POSITION ESTIMATION USING GRAVITY, SUN, PLANETS, AND STARS

A Thesis

by

ANTHONY CHARLES GARDNER

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Daniele Mortari
Co-Chair of Committee,	Gregory Chamitoff
Committee Members,	Suman Chakravorty
	Jean-Francois Chamberland-Tremblay
Head of Department,	Srinivas Rao Vadali

August 2021

Major Subject: Aerospace Engineering

ABSTRACT

Before humanity can safely venture away from their terrestrial roots, robotic probes will be required to survey distant planets and moons. The current methodology for interfacing with rovers on the Moon and Mars is through teleoperated transmissions, whose reliability comes at the expense of inherent communication delays between the transmitter and the receiver. As distances between space assets increase, which implies more pronounced time delays, so too will the need for offloading requirements in a way that promotes navigational independence. This thesis presents a novel approach to autonomous vehicle localization using celestial observations (e.g., Sun, stars, and the visible planets) and a reference measurement of the unit gravity vector provided by a higher-fidelity planetary shape model than the traditional axial-symmetric ellipsoid, i.e., the exact direction provided by the geoid. In doing so, the mapping between local and body reference frames was improved and initial position errors were mitigated by two or three orders of magnitude. This work also includes a literature review on topics such as space vehicle localization, optimal attitude estimation, navigation filtering, and measurement data processing/fitting using two-dimensional orthogonal polynomials, as well as derivations for a multivariate nonlinear least-squares technique and inclinometer covariance matrix with extended Kalman filter. Finally, a full-system sensitivity analysis was conducted using two test data sets, namely, the Himalayan mountain range on Earth and the Olympus Mons volcano on Mars, whose markedly different topologies were used to characterize the algorithm's performance.

DEDICATION

To M and P

ACKNOWLEDGMENTS

There is simply not enough white space to thank all of the wonderful people I have had the pleasure of meeting and working with over the last several years. That being said, I would like to acknowledge a few individuals that have had a profound impact on my personal development:

Dr. Mortari, thank you for giving me the opportunity to learn, fail, and grow at my own pace. I will always be grateful to you for the warmth and hospitality you have shown me. I would also like to thank Dr. Mortari's family for graciously welcoming me into their home for meals and conversation.

Dr. Chamitoff, thank you for welcoming me into your lab and giving me my first opportunity to perform graduate research.

Dr. Chakravorty and Dr. Chamberland, thank you for agreeing to serve on my committee and for teaching me about navigation filtering and signal processing theory, respectively.

The Guidance, Navigation, and Mission Analysis Group at the Marshall Spaceflight Center, and in particular, Dr. Evan Anzalone and Mr. Joel Amert, for providing me with two summer internship opportunities. Also, thank you Evan and Mr. Jared Leggett for providing me with a warm welcome and a place to stay during my first internship.

The Intelligent Systems Group at Lynntech, and in particular, Drs. Christian Bruccoleri, Stoian Borissov and Lance Kelly, for the opportunity to develop myself on their team through first-hand engineering experience.

My parents and Phil, thank you for supporting me as I pursued this goal. I love you all very much.

Carl, Hunter and Stoian, thank you for your enduring friendships and advice.

Maria, thank you for your encouragement and support. I love you.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professors Daniele Mortari and Gregory Chamitoff (co-advisors), Professor Suman Chakravorty of the Department of Aerospace Engineering, and Professor Jean-Francois Chamberland-Tremblay of the Department of Electrical & Computer Engineering.

The final version of this work would not be possible without significant contributions made by Drs. Daniele Mortari, Hunter Johnston, and Carl Leake to both the theory and testing of these ideas prior to me joining their research team. In particular, Dr. Mortari derived the nonlinear least-squares algorithm and the gravity covariance matrix. Dr. Johnston provided resources to include in the literature review section, as well as many useful suggestions that helped shape the sensitivity study chapter and the direction of the thesis at-large. Dr. Leake wrote code independently and assisted in debugging of algorithms that were ultimately used in the final numerical demonstration. All other work conducted for this thesis was completed by the student independently.

Funding Sources

Graduate study was supported by Texas A&M University teaching and research assistantships and by industry internships with the Marshall Spaceflight Center and Lynntech.

NOMENCLATURE

AU	Astronomical Unit
COTS	Commercial-of-the-Shelf
CSAC	Chip-Scale Atomic Clock
DOR	Differential One-way Ranging
DSN	Deep Space Network
ECEF	Earth-Centered Earth-Fixed
ECI	Earth-Centered Inertial
EGM2008	Earth Gravitational Model 2008
EKF	Extended Kalman Filter
ENU	East-North-Up
GPS	Global Positioning System
HCI	Heliocentric Inertial
IMU	Inertial Measurement Unit
LGM2011	Lunar Gravity Model 2011
MGM2011	Mars Gravity Model 2011
NASA	National Aeronautics and Space Administration
NDSIA	Non-Dimensional Star Identification Algorithm
NEN	Near Earth Network
NLLS	Nonlinear Least-Squares
PNT	Position, Navigation and Timing
SCaN	Space Communications and Navigation
SN	Space Network

SPICE	Spacecraft Planet Instrument “C-matrix" Events
SPS	Stellar Positioning System
VIPER	Visual Position Estimator for Rover
WGS84	World Geodetic System 1984

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION

1.1 Literature Review

Modern Earth-based autonomous navigation and localization is made possible by the Global Positioning System (GPS), which provides measurement accuracy of 3.9 meters vertically and 1.9 meters horizontally with 95% confidence [2]. While this accuracy is sufficient for the majority of human operations on Earth, ground-based extraterrestrial applications will require even greater precision in order to facilitate autonomous roving and other surface activities on distant planets and moons. The accuracy of GPS could be drastically improved using dual-frequency receivers (commonly found in military applications), which enable real-time measurements on the order of a few centimeters. The extent to which this system is used to support autonomous systems is demonstrated in the consumer smartphone, which has been shown to obtain position estimates within 4.9 meters of the user's actual location [3]. While this system enables wide-scale autonomy, it relies heavily on its network of pre-positioned satellites to triangulate (and track over time) an asset's global position. These networks require years of planning and billions of dollars to construct, which currently makes the technology infeasible for applications away from Earth. Therefore, it is clear that the success of future space operations will require a technology solution that promotes navigational independence from GPS (or GPS-spoofed) environments.

Currently, navigation throughout the Solar System is supported by a network of relay satellites that fall under the umbrella of the National Aeronautics and Space Administration's (NASA) Space Communications and Navigation (SCaN) network [4], which includes the Near Earth Network (NEN), Space Network (SN) and Deep Space Network (DSN). This architecture is capable of providing position measurements through sequential ranging and delta differential one-way ranging (Delta DOR) [5] with an accuracy of 2.5 nrad, which corresponds to position errors between 136 meters and 1 kilometer for surface assets on Mars, depending on the relative distance between the transmitter and receiver [5, 6]. For Martian rovers, two other state localization techniques have

been demonstrated to date. First, the VIPER algorithm was developed and analyzed with data from NASA's Mars Exploration Rover PANCAM [7]. This technique performed a state estimate by matching the rover's location with respect to the local horizon, as measured by optical sensors, to known surface digital elevation models. In general, this technique was able to determine Spirit's landing site with an accuracy of 51 meters, and Opportunity's landing site with an accuracy of 27 meters. The other localization technique leveraged orbital imagery, more specifically, the HiRISE Orthoimage-Based Rover Localization, to produce position estimates with errors as small as 3.5 meters [8, 9]. NASA's extensive probing of the Martian surface has provided a test-bed for other ground-based localization techniques as well. For instance, the Mars Pathfinder mission used a heading sensor and wheeler encoder to estimate changes in a vehicle's localized position [10, 11]; however, it also included an over-reliance on precision initialization. The Mars Exploration Rovers, on the other hand, used inertial measurement units (IMU) and mobility motor encoder information, coupled with Sun observations, to refine its state measurements [12, 13, 14, 15]. Another method leveraged visual odometry, where the motion of the vehicle is estimated by tracking terrain features between two pairs of stereo images [16]. Regardless of approach, these techniques are highly dependent on initial localization and are more specifically designed for relative navigation systems (as opposed to inertial). Due to the dependencies on precision initialization and/or orbital asset interfacing to define inertial positions, several other techniques have attempted to leverage celestial measurements [17, 18, 19, 20, 21]. In general, these methods utilize Sun, star, and planet imagery, coupled with measurements from an IMU or inclinometer array, to determine ground-based asset positions. Of all the celestial measurement-based approaches, the Stellar Positioning System (SPS) [22, 23] provides one of the most succinct and thorough investigations of this approach, and was used as the general framework for demonstrating the effectiveness of the geoid correction algorithm that will be introduced in Chapter 3.

1.2 Reference Frames

The orthonormal reference frame is fundamental for coordinatizing points in space with respect to a reference point, which is typically set as the origin of the frame. For the purposes of this work,

five unique frames of reference are formalized. A quick summary of definitions can be found in Table 1.1 below, while an expanded version is provided in the following bulleted list.

Name	Notation	x-axis	y-axis	z-axis
Earth-Centered Inertial (ECI)	i	Vernal Equinox	-	Spin Axis
Earth-Centered Earth-Fixed (ECEF)	f	Prime Meridian	-	Spin Axis
Local (ENU)	ℓ	Local East	Local North	-
Inclinometer (Body)	b	-	Vehicle Heading	Outward Normal
Camera	c	-	Vehicle Heading	Optical Axis

Table 1.1: Reference Frame Summary

- **Earth-Centered Inertial (ECI) reference frame, i.** The axes of the ECI are right-handed and fixed with respect to their origin at Earth's center of mass (Fig. 1.1). Neglecting the acceleration of Earth's motion about the Sun, the ECI is considered to be inertially fixed with respect to the stars. The xy-plane is coincident with Earth's equatorial plane, and the z-axis points to Earth's axis of rotation (i.e., the North Pole) and is orthogonal to the mean equator of epoch J2000. The x-axis points to the the vernal equinox where the Earth's equatorial plane intersects the ecliptic plane.

- **Earth-Centered Earth-Fixed (ECEF) reference frame, f.** The axes of the ECEF are right-handed and centered at their origin, Earth's center of mass (Fig. 1.2). The x and y-axes of the ECEF, which lie on Earth's equatorial plane, rotate in time with respect to Earth's spin axis. A point can be coordinatized with respect to the ECEF using either a Cartesian, $\{x, y, z\}$, or geodetic, $\{\varphi, \lambda, h\}$, representation. In the geodetic coordinate set, h is the distance above the reference spheroid, measured along the local normal at a given point. The x-axis of the ECEF points to the intersection of Earth's equator and prime meridian, i.e., the point of 0° latitude and longitude. The y-axis points to 0° latitude and 90° longitude. The z-axis points to 90° latitude, which is coincident with Earth's rotational axis.

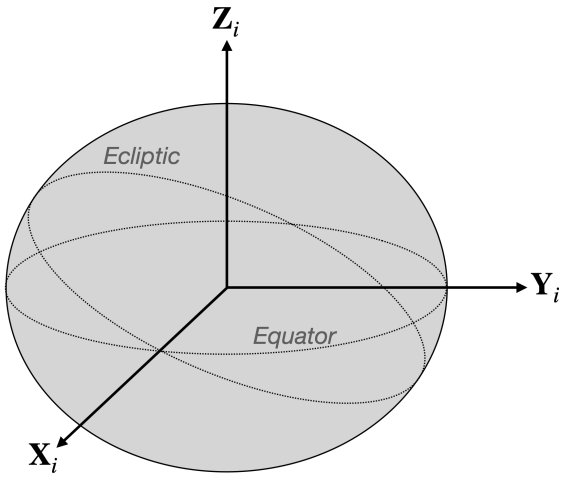

Figure 1.1: Earth-Centered Inertial Reference Frame

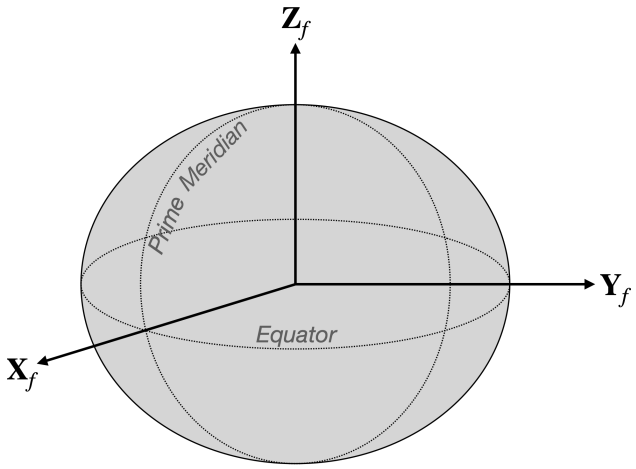

Figure 1.2: Earth-Centered Earth-Fixed Reference Frame

- **Local (ENU) reference frame, ℓ.** The axes of the local (geographic) reference frame are right-handed and centered at their origin, the point on Earth's surface at which the vehicle is located (Fig. 1.3). The orientation of axes is problem specific. For the purposes of this work, it is such that the x-axis points to the East, the y-axis points to the North, and the z-axis points vertically upward (i.e., ENU). By orienting the local reference frame in this way, the zenith direction of the frame would be co-linear with the direction of the local unit gravity vector, $\hat{\boldsymbol{g}}_\ell$, when considering a perfectly spherical celestial body with homogeneous density,

and the local level plane would be orthogonal to $\hat{\boldsymbol{g}}_\ell$.

