic Data.	39
Simulation Code Block Diagram.	40
Simulation Result - focal length	41
Simulation Result - Princ. Pt. m_o	41
Simulation Result - Princ. Pt. n_o	41
Simulation Result - Radial Distortion b_1	42
Simulation Result - Radial Distortion b_2	42
Simulation Result - Tilt Correction a_1	42
Simulation Result - Tip Correction a_2	43
Simulation Result - Residual.	43
Spread in Parameter a_2	44
Spread in Parameter a_2	44
Spread in Parameter b_1 .	45
Spread in Parameter b_2 .	45
Spread in Parameter $\overline{m_o}$	45
Spread in Parameter n_o	45
Spread in Parameter f.	45
	ST-16 star tracker (used in this project) [Enright, 2010] star tracker Schematic [Liebe, 1995]

3.18	Residual approx. 10^{-10}
3.19	Number of matched stars in Dataset
3.20	Consecutive Images
3.21	Different Scenes
3.22	Example of a Noisy Cost Function
3.23	Cost $J(f)$
3.24	Gradient $H(f)$
3.25	GD Algorithm
3.26	GN Algorithm
3.27	LM Algorithm
3.28	BFGS Algorithm
3.29	Code Block Diagram for Batch Estimation
3.30	Focal Length f Estimate Using Entire Batch
3.31	Principal Point m_o Estimate Using Entire Batch
3.32	Principal Point n_o Estimate Using Entire Batch
3.33	Radial Distortion b_1 Estimate Using Entire Batch
3.34	Radial Distortion b_2 Estimate Using Entire Batch
3.35	Detector Tilt a_1 Estimate Using Entire Batch
3.36	Detector Tip a_2 Estimate Using Entire Batch
3.37	Residual Using Entire Batch
3.38	Focal Length Estimate f Using Each Image
3.39	Principal Point m_o Using Each Image
3.40	Principal Point n_o Using Each Image
3.41	Radial Distortion b_1 Using Each Image
3.42	Radial Distortion b_2 Using Each Image
3.43	Detector Tilt a_1 Using Each Image. $\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots 57$
3.44	Detector Tip a_2 Using Each Image. $\ldots \ldots \ldots \ldots \ldots \ldots \ldots 57$
3.45	Residual Using Each Image
3.46	Consecutive Images of Same Scene
3.47	Focal Length Variance Vs. Num. Images 60
3.48	Std. Dev. n_0 Vs. Image Spacing. $\ldots \ldots \ldots$
3.49	Std. Dev. b_2 Vs. Image Spacing. $\ldots \ldots \ldots$
3.50	Std. Dev. b_2 Vs. Image Spacing. $\ldots \ldots \ldots$
3.51	Cost function in the space of m_o and a_2
3.52	Focal Length f Using Recursive Estimation
3.53	Principal Point m_o Using Recursive Estimation
3.54	Principal Point n_o Using Recursive Estimation
3.55	Radial Distortion b_1 Using Recursive Estimation 65

3.56	Radial Distortion b_2 Using Recursive Estimation 64	5
3.57	Detector Tip a_1 Using Recursive Estimation	6
3.58	Detector Tilt a_2 Using Recursive Estimation	6
3.59	Residual Using Recursive Estimation	6
3.60	Data - Sensor A.	7
3.61	Data - Sensor B	7
3.62	Focal Lengh(f)	
	Sensor A	8
3.63	Focal Lengh(f)	
	Sensor B	8
3.64	Principal Point (m_o)	
	Sensor A	8
3.65	Principal Point (m_o)	
	Sensor B	8
3.66	Principal Point (n_o)	
	Sensor A	9
3.67	Principal Point (n_o)	
	Sensor B	9
3.68	Radial Correction (b_1)	
	Sensor A	9
3.69	Radial Correction (b_1)	
	Sensor B	9
3.70	Radial Correction (b_2)	
	Sensor A	0
3.71	Radial Correction (b_2)	
	Sensor B	0
3.72	Detector Tip Angle (a_1)	
	Sensor A	0
3.73	Detector Tip Angle (a_1)	
	Sensor B	0
3.74	Detector Tilt Angle (a_2)	
	Sensor A	1
3.75	Detector Tilt Angle (a_2)	
	Sensor B	1
3.76	Cost Function Norm(J)	
	Sensor A	1
3.77	Cost Function $Norm(J)$	
	Sensor B	1

3.78	Attitude Data Before and After Calibration (Sensor A) 7	'4
3.79	Attitude Data Before and After Calibration (Sensor B) 7	'5
3.80	f - Recursive Estimation	<i>'</i> 6
3.81	m_o - Recursive Estimation	<i>'</i> 6
3.82	n_o - Recursive Estimation	<i>'</i> 6
3.83	b_1 - Recursive Estimation	7
3.84	b_2 - Recursive Estimation	7
3.85	a_1 - Recursive Estimation	7
3.86	a_2 - Recursive Estimation	'8

1 Introduction

PACECRAFT have matured a great deal since their inception in the mid twentieth century. The spacecraft of today are moving towards increasing autonomy, and working towards the capability of autonomously transporting payloads, acquiring mission data and even navigating themselves, with the attempt to eliminate any human interaction or ground support. Among the many problems encountered in space, the issue of spacecraft navigation and attitude determination is especially interesting and has been a constantly expanding field of study. Attitude control systems are an integral part of any spacecraft and have encouraged the development of a wide spectrum of orientation sensing devices that operate based on a host of different observations or physical phenomena. Common attitude sensing instruments include sun sensors, star sensors, horizon sensors, magnetometers, gyroscopes, or even Global Positioning System receivers for near earth space missions.

Although star sensors are rather costly in comparison to their counterparts, they certainly stand out in terms of their performance. The fact that a single instrument can provide three-axis attitude information at modestly high output frequency and an accuracy of a few arc-seconds is valuable. However, with high precision attitude sensors comes the added drawback of high precision calibration, which can be a costly and time consuming. In this study, a method of star camera calibration is presented, which employs the use of only images taken from the star camera, and an on-board star catalogue. This calibration comprises the determination of the intrinsic star tracker parameters that define star vectors from imaged co-ordinates. These parameters include the camera focal length, decentering distortions, radial distortions and detector pose with respect to the lens arrangement. Without any dependence on lab equipment, this method is suitable not only for a ground based implementation, but may also be implemented to autonomously calibrate star cameras while in orbit.

1.1 Problem Statement

Many factors may influence the quality of attitude readings produced by a star tracker not only during testing but also through its mission phase. With much effort going into developing and calibrating such a high precision device, it would certainly be problematic if a star tracker behaves differently in its working environment than it does in a pre-mission calibration environment. In addition, regardless of how well-designed and well-equipped high precision lab setups may be, they may never truly replicate the working conditions of a star tracker.

Driving factors affecting star tracker performance such as temperature or launch vibration are beyond control and the effects that they can have on parameters are unpredictable. An effective way to account for decline in star tracker performance, is through continuous re-calibration of the instrument. Having a star tracker continually re-calibrate itself in its working environment, and throughout its working life-time may significantly improve its measurement quality by accounting for dimensional changes caused by thermal expansion or structural warping or from any other source.

To account for dimensional changes, it is required to have a calibration method that is independent of a comfortable lab setup and one which can be implemented on-board the star tracker itself. This study presents a possible solution to this requirement, by attempting to determine the star camera's focal length, decentering distortions, detector tip, tilt and higher order radial distortions. The method uses only images acquired from the camera itself and the on-board star catalogue to revise parameters given reasonable initial estimates. This will allow for an accurate definition of star vectors from imaged co-ordinates, which in turn will increase the quality of attitude data and improve star matching. To reiterate, the objectives of this thesis are to:

- Present a lab-independent calibration method for a star tracker
- Track alterations in star tracker parameters throughout its lifetime
- Maintain the measurement quality of a star tracker

1.2 Prior Work

The study of online calibration for attitude determination systems has been the focus of many research projects in the aerospace field. The increasing precision of mission instruments has stressed the demand for accurate attitude determination during spacecraft operation stages. Attitude determination systems must therefore be able to provide and maintain high accuracy attitude measurements at a high data rate. To ensure this, online calibration techniques have been applied to a range of mission and navigation sensors on-board spacecraft in recent years in order to maintain the functional quality of measurements made throughout the working life of spacecrafts. Given the increase in sensor precision, and the availability of efficient computing means, these methods are becoming the norm for space missions. [Pittelkau 2007]

Misaligned or uncalibrated attitude sensors and gyros may cause large measurement residuals in an on-board attitude determination system. [Pittelkau 2007]. Therefore, attitude sensor calibration is one of the most critical mission support steps, and may be needed while satellites are deployed in orbit as instruments drift with age [Sedlak *et al.* 2003]. Groundbased calibration for attitude sensors, including star trackers has been well documented, [Sun *et al.* 2013], however ground calibrations cannot adequately simulate working conditions of on-board instruments. In addition, it requires the transmission of a large amount of data between satellites and ground stations. Therefore, the technical benefits of on-orbit calibration include more precise calibration, ability to track parameter variations due to thermal variations, less recorder and telemetry data, minimal interruption of science observations, greater autonomy, and less ground testing [Pittelkau 2007]. Reliable and efficient real-time software is required for autonomous on-board calibration. A properly designed calibration routine can reliably estimate attitude and calibration parameters and run in real-time on present-day space qualified computers [Sedlak & Hashmall 2004].

Online calibration methods have been applied in the past to a wide range of sensors. Typically, ground based methods utilize large amounts of data to minimize the residual in a least squares sense between sensor output and a known truth attained from lab equipment. Although this is an effective method on the ground, the dependence of this method on a lab setup and storage and processing of large volumes of data render it unfeasible for an online setting. To combat this, a recursive based estimation method is usually adopted to optimize sensor performance. There are several excellent sources on the subject [Crassidis & Junkins 2011] outlining reliable algorithms that may be adopted to solve the sensor optimization problem. In the past, researchers have used these recursive methods for sensor alignment [Kok-Lam Lai 2003], magnetometer calibration [Kim *et al.* 2004], thruster calibration [Wiktor 1996], gyro calibration [OShaughnessy & Pittelkau 2007], control systems [Chen *et al.* 2006] and a wide variety of different sensing elements.

The availability of powerful space qualified microcomputers and imaging technology has allowed for autonomous star trackers to be used in conjunction with traditional orientation sensors for attitude determination in order to meet these stringent mission requirements [Liebe 1995]. Although star trackers are slowly becoming the preferred attitude sensing device on most spacecraft, there is still much room for improvement, especially to keep up with increasing demands. High precision star trackers are subject to errors just as any other sensor. This error may be caused by not having an ideal sensor model, or an expired calibration of the camera and optics. Errors in the model include inaccurate focal length estimate, inaccurate intersection and angle of boresight with the focal plane and radial distortion corrections [Liebe 2002]. High accuracy star trackers will need to be re-calibrated while in-orbit, using matched star patterns to model their own optics in order to maintain high fidelity attitude information. Continual self calibration will help to correct for dimensional changes over time be they caused by thermal expansion or structural warping. [Enright 2012]. Given that a large amount of information regarding intrinsic camera parameters and distortion coefficients can be extracted from star tracker images and the known locations of matched stars, recursive estimation methods can be effectively

applied to star tracker calibration without the necessity of lab equipment to track post calibration parameter changes.

Perhaps the most influential and possibly the foremost research projects on online star tracker calibration were conducted by Junkins and his research team. In his first paper [Samaan et al. 2001] on the subject, Samaan proposed using the discrepancy between imaged star vectors attained from the star camera, and the matched star vectors from an on-board catalogue to determine the changes in internal camera parameters. This gave rise to the two basic types of calibration the attitude dependent and attitude independent methods. The former utilizes the errors in imaged and catalogued vectors themselves, and the latter using the discrepancy in angles between pairs of vectors from the camera and catalogue. Following up on this research, the team conducted simulations, as well as ground tests to evaluate the performance of each method. Since the attitude dependent method relies on potentially poor attitude information, it is no surprise that the attitude independent method was determined to be superior [Griffith *et al.* 2002]. Singla's evaluation of each methods performance was based on the estimates of the variance that can be encountered in the residuals between star tracker and catalogued data and the attitude estimates attained from the star tracker [Singla et al. 2002]. It is a well-known fact that recursive estimators can behave smugly after large volumes of data have been analyzed. To combat this, Griffith proposed using a modified recursive least squares algorithm, with additive process noise, and a forget factor in order to prevent co-variance wind-up, and to give preference to the most recent available data. This is especially useful when the parameters are expected to slowly drift over time [Griffith & Junkins 2009].

The notable contributions by Junkins sparked several extensions to the concept of online star tracker calibration. Firstly, Junkins separates his estimation in three steps; focal length and principal point estimation using a linearized pinhole camera model, filtering out noise using a Kalman filter, and lastly estimation of higher order focal plane distortions using a localized least squares approach. Although this method may be quite effective, the multiple loops in the algorithm may render it impractical for sensors with a smaller power and computational budget. Some researchers have proposed solving the focal length and principal point using a closed form solution, prior to determining distortion parameters [Pal & Bhat 2009] as opposed

Junkins least squares approach. Liu suggested determining the principal point offset and focal length through a least squares approach and then using the estimates as the measurement inputs for a recursive Kalman filter to eliminate the noise was unnecessary and instead can be done in a single step [Liu *et al.* 2011]. Furthermore, the work done by Shen demonstrates that the estimation of the focal length and higher order radial distortions can be done together as opposed to the method proposed by Junkins [Shen *et al.* 2010]. For a micro or nano-satellite star tracker such as the one used in this project, (See Fig. 1.1 below) these eliminations of multiple steps in the estimation process is key to feasibility.



Figure 1.1 ST-16 star tracker (used in this project) [Enright, 2010].

1.3 Contributions

In this project, the Junkins idea of attitude independent calibration is utilized in conjunction with a detailed camera model instead of the basic pinhole model to estimate focal length, principal point, detector pose and higher order distortions in one single step. The camera model chosen is an inverse formulation of the model proposed by Wang [Wang *et al.* 2008] required to estimate star vectors from illuminated pixels on the camera detector. Wangs model expresses lens distortion as radial distortion plus a transform from ideal image plane to real sensor array plane which effectively captures the net effect of decentering and tangential distortions. The transform is determined by two angular parameters describing the pose of the real sensor array plane with respect to the ideal image plane and two linear parameters locating the real sensor array with respect to the optical axis. Radial distortions are also considered. The fewer parameters to be calibrated, and more explicit physical meaning, simplify the calibrating process and reduce the possibility of attaining local minima. The project outlines results based on synthetic data as well as images captured from the star tracker itself. The telemetry attained from the star tracker was captured both on the ground, and from an in-service star tracker in orbit. Batch and recursive estimation schemes are presented and tested.

1.3.1 Scope of Project

This project aims to present a lab-independent calibration method for a star tracker to track changes in camera model parameters to maintain measurement quality throughout the life of the star tracker. To achieve this, both a batch and recursive algorithm are developed that may be used to calibrate parameters online, and track the changes in these parameters while the star tracker is in orbit.

Several Tests are performed to evaluate the effectiveness of the calibration routines. The tests employ both synthetic data, as well as images attained from the star tracker during ground tests and data obtained from a star tracker in service. By analyzing the results from these tests, factors that influence the effectiveness of the calibrated parameter estimates are explored.

These factors include the number of images required per batch to compute a accurate parameter estimate, as well as the required diversity of scenes necessary to ensure the observability of all parameters. The independence of parameters is also explored. The results from the tests are presented and analyzed, and some notes on implementation are discussed.

1.3.2 Outline

The remainder of Chapter 1 presents a basic summary of star tracking, including the working principle, sources of error and typical calibration ap-

proaches. In addition, an overview on online calibration procedures and function optimization are presented. Chapter 2 outlines the mathematical framework that define the online star tracker calibration routines. This includes a discussion of the star tracker model parameters, the formulation of the cost function and the derivation of the Jacobian matrix. In addition, the batch and recursive function optimization algorithms that are utilized in the calibration procedure are also discussed. Chapter 3 presents and discusses results attained from a number of tests performed to evaluate the effectiveness of the calibration procedure. The final chapter provides some concluding remarks, notes on implementation, and suggested directions of future research.

1.4 Background

1.4.1 Principles of Star Tracking

Stellar navigation is one of the oldest methods used for navigation and heading determination since stars provide a reliable and independent source of information about the observer's pointing direction. Although stellar navigation has been replaced by modern infrastructure like Global Positioning Systems here on earth, the fact that stars can provide valuable and accurate information for navigation regardless of the observer's position has remained unchanged. This fact has been especially exploited in the aerospace industry for navigating spacecrafts that operate beyond the comforts of earth. With the advent of modern computing technology, the manual star navigation methods of old have been completely replaced with fully autonomous star trackers that do not require external computing means for image processing and attitude determination. Star trackers are capable of imaging the night sky, identifying matching stars in an on-board catalog and determining the orientation of the observer with respect to an inertial frame. In essence, they are fully autonomous devices providing three axis attitude information for spacecraft navigation. Star trackers are capable of operating with prior attitude information available (tracking mode) and when no prior attitude information is available (Lost-in-Space mode). Modern star trackers are the only available stand alone devices that provide high resolution attitude measurements in all three axes. It is no surprise that autonomous star trackers are rapidly becoming the preferred attitude determination instruments on board most spacecraft. [Liebe 1995]



Figure 1.2 star tracker Schematic [Liebe, 1995].

1.4.2 Basic Components

A typical star tracking assembly consists of two basic components in addition to other supporting electronics that facilitate input voltage control, on-board memory, and external communications. A schematic illustrating these components is depicted in Fig. 1.2. All of these components are encompassed in a mechanical housing [Liebe 2002]. The two main components include:

- Lens Array and Image Detector Plane
- On-board Processing Unit

The lens array is designed based on the availability of stars being captured in the field of view, and the sensitivity of the detector to image poorly lit stars. At least three stars must be captured in the field of view in order to identify the stars with a high degree of probability and to produce a high fidelity attitude solution. Star sensors typically employ a range of detectors such as CMOS or CCD arrays.

The on-board processor facilitates the readout of images from the image detector, contains enough memory to store the sensor firmware and the star catalogue, and is used to process the images and perform the necessary calculations required in order to produce an attitude solution from the acquired data.

1.4.3 Principle of Operation

The firmware uploaded onto the star tracker plays an important role in the functioning of the sensor and is key to appropriately transforming the measurements acquired from the camera into the attitude information that is desired. The basic routines carried out by the star tracker include centroid determination, geometry calculations, star matching and attitude determination. [Enright John 2010]

• Centroid Determination

When an image is captured by the detector, the firmware on the star tracker discounts any lit pixels that are most likely not caused due to the presence of a star in the field of view and performs a correction to minimize the response from unlit pixels. Based on the focusing of the camera, a single star usually illuminates an array of pixels. The centroid detection routine refines the array of lit pixels to produce a star centroid location to an accuracy better than a single pixel. In most cases, an additional compensation for the motion of the star camera is also taken into account.

• Geometry Calculations

Once the centroid locations of each star are determined on the detector plane, it is necessary to compute a three dimensional unit-vector corresponding to the location of each star in relation to the star sensor. This process of determining the three-dimensional vectors from illuminated centroids requires an inverse model of the star sensor and is a function of the internal camera geometry. Essentially, this calculation employs the focal length and array dimensions to relate detector array co-ordinates to directional vectors. However, due to imperfections, other distortion corrections are also taken into account. Determining these parameters is key to calibrating the camera since the accuracy of the resulting attitude is closely related to the accuracy in relating a star image on the focal plane to a star vector in space. The inverse model will be re-visited in greater detail in the Chapter 2.

• Star Matching

Star matching is perhaps the most critical step in the attitude acquisition procedure. Imaging a variety of stars cannot produce an atti-

tude solution without the ability to determine which stars are captured in the image, and where each of these stars lie in an inertial frame. The star matching routine takes the observed star vectors and identifies the corresponding stars from an on-board catalogue based on the spherical triangles formed by triplets of stars detected in the field of view. Several robust algorithms have been developed for this process [Samaan *et al.* 2003].

• Attitude Determination

The final step conducted by the star tracker firmware is the attitude determination. Using the centroids captured by the star camera, their corresponding vectors mapped through the inverse model, and the matching vectors acquired from the star catalogue, an attitude solution can be determined. This is achieved by employing several robust algorithms that determine the appropriate rotation matrix from two sets of vectors in two frames. Some algorithms that achieve this include the Q-method, FOAM or ESOQ [Crassidis & Junkins 2011].

1.4.4 Sources of Error

Changes in environmental conditions may significantly alter the camera's internal geometry. This includes changes in the focal length, the intersection of the boresight axis with the detector plane, radial distortions and the tip and tilt of the detector plane with respect to the lens array. These changes are likely to occur and their magnitude cannot be predicted. In order to carry out the calibration process it is important to analyze the possible driving factors influencing the behavior of the star camera, how these factors affect the parameters defining the geometry of the camera, and how changes in the parameters may affect the sensed measurement.

All spacecraft equipment is subject to a hostile launch and a harsh space environment that may include vibration, drastic temperature variations as well as solar radiation. Like all spacecraft equipment, star trackers are designed to function with all these adverse effects. However, these factors may pose danger to the functioning of the instrument, and will have a significant impact on its performance affecting the accuracy of attitude measurements. The driving factors affecting camera parameters can either be single-event or ongoing. Examples of single event shifts include launch vibration and vacuum operation, while thermal deformation, and instrument aging can continually affect parameters throughout the star-trackers working period.

Temperature changes that occur in a space environment may alter the optical parameters of the star tracker. Changes in the temperature of an optical system can cause thermal expansion in the glass lens elements or in the metal structure that holds them in place. The index of refraction of the glass itself may also change. Temperature gradients may also cause bending of lens and fastening components in the star tracker. This can cause a significant shift in camera parameters and affect the accuracy of vectors mapped by the star tracker.

High intensity vibrations during launch can disrupt the alignment delicate components. Star trackers in particular can be greatly affected by these large vibrations since they rely on a set of precisely aligned optics. Like vibration, impulsive loading during satellite separation or maneuvers also has the potential to disturb instrument focus and camera geometry.

Other factors that may influence the accuracy and performance of star trackers include radiation, instrument aging, refraction in the lens, and star motion. Although the errors caused from radiation and aging are considerable in high precision star trackers, it is highly unlikely that these sources can cause any dimensional alterations within the camera itself; instead these errors are likely to influence the noise patterns seen in the star camera. Refraction and star motion can both be modeled, however, an online calibration procedure may not be the best approach here.

1.4.5 Typical Calibration Approaches

Star trackers undergo rigorous calibration procedures during their development and testing. Ground-based calibration procedures are either used as an initial starting point for future autonomous calibrations or in most cases, the first and final step in calibration. The typical calibration method for any camera is to acquire images of some object with a well known geometry and structure. This can either be a physical object of in the case of star trackers, the location of a point light source or the geometry of a known star constellation. The ground-based testing of star trackers involves a lab calibration procedure and night sky tests.

• Lab Calibration:

In the lab calibration the use of a single star light simulator and a high precision motion platform is employed. Moving the star sensor through several known orientations will illuminate different pixels on the detector. Scenes of actual star constellations may be simulated. Using the spatial knowledge of the light source and the corresponding lit pixels, a set of best fit camera parameters values can be determined based on a large set of data.

• Night Sky Test:

In night sky tests, the stars imaged by the star sensor must either be manually identified, or identified by some highly robust star ID algorithm that is tolerant of calibration errors. Using a large number of acquired images, and the corresponding inertial star vectors from the catalogue matched to the data, a set of best fit parameter values can be determined to verify the results attained through the lab calibration. Ideally, the lab calibration is sufficient to produce matches and night sky-tests serve only as a validations step.

1.5 Online Calibration Overview

An important problem in spacecraft sensor calibration is the frequently occurring situation in which a previously calibrated instrument behaves differently in its working environment as it did during ground calibration either because it encounters unpredictable magnitudes of changes in its geometry, or due to the inability of the testing environment to efficiently replicate its working conditions [Shen *et al.* 2010]. In addition, a lab calibration procedure makes the assumption that calibration parameters remain constant or remain within allowed tolerances throughout the working life of the instrument; this may not necessarily be the case. All these factors contribute to the necessity of an online calibration.

Online calibration can easily be carried out if accurate locations or angles between imaged points are known. This is exactly the case with star trackers, since the on-board star catalogue contains inertially referenced star-vectors that are known to a high degree of accuracy. Of course it is necessary to assume that the initial estimate of the camera model is sufficient to accurately map the star centroids into star vectors, and identify the matching stars on board the catalogue. By comparing the star vectors from the image and catalogue, two pieces of information are directly available:

- The star vectors themselves are obviously identical if expressed in the same frame
- The angles between two catalogued star vectors expressed in an inertial frame, should be identical to the angle between star vectors determined by the camera

The discrepancy in these values attained from each source is either a product of measurement noise or the inadequate knowledge of the current state of the calibration parameters.

Multiple star vectors or vector pairs from either a single or multiple images can be utilized to generate an over-determined system of equations that can be solved using well established non-linear numerical methods in order to provide revised estimates of calibration parameters. This method of calibration has significant potential for tuning star cameras on-board spacecrafts if its original set of calibration parameters has drifted beyond tolerance due to hardware re-configurations experienced during launch and to compensate for deformations caused due to thermal cycles or material fatigue. Such online star tracker calibration methods have recently become the focal point of many research projects, most notably the work done by Junkins [Samaan *et al.* 2001] and his research team.

1.5.1 Attitude Dependent vs Attitude Independent Calibration

The method in which catalogued and imaged vectors are related to each other is an important question to answer. As mentioned above, the sets of star vectors determined from the catalog or from the star sensor should be identical if they are represented in the same frame. Additionally, the angles between a pair of corresponding vectors remain constant regardless of the frame of reference. Either of these two pieces of information can be utilized to formulate a calibration algorithm. This gives rise to two distinct methods of online calibration, the attitude dependent (ADC) and independent (AIC) approach [Samaan *et al.* 2001].

• Attitude Dependent Calibration

In the attitude dependent approach, the inertial vectors from the catalog can be rotated to the star sensor frame using the current estimate of the attitude matrix. It has been established that if the measurements are ideal, these two sets of vectors should be identical, and any discrepancy in the vectors can be linked to inadequate knowledge of the camera parameters. The advantage of this approach is the smaller amount of computation required in the estimation process. Each vector determined from the camera counts as one data-point, whereas if pairs of vectors are used, each set of m vectors mapped by the camera correspond to p combinations of pairs, where p is the binomial co-efficient of m, and 2. Therefore, the attitude dependent approach provides the same information about camera parameters, with a smaller number of equations to solve. The obvious disadvantage in this approach is that the current attitude estimate itself will be erroneous with a poor estimate of camera parameters. Although this method is computationally efficient compared to attitude independent method, the coupled nature of the parameter estimate with the attitude solution renders this method inferior.