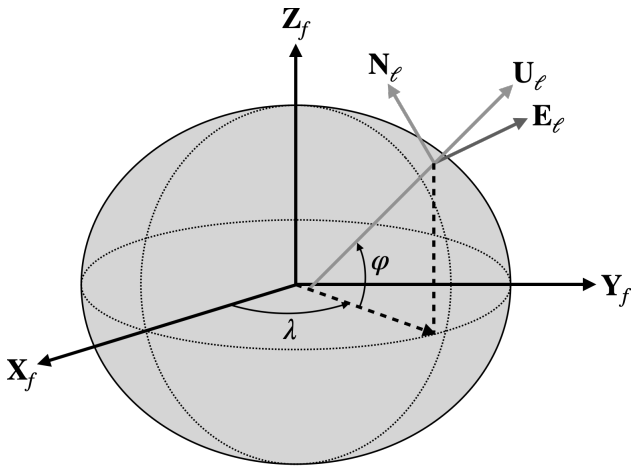

Figure 1.3: Local (East-North-Up) Reference Frame

- **Inclinometer (body) reference frame, b.** The axes of the inclinometer (body) reference frame are right-handed and centered at their origin, the vehicle's center of mass (Fig. 1.4). The xy-plane of this reference frame is defined by a pair of inclination sensors that are rigidly mounted to a level platform onboard the vehicle. The inclination sensors are oriented such that they are nearly perpendicular to each other (the exactness of which can be determined through laboratory calibration prior to the mission date), with the y-axis pointing in the heading direction of the vehicle. Assuming a small misalignment angle, ε, exists between the mounted inclinometer set, the orientation of one of the in-plane reference axes must be written as a function of ε. For instance,

$$\boldsymbol{x}_b = \left\{ \begin{array}{c} 1 \\ 0 \\ 0 \end{array} \right\} \rightarrow \boldsymbol{y}_b = \left\{ \begin{array}{c} \sin\varepsilon \\ \cos\varepsilon \\ 0 \end{array} \right\} \rightarrow \boldsymbol{z}_b = \frac{\boldsymbol{x}_b \times \boldsymbol{y}_b}{\|\boldsymbol{x}_b \times \boldsymbol{y}_b\|_2} = \frac{1}{\sqrt{\cos^2\varepsilon - \sin^2\varepsilon}} \left\{ \begin{array}{c} 0 \\ -\sin\varepsilon \\ \cos\varepsilon \end{array} \right\}. \tag{1.1}$$

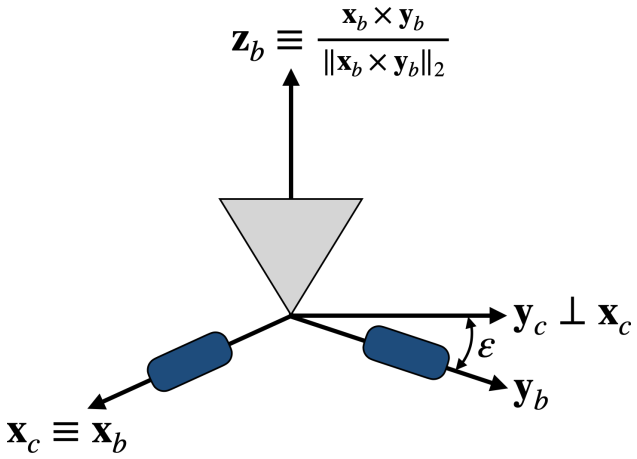

Figure 1.4: Inclinometer (Body)/Camera Reference Frames

- **Camera reference frame,** c**.** The axes of the camera reference frame are right-handed and centered at their origin, the center of mass of a sensor platter that is rigidly mounted to the vehicle (Fig. 1.4). The xy-plane of this reference frame is defined by the idealized orthogonal mount of the inclinometer set to a level platform onboard the vehicle, such that $\boldsymbol{x}_c \perp \boldsymbol{y}_c$. The optical camera is assumed to be rigidly mounted to the level platform such that the aperture points upward. In other words, the zenith direction of this frame of reference, $\boldsymbol{z}_c$, points to the camera's optical axis.

1.3 Transformation Matrices

We also introduce the following five transformation matrices, which are ultimately used to relate measurement vectors taken in the inclinometer and camera reference frames with reference vectors defined in the local and inertial reference frames. Here, we standardize subscript notation such that the new reference frame is introduced first, followed by the old. For example, $C_{A,B}$ (or more explicitly, $C_{A \leftarrow B}$) denotes the transformation of a point in reference frame B to reference frame A. Furthermore, whenever reference is made to a specific element of a transformation matrix, the indexing is defined such that the row number preceeds the column number, i.e., $C_{A,B}[1, 2]$ would be the element located in the first row and second column of $C_{A,B}$.

- **Inertial-to-fixed transformation matrix, $C_{f,i}$.** This transformation matrix accounts for a celestial body's near-constant rotation rate about its polar axis, which affects the pointing direction of the ECEF's x and y-axes over time (Fig. 1.5). The rotation rates for bodies of interest, namely, Earth, Mars, and the Moon, are provided in Appendix A.

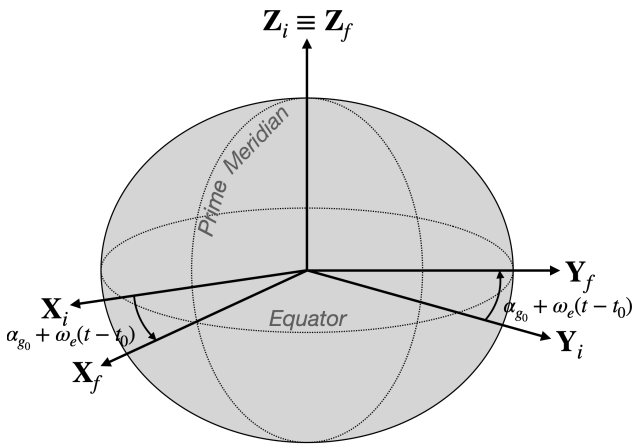

Figure 1.5: Description of ECI/ECEF Relationship

By neglecting higher order perturbations, such as nutation and precession of the Earth's spin axis, we have

$$C_{f,i}(t) = \begin{bmatrix} \cos(\alpha_{g_0} + \omega_e(t - t_0)) & -\sin(\alpha_{g_0} + \omega_e(t - t_0)) & 0 \\ \sin(\alpha_{g_0} + \omega_e(t - t_0)) & \cos(\alpha_{g_0} + \omega_e(t - t_0)) & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where α_{g_0} is the longitude of Greenwich at the reference time, t_0, and t is the current time.

- **Inertial-to-camera transformation matrix, $C_{c,i}$.** This transformation matrix is estimated by mapping unit measurement vectors provided in the camera reference frame with their known inertial counterparts. Here, a method for determining the best estimate of the attitude matrix is required. While a number of techniques exist, this work employs Davenport's $\boldsymbol{q}$-method to find the unit quaternion that minimizes Wahba's loss function. While the at-

titude quaternion, $\boldsymbol{q}_{c,i}$, provides a sufficient description of the relationship between inertial and camera reference frames, the attitude matrix can be derived from the following transformation given a unit quaternion of the form $\boldsymbol{q} = [q_w, \boldsymbol{q}_v]^{\mathrm{T}} = [q_w, q_x, q_y, q_z]^{\mathrm{T}}$,

$$C_{c,i}(\boldsymbol{q}) = \begin{bmatrix} q_w^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_w q_z) & 2(q_w q_y + q_x q_z) \\ 2(q_x q_y + q_w q_z) & q_w^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_w q_x + q_y q_z) & q_w^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix}.$$

- **Fixed-to-local transformation matrix, $C_{\ell,f}$.** This transformation matrix is used to localize a point on the Earth's surface with respect to Earth's center of mass, i.e., the origin of the ECEF. By considering two angles, one in-plane (longitude) and one out-of-plane (latitude), we can write the local reference frame with respect to the ECEF as a two-rotation Euler sequence (Fig. 1.6).

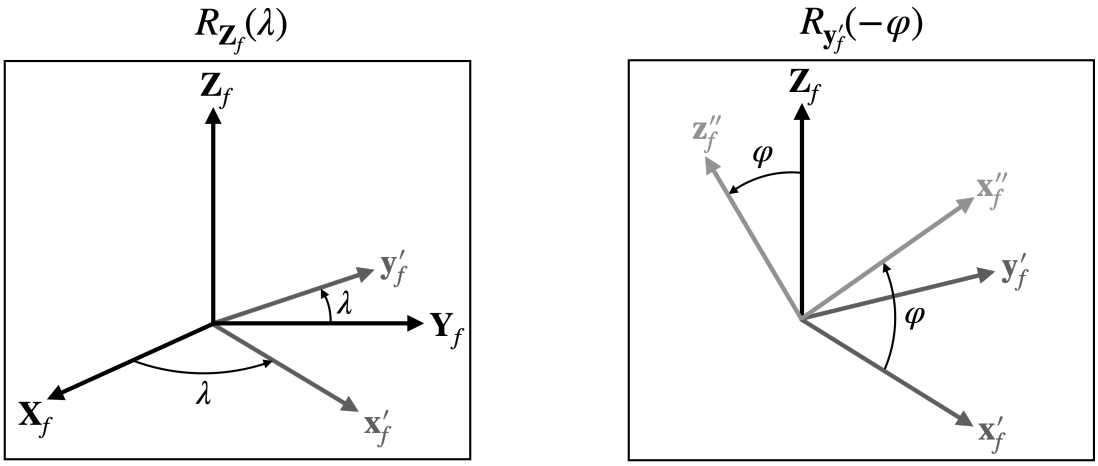

Figure 1.6: Two-Rotation Sequence of the ECEF

The specific orientation of the local reference frame axes is problem specific. For the purposes of this work, an "East-North-Up" (ENU) orientation of the local reference frame was required. Using a permutation matrix, P, we pre-multiply the Euler sequence such that the

intended local reference frame axes are obtained (Fig. 1.7). Thus, we have

$$
\begin{aligned}
C_{\ell,f}(\varphi,\lambda) &= P\; R_{\boldsymbol{y}'_f}(-\varphi)\; R_{\boldsymbol{Z}_f}(\lambda) \\
&= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos(-\varphi) & 0 & -\sin(-\varphi) \\ 0 & 1 & 0 \\ \sin(-\varphi) & 0 & \cos(-\varphi) \end{bmatrix} \begin{bmatrix} \cos\lambda & \sin\lambda & 0 \\ -\sin\lambda & \cos\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix}.
\end{aligned}
$$

If we consider the negative angle identities for sine and cosine, $\sin(-\theta) = -\sin\theta$ and $\cos(-\theta) = \cos\theta$, we have

$$
\begin{aligned}
C_{\ell,f}(\varphi,\lambda) &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix} \begin{bmatrix} \cos\lambda & \sin\lambda & 0 \\ -\sin\lambda & \cos\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos\varphi\cos\lambda & \cos\varphi\sin\lambda & \sin\varphi \\ -\sin\lambda & \cos\lambda & 0 \\ -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \end{bmatrix} \\
&= \begin{bmatrix} -\sin\lambda & \cos\lambda & 0 \\ -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ \cos\varphi\cos\lambda & \cos\varphi\sin\lambda & \sin\varphi \end{bmatrix}.
\end{aligned}
\tag{1.2}
$$