• Attitude Independent Calibration

Inter-star angles remain invariant to an orthogonal transformation between two frames. This fact is exploited in the attitude independent approach. In this method, the attitude estimate is not needed to estimate camera parameters. This is a major advantage over the attitude dependent approach. Regardless of its relatively lower computational efficiency, the attitude independent approach can provide more reliable and accurate estimates of camera parameters when compared to the attitude dependent method [Griffith *et al.* 2002].

1.5.2 Unconstrained Nonlinear Optimization

The basic premise of the unconstrained optimization problem is to minimize a real-valued function of one or more real variables. Several engineering problems can be posed as optimization problems, possibly in the presence of constraints on the decision variables. In fact, an amazing variety of problems including solving systems of nonlinear equations or parameter estimation can be posed as optimization problems. Having appropriately posed an engineering problem mathematically as an optimization problem one can go about attaining the solution to the problem using one of many well established algorithms. With the rapid increase of micro-computers embedded in devices, a rapid growth in embedded optimization has occurred, whereby devices or instruments can optimize themselves all without any human intervention.[Boyd & Vandenberghe 2004]

Measurements are seldom perfect, since there will always be some inherent noise and bias associated with the measurement. This imperfection in measurements becomes increasingly important in high precision sensors and applications. In the interest of attaining high fidelity measurements, the measurement error must be minimized through calibration. The optimization problem here is to find the appropriate sensor model parameter values that minimize the error between the measured value and what is known to be true. In the case of the star tracker, the goal is to estimate the optimal values of the parameters defining the camera's internal geometry such that the difference in the angles between vectors imaged by the star tracker and those defined in the star catalog is minimized. This can be achieved using the concept of least squares for parameter identification.

A least-squares problem is an optimization problem with no constraints, that attempts to minimize the square of the error between a large number of measurements and true values. Several well-established algorithms such as the Gauss-Newton and Levenberg Marquardt method may be used to solve the least squares problem [Crassidis & Junkins 2011]. Least squares can either be used to solve a set of linear equations, or even more importantly, be used iteratively to locally solve a set of nonlinear equations with such high reliability that they may be implemented for embedded optimization routines [Boyd & Vandenberghe 2004]. Thus, it is appropriate to apply this method for an online calibration for a star tracker.

1.5.3 Batch Estimation & Recursive Estimation

The method of least squares optimal estimation has proved to be an extremely powerful technique and has historically been used extensively for sensor parameter identification algorithms. There are many excellent sources on the subject which provide algorithms to solve the least squares problem. These solutions are either batch estimation algorithms or recursive estimation algorithms.

• Batch Estimation

The batch method is employed when a large number of datasets have been accumulated over time. On the one hand, batch methods are easy to understand and implement, and lay the foundation for recursive algorithms. Although the batch least squares methods work remarkably well, the drawback is the need for a large memory capacity to store acquired data and the large amount computation required to attain a solution from large datasets. This is unfeasible for an online implementation.

• Recursive Estimation

The recursive least squares algorithm is just as capable of estimating calibration parameters as the batch method. This approach has the advantage that a large amount of data need not be stored and that it is computationally feasible even for real time applications. The calibration parameters are updated with each dataset, and the degree to which these parameters are known, their covariance, is stored. At each update, the dataset is discarded. The well-known drawback to the recursive method is that once a large number of updates have been performed, the covariance may become small enough to cause smugness in the estimator leading to a divergence in the parameter estimates from their optimal values [Griffith & Junkins 2009].

2 Mathematical Framework

This chapter presents the mathematical foundations required to obtain estimates for focal length, principal point offset and distortion parameters for a star camera. The estimation algorithm is based on an inverse formulation of the camera model presented by Wang [Wang *et al.* 2008]. The chapter first outlines the basic sensor geometry, presents the mathematical formulation of the inverse sensor model and illustrates the effects that changes in camera parameters will have on measurements. Next, the optimization problem is formulated, the objective function is defined. Using this objective, and the analytically derived partial derivatives of the seven camera optimization parameters with respect to the star tracker inverse model, the parameters can be estimated efficiently using a non linear least squares batch or recursive estimator.

2.1 Basic Sensor Geometry

This section presents the basic geometry that allows for a definition of star vectors from imaged centroids. The frames of reference in which measurements are made, and in which the sensor truth are defined. In addition, the errors in centroid locations caused by an inadequate knowledge of camera parameters as observed in measurements are presented.

2.1.1 Frames of Reference

There are two important frames to consider in the calibration of this instrument: the frame in which measurements are made, and the frame in which the true values of the measurements are known. The internal geometry of the star tracker and its measurements are defined in the star tracker (D) frame, while the cataloged inertial vectors that describe star locations in space are expressed in the Earth Centered Inertial (I) frame. Additionally the attitude of the star sensor is determined by finding the best fit rotation from the star tracker frame (D) to the inertial frame (I). Fig. 2.1 illustrates the two frames of reference.



Figure 2.1 Inertial and star tracker Frame.

- star tracker Frame ('D') The star tracker frame is defined by the detector array; its x and y axes correspond to the x and y axes of the detector, while the z axis is the normal to the detector plane coming out the lens of the camera. The tip and tilt of the detector with the lens array are also defined in this frame.
- Inertial Frame ('I') The x-axis of the inertial frame is defined by the vernal equinox, and the z-axis is defined by the north rotational pole of the earth. The y-axis chosen to maintain the right hand rule.

2.1.2 Star Camera Inverse Model

To calculate the star vector, given the chief ray intersection point, an appropriate star tracker inverse model was adopted [Wang *et al.* 2008]. The equations below are an inverse formulation of the model provided by Wang. The inputs to this model comprise of the co-ordinates of each intersection point of the chief ray of each star with the detector array, m_c and n_c attained from the image analysis performed by the star tracker firmware. The output is a vector expressed in the detector frame describing the spatial location of each star captured by the image with respect to the star camera (See Fig. 2.2 below).



Figure 2.2 Star Camera Model.

Given the centroid location on the detector array, m_c and n_c , the uncorrected physical distances between the centroid and the optical axis, u and v, can be determined in terms of the x and y axes of the detector frame.

$$u = \delta x \left(m_c - m_o \right) \tag{2.1}$$

$$v = g_y \,\delta y \,(n_c - n_o) \tag{2.2}$$

The variables δx and δy represent the pixel dimensions. To account for the discrepancy in pixel dimensions in each axis, a scale factor g_y is applied.

Next, the tip, tilt and radial distortion corrections a_1 , a_2 & b_1 , b_2 are considered. Note that the terms a_1 and a_2 are terms representing not simply the physical tip and tilt of the detector plane with the lens system, but also other highly coupled distortions such as tangential and small prism distortion.

Let, U, V be the corrected distances from the boresight axis to the illuminated centroid. They are defined by the expressions:

$$U = \frac{u f}{a_1 v + a_2 u + f}$$
(2.3)

$$V = \frac{v f}{a_1 v + a_2 u + f}$$
(2.4)

To apply the radial correction, the radial distance from the boresight axis to the illuminated pixel ρ , must be defined:

$$\rho = \sqrt{U^2 + V^2} \tag{2.5}$$

The radial correction can be modeled in terms of ρ using the basis polynomial below.

$$B = 1 + b_1 \rho^4 + b_2 \rho^2 \tag{2.6}$$

Once the correction terms have been applied, the vector \mathbf{r} defining the location of the star in the sensor frame can be attained using the following expression.

$$\mathbf{r} = \begin{bmatrix} BU\\BV\\f \end{bmatrix}$$
(2.7)

The unit vector describing the star location \mathbf{s} is therefore given by

$$\mathbf{s} = \frac{\mathbf{r}}{||\mathbf{r}||} \tag{2.8}$$

Calibrating the star camera is simply a matter of finding the optimal values for the focal length, principal point and distortion parameters as implemented in the inverse model. This camera calibration model of lens distortion expresses the lens distortion as radial distortion plus a transformation from the ideal image plane to the real sensor array plane, and the small number of parameters can simplify the calibration process and reduce the possibility of the optimization algorithm being hindered by local minima.

2.1.3 Systematic Errors

The effects that the changes in some of the important parameters defining the internal camera geometry have on the image captured by the camera are illustrated below.

• Focal Length f

Focal Length Errors distort the scale of the image. Pixels illuminated by the same source of light will be pushed radially inward or outward. See Fig. 2.3



Figure 2.3 Radial Errors.

• Principal Point m_o, n_o

The Principle Point is the intersection of the boresight of the camera with the detector array. Errors in the knowledge of this point translate sensed pixels along the appropriate axis. See Fig. 2.4 and 2.5



Figure 2.4 Translation Errors.

• Radial Distortion b_1, b_2

Radial Distortions appear much like focal length errors but are higher-order aberrations and a function of the distance from the principle point. See Fig. 2.3

• Detector Tip, Tilt Angle a_1, a_2

Detector tip and tilt angles appear similar to the translation errors caused by the principle point, however they are caused by a rotation of the detector plane instead of a translation with respect to the lens array. See Fig. 2.4, 2.5

It is important to mention that the illustrations above depict how a ray of light will illuminate different pixels on the detector array based on the camera geometry, i.e. forward model, simply because this is easier to visualize. In the calibration process, an inverse model is used, whereby the illuminated pixels on the detector govern where the ray of light originated from, based on the estimates of the camera's internal geometry and distortion parameters.



Figure 2.5 Translation Errors.

2.2 Problem Formulation

This section outlines the mathematical framework that defines the sensor calibration procedure as an optimization problem. The optimization parameters are defined, in addition to a formulation of the cost function that must be minimized and the analytical derivation of the Jacobian matrix. Some gradient based algorithms used to minimize the objective function are also discussed.

2.2.1 Optimization Parameters

The main parameters that affect the accuracy of the star tracker model are the focal length and the principle point offset. In addition to changes in these parameters, the changes caused by any variation in the distortion parameters in the model may also have a significant effect for high accuracy applications. Since the pixel dimensions are unlikely to change, and will not have significant effect on the imaged vectors, they are ignored in the online calibration procedure. For this procedure, the seven parameters were chosen to optimize are outlined in Table: 2.1.

Description	Symbol
Focal Length	f
Principal axis x-offset	m_o
Principal axis y-offset	n_o
Radial correction	b_1
Radial correction	b_2
Tip correction	a_1
Tilt correction	a_2

Table 2.1 OPTIMIZATION PARAMETERS

$$\mathbf{x} = [f \ m_o \ n_o \ b_1 \ b_2 \ a_1 \ a_2] \tag{2.9}$$

The optimization routine seeks to select appropriate values of these optimization parameters that will minimize the error between the measurements attained from the camera and the known truth from the catalog in a least squares sense.

2.2.2 Cost Function

In order to formulate a non linear least squares estimator, the first requirement is to mathematically define the error function that must be minimized. For this estimator the error was defined as the difference in the cosines of the angles between two star vectors computed from the image, and the cosines of the angles between the corresponding star vectors attained from the Yale Bright Star catalogue. The cosines of the angle were chosen instead of the physical angle for simplicity.

Given two star unit vectors $\mathbf{s_i}$ and $\mathbf{s_j}$, the cosine of the angle between them can be determined using the inner product of the two vectors.

$$\cos\theta_{ij} = \mathbf{s_i}^T \, \mathbf{s_j} \tag{2.10}$$

Suppose the initial parameter estimates are sufficient to identify the stars captured in the image, and each observed star vector determined through the
inverse model from image co-ordinates, can be matched to a corresponding true star vector from the catalog. The error between the angle in a pair of vectors in the image, and the corresponding angle between the same pair of vectors as identified in the catalog can be expressed as

$$\psi_{ij} = (\cos \theta_{ij})_{img} - (\cos \theta_{ij})_{cat} \tag{2.11}$$

If n stars are captured in one image, the number of pairs that can be formed between the set of imaged stars is given by p, where

$$p = \binom{n}{2} = \frac{n!}{2!(n-2)!}$$
(2.12)

Appending all of the errors in cosine angles between each of the p pairs of star vectors, provides the $[p \times 1]$ error vector **e**. The cost function J is simply the one half the square of the vector.

$$\mathbf{e} = \begin{bmatrix} \psi_{ij_1} \\ \psi_{ij_2} \\ \vdots \\ \psi_{ij_p} \end{bmatrix}$$
(2.13)

$$J = \frac{1}{2} \mathbf{e}^T \mathbf{e} \tag{2.14}$$

The next step required to minimize the scalar function J, by appropriately selecting values for the optimization parameters given by \mathbf{x} , is to determine the Jacobian matrix of \mathbf{e} with respect to \mathbf{x} .