The vehicle's latitude and longitude can be extracted from Eq. (1.2) using

$$
\begin{aligned}
\tan\varphi = -\frac{C_{\ell,f}[2,1]}{C_{\ell,f}[3,1]} \quad &\Longrightarrow \quad \varphi = -\arctan2\left(C_{\ell,f}[2,1], C_{\ell,f}[3,1]\right) \\
\tan\lambda = \frac{C_{\ell,f}[3,2]}{C_{\ell,f}[3,1]} \quad &\Longrightarrow \quad \lambda = \arctan2\left(C_{\ell,f}[3,2], C_{\ell,f}[3,1]\right)
\end{aligned}
$$

- **Inclinometer-to-camera transformation matrix, $C_{c,b}$.** By revisiting Fig. 1.4, it is trivial to write this transformation matrix as a function of the mounting misalignment angle, ε, which we assume to be statistically quantified through laboratory calibration. Using the vector

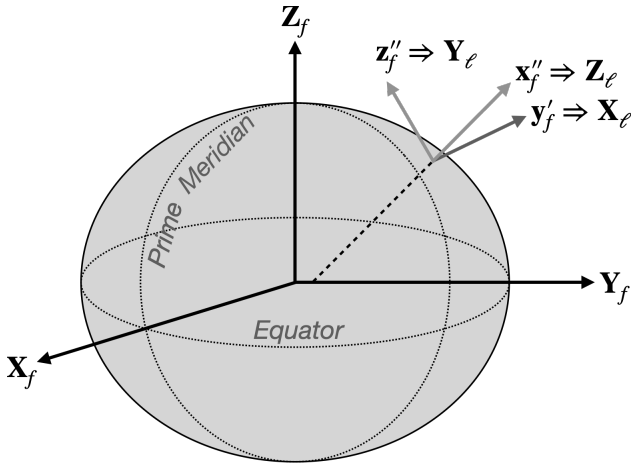

Figure 1.7: Description of ECEF/ENU Relationship

definitions of the inclinometer (body) reference axes with respect to the camera reference axes provided by Eq. (1.1), we have

$$C_{c,b}(\varepsilon) = \begin{bmatrix} \vdots & \vdots & \vdots \\ \boldsymbol{x}_b & \boldsymbol{y}_b & \boldsymbol{z}_b \\ \vdots & \vdots & \vdots \end{bmatrix} = \begin{bmatrix} 1 & \sin\varepsilon & 0 \\ 0 & \cos\varepsilon & -\frac{\sin\varepsilon}{\sqrt{\cos^2\varepsilon - \sin^2\varepsilon}} \\ 0 & 0 & \frac{\cos\varepsilon}{\sqrt{\cos^2\varepsilon - \sin^2\varepsilon}} \end{bmatrix}.$$

- **Inclinometer-to-local transformation matrix, $C_{\ell,b}$.** As shown in Fig. 1.8, this transformation matrix can be derived from the previous four, such that

$$C_{\ell,b} = C_{\ell,f}(\varphi,\lambda)\; C_{f,i}(t)\; C^{\mathrm{T}}_{c,i}(\boldsymbol{q})\; C_{c,b}(\varepsilon).$$

Alternatively, $C_{\ell,b}$ can be written as a function of the inclination sensor measurement angles, α_x and α_y, and the body azimuth angle, γ, which we define as the angular deviation between the y-axes of the local ($\boldsymbol{N}_\ell$) and inclinometer ($\boldsymbol{y}_b$) reference frames, after projecting $\boldsymbol{y}_b$ onto the local level plane,

$$\gamma = \arccos\left(\boldsymbol{N}_\ell^{\mathrm{T}} \cdot \boldsymbol{y}_{b,proj.}\right).$$

The inclination angle definitions, α_x and α_y, are a more realistic representation of the raw

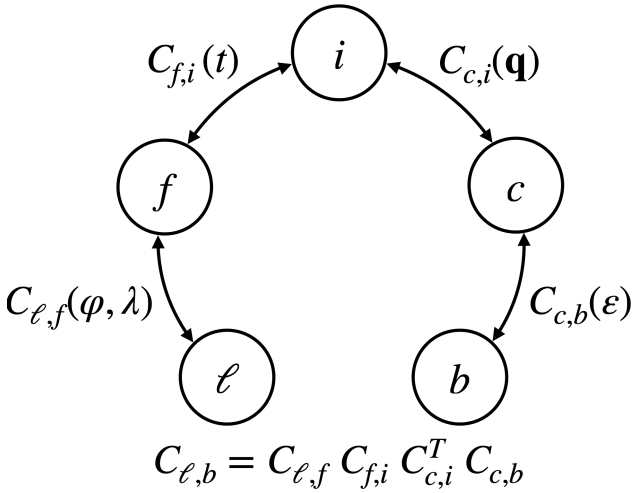

Figure 1.8: Reference Frame Relationships

sensor data output of a typical COTS inclinometer. For analytical purposes, however, we also introduce a second definition of the inclination angles, ϑ_x and ϑ_y, which are used to perform a proper covariance analysis. As we will show in a later section, the covariance derivation requires that we re-frame the inclination angle definitions in order to avoid null values of the expectation operator associated with the "odd" sine function. Fig. 1.9 highlights the

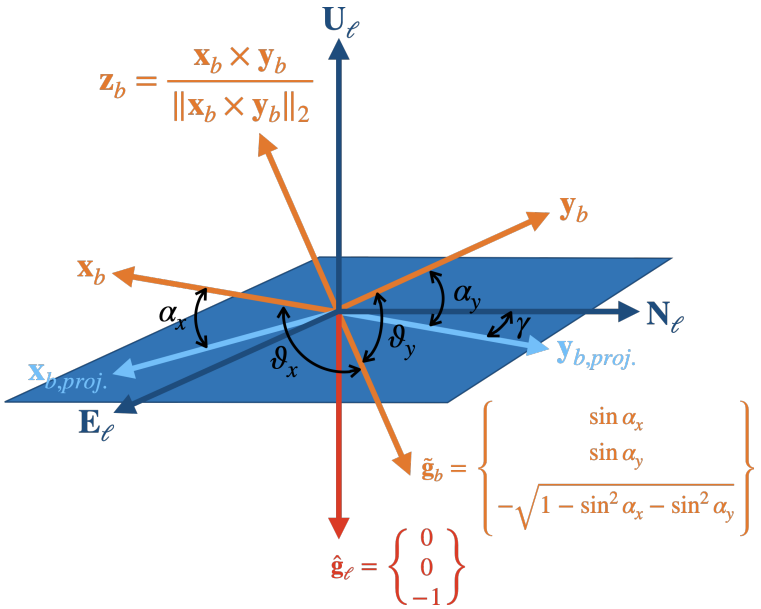

Figure 1.9: Description of ENU/Inclinometer (Body) Reference Frame Relationship

relationship between inclination angle definitions, in particular,

$$\theta_x = \frac{\pi}{2} - \alpha_x \quad \text{and} \quad \theta_y = \frac{\pi}{2} - \alpha_y.$$

We note here that a positive deflection angle with respect to either $\boldsymbol{E}_\ell$ or $\boldsymbol{N}_\ell$ describes the body reference frame tilting toward local zenith. This is a consequential definition, as we can now derive $C_{\ell,b}$ as a three-rotation Euler sequence that assumes positive counterclockwise rotations about each reference axis, given by

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}, \; R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \; R_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{1.3}$$

From Eq. (1.3), and with Fig. 1.9 in mind, we have

$$
\begin{aligned}
C_{\ell,b} &= R_{\boldsymbol{U}_\ell}(\gamma)\; R_{\boldsymbol{N}_\ell}(-\alpha_x)\; R_{\boldsymbol{E}_\ell}(\alpha_y) \\
&= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(-\alpha_x) & 0 & \sin(-\alpha_x) \\ 0 & 1 & 0 \\ -\sin(-\alpha_x) & 0 & \cos(-\alpha_x) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_y & -\sin\alpha_y \\ 0 & \sin\alpha_y & \cos\alpha_y \end{bmatrix} \\
&= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\alpha_x & 0 & -\sin\alpha_x \\ 0 & 1 & 0 \\ \sin\alpha_x & 0 & \cos\alpha_x \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_y & -\sin\alpha_y \\ 0 & \sin\alpha_y & \cos\alpha_y \end{bmatrix} \\
&= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\alpha_x & -\sin\alpha_x\sin\alpha_y & -\sin\alpha_x\cos\alpha_y \\ 0 & \cos\alpha_y & -\sin\alpha_y \\ \sin\alpha_x & \cos\alpha_x\sin\alpha_y & \cos\alpha_x\cos\alpha_y \end{bmatrix} \\
&= \begin{bmatrix} \cos\gamma\cos\alpha_x & -\cos\gamma\sin\alpha_x\sin\alpha_y - \sin\gamma\cos\alpha_y & -\cos\gamma\sin\alpha_x\cos\alpha_y + \sin\gamma\sin\alpha_y \\ \sin\gamma\cos\alpha_x & -\sin\gamma\sin\alpha_x\sin\alpha_y + \cos\gamma\cos\alpha_y & -\sin\gamma\sin\alpha_x\cos\alpha_y - \cos\gamma\sin\alpha_y \\ \sin\alpha_x & \cos\alpha_x\sin\alpha_y & \cos\alpha_x\cos\alpha_y \end{bmatrix}.
\end{aligned}
\tag{1.4}
$$

The tilt angles can be extracted from Eq. (1.4) using

$$
\alpha_x = \arcsin\left(C_{\ell,b}[3,1]\right) \quad \text{and} \quad \alpha_y = \arctan2\left(C_{\ell,b}[3,2], C_{\ell,b}[3,3]\right),
$$

along with the body azimuth angle,

$$
\gamma = \arctan2\left(C_{\ell,b}[2,1], C_{\ell,b}[1,1]\right).
$$

1.4 Optimal Attitude Estimation

Consider a finite set of unit vectors, $\boldsymbol{r}_k$, where $k = 1, \cdots, N$, that exist in a known frame of reference. Now, suppose the same unit vectors are measured in a second frame, e.g., the body

reference frame of a vehicle, $\boldsymbol{b}_k,$ where $k = 1, \cdots, N$. Then, the vehicle's orientation with respect to the known reference frame can be described by an orthogonal matrix C, with $\det C = +1$, that satisfies

$$\boldsymbol{b}_k = C\boldsymbol{r}_k.$$

Determination of the optimal attitude matrix for a given set of measurement vectors has been a topic of research for decades since Grace Wahba [24] first posed the problem using a simplification of the following least-squares loss function,

$$\mathbb{L}(C) = \frac{1}{2}\sum_{k=1}^{N} w_k \left\|\boldsymbol{b}_k - C\,\boldsymbol{r}_k\right\|_2^2.$$

Here, $\|\cdot\|_2$ denotes the Euclidean norm operator, w_k is a scalar weight associated with the kth measurement vector, and C is the attitude matrix that minimizes $\mathbb{L}$. Throughout this work, we estimate the vehicle's attitude, $C_{c,i}$, from Wahba's problem using Paul Davenport's $\boldsymbol{q}$-method [25], which is summarized in the following sentences. First, the known inertial position vectors of the Sun, stars, and visible planets are concatenated into a matrix, R, such as

$$R = \left[\sqrt{w_1}\,\boldsymbol{r}_1, \sqrt{w_2}\,\boldsymbol{r}_2, \sqrt{w_3}\,\boldsymbol{r}_3, \sqrt{w_4}\,\boldsymbol{r}_4, \sqrt{w_5}\,\boldsymbol{r}_5\right],$$

while their observed counterparts, provided by the onboard imager, are arranged in a separate matrix, B, such as

$$B = \left[\sqrt{w_1}\,\boldsymbol{b}_1, \sqrt{w_2}\,\boldsymbol{b}_2, \sqrt{w_3}\,\boldsymbol{b}_3, \sqrt{w_4}\,\boldsymbol{b}_4, \sqrt{w_5}\,\boldsymbol{b}_5\right].$$

Davenport's $\boldsymbol{q}$-method parameterizes the attitude matrix as a unit quaternion such that the eigenvector associated with the largest eigenvalue of the matrix K is precisely the optimal attitude quaternion, i.e.,

$$K\boldsymbol{q}_{opt} = \lambda_{max}\boldsymbol{q}_{opt},$$

where

$$K = \begin{bmatrix} Q - \sigma \boldsymbol{I}_{3\times 3} & Z^{\mathrm{T}} \\ Z & \sigma \end{bmatrix}.$$

Here, $\boldsymbol{I}_{3\times 3}$ is the identity matrix. The remaining terms in K are defined by

$$\begin{cases} Q = A + A^{\mathrm{T}} \\ Z = \begin{bmatrix} A[2,3] - A[3,2] & A[3,1] - A[1,3] & A[1,2] - A[2,1] \end{bmatrix} \\ \sigma = \mathrm{tr}(A) \end{cases}$$

where $\mathrm{tr}\,(\cdot)$ denotes the matrix trace operator and A is the attitude profile matrix, given by

$$A = BR^{\mathrm{T}}. \tag{1.5}$$

1.5 Navigation Filtering

The field of guidance, navigation, and control in aerospace engineering leans heavily on linear estimation theory to model the time evolved behavior of nonlinear systems. The Extended Kalman Filter (EKF) provides a mathematical framework to achieve this, as was first demonstrated in the Apollo program, by integrating sensor measurement data (when available) such that uncertainties in the dynamics model of the vehicle are mitigated. It should be emphasized here that the EKF assumes a first-order linear approximation of both the nonlinear state and measurement equations in order to propagate the entire system forward in time. Given that this is a simplification of the true dynamics, the filter can drift over time when measurements are sparse or parameters such as the covariance matrices are not properly tuned. In general, the discrete EKF can be summarized in two steps:

1. **Predict.** During the prediction step, the EKF propagates the dynamics forward in time by passing nonlinear state information through a linearization function, f, subject to additive zero-mean Gaussian noise, w, whose variance, σ_w^2, is known. In doing so, the best estimate

of the state is given by

$$\hat{x}_k = f(\hat{x}_{k-1}) + w_{k-1}, \quad w \sim \mathcal{N}(0, \sigma_w^2).$$

Errors associated with the linearization of the state and any unfiltered noise are collected in a covariance matrix, P, which is effectively a measurement of the filter's confidence in its estimate of the state. The covariance matrix is also propagated forward in time using a linearization of the dynamical model as defined by a Jacobian matrix of partial derivatives, F, and an uncertainty matrix, Q, whose diagonal elements are the variances associated with each element of the state vector,

$$P_k = F_k \, P_{k-1} \, F_k^{\mathrm{T}} + Q_k, \quad F_k = \left. \frac{\partial f}{\partial x} \right|_{x=\hat{x}_{k-1}}.$$

2. **Update.** The navigation filter's ability to maintain reasonable estimates of the state over time depends on the availability and accuracy of measurement data. When an onboard sensor returns a measurement of the state (the frequency of which is constrained by the sampling rate of the device), the EKF update step is initiated in order to temper the covariance error of the state model that has accrued over a propagation period. This is achieved by minimizing the difference between a measurement model, z, (subject to additive zero-mean Gaussian noise, v, with known variance, σ_v^2) and a linearized measurement model provided by the best estimate of the state, $h(\hat{x})$. More explicitly, the measurement model is given by

$$z_k = h(\hat{x}_k) + \nu_k, \quad \nu_k \sim \mathcal{N}\left(0, \sigma_v^2\right).$$

which is used, along with the Kalman gain defined by a Jacobian matrix of partial derivatives (H) and an uncertainty matrix (R) whose diagonal elements are the variances associated with

each element of the measurement vector,

$$K_k = P_{k-1}\, H_k^{\mathrm{T}}\, (H_k\, P_{k-1}\, H_k^{\mathrm{T}} + R_k)^{-1}\,, \quad H_k = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}_{k-1}},$$

to update the state estimate and covariance matrix as follows,

$$\begin{cases} \hat{x}_k = \hat{x}_{k-1} + K_k(z_k - h(\hat{x}_{k-1})) \\ P_k = (I - K_k\, H_k)\; P_{k-1} \end{cases}.$$

2. MEASUREMENT DATA PROCESSING

2.1 Introduction

Numerical simulations provide a means for demonstrating an algorithm's capabilities and defining operational limits through statistical analyses. This procedure typically involves the preparation and processing of reference data sets, which are used to stress test the algorithm using a random sampling approach such as the Monte Carlo method. This work utilized several data sets, including the Hipparcos star catalog, SPICE data kernels, and binary files containing two-component vertical deflection data for Earth (EGM2008), Moon (LGM2011), and Mars (MGM2011). The following sections describe in greater detail how each data set was utilized within simulations.

2.2 Sun

Ephemerides for the Sun, along with the visible planets, were generated in the ECI from data kernels provided by NASA SPICE [26] using a Pythonic wrapper, SpiceyPy [27]. In the ECI reference frame, the unit position vector of the Sun is defined as $\hat{\boldsymbol{s}}_i$. A Sun sensor can be used to measure the Sun's unit position vector in the camera reference frame, $\tilde{\boldsymbol{s}}_c$. From here, a relationship is established between inertial unit reference vectors and camera measured unit vectors through an attitude matrix,

$$\tilde{\boldsymbol{s}}_c = C_{c,i}\, \hat{\boldsymbol{s}}_i$$

which is simply a restatement of Wahba's problem, as described in Section 1.2.

2.3 Stars

Modern star trackers have earned a reputation for being the first-choice of space mission designers who require precision attitude estimates. From a mechanical perspective, both the Sun sensor and the star tracker can be considered digital cameras that measure point light sources within their fields-of-view (Fig. 2.1). Photons emitted from visible light sources occupying three-dimensional space are mapped onto a two-dimensional image plane that is populated by either CCD (charge-

coupled device) or CMOS (complementary metal-oxide semiconductor) pixels, which convert photons to electrons in order to characterize individual pixel intensities. After passing through the camera's open aperture, wherein the trajectory of incoming photons is slightly distorted by the shape and surface imperfections of the lens, the star tracker uses onboard centroiding algorithms, such as Run-Length-Encode (RLE), to determine the location of objects in a scene on the image plane [28]. Several peak finding approaches exist to estimate the centroid, the most popular of which are center-of-mass and topological. The RLE is a topological centroiding algorithm, which performs an iterative search over rows or columns of pixels, clustering those of similar intensities, and merging them into updated centroid locations. Once a centroid is estimated, the pinhole camera model can be used to project the centroid's pixel coordinates on the image plane into three-dimensional space using knowledge of the camera's focal length. From here, the star tracker implements a star identification algorithm, such as Pyramid [29], in order to relate interstellar angles of measurement vectors in the body reference frame of the camera, defined by

$$\tilde{\boldsymbol{b}}_c = \frac{1}{\sqrt{x^2 + y^2 + f^2}} \begin{Bmatrix} x \\ y \\ f \end{Bmatrix},$$

with a reference star catalog (e.g., Hipparcos), which tabulates unit star vectors in the ECI reference frame based on their right ascension (α) and declination (β) angles,

$$\hat{\boldsymbol{r}}_i = \begin{Bmatrix} \cos\alpha \ \ \sin\beta \\ \sin\alpha \ \ \sin\beta \\ \cos\beta \end{Bmatrix}.$$

It is worth mentioning here as well that a common practice for star identification is star catalog preparation, which tailors the reference catalog to the mission timeline by implementing proper star motion and merging or throwing out stars based on a visual magnitude thresholding criteria. In doing so, the overhead required to carry a reference star database onboard is reduced, and the

range searching algorithms become faster and more robust. The searching procedure of this sub-catalog is conducted until four or more stars are identified in the scene, upon which an attitude estimate is made using a quaternion solution of Wahba's problem (e.g., $\boldsymbol{q}$-method [25], QUEST [30], ESOQ [31], ESOQ2 [32]) in order to satisfy

$$\tilde{\boldsymbol{b}}_c = C_{c,i}\hat{\boldsymbol{r}}_i.$$

Figure 2.1: Real Star Image Taken with a Digital Camera in College Station, TX

2.4 Planets

The astronomical distances of the Sun, stars, and planets from Earth requires particular attention when attempting to leverage them for navigation. The first problem relates to the visibility of the observation, measured in terms of its brightness. A number of factors can influence the visibil-

ity of a celestial observation from an Earth-bound observer, including atmospheric scintillation and certain operational settings of the device used for imaging (e.g., exposure time). Additional problems arise due to the dynamic motion of celestial observations. Light, either emitted or reflected, travels through space from a source to an observer at a finite constant speed, c. Two angular corrections to the measured position vectors, commonly referred to as *light-time* and *stellar aberration* corrections, are required in order to resolve discrepancies between the observed and actual positions of an optical observation for a given time, t. The following subsections describe in detail how these corrections are implemented. Finally, we note here that the derivation provided for stellar aberration of planets is also applicable to the stars.

2.4.1 Visual Magnitude

The visibility of a planet can be characterized in several ways. For instance, the absolute magnitude, as defined in Ref. [33], corresponds to full solar illumination of a body (modeled as a Lambertian disk) when the source and observer are distanced by one astronomical unit, and their phase angle, α, is zero. Eq. (2.1) defines the absolute magnitude of a planet with respect to the apparent magnitude of the Sun in the Earth-Sun system,

$$M_p = m_{Sun} - 5\log_{10}\left(\frac{r_p}{d_0}\sqrt{a_p}\right) \tag{2.1}$$

where $m_{Sun} = -26.73$ and $d_0 = 1$ AU $\approx 1.49598 \cdot 10^8$ km. Values for the radius and geometric albedo of the visible planets, denoted in Eq. (2.1) as r_p and a_p, respectively, are provided in Table A.2. From the absolute magnitude the apparent (visual) magnitude of a planet is then computed by,

$$m_p = M_p + 2.5\log_{10}\left(\frac{d_{s,b}^2 d_{o,b}^2}{d_0^4 I(\alpha)}\right),$$

where $I(\alpha)$ is the phase integral, $d_{s,b}$ and $d_{o,b}$ are the body-to-Sun and body-to-observer distances shown in Fig. 2.2, and α is derived from the law of cosines,

$$\cos\alpha = \frac{d_{o,b}^2 + d_{s,b}^2 - d_{s,o}^2}{2\, d_{o,b}\, d_{s,b}}.$$

If we assume a planet can be modeled as an ideal diffuse reflecting sphere, then the phase integral becomes

$$I(\alpha) = \frac{2}{3}\left[\left(1 - \frac{\alpha}{\pi}\right)\cos\alpha + \frac{1}{\pi}\sin\alpha\right], \quad \alpha \in [0, \pi].$$

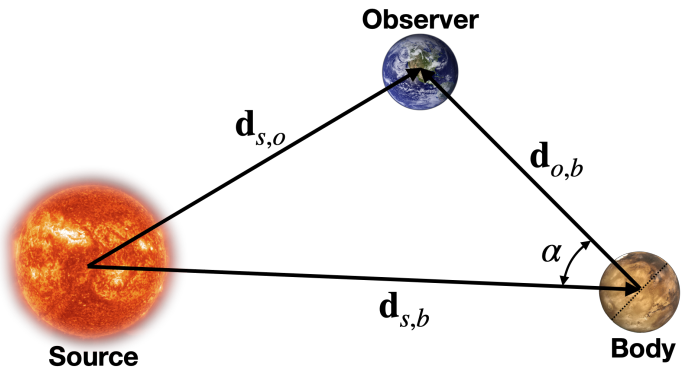

Figure 2.2: Geometry of Source-Body-Observer Phase Angle

2.4.2 Light-time Correction

The light-time phenomenon exists due to the fact that light speed is finite and not instantaneous, as was first argued by Ole Rømer in 1676 [34]. The light-time correction accounts for the displacement of a celestial observation as light travels from source-to-observer. For the visible planets, a reasonable approximation of this correction can be made by first estimating the elapsed light-time,

$$\Delta_t \approx \frac{\|\boldsymbol{r}_p(t)\|_2}{c},$$

where $c \approx 2.99792 \cdot 10^5$ km/s and $\boldsymbol{r}_p(t)$ is the position vector of a planet as measured from an Earth-bound observer (Fig. 2.3). The linear displacement of the planet over the elapsed light-time is then determined from