2.2.3 Forming the Jacobian

The Jacobian matrix \mathbf{H} is defined as the change in the error with respect to the parameters.

$$\mathbf{H} = \frac{\partial \mathbf{e}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial \psi_{\mathbf{ij}}}{\partial \mathbf{x}} & \mathbf{1} \\ \frac{\partial \psi_{\mathbf{ij}}}{\partial \mathbf{x}} & \mathbf{2} \\ \vdots \\ \frac{\partial \psi_{\mathbf{ij}}}{\partial \mathbf{x}} & p \end{bmatrix}$$
(2.15)

Expanding one row of the Jacobian gives:

$$\frac{\partial \psi_{\mathbf{ij}}}{\partial \mathbf{x}} = \left[\frac{\partial \mathbf{s_i}^T}{\partial \mathbf{x}} \mathbf{s_j} + \frac{\partial \mathbf{s_j}^T}{\partial \mathbf{x}} \mathbf{s_i}\right]_{img}^T - \left[\frac{\partial \mathbf{s_i}^T}{\partial \mathbf{x}} \mathbf{s_j} + \frac{\partial \mathbf{s_j}^T}{\partial \mathbf{x}} \mathbf{s_i}\right]_{cat}^T$$
(2.16)

The subscripts 'img' and 'cat' denote vectors measured by the camera, and acquired from the catalog respectively. Since the angles between the catalogue vectors, are clearly independent of the optimization parameters \mathbf{x} , all partial derivatives are 0. Eliminating the 0 terms, gives

$$\frac{\partial \psi_{\mathbf{ij}}}{\partial \mathbf{x}} = \left[\frac{\partial \mathbf{s_i}^T}{\partial \mathbf{x}} \mathbf{s_j} + \frac{\partial \mathbf{s_j}^T}{\partial \mathbf{x}} \mathbf{s_i} \right]_{img}^T$$
(2.17)

The expression above describes one row of the $[p \times m]$ Jacobian matrix **H**. Appending the rows formed by each pair of star vectors, as done with the error vector, provides the Jacobian for all measurements.

The first step required to form the Jacobian matrix is to determine the partial derivatives of each unit vector with respect to the optimization parameters:

$$\frac{\partial \mathbf{s}}{\partial \mathbf{x}} = \left[\frac{\partial \mathbf{s}}{\partial f} \frac{\partial \mathbf{s}}{\partial m_o} \frac{\partial \mathbf{s}}{\partial n_o} \frac{\partial \mathbf{s}}{\partial b_1} \frac{\partial \mathbf{s}}{\partial b_2} \frac{\partial \mathbf{s}}{\partial a_1} \frac{\partial \mathbf{s}}{\partial a_2} \right]$$
(2.18)

In general, if $\mathbf{s} = \frac{\mathbf{r}(\mathbf{x})}{\|\mathbf{r}(\mathbf{x})\|}$, then the partial derivative of the unit vector $\frac{\partial \mathbf{s}}{\partial \mathbf{x}}$ can be expressed in terms of $\frac{\partial \mathbf{r}}{\partial \mathbf{x}}$ as.

$$\frac{\partial \mathbf{s}}{\partial \mathbf{x}} = \frac{\partial}{\partial \mathbf{x}} \left(\frac{\mathbf{r}}{\sqrt{\mathbf{r}^T \mathbf{r}}} \right)$$
(2.19)

It can be shown that this reduces to:

$$\frac{\partial \mathbf{s}}{\partial \mathbf{x}} = \left(\frac{1}{||\mathbf{r}||}\right) \frac{\partial \mathbf{r}}{\partial \mathbf{x}} - \frac{\mathbf{r}}{||\mathbf{r}^3||} \left(\mathbf{r}^T \frac{\partial \mathbf{r}}{\partial \mathbf{x}}\right)$$
(2.20)

Hence the next step is to determine $\frac{\partial \mathbf{r}}{\partial \mathbf{x}}$. The partial derivatives are derived analytically since a finite difference approach is both inaccurate and computationally unfeasible. The required partials are:

$$\frac{\partial \mathbf{r}}{\partial \mathbf{x}} = \left[\frac{\partial \mathbf{r}}{\partial f} \frac{\partial \mathbf{r}}{\partial m_o} \frac{\partial \mathbf{r}}{\partial n_o} \frac{\partial \mathbf{r}}{\partial b_1} \frac{\partial \mathbf{r}}{\partial b_2} \frac{\partial \mathbf{r}}{\partial a_1} \frac{\partial \mathbf{r}}{\partial a_2} \right]$$
(2.21)

Finding these from the inverse model are fairly straight forward. The partial derivatives of a star vector with respect to each optimization parameter are derived analytically below.

Focal Length f

$$\frac{\partial U}{\partial f} = \frac{u (a_1 v + a_2 u)}{(a_1 v + a_2 u + f)^2}$$
(2.22)

$$\frac{\partial V}{\partial f} = \frac{v (a_1 v + a_2 u)}{(a_1 v + a_2 u + f)^2}$$
(2.23)

$$\frac{\partial \rho}{\partial f} = \frac{U \frac{\partial U}{\partial f} + V \frac{\partial V}{\partial f}}{\rho}$$
(2.24)

$$\frac{\partial B}{\partial f} = 4 b_1 \rho^3 \frac{\partial \rho}{\partial f} + 2 b_2 \rho \frac{\partial \rho}{\partial f}$$
(2.25)

$$\frac{\partial \mathbf{r}}{\partial f} = \begin{bmatrix} \frac{\partial B}{\partial f} U + \frac{\partial U}{\partial f} B\\ \frac{\partial B}{\partial f} V + \frac{\partial V}{\partial f} B\\ 1 \end{bmatrix}$$
(2.26)

Principal Point m_o

$$\frac{\partial u}{\partial m_o} = -\delta x \tag{2.27}$$

$$\frac{\partial U}{\partial m_o} = \frac{f \frac{\partial u}{\partial m_o} (a_1 v + f)}{(a_1 v + a_2 u + f)^2}$$
(2.28)

$$\frac{\partial V}{\partial m_o} = \frac{-a_2 \frac{\partial u}{\partial m_o} v f}{(a_1 v + a_2 u + f)^2}$$
(2.29)

$$\frac{\partial \rho}{\partial m_o} = \frac{U \frac{\partial U}{\partial m_o} + V \frac{\partial V}{\partial m_o}}{\rho}$$
(2.30)

$$\frac{\partial B}{\partial m_o} = 4 b_1 \rho^3 \frac{\partial \rho}{\partial m_o} + 2 b_2 \rho \frac{\partial \rho}{\partial m_o}$$
(2.31)

$$\frac{\partial \mathbf{r}}{\partial m_o} = \begin{bmatrix} \frac{\partial B}{\partial m_o} U + \frac{\partial U}{\partial m_o} B\\ \frac{\partial B}{\partial m_o} V + \frac{\partial V}{\partial m_o} B\\ 0 \end{bmatrix}$$
(2.32)

Principal Point n_o

$$\frac{\partial v}{\partial n_o} = -g_y \,\delta x \tag{2.33}$$

$$\frac{\partial U}{\partial n_o} = \frac{-a_1 \frac{\partial v}{\partial n_o} u f}{(a_1 v + a_2 u + f)^2}$$
(2.34)

$$\frac{\partial V}{\partial n_o} = \frac{f \frac{\partial v}{\partial n_o} (a_2 u + f)}{(a_1 v + a_2 u + f)^2}$$
(2.35)

$$\frac{\partial \rho}{\partial n_o} = \frac{U \frac{\partial U}{\partial n_o} + V \frac{\partial V}{\partial n_o}}{\rho}$$
(2.36)

$$\frac{\partial B}{\partial n_o} = 4 b_1 \rho^3 \frac{\partial \rho}{\partial n_o} + 2 b_2 \rho \frac{\partial \rho}{\partial n_o}$$
(2.37)

$$\frac{\partial \mathbf{r}}{\partial n_o} = \begin{bmatrix} \frac{\partial B}{\partial n_o} U + \frac{\partial U}{\partial n_o} B\\ \frac{\partial B}{\partial n_o} V + \frac{\partial V}{\partial n_o} B\\ 0 \end{bmatrix}$$
(2.38)

Radial Correction b_1

$$\frac{\partial B}{\partial b_1} = \rho^4 \tag{2.39}$$

$$\frac{\partial \mathbf{r}}{\partial b_1} = \begin{bmatrix} \frac{\partial B}{\partial b_1} & U\\ \frac{\partial B}{\partial b_1} & V\\ 0 \end{bmatrix}$$
(2.40)

Radial Correction b_2

$$\frac{\partial B}{\partial b_2} = \rho^2 \tag{2.41}$$

$$\frac{\partial \mathbf{r}}{\partial b_2} = \begin{bmatrix} \frac{\partial B}{\partial b_2} U\\ \frac{\partial B}{\partial b_2} V\\ 0 \end{bmatrix}$$
(2.42)

Distortion Correction a_1

$$\frac{\partial U}{\partial a_1} = \frac{-v \, u \, f}{(a_1 \, v + a_2 \, u + f)^2} \tag{2.43}$$

$$\frac{\partial U}{\partial a_1} = \frac{-v \, v \, f}{(a_1 \, v + a_2 \, u + f)^2} \tag{2.44}$$

$$\frac{\partial \rho}{\partial a_1} = \frac{U \frac{\partial U}{\partial a_1} + V \frac{\partial V}{\partial a_1}}{\rho} \tag{2.45}$$

$$\frac{\partial B}{\partial a_1} = 4 b_1 \rho^3 \frac{\partial \rho}{\partial a_1} + 2 b_2 \rho \frac{\partial \rho}{\partial a_1}$$
(2.46)

$$\frac{\partial \mathbf{r}}{\partial a_1} = \begin{bmatrix} \frac{\partial B}{\partial a_1} U + \frac{\partial U}{\partial a_1} B\\ \frac{\partial B}{\partial a_1} V + \frac{\partial V}{\partial a_1} B\\ 0 \end{bmatrix}$$
(2.47)

Distortion Correction a_2

$$\frac{\partial U}{\partial a_2} = \frac{-u \, u \, f}{(a_1 \, v + a_2 \, u + f)^2} \tag{2.48}$$

$$\frac{\partial U}{\partial a_2} = \frac{-u v f}{(a_1 v + a_2 u + f)^2}$$
(2.49)

$$\frac{\partial \rho}{\partial a_2} = \frac{U \frac{\partial U}{\partial a_2} + V \frac{\partial V}{\partial a_2}}{\rho}$$
(2.50)

$$\frac{\partial B}{\partial a_2} = 4 b_1 \rho^3 \frac{\partial \rho}{\partial a_2} + 2 b_2 \rho \frac{\partial \rho}{\partial a_2}$$
(2.51)

$$\frac{\partial \mathbf{r}}{\partial a_2} = \begin{bmatrix} \frac{\partial B}{\partial a_2} U + \frac{\partial U}{\partial a_2} B\\ \frac{\partial B}{\partial a_2} V + \frac{\partial V}{\partial a_2} B\\ 0 \end{bmatrix}$$
(2.52)

2.3 Unconstrained Optimization

Given a smooth function $J(x) \in \mathbb{R}$ with parameters $x \in \mathbb{R}^m$, the general unconstrained optimization problem has the simple mathematical formulation

$$\underset{\mathbf{x}\in\mathbb{R}^m}{\mininize} : J(\mathbf{x}):\mathbb{R}^m\to\mathbb{R}$$
(2.53)

If x^* is a local minimizer of J, then

$$\nabla J(\mathbf{x}^*) = 0 \tag{2.54}$$

In general, using the information expressed in Eq. (2.54), gradient based algorithms for unconstrained nonlinear optimization generate a sequence of iterates \mathbf{x}_k where,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \delta_{\mathbf{x}} \tag{2.55}$$

with,
$$J(\mathbf{x}_{k+1}) < J(\mathbf{x}_k)$$
 (2.56)

2.3.1 Gradient Descent Method:

The first derivative of a function provides information regarding the local rate of increase. If a minimization of the function is required, then a natural choice for an update to a current estimate of the function variables is simply the negative of the gradient. This is known as the gradient descent method. [Boyd & Vandenberghe 2004]

For a Gradient Descent solver, the update equation is simply

$$\delta_{\mathbf{x}} = -\alpha \nabla J = \mathbf{H}^T \mathbf{e} \tag{2.57}$$

where α is an appropriate step-length, **e** is the error vector and **H** is the Jacobian Matrix.

Naturally the stopping criterion for this method is when the norm of the gradient is sufficiently close to 0. Although this method is easy to implement and contains a minimal number of computations, it is not suitable and may not optimize quickly with ill-conditioned problems.

2.3.2 Gauss Newton Method:

The gradient of the function f can be approximated using a Taylor expansion about the current value of \mathbf{x} .

$$\nabla J(\mathbf{x} + \delta_{\mathbf{x}}) \approx \nabla J(\mathbf{x}) + \nabla^2 J(\mathbf{x})\delta_{\mathbf{x}} = 0$$
(2.58)

Solving for δ_x in (2.58) above yields the Newton Method equation:

$$\delta_{\mathbf{x}} = -\left(\nabla^2 f(\mathbf{x})\right)^{-1} \nabla f(\mathbf{x}) \tag{2.59}$$

Since analytically deriving the second order term, the Hessian matrix, is tedious, and attaining it numerically is inefficient, several approximations may be made to deal with this matter.

Given Measurement Model:

$$\tilde{y} = h(\mathbf{x}) + \mathbf{v} \tag{2.60}$$

and least squares cost function,

$$J = \frac{1}{2} \mathbf{e}^T \mathbf{e} = \frac{1}{2} (\tilde{y} - h(\mathbf{x}))^T (\tilde{y} - h(\mathbf{x}))$$
(2.61)

the gradient of J, is clearly

$$\nabla J = \frac{1}{2} \left(\frac{\partial \mathbf{e}^{T}}{\partial \mathbf{x}} \mathbf{e} + \frac{\partial \mathbf{e}}{\partial \mathbf{x}} \mathbf{e} \right) = \frac{\partial \mathbf{e}^{T}}{\partial \mathbf{x}} \mathbf{e} = -\mathbf{H}^{T} \mathbf{e}$$
(2.62)

In the Gauss Newton (GN) Method [Crassidis & Junkins 2011], the Hessian of the function J is approximated as

$$\nabla^2 f(x) = \mathbf{H}^T \mathbf{H} \tag{2.63}$$

Thus, for a Gauss Newton solver, Eq. 2.59 becomes:

$$\delta_{\mathbf{x}} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{e}$$
(2.64)

2.3.3 Levenberg Marquardt Method:

The Levenberg Marquardt (LM) method is an extension of Gauss-Newton, which combines the ability of gradient methods to converge from a bad initial guess of function parameters with the ability of Gauss-Newton to close in on the converged parameters rapidly and accurately leading to a robust algorithm [Crassidis & Junkins 2011].