$$\boldsymbol{d}(t) = \boldsymbol{v}_p(t)\Delta_t,$$

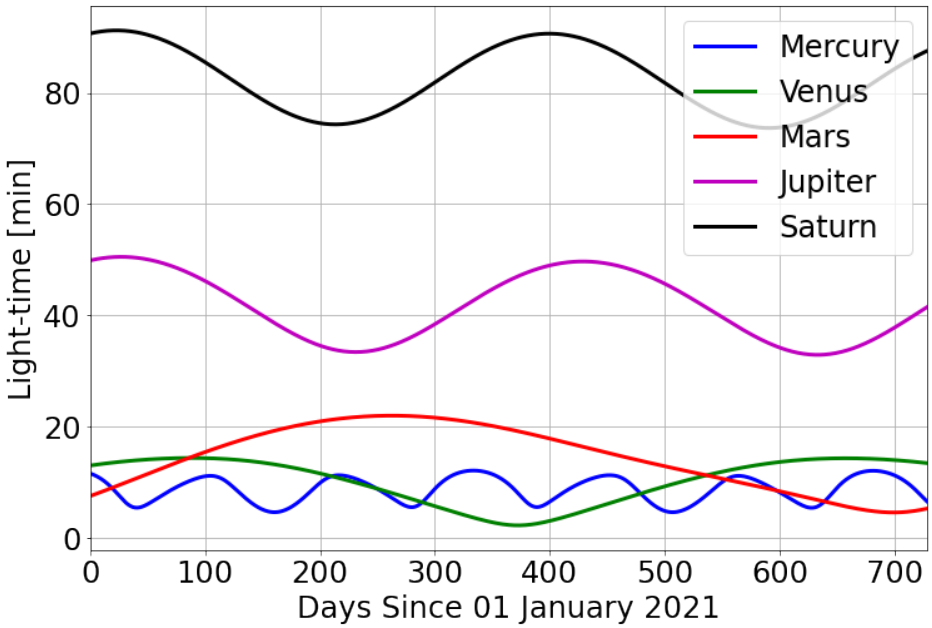

Figure 2.3: Travel-time of Light for the Visible Planets with Respect to an Earth-bound Observer

where $\boldsymbol{v}_p(t)$ is the orbital velocity vector of the planet (see Table A.3). Finally, if we assume that the observer's location is fixed for the duration of the elapsed light-time, Δ_t, the observed planet's position vector can be updated using the light-time correction, such that

$$\boldsymbol{r}_p(t - \Delta_t) = \boldsymbol{r}_p(t) - \boldsymbol{d}(t),$$

where $\boldsymbol{r}_p(t - \Delta_t)$ represents the actual location of a planet at the time of observation. Moreover, the light-time angle, can be computed using the geometry of Fig. 2.4,

$$\beta_{\ell t} = \arccos\left(\boldsymbol{r}_p^{\mathrm{T}}(t) \cdot \boldsymbol{r}_p(t - \Delta_t)\right).$$

2.4.3 Stellar Aberration Correction

The stellar aberration was first conjectured by James Bradley in 1727 [35] after he discovered discrepancies in his parallax measurements of Gamma Draconis, the brightest star in the

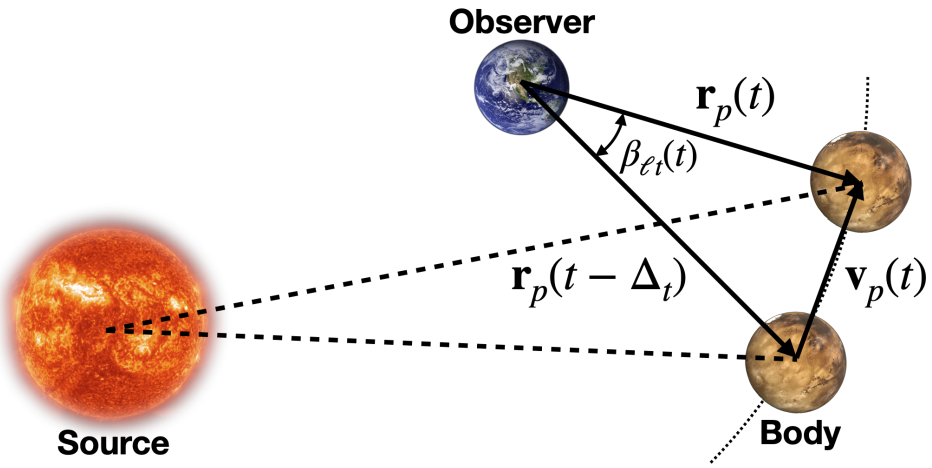

Figure 2.4: Light-time Geometry

Draco constellation. Bradley recognized that during a measurement period, the observation location moves with nonzero velocity due to the Earth's rotation rate and relative motion around the Sun. To account for this effect, Bradley considered a finite speed of light and derived the stellar aberration geometry and velocity triangle shown in Figs. 2.5 and 2.6, respectively. In doing so, he quantified the effect of the stellar aberration through a correction angle, β_{sa}, which is determined from the following procedure.

Let $\boldsymbol{v}_p$ and $\boldsymbol{v}_o$ be the velocities of the planet and the observer in the Heliocentric Inertial (HCI) reference frame, whose definition is such that its x-axis points to the ascending node of the Sun's equatorial plane on the ecliptic plane of J2000 and its z-axis points to the Sun's rotation axis. With reference to Fig. 2.5, which provides the stellar aberration geometry, the velocities can be decomposed into two orthogonal terms, $\boldsymbol{v}_{\parallel}$ (aligned with the direction of the observer-to-planet unit vector, $\hat{\boldsymbol{d}}_{p,o}$) and $\boldsymbol{v}_{\perp}$ (orthogonal to $\hat{\boldsymbol{d}}_{p,o}$). The vector components of each are provided by

$$\begin{cases} \boldsymbol{v}_{\parallel} = \left(\boldsymbol{v} \cdot \hat{\boldsymbol{d}}_{p,o}\right) \hat{\boldsymbol{d}}_{p,o} \\ \boldsymbol{v}_{\perp} = \boldsymbol{v} - \boldsymbol{v}_{\parallel} \end{cases} .$$

An Earth-bound observer could expect to have a linear velocity defined by

$$\boldsymbol{v}_o = \boldsymbol{v}_E + \boldsymbol{\omega}_E \times \boldsymbol{r}_o,$$

where $\boldsymbol{v}_E$ is the orbital velocity vector of Earth, $\boldsymbol{\omega}_E$ is the angular velocity vector of Earth, and $\boldsymbol{r}_o$ is the radial vector pointing from the Earth's center of mass to the observer, all of which are expressed in the HCI reference frame. Thus, the stellar aberration deflection angle, β_{sa}, is known from the velocity triangle shown in Fig. 2.6, whereby

$$\beta_{sa} = \arctan\left(\frac{\|\boldsymbol{v}_{o_\perp} - \boldsymbol{v}_{p_\perp}\|_2}{c}\right).$$

An iterative approach for implementing both light-time and stellar aberration corrections is outlined in [36].

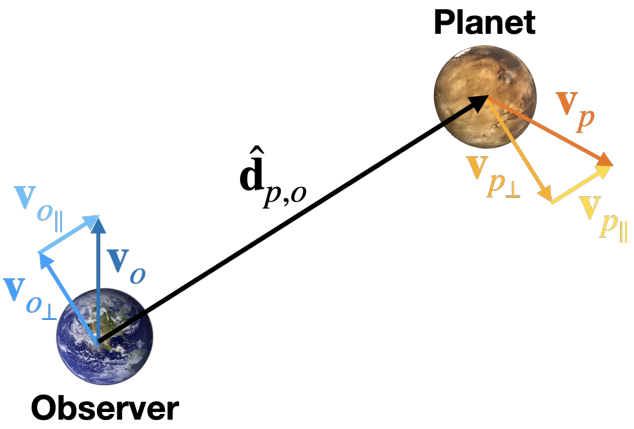

Figure 2.5: Stellar Aberration Geometry

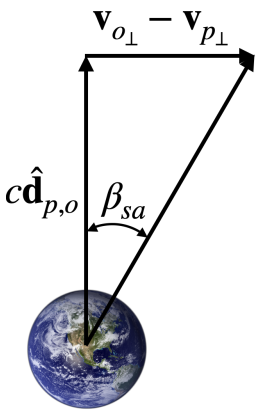

Figure 2.6: Stellar Aberration Velocity Triangle

2.5 Gravity

The Earth's gravitational field is a constantly evolving dynamical system that cannot be analytically defined by modern scientific frameworks. The local gravity vector that one would feel on

Earth is simply a static sampling of the ever-fluctuating mass around the observer. In the decades since launching several satellite surveillance missions, such as the GRACE [37] and GRACE-FO [38] orbiters, NASA, ESA, and others have managed to generate best-fit models for both Earth's shape and its gravitational field that are orders of magnitude more accurate than the earliest assumed representations of each.

2.5.1 Celestial Body Shape Models

The irregular shape of Earth has presented significant challenges to surveyers and cartographers throughout history. The early map makers considered the Earth to be a perfect sphere prior to Isaac Newton positing an oblate spheroid shape model in *Principia* [39]. The unit sphere can be thought of as a subclass of the ellipsoid, subject to the following governing equation,

$$x^2 + y^2 + z^2 = 1,$$

where $\{x, y, z\}$ are the Cartesian coordinates of a point on the surface. The ellipsoid generalizes the unit sphere by considering affine transformations with respect to its reference axes, and is typically used as a first approximation when modeling the shape of celestial bodies. The equation of the ellipsoid is

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1,$$

where $\{a, b, c\}$ are scalar distortions of its surface with respect to a particular reference direction. A common assumption is that $a \equiv b$, which is the mathematical definition of an axial-symmetric ellipsoid, whose governing equation is

$$\frac{x^2 + y^2}{a^2} + \frac{z^2}{c^2} = 1,$$

where a and c define the semi-major and semi-minor axes, respectively. The axial-symmetric ellipsoid description of Earth, which has been formalized as the WGS84, is reliably used as the standard shape model for most of today's GPS applications [40]. The WGS84 (as provided in

Table A.4) is defined by estimates of the Earth's equatorial and polar radii, and a flattening factor, f, which denotes the relative ratio between two ellipsoidal reference axes, such that

$$f = \frac{a - c}{a}.$$

Describing a point on the surface of these three-dimensional objects can be done in a number of ways. The traditional Cartesian coordinate set could be used; however, geodesists have historically opted for the angular definition using latitude and longitude. Referenced from the ECEF x-axis, where the mean equator intersects the line of zero meridian, the longitude, λ, measures the angular distance along Earth's equatorial plane. Referenced from the equatorial plane, the latitude measures the angular distance to the surface normal, whose definition depends on how the Earth is modeled. Fig. 2.7 illustrates this point by considering three definitions of the latitude. Consider a body whose center of mass is defined as its geocenter. For a perfect sphere, the surface normal at all points extends through the geocenter. On the other hand, the surface normal for points on the ellipsoid will not extend exactly through the geocenter unless they lie on the equatorial plane or exist at the poles. Thus, two unique definitions of latitude are required to differentiate a perfect sphere from an ellipsoid, namely, geocentric (θ) and geodetic (φ). As proved by Snyder in Ref. [41], a geometric relationship exists between the two latitudes. In particular,

$$\theta = \arctan\left((1 - f)^2 \tan\varphi\right). \tag{2.2}$$

Eq. (2.2) can be used to demonstrate the effectiveness of the ellipsoidal refinement to Earth's shape, as opposed to assuming a perfect sphere (Fig. 2.8). At worst, the angular deviation in latitudes is nearly 700 arcseconds, a discrepancy that would introduce over 21 km of position error on Earth.

Given that the ellipsoid is still a modest simplification of Earth's true shape, a third definition of latitude, the astronomic latitude (Φ), is also considered. Astronomic latitude defines the angle between Earth's equatorial plane and the true zenith of an observer, and is estimated using reference stars whose declination is known with extreme precision. A final definition falls out of the

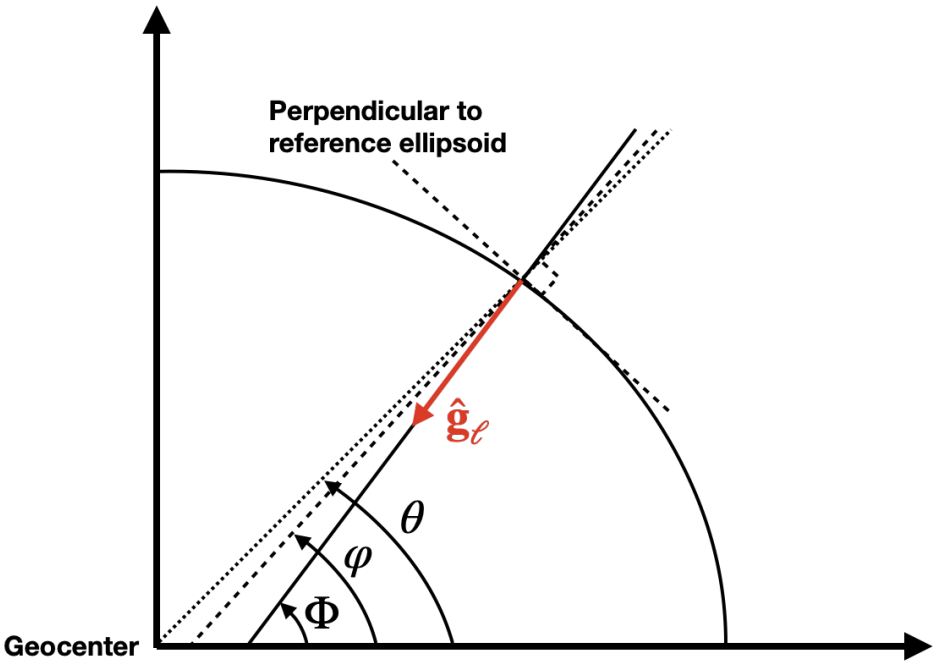

Figure 2.7: Definitions of the Latitude Angle

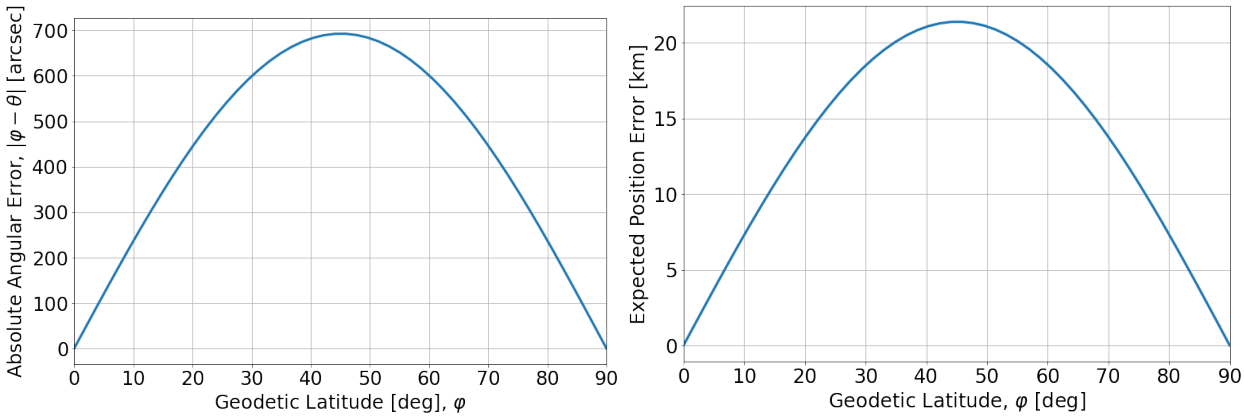

Figure 2.8: Expected Angular and Position Errors Due to Sphere/Ellipsoid Assumption

relationship between astronomic and geodetic latitudes, that is, the "deflection of vertical," which is comprised of North-South (ξ) and East-West (η) components,

$$\xi = \Phi - \varphi \quad \text{and} \quad \eta = (\Lambda - \lambda)\cos\varphi,$$

where Λ is the astronomic longitude of an observer with respect to the stars. Astronomic latitude and longitude are considered to be the best descriptors for both an observer's location on Earth and the pointing direction of the local unit gravity vector.

2.5.2 Continuous Description of the Geoid

The geoid was first described by Gauss [42] as the shape that the ocean surface would take if the only acceleration forces acting on it were Earth's gravitational field and the rotational dynamics of the planet. In reality, this is still a simplification of Earth's true topography. The geoid is particularly useful for describing the effect that mass anomalies (such as mountain ranges or caverns) have on the local pointing direction of gravity, which can deviate from the reference ellipsoid by tens, hundreds, or even thousands of arcseconds, depending on the celestial body (see Figs. 2.10, 2.13, 2.16). It is also worth pointing out that, unlike the axial-symmetric ellipsoid and other simplified shape models, there is no true mathematical description of the geoid. For Earth, the EGM2008 provides an approximation of Earth's gravitational potential surface by interpolating satellite measurement data using a normalized basis combination of Legendre orthogonal polynomials and harmonic coefficients, such that

$$V = \frac{GM}{r}\left(\sum_{n=0}^{N}\left(\frac{a}{r}\right)^n \sum_{m=0}^{n} \bar{P}_{nm}\sin\varphi\left[\bar{C}_{nm}\cos(m\lambda) + \bar{S}_{nm}\sin(m\lambda)\right]\right). \tag{2.3}$$

Here, GM is the geocentric gravitational constant, a is the semi-major axis, (φ, λ, r) are the spherical coordinates at which the evaluation is made, $\bar{P}_{nm}$ are the normalized Legendre basis functions of degree n and order m, and $\bar{C}_{nm}$ and $\bar{S}_{nm}$ are the normalized harmonic coefficients. Taking normalized partial derivatives of Eq. (2.3) with respect to latitude and longitude, we obtain a mathematical definition for the vertical deflections with respect to North-South and East-West reference directions,

$$\xi = -\frac{1}{\gamma r}\frac{\partial V}{\partial \varphi} = -\frac{GM}{\gamma r^2}\left(\sum_{n=2}^{N}\left(\frac{a}{r}\right)^n \sum_{m=0}^{n} \frac{\mathrm{d}\bar{P}_{nm}\sin\varphi}{\mathrm{d}\varphi}\left[\bar{C}_{nm}\cos(m\lambda) + \bar{S}_{nm}\sin(m\lambda)\right]\right), \tag{2.4}$$

and

$$\eta = -\frac{1}{\gamma r \cos\varphi}\frac{\partial V}{\partial \lambda} = -\frac{GM}{\gamma r^2 \cos\varphi}\left(\sum_{n=2}^{N}\left(\frac{a}{r}\right)^n \sum_{m=0}^{n} m\bar{P}_{nm}\sin\varphi\left[-\bar{C}_{nm}\sin(m\lambda) + \bar{S}_{nm}\cos(m\lambda)\right]\right), \tag{2.5}$$

where γ is a scalar value representing the theoretical normal gravity [40]. The analytical complexities associated with these infinite series (Eqs. (2.4), (2.5)) make them difficult to evaluate numerically. Through nonlinear regression, observation data can be approximated as a linear combination of coefficients, whose values are optimized using least-squares and a set of basis functions, such as Chebyshev orthogonal polynomials of the first kind. In one dimension, the Chebyshev basis set can be computed recursively from

$$\begin{cases} S_0(x) = 1 \\ S_1(x) = x \\ S_{k+1}(x) = 2xS_k(x) - S_{k-1}(x),\ \forall k > 1 \end{cases}. \tag{2.6}$$

The first derivatives of Eq. (2.6), which can be used to prove the existence of a critical point in a given data set, have a similar recursive form. Namely,

$$\begin{cases} \dfrac{dS_0(x)}{dx} = 0 \\ \dfrac{dS_1(x)}{dx} = 1 \\ \dfrac{dS_{k+1}(x)}{dx} = 2\left(S_k(x) + x\dfrac{dS_k(x)}{dx}\right) - \dfrac{dS_{k-1}(x)}{dx},\ \forall k > 1 \end{cases}.$$

Before performing polynomial regression, the independent coordinates of each data set must be mapped into the operational domain of the Chebyshev orthogonal polynomials, $x \in [-1, +1]$. Two mappings can be applied, the first (and simplest) is a linear map,

$$x = 2\left(\frac{\varphi - \varphi_{\min}}{\varphi_{\max} - \varphi_{\min}}\right) - 1 \quad \text{and} \quad y = 2\left(\frac{\lambda - \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right).$$

The second is the Gauss-Lobatto-Chebyshev nodes, which provide the optimal point spacing distribution for the Chebyshev orthogonal polynomials, and are given by

$$x_k = \cos\left(\pi \frac{2k-1}{2n}\right), \quad k = 1, \cdots, n$$

where n represents the number of discrete data points to be mapped within a given domain of interest. Since the values of ξ and η depend on two coordinates, φ and λ, the following bi-variate functions must be developed independently,

$$\begin{cases} \tilde{\xi} = f(\varphi, \lambda) = \sum_{i=0}^{N-j} \sum_{j=0}^{N} \alpha_{ij} S_i(\varphi) S_j(\lambda) \\ \tilde{\eta} = g(\varphi, \lambda) = \sum_{i=0}^{N-j} \sum_{j=0}^{N} \beta_{ij} S_i(\varphi) S_j(\lambda) \end{cases} . \tag{2.7}$$

where N represents the maximum degree of fit. Re-writing Eq. (2.7) in matrix-vector form, we have

$$\mathbb{S}\boldsymbol{\alpha} = \boldsymbol{\xi} \quad \text{and} \quad \mathbb{S}\boldsymbol{\beta} = \boldsymbol{\eta},$$

which are linear systems with respect to their unknown coefficients. Thus, from least-squares, we have

$$\boldsymbol{\alpha} = (\mathbb{S}^{\mathrm{T}}\mathbb{S})^{-1}\mathbb{S}^{\mathrm{T}}\boldsymbol{\xi} \quad \text{and} \quad \boldsymbol{\beta} = (\mathbb{S}^{\mathrm{T}}\mathbb{S})^{-1}\mathbb{S}^{\mathrm{T}}\boldsymbol{\eta}.$$

Finally, we can express Eq. (2.7) in terms of the now known coefficients, using

$$\begin{cases} \tilde{\xi} = f(\varphi, \lambda) = \sum_{k=1}^{N} \alpha_k \mathbb{S}_k(\varphi, \lambda) \\ \tilde{\eta} = g(\varphi, \lambda) = \sum_{k=1}^{N} \beta_k \mathbb{S}_k(\varphi, \lambda) \end{cases} . \tag{2.8}$$

2.5.3 Earth Gravitational Model (EGM2008)

The EGM2008 [43] is a gravitational model of Earth developed by the National Geospatial-Intelligence Agency (NGA). It uses a degree 2159 spherical harmonic best-fitting of discrete measurement data provided by the GRACE satellite missions to refine the local pointing direction of the unit gravity vector with respect to the standard axial-symmetric ellipsoid model of Earth, the WGS84 (see Table A.4). The EGM2008, at the advent of this work, was considered to be the most precise geoid model of Earth, providing 2.5 arcminute resolution over the entirety of the world's surface. Fig. 2.9 was generated by reading two binary files that contained discretized North-South and East-West vertical deflection data in units of arcseconds. As standalone grids, it is difficult to identify any of the Earth's prominent topological features. Therefore, the total deflection ($\sqrt{\xi^2 + \eta^2}$) was computed and plotted as shown in Fig. 2.10. From this perspective, the mountainous regions of the world are evident. In particular, the most prominent vertical deflections were found to be in the Himalayas of Asia and the Andes of South America. In order to demonstrate the effectiveness of the geoid correction, a sub-grid around the Himalayan mountain range was selected, and the associated vertical deflections were stored in separate sub-grids. Fig. 2.11 shows the highly nonlinear behavior of the vertical deflection values within the Himalayas. Properly fitting this region with two-dimensional Chebyshev orthogonal polynomials required a study of how the maximum absolute fitting error across the entire sub-grid changed with respect to the degree of polynomial fit. Fig. 2.11 also shows how these error trends evolve with increasing degree of fit. For preliminary testing purposes, a degree 20 fit was determined to be sufficient, as it is not overly-expensive to compute and clearly avoids the over-fitting behavior defined by the discontinuity that occurs in both error trends around degree 37.

2.5.4 Mars Gravitational Model (MGM2011)

The MGM2011 [44] is a gravitational model of Mars developed by researchers at the Curtin University of Technology in Austalia. It uses discrete measurement data provided by the MOLA satellite mission to estimate vertical deflections of the local unit gravity vector from the Mars ref-

erence ellipsoid at 3 arcminute resolution over the entire Martian surface. Fig. 2.12 was generated by reading two binary files that contained discretized North-South and East-West vertical deflection data in units of arcseconds. As standalone grids, the Martian mountains (Olympus Mons, in particular, as well as Ascraeus Mons, Pavonis Mons, and Arsia Mons) are easy to identify due to their significant masses inducing large deflections of the vertical plumbline. Just as we did in the previous subsection, the total deflection ($\sqrt{\xi^2 + \eta^2}$) was computed and plotted as shown in Fig. 2.13. Using this sub-grid of values, the most significant variation in vertical deflections were found to be in and around Olympus Mons. In order to demonstrate the effectiveness of the geoid correction, a sub-grid around Olympus Mons was selected, and the associated vertical deflections were stored in separate sub-grids. Fig. 2.14 describes this region in terms of vertical deflections, wherein the steep ascent of the volcano walls have been captured beautifully. A similar study, as described in the previous subsection, was conducted to determine how the maximum absolute fitting error across the entire sub-grid changes with respect to the degree of fit. Fig. 2.14 also shows how these error trends evolve with increasing degree of fit. For preliminary testing purposes, a degree 20 fit was determined to be sufficient, as it is not overly-expensive to compute and clearly avoids the over-fitting behavior defined by the discontinuity that occurs in both error trends around degree 38.