The LM update equation is expressed as:

$$\delta_{\mathbf{x}} = (\mathbf{H}^T \ \mathbf{H} + \lambda \ \mathbf{I})^{-1} \ \mathbf{H}^T \mathbf{e}$$
(2.65)

The choice of parameter λ is heuristic, however, it is clear that a large λ will produce an update similar to gradient method, whereas a small λ produces an update similar to Gauss-Newton. Typically, problems start with a large value of λ and the value is reduced as the solution converges.

In some cases, the identity matrix is replaced with a scaling matrix that can adjust the scale of each parameter according to its curvature. The resulting update equation is:

$$\delta_{\mathbf{x}} = (\mathbf{H}^T \mathbf{H} + \lambda \, diag(\mathbf{H}^T \mathbf{H}))^{-1} \, \mathbf{H}^T \mathbf{e}$$
(2.66)

2.3.4 BFGS Method:

Since the computation of the inverse is expensive, several algorithms have been proposed to estimate the inverse of the Hessian matrix required by the newton method. The most prominent method is the Broyden-Fletcher-Goldfarb-Shanno Method. This method employs the use of a inverse Hessian update formula at every iteration that eliminates the computation of an inverse. The update may also converge to a better approximation of the Hessian of a function than the Gauss-Newton Method. [Kelley 1999]

Let the gradient method step be defined as:

$$\mathbf{s}_k = -\alpha \, \nabla J_k \tag{2.67}$$

Where, α is an appropriate scalar step size.

Let the change in gradient of J after update iteration be defined as

$$\mathbf{y}_k = \nabla J_{k+1} - \nabla J_k \tag{2.68}$$

Then, the inverse Hessian update \mathbf{B}^{-1} at step k is given by:

$$\mathbf{B}_{k+1}^{-1} = \mathbf{B}_{k}^{-1} + \frac{(\mathbf{s}_{k}^{T}\mathbf{y}_{k} + \mathbf{y}_{k}^{T}\mathbf{B}_{k}^{-1}\mathbf{y}_{k})(\mathbf{s}_{k}^{T}\mathbf{s}_{k})}{(\mathbf{s}_{k}^{T}\mathbf{y}_{k})^{2}} - \frac{(\mathbf{B}_{k}^{-1}\mathbf{y}_{k}\mathbf{s}_{k}^{T} + \mathbf{s}_{k}\mathbf{y}_{k}^{T}\mathbf{B}_{k}^{-1})}{(\mathbf{s}_{k}^{T}\mathbf{y}_{k})^{2}}$$
(2.69)

The initial Hessian estimate can be set as the identity matrix to force an initial gradient descent step, or may be appropriately set to force an initial Gauss-Newton Step.

Either of these methods can be used to minimize f efficiently. However, this particular problem deals with parameters that have grossly different magnitudes and sensitivities to the cost. A gradient method may be unfeasible here. In addition, the dependence of the BFGS on a an appropriate step size estimation routine such as a backtracking or quadratic line search increases the amount of computation required for this method to be implemented. Therefore the Gauss-Newton algorithm is the most simplest, effective and most appropriate choice.

2.3.5 Recursive Least Squares:

As discussed in Chapter 1, the Recursive Least Squares (RLS) method can be used to update parameters in real time as new data becomes available in a computationally efficient manner [Stengel 1986]. The RLS method is expressed in the equations below.

Given two observations:

$$\tilde{\mathbf{y}}_1 = h(\mathbf{x}_1) + \mathbf{v}_1 \tag{2.70}$$

$$\tilde{\mathbf{y}}_2 = h(\mathbf{x}_2) + \mathbf{v}_2 \tag{2.71}$$

The new state of \mathbf{x}_2 and its covariance, \mathbf{P}_2 , can be calculated based on some prior knowledge of the state, \mathbf{x}_1 and a prior covariance, \mathbf{P}_1 using three equations:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k(\mathbf{e}) \tag{2.72}$$

$$\mathbf{K}_{k} = \mathbf{P}_{k-1}\mathbf{H}_{k}^{T}(\mathbf{H}_{k}\mathbf{P}_{k-1}\mathbf{H}_{k}^{T} + \mathbf{R}_{k})^{-1}$$
(2.73)

$$\mathbf{P}_{k} = (\mathbf{P}_{k-1} + \mathbf{H}_{k}^{T} \mathbf{R}_{k}^{-1} \mathbf{H}_{k})^{-1}$$

$$(2.74)$$

The subscript 'k' denotes the current step, and K_k is the gain matrix. The matrix **R** and **P** are the estimated measurement covariance, and state covariance respectively.

$$\mathbf{R} = \mathbb{E}\left\{\mathbf{e}^{T}\mathbf{e}\right\} = \mathbb{E}\left\{(\tilde{y} - h(\mathbf{x}))^{T}(\tilde{y} - h(\mathbf{x}))\right\}$$
(2.75)

$$\mathbf{P} = \mathbb{E}\left\{ (\mathbf{x}^* - \hat{\mathbf{x}})^T (\mathbf{x}^* - \hat{\mathbf{x}}) \right\}$$
(2.76)

where, $\mathbb{E}{V}$ denotes the Expected Value of V.

Since this is a non-linear implementation, the updates $\hat{\mathbf{x}}_k$ may be iterated until they have converged, or updated once per dataset. An additional advantage of the RLS algorithm apart from its efficiency, is that even a single datapoint, can be used to update the parameters based on the information that it provides. This is useful in a random setting where the number of pairs of stars captured in a star sensor image is unknown.

Covariance Estimation

The initial co-variance of the parameters \mathbf{P}_{o} is typically assumed to be an identity matrix multiplied by a large number. In the case of this problem, the different parameters are several orders of magnitude apart and have different co-variances. A reasonable and conservative initialization of parameter co-variance can be a diagonal matrix of the current estimate of the parameters themselves. The measurement co-variance can be estimated differently. Based on prior knowledge of the star tracker used in this work, the typical standard deviation in the centroid positions σ_c experienced in the detector due to noise is about 0.2 pixels. Based on camera geometry, this variance must be translated to a variance in vector angles, and then the difference in the cosines between two vectors that are mapped using the noisy centroids attained from an image. See figure below.



Figure 2.6 Co-variance Propagation Through Camera Geometry.

Based on the camera geometry, the angular variance in each a vector can be determined using a small angle approximation:

$$\sigma_{\theta}^2 = \frac{2 \gamma}{f} \sigma_c^2 \tag{2.77}$$

where, σ_c^2 is the estimated centroid variance σ_{θ}^2 is the angular variance in each a vector

- fis the nominal focal length
- is the nominal pixel dimension

Next, based the estimated angular variance, the variance in the measurement residual can be determined. Consider the following:

For a perfect measurement

$$\hat{y} = \hat{s_i}^T \hat{s_j} = \cos(\hat{\theta_{ij}}) \tag{2.78}$$

However, a perturbation can be expected in vectors s_i and s_j ,

$$\tilde{\mathbf{s}}_i = \mathbf{s}_i + \delta \mathbf{s}_i \tag{2.79}$$

$$\tilde{\mathbf{s}}_j = \mathbf{s}_j + \delta \mathbf{s}_j \tag{2.80}$$

Therefore, the corrupt measurement can be expressed as

$$\tilde{y} = \cos(\tilde{\theta_{ij}}) \tag{2.81}$$

$$y + \delta y = \cos(\theta_{ij} + \delta \theta_{ij}) \tag{2.82}$$

$$= \cos(\theta_{ij})\cos(\delta\theta_{ij}) - \sin(\theta_{ij})\sin(\delta\theta_{ij})$$
(2.83)

$$= \cos(\theta_{ij}) - \sin(\theta_{ij}) \,\delta\theta_{ij} \tag{2.84}$$

The error caused by noisy vectors can therefore be expressed as

$$\delta y = -\sin(\theta_{ij}) \,\,\delta\theta_{ij} \tag{2.85}$$

The measurement variance can be expressed as

$$\sigma_m^2 = var(\delta y) = \mathbb{E}\left\{\delta y^2\right\} - \mathbb{E}\left\{\delta y\right\}^2$$
(2.86)

(2.87)

Given that the noise is 0 mean,

$$\sigma_{m_{ij}}^2 = var(\delta y) = \mathbb{E}\left\{\delta y^2\right\}$$
(2.88)

$$= \sin^2(\theta_{ij}) \mathbb{E}\left\{\delta\theta_{ij}^2\right\}$$
(2.89)

$$= \sin^2(\theta_{ij}) \sigma_{\theta}^2 \tag{2.90}$$

$$= (1 - (s_i^T s_j)^2) \sigma_{\theta_{ij}}^2$$
 (2.91)

Ignoring the effects of centroid coupling between two pairs for simplicity, the co-variance matrix \mathbf{R} , can be estimated as:

$$\mathbf{R} = Diag(\sigma_{m_{ij}}^2) \tag{2.92}$$

where, Diag is a diagonal matrix in the span of the error vector, and subscripts ij denote angle measurements between imaged vectors i and j respectively. In the case of a batch estimation, where the co-variance update equation (Eq. 2.74) is not used, the measurement co-variance matrix **P** can be estimated using the matrix **R** and the Jacobian **H**

$$\mathbf{P} = (\mathbf{H}^T \ \mathbf{R} \ \mathbf{H})^{-1} \tag{2.93}$$

A test was conducted to verify the formulation of the measurement covariance and its dependence on the spacing between centroids. A synthetic image with two closely spaced centroids was generated. The vectors attained from these centroids were taken as the true vectors. Next white noise with a standard deviation of 0.2 pixels was introduced to the centroids, and vectors attained from the noisy centroids served as noisy measurements. Next, the centroids were shifted further apart from each other until the centroids were at opposite ends of the detector. The discrepancy between cosines of the pair of true vectors and simulated measurements was calculated multiple times per centroid location. The variance in the error vector is illustrated in Fig. 2.7 below.



Figure 2.7 Measurement Variance vs Centroid Spacing.

3 Testing and Results

Chapter 3 outlines the tests performed to evaluate the effectiveness of the calibration procedure. The first test is simply a simulation and utilizes synthetic data to test the working of the calibration routine in ideal conditions and whether this procedure would perform well with noisy measurements. The next group of tests utilize images attained from the star tracker during ground tests and data obtained from a star tracker in service. The results from these tests are analyzed to explore factors that influence the effectiveness of procedure and the accuracy of the calibrated parameter estimates.

3.1 Simulated Data

To test the implementation of the mathematical model and to verify the concept of sensor calibration using least squares optimization, a simulation was performed using synthetic data. For the synthetic data an artificial grid of 8 stars was compiled (See. Fig. 3.1).



Figure 3.1 Synthetic Data.



Figure 3.2 Simulation Code Block Diagram.

Next, a "true" sensor model was generated to simulate true star vectors, and a perturbed sensor model was generated in the same fashion to simulate measurements made by an uncalibrated star tracker. The aim of this experiment was to verify whether the estimation routine was able to use the initial guesses of the parameters from the perturbed model and generate updates to the parameters which converge to the true model under ideal conditions. The Gauss Newton Algorithm was implemented for the simulation. A code schematic is illustrated in the flow diagram above (See Fig. 3.2). The results from the simulation are outlined in Figures 3.3 through 3.10 below.



Figure 3.3 Simulation Result - focal length.



Figure 3.4 Simulation Result - Princ. Pt. m_o .



Figure 3.5 Simulation Result - Princ. Pt. no.

Simulation Results continued.



Figure 3.6 Simulation Result - Radial Distortion b_1 .



Figure 3.7 Simulation Result - Radial Distortion b_2 .



Figure 3.8 Simulation Result - Tilt Correction a_1 .

From the results above, it is evident that the calibration routine was able to estimate the true sensor parameters using the simulated star measurements and the known true star vectors with a relatively small numerical error. In



Figure 3.9 Simulation Result - Tip Correction a₂.



Figure 3.10 Simulation Result - Residual.

Description	Symbol	Estimate	Initial	Numerical Error
Focal Length [m]	f	0.0161296	0.016	4.3×10^{-14}
Detector x-offset [pix]	\tilde{m}_o	939.455	972	9.2×10^{-13}
Detector y-offset [pix]	n_o	1261.578	1296	8.2×10^{-14}
Radial correction	b_1	-2.126×10^7	-1e8	2×10^{-10}
Radial correction	b_2	-996.872	-500	2×10^{-11}
Tip correction	a_1	0.007765	0.00	3.5×10^{-11}
Tilt correction	a_2	-0.01793	0.00	1.9×10^{-12}

Table 3.1 SIMULATION RESULTS

each of the above plots it can be seen that the parameters converged to their true values within a few iterations. With reasonable results attained from the noise free simulation, an angular white noise with a conservative standard deviation of 1×10^{-6} was introduced to the simulated measurements to check estimator performance in the presence of noisy star vector measurements. The simulation test was carried out for 1000 runs to check whether the

true parameters can be recovered, and the variance that can be expected in these estimates. The spread in the parameter estimates from these noisy measurements attained from 25000 test runs are illustrated in the plots below and summarized in Table 3.2.