2.5.5 Lunar Gravitational Model (LGM2011)

The LGM2011 [45] is a gravitational model of the Moon developed by researchers at the Curtin University of Technology in Austalia. It uses discrete measurement data provided by the SELENE and LOLA satellite missions to estimate vertical deflections of the local unit gravity vector from the Moon's homogeneous mass-sphere, as opposed to the reference ellipsoids used in the Earth and Mars cases, at 3 arcminute resolution over the entire Lunar surface. Fig. 2.15 was generated by reading two binary files that contained discretized North-South and East-West vertical deflection data in units of arcseconds. Just as we did in the previous subsections, the total deflection ($\sqrt{\xi^2 + \eta^2}$) was computed and plotted as shown in Fig. 2.16. Using this sub-grid of values, the most significant variation in vertical deflections were found to be associated with an unidentifiable

crater in the North-Eastern region of the grid. The variation in vertical deflection extrema for the Moon, as well as the resolution of the data set, are quite similar to the Martian data sets described in the prior subsection. To this end, a numerical analysis of the LGM2011 was not considered for this work.

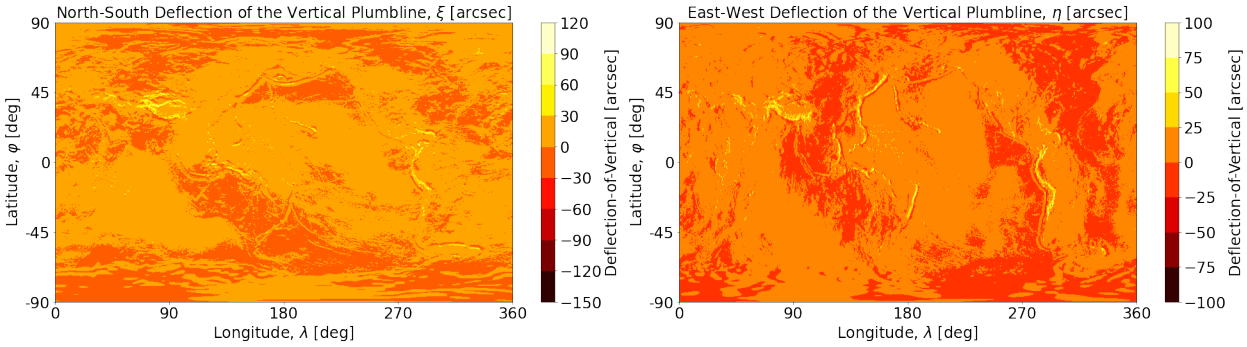

Figure 2.9: EGM2008 Discrete Data Description; North-South and East-West Deflections of the Vertical Plumb-line with Respect to WGS84 Reference Ellipsoid

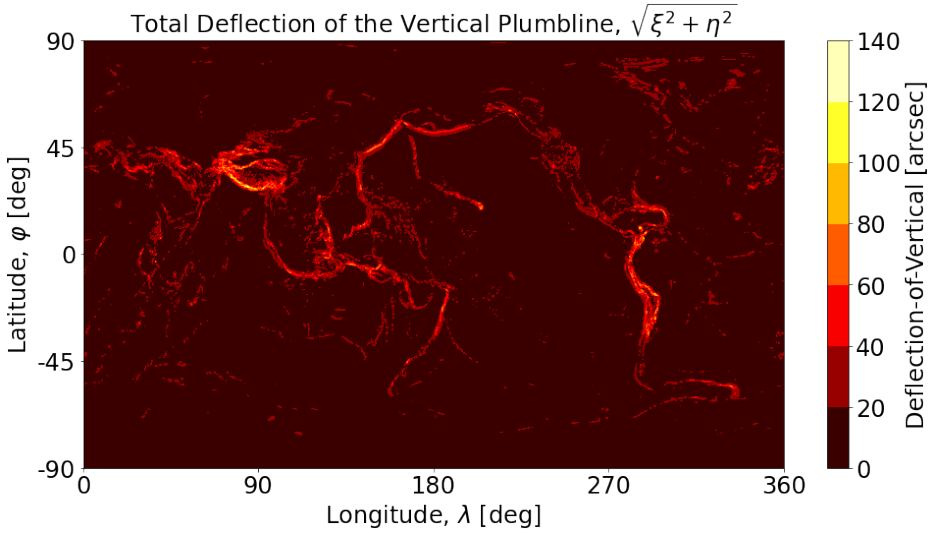

Figure 2.10: EGM2008 Discrete Data Description; Total Deflection of the Vertical Plumb-line with Respect to WGS84 Reference Ellipsoid

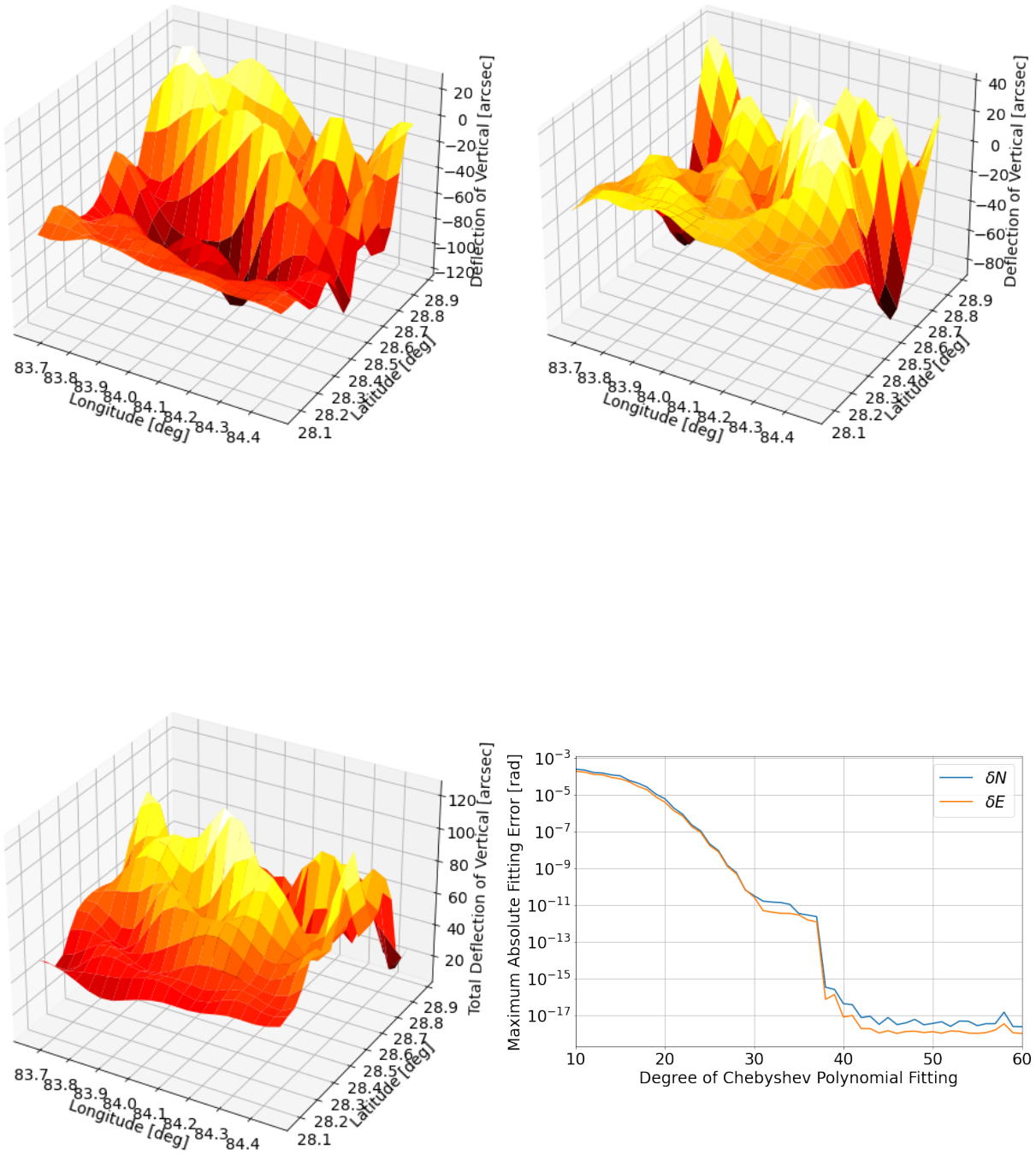

Figure 2.11: Vertical Deflection Sub-grids, and Maximum Absolute Fitting Error with Respect to Degree of Chebyshev Basis Functions, for the Himalayas

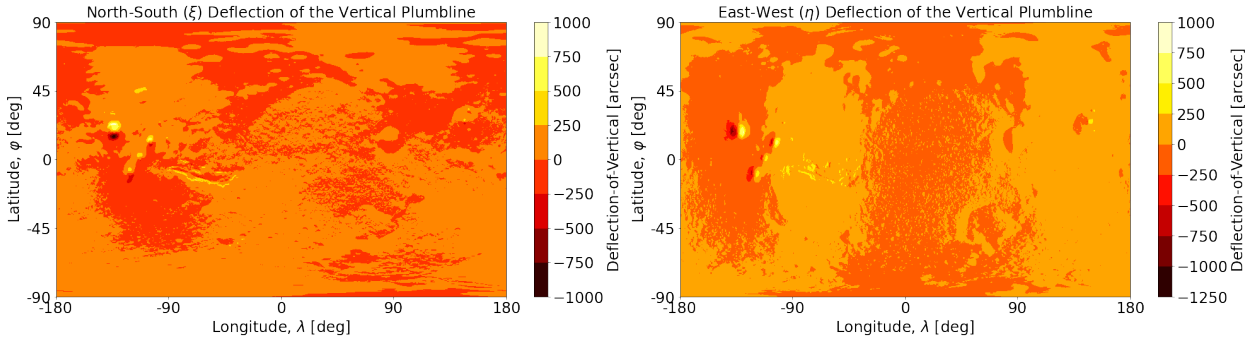

Figure 2.12: MGM2011 Discrete Data Description; North-South and East-West Deflections of the Vertical Plumb-line with Respect to Mars Reference Ellipsoid

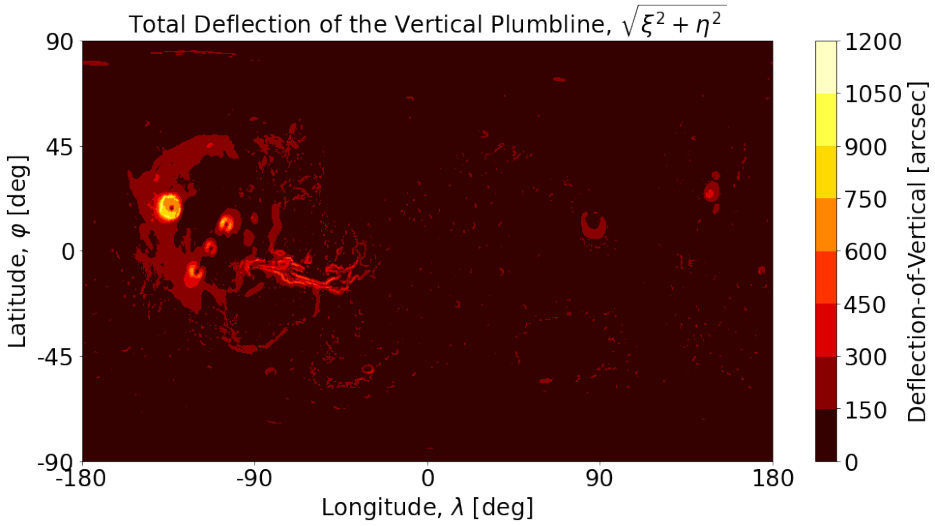

Figure 2.13: MGM2011 Discrete Data Description; Total Deflection of the Vertical Plumb-line with Respect to Mars Reference Ellipsoid