Description	Symbol	Mean	Std. Dev	Mean Error
Focal Length [m]	f	0.0161299	$5.1775e^{-6}$	2.56×10^{-7}
Detector x-offset [pix]	m_o	939.504	13.091 [pix]	0.04927
Detector y-offset [pix]	n_o	1261.308	13.228 [pix]	0.2702
Radial correction	b_1	-2.28×10^7	$5.741e^{7}$	1.548×10^6
Radial correction	b_2	-988.827	277.39	8.04536
Tip correction	a_1	0.00769	0.00314	7.5226×10^{-5}
Tilt correction	a_2	-0.0179	0.00293	6.6553×10^{-6}

 Table 3.2 NOISE SIMULATION RESULTS



Figure 3.11 Spread in Parameter a_2 . Figure 3.12 Spread in Parameter a_2 .

The results in Table 3.2 indicate that the parameters can be effectively recovered from noisy measurements. In each case, the cost function was minimized to approximately 10^{-10} (See Fig. 3.18. The focal length and principal point offsets were estimated with a low variance while the distortion terms were estimated to a considerably higher variance. This can be expected since distortion parameters have considerably lower sensitivities to the cost function.

 $\mathbf{44}$



Figure 3.13 Spread in Parameter b_1 .



Figure 3.15 Spread in Parameter m_o .



Figure 3.17 Spread in Parameter f.



Figure 3.14 Spread in Parameter b_2 .



Figure 3.16 Spread in Parameter n_o .



Figure 3.18 Residual approx. 10^{-10} .

Nonetheless, the results attained with a low mean error from the noisy simulated data suggest that the approach is feasible for testing using actual hardware. Similar tests were then conducted using images downloaded from an ST16 star tracker and corresponding matched star vectors from the Yale Bright Star Catalog. Both batch and recursive estimation techniques were tested using ground and orbit data.

3.2 Ground Test Data

The ground test data for the following set of test procedures was captured from an ST-16 star over the traverse of a rover test. The test, conducted on earth, consisted of a rover traversing over an uneven terrain. The star tracker implemented on the rover gathered star images and conducted a star matching process throughout the traverse of the rover. The images and matching information from this test were used to calibrate the star tracker's internal parameters. This test can be used to indicate whether the calibration procedure is effective in determining revised parameter estimates from initial lab calibration estimates that minimize the error between true vectors in the on-board star catalogue and measurements made using the star tracker hardware.

3.2.1 Data Analysis

Before optimizing the star tracker parameters it is important to analyze the available data. The larger the dataset the more observable each parameter is, and the less susceptible each estimate is to the effects of noise. The dataset utilized in this project consisted of 20,000 images captured by the ST16 star tracker at a variety of different orientations over the traverse of the rover. It is important to note that since these images were taken on earth it is susceptible to a higher noise level than in space due to refraction from the atmosphere and due to stray light.

The number of stars matched in these images are illustrated in Fig. 3.19 below. At least three stars are required to identify a match, although a four star match significantly reduces the probability of a false matching and large datasets reduce the effects of measurement noise. It can be seen that there are a significant number of useful images that may be utilized for calibration.



Figure 3.19 Number of matched stars in Dataset.

In addition to the length of the data set, the variety of scenes captured in the star images may also affect the quality of parameter estimates. It is important to verify that images with a wide variety of centroid positions, possibly covering the entire frame of the detector are available. Images capturing significantly different star patterns are likely to increase the capability to observe each parameter.

When performing a batch optimization, a appropriate choice for a dataset is to use all the available data. Using all images in a dataset will ensure that significantly different scenes are available, and ensure the observability of all parameters, in addition to minimizing the influence of measurement noise on the estimates. However for an online implementation, larger datasets equates to lower computational efficiency. For an online implementation, saving large batches of centroids and matched vectors on-board is unfeasible and hence batches must be strategically selected.

Large batches of data will reduce the effects of noise, and large variety of scenes in captured images will increase the observability (See Fig. 3.20 and 3.21. Therefore, a good balance between batch size and batch diversity must be selected such that parameters may be determined with as high a fidelity

as possible without involving the entire batch in the calibration procedure. The appropriate selection of images is discussed further in the section 3.4.



Figure 3.20 Consecutive Images.

Figure 3.21 Different Scenes.

It is also important to ensure that the measurements are not noisy enough to cause objective function to be noisy as well (See Example in Fig. 3.22). This will significantly reduce the ability of the optimization methods outlined in Chap. 2



Figure 3.22 Example of a Noisy Cost Function.

To ensure that the objective function has been adequately formulated and the gradient methods outlined in Chapter 2 may optimize the function effectively, the cost is can be plotted in the space of each parameter. In addition, the analytically derived gradients can then be compared to a finite difference calculation to ensure its accuracy. This process was done for each parameter using several images for verification. As an example, figure 3.23 and 3.24 below illustrates the objective in the space of the focal length, and the analytically and numerically calculated gradient. This serves as an important debugging step.



Figure 3.23 Cost J(f).

Figure 3.24 Gradient H(f).

A similar check was performed for each parameter, and expected results were attained.

3.2.2 Algorithm Selection

In this section, the four candidates for function optimization are explored. These include the Gradient descent method (GD), the Gauss-Newton method (GN), the Levenberg-Marquardt method (LM), and the Broyden-Fletcher-Goldfarb-Shanno method (BFGS). The tests conducted in this section aim to verify whether the function is effectively minimized by each algorithm, and which algorithm works best.

To demonstrate the proper working of the estimator and to verify its implementation, an optimization routine was conducted using two parameters for visualization - m_o, n_o . Both have similar magnitudes and sensitivities to the cost function and hence prove to be the best choice to check if the estimator indeed optimizes the objective. A cost map was generated for a small batch of images, and the estimator path was superimposed on the cost map to verify if the estimator efficiently obtains the minimum value of the objective function. The four different algorithms were evaluated for this purpose. This is illustrated in Fig. 3.25 through 3.28 below.



1500

1400

1350 2⁰2

1250

1200 1150



Figure 3.27 LM Algorithm.

M 0 Figure 3.28 BFGS Algorithm.

1000

The gradient method is clearly inferior to the remaining three Quasi-Newton methods. Although it requires the least amount of computation, it is not well suited for ill-conditioned problems as in this case. Distortion parameters have a significantly lower effect on the cost when compared to the focal length and

principal point. In addition, the parameters themselves are orders of magnitude apart. The Levenberg Marquardt and BFGS methods are both on par if not better than Gauss-Newton. However, the GN algorithm is sufficient to optimize the objective, and both LM and QN require a slightly increased amount of computation without providing any significant benefit. The LM algorithm is more robust if parameters are poorly known, but provides no advantage if parameters are fairly close to their optimum values as in this case. The BFGS algorithm offers some computational advantages by eliminating the inverse calculation and converging in fewer iterations but relies on a line-search algorithm. Thus the GN algorithm is the method of choice for this problem. This choice is consistent with other research projects focusing on autonomous star-tracker calibration [Samaan *et al.* 2001].

3.3 Batch Results

Once the mathematical model was tested and provided favorable results, a batch estimator was implemented with the selected algorithm to optimize all the parameters. Fig. 3.29 below illustrates a code flow diagram of the optimization routine. Initial values of the parameters attained from a lab calibration procedure were passed to the estimator with the aim of attaining an updated set of parameters that minimize the residual between the cosine angles between imaged and catalogued vectors. The centroiding and matching phases of the firmware were pre-processed by the star tracker. Two tests were performed:

- 1. Estimate Parameters using the entire batch
- 2. Estimate Parameters using a single image at a time

The first test provides a set of parameters that best match all the data. The disadvantage of using the entire batch, is that an optimization using such a large number measurements would never be feasible online. Using single image shave the advantage of computational efficiency, however a single image cannot guarantee the adequate observability of all parameters and is highly susceptible to measurement noise. The result is that the estimates themselves will be highly influenced by noise and will not represent the true state of camera co-efficients. Figures 3.30 through 3.37 illustrate the entire batch solution, where as Figures 3.38 through 3.45 illustrate solutions using each image at a time. Images providing rank deficient measurements were ignored.



Figure 3.29 Code Block Diagram for Batch Estimation.



Figure 3.30 Focal Length f Estimate Using Entire Batch.



Figure 3.31 Principal Point m_o Estimate Using Entire Batch.



Figure 3.32 Principal Point n_o Estimate Using Entire Batch.



Figure 3.33 Radial Distortion b_1 Estimate Using Entire Batch.



Figure 3.34 Radial Distortion b_2 Estimate Using Entire Batch.



Figure 3.35 Detector Tilt a_1 Estimate Using Entire Batch.



Figure 3.36 Detector Tip a_2 Estimate Using Entire Batch.



Figure 3.37 Residual Using Entire Batch.



Figure 3.38 Focal Length Estimate f Using Each Image.



Figure 3.39 Principal Point m_o Using Each Image.



Figure 3.40 Principal Point n_o Using Each Image.



Figure 3.41 Radial Distortion b_1 Using Each Image.

In both cases, the residual between true and measured star vectors was reduced by over an order of magnitude. In addition, since the mean value of the single image estimates was very much the same as the batch value, and slightly different from the lab calibration estimates. This suggests that the parameters underwent a slight shift from their lab calibration values, either due to shifted components or due to drawbacks of using lab equipment as opposed to actual star images. In addition, the single-image estimates overlap the estimates attained using the entire batch, with no visible motion in single image tests. This suggests that the parameters did not shift during the period of the test. This is expected since parameters are unlikely to change on earth without any drastic temperature variations. The significantly higher variance in the single image results are hence less likely due to be physical changes in the parameters, and can be attributed either to a significant magnitude of noise in the measurements, or the lack of observability of the parameters given a single image. The approximated variance in the focal length estimate acquired using the entire batch of data is 6×10^{-10} , whereas the variance in focal length estimates attained using single images is 6×10^{-7} . Using the more feasible single image approach provides estimates with an unacceptable variance, and including the entire batch in the computation to achieve a low variance result is unfeasible online. Hence, a batch selection strategy is required to make appropriate choices of images to include in a smaller, more feasible subset of the entire batch, that will allow for estimates with an acceptable variance without being computationally intensive.



Figure 3.42 Radial Distortion b_2 Using Each Image.



Figure 3.43 Detector Tilt a_1 Using Each Image.



Figure 3.44 Detector Tip a_2 Using Each Image.



Figure 3.45 Residual Using Each Image.

3.4 Batch Selection

Single image solution solutions are highly varied and present challenges for parameter observability and solutions from large batches are computationally unfeasible; therefore an appropriate batch selection criteria must be developed. Selecting an appropriate batch involves determining the best possible combination of batch size and scene diversity that will allow the estimator to efficiently optimize the camera parameters while minimizing the need for computational power. An appropriate metric that can allow for the evaluation of estimate quality can be attained by analyzing the covariance matrix. In essence it is required to select an optimum subset of images from the entire batch while achieving almost as low a variance in estimates and covariance between parameters as though if the entire batch was used in the optimization routine.

The batch of images must be selected to improve:

• Estimate Variance due to noise:

Given that the ground test data was acquired on earth, the parameters are not expected to change or if so with a small magnitude and with a small rate of change. A large variance in a set of estimates provide evidence that the estimate from a given subset of data is largely driven by measurement noise and not and actual physical change.

• Observability: Ability to resolve each parameter.

Parameters with high variance are likely un-observable given the current set of data. The trace of the covariance matrix or the 'Cramer-Rao Lower-Bound' serves as an approximate minimum achievable variance in each parameter estimate.

• Separability: Ability to find a unique minimum between highly coupled parameters.

Parameters like m_o and a_2 are highly coupled and have very similar effects on the cost function. Selecting a large enough batch size is key

to resolving these two parameters. The off-diagonal values in the covariance matrix provide insight into the separability of parameters. Parameters that are highly coupled, such as m_o and a_2 will have higher off-diagonal values than those that are loosely coupled.

The first test conducted to determine an appropriate batch involved finding focal length variance using data with an increasing number of images. For this test, 50 images of the same scene was used (See. Fig. 3.46).



Figure 3.46 Consecutive Images of Same Scene

The aim of this test was to determine a batch size at which adding more images to reduce the effect of measurement noise does not provide a substantial drop in estimate variance. Although only focal length was considered since similar trends are expected in all parameters. For consistency, images with same number of matched stars (10 star match) were used in these tests. Estimates are less likely to be driven by measurement noise as the number of images included in batches increase. The result is plotted in Fig. 3.47. The results indicate that a significant drop in variance is not achieved after a set of 25 images is utilized.

Apart from being influenced by measurement noise, sensor parameters are likely to be unobservable in a single scene. Lack of observability increases the variance in parameters. To ensure that these parameters are observable in batches of 25 images, the spacing between images included in the batch was gradually increased and the variance in poorly observable parameters was approximated. The lower the approximated variance in each estimate, the



Figure 3.47 Focal Length Variance Vs. Num. Images.

greater the observability of the parameter in question. To demonstrate, the variance in n_o and b_1 are approximated and plotted as a function of image spacing in the figures below. The parameters n_o and b_1 are moderately and poorly observable. The size of the batch was kept constant at 25 images as suggested from the test outlined above. The variance is expected to reduce as scenes with different star patterns are encountered.



Figure 3.48 Std. Dev. n_0 Vs. Image Spacing.

The final test conducted involved determining the off-diagonal elements of the covariance matrix, corresponding to m_o and a_2 . The effects of m_o and a_2 on the cost are very similar posing challenges for separability of the two

60



Figure 3.49 Std. Dev. b₂ Vs. Image Spacing.

parameters. To evaluate the separability of these parameters a similar test was conducted to determine the covariance between m_o and a_2 as a function of batch size and diversity. The results are plotted below.



Figure 3.50 Std. Dev. b_2 Vs. Image Spacing.

By analyzing the plotted results from the tests above, an acceptable subset of the batch was chosen. This batch must contain atleast 25 images to eliminate the effects of measurement noise. A greater separation between images increases observability and separability of parameters. Using an optimal subset of images from the entire batch allows for the estimation of camera parameters with almost as high a fidelity as with the entire batch by ensuring observability and separability of parameters, and determining

Brendon Vaz
estimates that are less influenced by measurement noise. The added benefit is a lower computational expense. The table below lists the focal length estimate variance attained using using a single image, an optimal set of images and all images. From table 3.3 it can be seen that although the variance in the focal length estimate using all images isn't quite reached, a significant drop in variance is achieved using a diverse subset of images and a batch size of one fifth of the total data.

Description	Variance
Single Image	6.1328×10^{-7}
All Images	5.9980×10^{-10}
Optimal Set	4.9710×10^{-9}

 Table 3.3 ESTIMATED VARIANCE FROM SUBSETS OF DATA

Although a post processing method as implemented here is effective, the ultimate goal would be to determine a batch selection method that can be implemented online. This would require a means to determine the estimate variance before any computationally intensive calculations are performed. One parameter affecting the estimate quality that may be explored include the variance of the centroids in each image to attain an understanding of the diversity of scenes in the image. In addition, the measurement covariance matrix \mathbf{R} may also be a good source of information regarding the accuracy that may be expected in parameter estimate.

An issue that requires further investigation with regards to parameter separability is the effects that m_o and a_2 have on the cost function (See Fig. 3.51) below. The figure illustrates that a unique minimum does not exist in the space of these two parameters. This can be expected since both parameters have a similar translation effect on the centroids imaged by the star camera. Fixing this issue may be carried out by minimizing the cost in focal length and principal point initially, and then finding best fit distortion coefficients that further minimize the residual between imaged and catalogued vectors. In addition, a different formulation of the cost function, or a different camera model can be explored with independent parameters reducing the possibility of attaining local minima.



Figure 3.51 Cost function in the space of m_o and a_2 .

3.5 Recursive Estimation

Perhaps the most feasible means calibrate parameters online is to adopt a recursive algorithm. Recursive estimation allows for the update of the parameter estimate and co-variance in a feasible manner by processing a large batch of data one image at a time, as images become available. Even though a single image is used for each update sequence, the co-variance matrix and prior estimate preserve the information provide by the previous images. Hence each update provides the average value for all data up to and including the current image. Figure 3.52 through 3.59 illustrate the recursively estimated camera parameters. For problems involving constant parameters such as a ground calibration, batch and recursive estimates are expected to be the same.

The recursive least squares method is a feasible approach for online parameter estimation, but must be implemented with caution. In the case of



Figure 3.52 Focal Length f Using Recursive Estimation.



Figure 3.53 Principal Point m_o Using Recursive Estimation.



Figure 3.54 Principal Point n_o Using Recursive Estimation.

the ground test, where parameters don't change, the batch and recursive methods provide very similar revised estimates (See Table 3.4). The recursive estimate has the added benefit that only a single image may be used in one update as opposed to involving all images as done with the batch solver. However, in an orbital setting, recursive least squares estimation methods are very likely to encounter a "co-variance windup" after several updates whereby the estimator becomes too confident in its own estimates causing updates to be ignored and the estimator to behave smugly leading estimates to diverge from their true state.

Description	Symbol	Batch	Recursive
Focal Length	f	0.0160816	0.0160796
Principal axis x-offset	m_o	1041.231	1036.655
Principal axis y-offset	n_o	1271.173	1293.674
Radial correction	b_1	-5.629×10^{6}	-2.802×10^{6}
Radial correction	b_2	-1368.422	1403.531
Tip correction	a_1	0.00268	0.000910
Tilt correction	a_2	0.0184	0.0175

 Table 3.4 BATCH VS RECURSIVE RESULTS



Figure 3.55 Radial Distortion b_1 Using Recursive Estimation.



Figure 3.56 Radial Distortion b_2 Using Recursive Estimation.

Although all parameters approached the batch estimate and smugness may not necessarily be evident in a ground based calibration since parameters are likely to remain constant, it may certainly pose an issue for online calibration



Figure 3.57 Detector Tip a_1 Using Recursive Estimation.



Figure 3.58 Detector Tilt a_2 Using Recursive Estimation.



Figure 3.59 Residual Using Recursive Estimation.

in space where the environment is constantly changing, and the parameters are possibly prone to drift. This can be avoided by introducing artificial process noise into the system or by using a sliding window approach where estimator gain is strategically altered to ignore previous estimates and give preference to the most recent measurement. A direction of further study may include developing a simple Extended Kalman filter driven by process noise as opposed to the recursive least squares approach implemented in this thesis. This will allow for the estimation of not only constant parameters, but also to track parameters as they slowly drift in time.

3.6 On - Orbit Data

With both batch and recursive estimators performing effectively using ground calibration data, further testing was performed using on-orbit telemetry from two in-service star trackers on-board an imaging satellite. The star trackers on-board the satellite were subject to post launch changes in their camera parameters reducing the number of expected star matches and the frequency of attitude estimates. A calibration procedure was required to re-calibrate the camera and determine the parameter values that would minimize the residual between imaged star vectors and vectors in the on-board star catalogue. The minimized residual would allow for an increase in the rate of star matching, reducing dropouts in attitude readings, and provide more frequent and more accurate attitude data. The figure below depicts the on-orbit data that was captured by the two in-service star trackers Sensor A and Sensor B.



Figure 3.60 Data - Sensor A.

Figure 3.61 Data - Sensor B.

The images captured by the star trackers contain divserse scenes of star patterns. Since the dataset attained from the satellite was fairly small, (approximately 150 images), and the processing was done offline, the entire batch was included in the calibration and the new parameters that best fit the entire dataset was attained using a batch calibration procedure. The results from the re-calibration are illustrated in figures 3.62 through 3.75 and summarized in Table 3.5 and 3.6 below.









=

Description	Symbol	Before	After
Focal Length	f	0.016097	0.016049
Principal axis x-offset	\dot{m}_o	894.441	1048.298
Principal axis y-offset	n_o	1210.795	1127.392
Radial correction	b_1	-1.224×10^{7}	1.016×10^{8}
Radial correction	b_2	-827.404	-2168.390
Tip correction	a_1	-0.0169	-0.0561
Tilt correction	a_2	-0.00680	0.0436

 Table 3.5 SENSOR A CALIBRATION RESULTS

escription	Symbol	Before	After

 Table 3.6 SENSOR B CALIBRATION RESULTS

Description	Symbol	Before	After
Focal Length	f	0.016113	0.016025
Principal axis x-offset	m_o	1040.559	1001.009
Principal axis y-offset	n_o	1316.122	1221.169
Radial correction	b_1	-1.06×10^{7}	2.31×10^{8}
Radial correction	b_2	-941.924	-2.728
Tip correction	a_1	0.0421	0.0158
Tilt correction	a_2	-0.00644	-0.0192

From the result plots, it is evident that the star tracker parameters had undergone a significant change post launch. The calibration procedure was able to determine the magnitude of this change. The revised parameters reduced the residual between imaged and catalogued star vectors in both star trackers by almost an order of magnitude (See Fig. 3.77). The reduction in the residual will allow the star trackers to match an increased number of stars reducing drop outs in attitude data. In addition, a reduction in the residual between imaged and catalogued vectors suggests an increase in the accuracy of attitude readings.

To evaluate the quality of estimates, the matching and attitude data from the telemetry was re-processed before and after calibration with the expectation of seeing a greater number of matched stars after calibration. The improvements are illustrated in figures 3.78 and 3.79 below. This plot illustrates the four elements of the best fit quaternion solution determined by the star tracker describing space craft orientation. Upon examining the attitude data it is clear that the on-orbit calibration procedure performed the necessary task.

The significant gaps in the 'Before Calibration' plots indicate that are several instances of invalid attitude readings that result from a false star match. False star matches occur when a large residual exists between imaged, and catalogued star vectors. The revised parameters attained from the batch calibration procedure were used to re-determine the imaged star vectors and the matching process was performed again. This test serves as a verification as to whether the updated parameters increased matching efficiency.

In the 'After Calibration' plot, a significantly lower number of gaps can be observed in the plots. This indicates that a fewer number of false matches was encountered allowing for a higher frequency of valid attitude measurements. The batch method used here worked well to provide revised parameter estimates and allowing the star tracker to effectively match a significantly higher number of stars.

In addition to the batch calibration, the sensor parameters were re-calibrated using a recursive estimator. The results for Sensor B are shown in figures 3.80 through 3.86 below.

The results from the recursive estimation routine were very similar to the batch estimates in the case of f, n_o and a_1 . Parameters b_1 and b_2 are poorly observable, and hence require several more images to be estimated with a low variance. Nonetheless, both b_1 and b_2 can be seen approaching batch estimates. Parameters m_o and a_2 are prone to have local minima, and hence were poorly resolved. Recursive estimates should ideally approach batch values as all images are processed. The discrepancy in the values can be attributed to an insufficient number of iterations at each update in the recursive formulation, or an insufficient drop in the covariance matrix with each update. It may also suggest that the parameters may not be constant and subject to change.







Figure 3.80 f - Recursive Estimation.



Figure 3.81 m_o - Recursive Estimation.



Figure 3.82 n_o - Recursive Estimation.

The results from the on-orbit data tests indicate that online calibration may be used to continually re-calibrate parameters online should they drift beyond tolerance. However, some further study is required. The batch method used here may be an unfeasible means to update camera parameters online if parameters undergo further changes and the recursive method is effective only in cases with constant parameters. A simple Extended Kalman Filter driven by process noise must be implemented on the star tracker to not only adapt to post launch changes in parameters, but track the motion of the parameters through out the orbital period.



Figure 3.83 b_1 - Recursive Estimation.



Figure 3.84 b_2 - Recursive Estimation.



Figure 3.85 a_1 - Recursive Estimation.



Figure 3.86 a_2 - Recursive Estimation.

4 Conclusion

Star trackers have become the most accurate and most popular instruments for measuring spacecraft attitude. A single device providing three axis attitude information is invaluable. However star-trackers rely on a set of precisely aligned optics that are likely to drift over the course of the star-trackers mission life. Without adequate knowledge of camera parameters, imaged starvectors will be inaccurately defined, reducing the star cameras ability to accurately identify stars in its field of view. This causes an increase in the number of false matches and therefore reduction in the quality of attitude information. To combat this issue, a method of online calibration was presented in this project. The method outlines a routine that utilizes images captured by the star tracker and the on-board star catalogue to determine the constantly drifting star tracker parameters. The approach aims to minimize the residual between the angular distance between pairs of imaged and cataloged stars. This method was tested using simulated data, ground test data gathered over the traverse of a rover as well as on-orbit telemetry gathered from an in-service star tracker.

4.1 Summary

The results outlined in this thesis demonstrate that an online calibration procedure has the potential to minimize the residuals in star tracker measurements allowing for a greater matching efficiency and a higher rate and accuracy of attitude data. The project outlined two different approaches for online calibration calibration: a batch algorithm using a Gauss Newton solver and and a sequential algorithm utilizing a recursive least squares solver. Both algorithms were tested using ground data gathered from a star tracker implemented on a rover. In addition, further testing was done using orbital data gathered from an in-service star tracker.

4.1.1 Simulated Data

The simulated test outlined at the onset of chapter 3 demonstrated that the calibration method may be used to calculate revised parameters from a reasonable initial guess under ideal conditions. Next, a similar test using synthetic noisy measurements was performed to evaluate whether the calibration method may perform in a more realistic situation. The noise test provided parameter estimates with a low mean error, however, the radial distortion parameters estimates were highly varied. This is expected since the cost function is relatively insensitive to changes radial distortion parameters. Nonetheless, the results from the calibration procedure provided reasonable evidence that the procedure would effectively determine parameters from data gathered using actual star tracker hardware.

4.1.2 Ground Data Tests

In the ground test, star data was gathered over the traverse of a rover. Both Batch and Recursive methods were used to calibrate the star tracker. Although the measurements are noisy, both batch and recursive methods produced camera parameter estimates that are in accordance with each other and reduced the residual in measurements by an over order of magnitude.

Batch Algorithm Tests

Four different minimization routines were tested - gradient descent, Gauss-Newton, Levenberg Marquardt and the Broyden-Fletcher-Goldfarb-Shanno methods. Of the four methods, the gradient method was clearly the least efficient method. From the three remaining, the Gauss-Newton method proved to be the simplest and hence most promising since all three behave in relatively the same fashion. The LM algorithm is usually more robust compared to GN, especially in cases with a bad initial estimate. However, in this case, since the nominal paramter values or those provided by lab calibration are fairly close to the optimal value, the LM algorithm provided

no significant benefit. The BFGS method has a slight computational advantage by converging in fewer iterations and the elimination of the inverse hessian calculation, but relied on a line-search algorithm to determine an appropriate step size. Thus the GN method was the chosen method, due to its simplicity, and is sufficient to perform the necessary minimization.

Two different tests were performed with the batch calibration procedure. In the first test, all available images were utilized in the calibration. In the second test, each image was used individually to attain revised parameter estimates. In both cases, the residual was significantly minimized and revised estimates were attained. The Batch calibration works best in cases where a large amount of data is available. This is certainly a preferred calibration method for ground tests especially since the parameters are unlikely to change, and the large sets of data may be processed offline. Unexpectedly, using this method with a single image provides little information about parameter motion, instead the estimates attained are highly influenced by noise and hence have an unacceptable variance. To combat this issue, two different directions were taken. The first, was to develop a batch selection strategy that allows for the calculation of revised parameter estimates using a well selected feasible set of data that minimizes the effect of noise and ensures paramter observability. The second was to implement a recursive algorithm.

Batch Selection Tests

The batch selection strategy was developed based on some key factors that affect parameter estimates. These factors include the dependence on estimates to noise, parameter observability and separability. All parameters are subject to the influence of measurement noise. Adding more images to a batch of data will cause parameters to be driven less by noise, and hence provide the true state of the parameters. In addition, parameters that have low observability, such as radial and tangential distortions require a variety of scenes to be estimated with an acceptable variance. In addition, parameters that are strongly coupled such as tangential distortions a_1 and a_2 and principal point offsets n_o and m_o are likely to have a very high covariance reducing the ability to acquire good estimates. To avoid these issues, the estimation routine was performed using batches with varying lengths, and separation. The results confirm that longer sets will allow for a reduction in the influence of noise, whereas a higher separation between images guarantee a variety of a diversity scenes in the dataset increasing both observability and separability. The tests conducted in this thesis suggest that beyond 25 images the effects of noise is minimal. In addition, having atleast two different scenes will ensure observability of parameters but the higher the number of scenes the better the observability.

Recursive Algorithm Tests

The Recursive Calibration converged to the same estimate as the batch calibration in the ground test with the added advantage of utilizing single images as they would become available. This is clearly the best method to implement on-line on the star tracker due to its computational feasibility. However the recursive least squares is efficient in estimating constant vectors given a set of data. In a mission setting in orbit, the parameters may undergo changes due to vibration from launch, and continue to drift slowly due to temperature gradients experienced throughout its orbital period. In this case the recursive methods must be implemented with caution to prevent estimator smugness. Junkins, [Griffith & Junkins 2009], proposed that introducing artificial process noise into the estimator can efficiently prevent the co-variance windup problem, and using a forget factor will give preference to the most recent measurements allowing for a more accurate tracking of changes in parameters. This is an area that requires further investigation in order to test the recursive method with on-orbit data, and be fully implemented on the star tracker.

4.1.3 Orbital Data Tests

An on-orbit test was performed to verify if the method works as efficiently in flight, as it does on the ground. The data utilized in this experiment was gathered from two in service star trackers on-board an imaging satellite. (Sensor - A and Sensor B). Significant discrepancies in the camera parameters can be expected in flight when compared to the ground test due to vibration and temperature changes in the environment. Using the data from these in-service star trackers, a calibration was performed to provide revised parameter estimates. Next, the matching process was conducted to evaluate the new parameters. The re-calibration significantly reduced the number of false matches that were experienced in flight allowing for an increased rate of attitude data provided by the star tracker. In addition, the residual between imaged vectors and catalogued vectors reduced by approximately an order of magnitude in both cases directly increasing attitude accuracy.

4.2 Future Work

Some areas of future study include:

• Better Measurement Co-variance Estimation:

The co-variance approximation outlined in this thesis is a crude estimate. The coupling between centroid locations and the resulting arc lengths between star vectors was ignored. The diagonal covariance used here allows for analytic results, however a co-variance matrix taking into account the coupling between pairs of vectors is more analogous to real life and can more adequately depict model measurement noise. This will provide better performance numerically.

• Better Batch Selection Strategy:

The batch selection strategy outlined in this project is efficient but only in retrospect. It provides information about the data by analyzing the expected variance in the parameters once they have been estimated. Ideally, batches should be selected by observing the data itself, before any computationally intensive calculations are conducted. This process may involve evaluating the quality of the data by examining the measurement co-variance, or the variance in the centroids per image or over the course of all images in the batch to understand the diversity of star vectors the scene. Determining the quality of the data before estimating the parameters can significantly increase the performance of the estimator.

• Introduce artificial process noise to Recursive Estimator

Batch methods are too noisy and computationally inefficient to be implemented online, while recursive estimators tend to act smugly after a significant number of updates and give preference to the current estimate while ignoring new data. In order to take properly track parameter changes an appropriate process noise must be determined to tune the estimator allowing for the filtering of noise while giving preference to the incoming data.

• Design a parameter Health Monitoring Strategy

Since the star tracker has some limitations in its power and processing budget, it is in-efficient to have an estimator running at all times. Ideally an estimation routine should only be invoked if there is significant reason to believe that the parameters have shifted beyond an effective region. Perhaps an efficient means to achieve this is to monitor the mean measurement residual over the course of the star tracker's life. If the residual is unexpectedly high, it is evident that a calibration procedure is required.

• Implement and Test Algorithm On-line

Of course the final step is to implement the algorithms on the star tracker itself. Currently all processing was done offline. An online approach would eliminate the need for ground support and add to the autonomy of the star tracker.

4.3 Concluding Remarks

In conclusion, the results outlined in this thesis suggest ground calibration may be sub-optimal post-launch and that an online calibration procedure may be used to to maintain the performance of a star tracker. Although an initial lab calibration and night sky testing is key to attaining initial catalog matches from imaged star data, they may not be sufficient, either due to the inability of the lab testing environment to efficiently simulate its working conditions or due to atmospheric effects and noise caused from stray light during night sky tests. Parameters may also change due to other factors including hostile launch vibrations and operation in the vacuum of space, and may continue drift over time due to aging and temperature variations. An online calibration is can allow the star tracker to accurately image star vectors with the attempt to maintain the accuracy of the instrument throughout its mission life. The procedure outlined in this project was subject to several tests using ground data, and its performance was evaluated using on-orbit telemetry. The calibration procedure considerably increased star matching efficiency and minimized measurement residual as demonstrated in results. The minimized residual increases the accuracy of the attitude data gathered from the star tracker, where as the increased matching efficiency will significantly reduce the number of false matches, thereby decreasing the frequency of drop outs in attitude data. The technical benefit of on-orbit calibration includes a more precise calibration, ability to track parameter changes, less telemetry data, minimal interruption of science observations, greater autonomy, and less ground testing and support.

Bibliography

- [Boyd & Vandenberghe 2004] Boyd, Stephen Poythress, & Vandenberghe, Lieven, *Convex optimization*, Cambridge university press, 2004.
- [Chen et al. 2006] Chen, Xueqin, Wang, Feng, Geng, Yunhai, & Zhang, Yingchun, "An on-orbit calibration system for satellite attitude control," Systems and Control in Aerospace and Astronautics, 2006. ISS-CAA 2006. 1st International Symposium on, 2006, Pages: 6 pp.-435.
- [Crassidis & Junkins 2011] Crassidis, John L, & Junkins, John L, Optimal estimation of dynamic systems, Vol. 24, CRC press, 2011.
- [Enright John 2010] Enright John, Tom Dzamba, Geoff Mcvittie, "Commissioning the S3S Nanosatellite Star Tracker," *International Astronau*tical Congress, 2010.
- [Enright 2012] Enright, John, Doug Sinclair Tom Dzamba, "The Things You Can't Ignore: Evolving a Sub-Arcsecond Star Tracker," Small Statellite, 26th Annual AIAA/USU Conference, 2012.
- [Griffith & Junkins 2009] Griffith, D Todd, & Junkins, John L. "Recursive on-orbit calibration of star sensors,", 2009.
- [Griffith et al. 2002] Griffith, D Todd, Singla, Puneet, & Junkins, John L, "Autonomous on-orbit calibration approaches for star tracker cameras," Advances in the Astronautical Sciences, Vol. 112, 2002, Pages: 39–57.
- [Hashmall et al. 2000] Hashmall, Joseph A, Radomski, Mark, Sedlak, Joseph, & Harman, Richard, "On-orbit calibration of satellite gyroscopes," 2000.

- [Kelley 1999] Kelley, Carl T, Iterative methods for optimization, Vol. 18, Siam, 1999.
- [Kim et al. 2004] Kim, YV, Di Filippo, KJ, & Ng, A, "On the autonomous in orbit calibration of satellite attitude sensors," AIAA Guidance, Navigation, and Control Conference and Exhibit (AIAA 2004-5125), Providence, Rhode Island, 2004.
- [Kok-Lam Lai 2003] Kok-Lam Lai, John L. Crassidis, Richard R. Harman, "In-Space Spacecraft Alignment Calibration using Unscented Filter," Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit, 2003.
- [Liebe 1995] Liebe, C.C., "Star trackers for attitude determination," Aerospace and Electronic Systems Magazine, IEEE, Vol. 10, No. 6, 1995, Pages: 10–16.
- [Liebe 2002] Liebe, C.C., "Accuracy performance of star trackers a tutorial," Aerospace and Electronic Systems, IEEE Transactions on, Vol. 38, No. 2, 2002, Pages: 587–599.
- [Liu et al. 2011] Liu, Hai-bo, Wang, Jiong-qi, Tan, Ji-chun, Yang, Jian-kun, Jia, Hui, & Li, Xiu-jian, "Autonomous on-orbit calibration of a star tracker camera," Optical Engineering, Vol. 50, No. 2, 2011, Pages: 023604–023604.
- [Lu Jing-Hui] Lu Jing-Hui, Wang Hong-Li, Wen Tao Zhan Qiao-Lin, ," .
- [OShaughnessy & Pittelkau 2007] OShaughnessy, Dan, & Pittelkau, Mark E, "Attitude Sensor and Gyro Calibration for MESSENGER," 20th International Symposium on Space Flight Dynamics, Annapolis, MD, 2007, Pages: 24–28.
- [Pal & Bhat 2009] Pal, Madhumita, & Bhat, M Seetharama, "Star camera calibration combined with independent spacecraft attitude determination," American Control Conference, 2009. ACC'09. IEEE, 2009, Pages: 4836–4841.
- [Pittelkau 2007] Pittelkau, Mark E, "Autonomous On-Board Calibration of Attitude Sensors and Gyros," 2007.

- [Samaan et al. 2001] Samaan, Malak A, Griffith, Todd, Singla, Puneet, & Junkins, John L, "Autonomous on-orbit calibration of star trackers," Core Technologies for Space Systems Conference (Communication and Navigation Session), 2001.
- [Samaan et al. 2003] Samaan, Malak A, Mortari, Daniele, & Junkins, John L, "Non-dimensional star identification for uncalibrated star cameras," Advances in the Astronautical Sciences, Vol. 114, 2003, Pages: 477–490.
- [Sedlak & Hashmall 2004] Sedlak, Joseph, & Hashmall, Joseph, "Automated Attitude Sensor Calibration: Progress and Plans," Paper No. AIAA-2004-4854, AIAA/AAS Astrodynamics Specialist Conference, Providence, RI, Vol. 2, 2004, Page: 5.
- [Sedlak et al. 2003] Sedlak, Joseph, Welter, Gary, & Ottenstein, Neil, "Towards Automating Spacecraft Attitude Sensor Calibration," 54th International Astronautical Congress of the International Astronautical Federation, Bremen, Germany, 2003, Pages: 29–30.
- [Shen et al. 2010] Shen, Juan, Zhang, Guangjun, & Wei, Xinguo, "Star sensor on-orbit calibration using Extended Kalman Filter," Systems and Control in Aeronautics and Astronautics (ISSCAA), 2010 3rd International Symposium on. IEEE, 2010, Pages: 958–962.
- [Singla et al. 2002] Singla, Puneet, Griffith, D Todd, Crassidis, JL, & Junkins, JL, "Attitude determination and autonomous on-orbit calibration of star tracker for the gifts mission.," Advances in the Astronautical Sciences, Vol. 112, 2002, Pages: 19–38.
- [Stengel 1986] Stengel, Robert F, Optimal control and estimation, Dover publications, 1986.
- [Sun et al. 2013] Sun, Ting, Xing, Fei, & You, Zheng, "Optical System Error Analysis and Calibration Method of High-Accuracy Star Trackers," Sensors, Vol. 13, No. 4, 2013, Pages: 4598–4623.
- [Wang et al. 2008] Wang, Jianhua, Shi, Fanhuai, Zhang, Jing, & Liu, Yuncai, "A new calibration model of camera lens distortion," *Pattern Recognition*, Vol. 41, No. 2, 2008, Pages: 607–615.

- [Wang et al. 2010] Wang, HT, Luo, CZ, Wang, Yu, & Zhao, SF, "Star sensor model parametric analysis and calibration method study," Journal of University of Electronic Science and Technology of China, Vol. 39, No. 6, 2010, Pages: 880–885.
- [Wiktor 1996] Wiktor, Peter J., "On-orbit thruster calibration," Journal of Guidance, Control, and Dynamics, Vol. 19, No. 4, 1996, Pages: 934– 940.
- [Xing et al. 2005] Xing, Fei, Dong, Ying, Wu, Yanpeng, & You, Zheng, "Star tracker parametric analysis for autonomous calibration [J]," Journal of Tsinghua University (Science and Technology), Vol. 11, 2005, Page: 011.