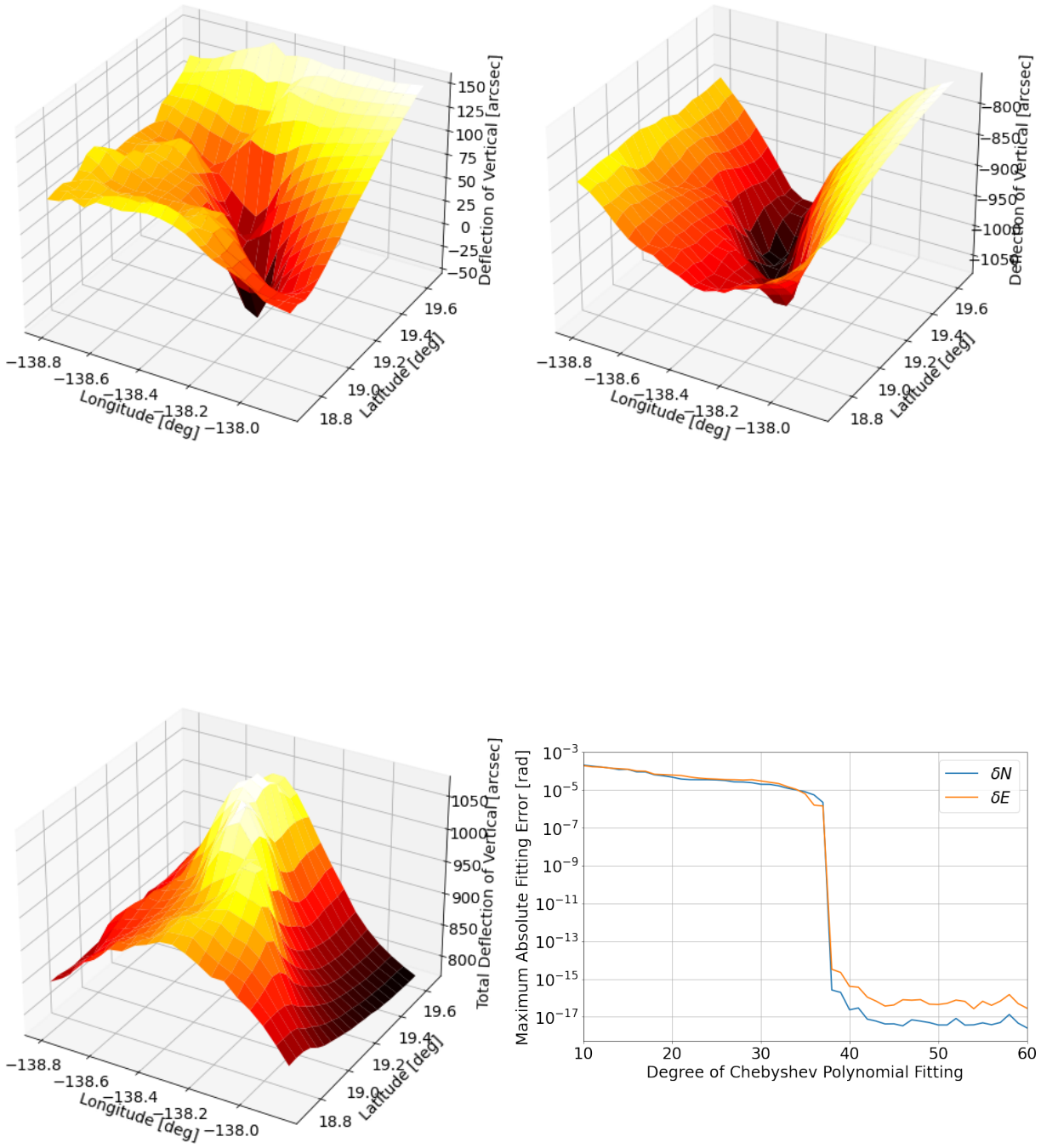

Figure 2.14: Vertical Deflection Sub-grids, and Maximum Absolute Fitting Error with Respect to Degree of Chebyshev Basis Functions, for Olympus Mons

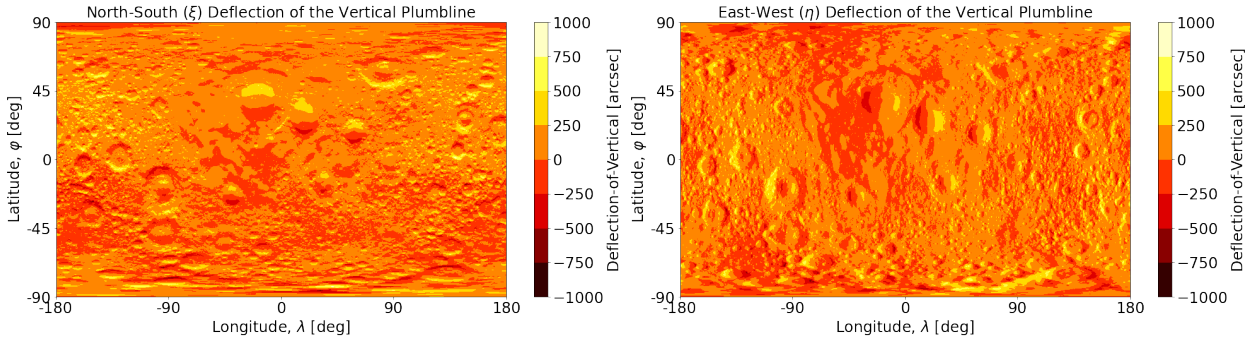

Figure 2.15: LGM2011 Discrete Data Description; North-South and East-West Deflections of the Vertical Plumb-line with Respect to Moon Reference Sphere

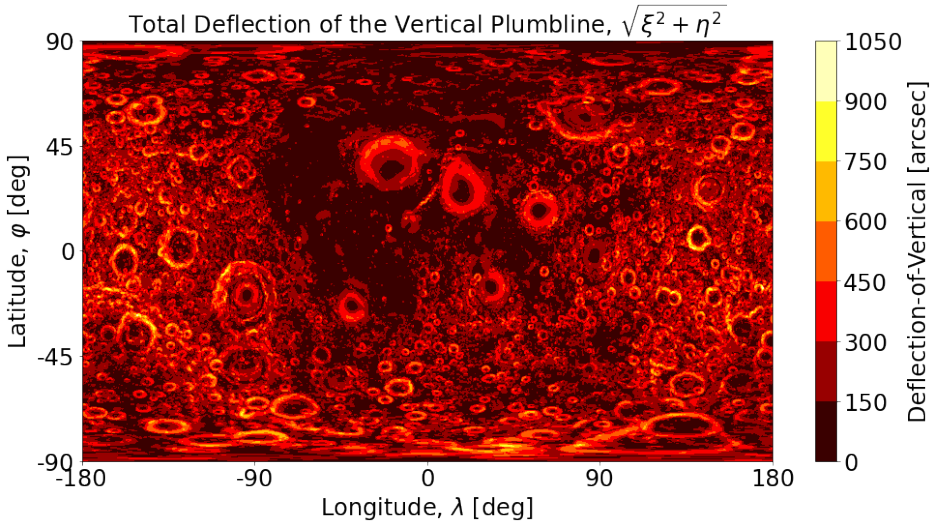

Figure 2.16: LGM2011 Discrete Data Description; Total Deflection of the Vertical Plumb-line with Respect to Moon Reference Sphere

3.1 System Description

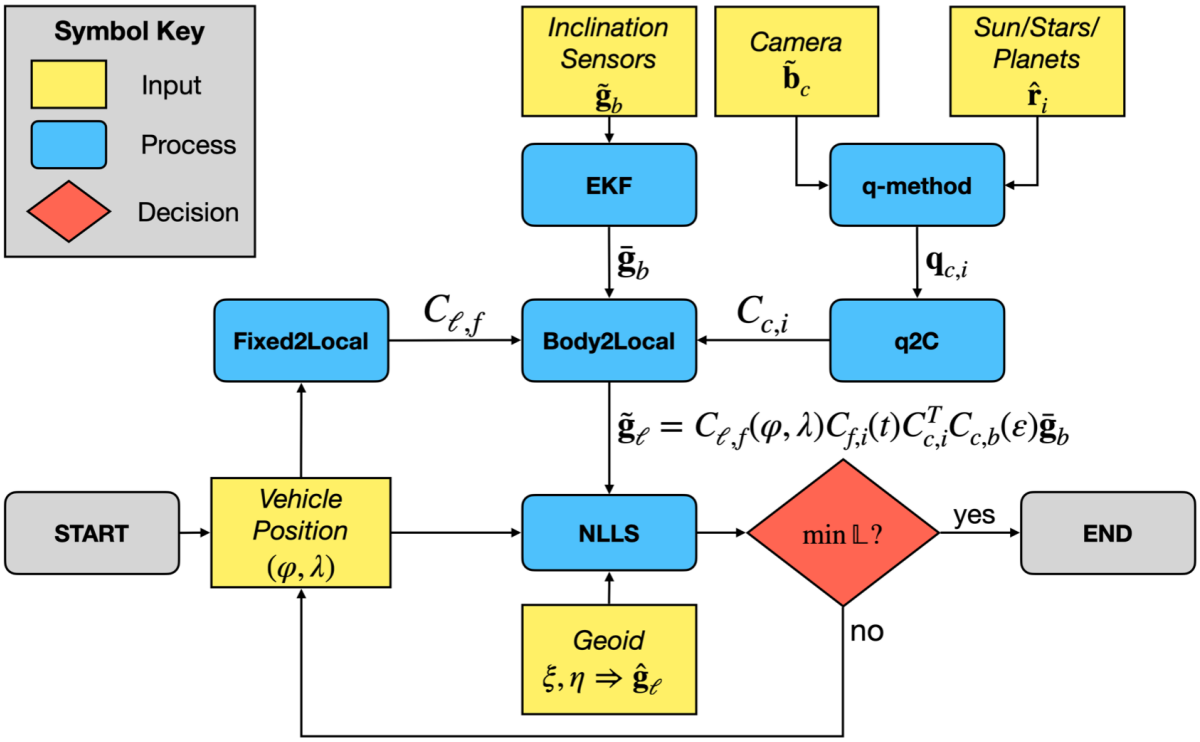

Figure 3.1: Flowchart of Geoid Correction Algorithm

The idea for the geoid correction, which we formalize now as the iterative sequence shown in Fig. 3.1, was born out of the derivations and discussion provided by Parish et al. in Ref. [22] (and later field-tested in Ref. [23]). This work considers the same real-world sensor suite that was proposed in these seminal papers (i.e., a pair of inclinometers and a digital camera) for performing position estimation. The inclinometer set measures relative tilt of the platform with respect to the local horizon defined by the unit gravity vector. The drawback in using these sensors is that they measure static tilt, which makes dynamic motion of the vehicle difficult to estimate. In theory,

one could swap out these sensors for an inertial measurement unit (IMU), which measures both relative tilt and rotation rates of the platform. In doing so, the same tilt angles provided by the inclinometer set could be dynamically estimated through sensor fusion of accelerometer (tilt sensor) and gyroscope (rate sensor) measurements. Irrespective of the artifacts that comprise the suite, measurement biases and random noise are inherent to all sensory devices (primarily attributable to thermal cycling and electrical fluctuations). Mitigating these measurement errors is typically done using a Kalman filter, whose derivation is provided in the next chapter. As Fig. 3.1 shows, the extended Kalman filter, labeled as `EKF`, is just one of several functions that make up the entire geoid correction algorithm. Other prominent functions include `Body2Local`, `Fixed2Local`, `NLLS`, `q2C`, and `q-method`. Table 3.1 provides a brief summary of each of these functions, along with their expected input(s) and output(s), and the remaining sections of this chapter include the derivations necessary for implementing the geoid correction.

Name	Description	Input(s)	Output(s)
`Body2Local`	Maps unit gravity measurement vectors from the inclinometer (body) reference frame to the local (ENU) reference frame	$\tilde{\boldsymbol{g}}_b$, $C_{c,b}(\varepsilon)$, $C_{c,i}(\boldsymbol{q})$, $C_{f,i}(t)$, $C_{\ell,f}(\varphi,\lambda)$	$\tilde{\boldsymbol{g}}_\ell$
`EKF`	Filters zero-mean Gaussian noise out of inclinometer measurement angles using a first-order linearization of the system dynamics	$\tilde{\boldsymbol{g}}_b(\theta_x,\theta_y)$, σ_θ	$\bar{\boldsymbol{g}}_b$
`Fixed2Local`	Constructs fixed-to-local transformation matrix	φ_0, λ_0	$C_{\ell,f}$
`NLLS`	Applies iterative nonlinear least-squares algorithm detailed in Section 3.4	φ_0, λ_0, $\hat{\boldsymbol{g}}(\xi,\eta)$, $\tilde{\boldsymbol{g}}_\ell$	φ, λ
`q2C`	Maps a quaternion to a direction cosine matrix	$\boldsymbol{q}$	C
`q-method`	Applies $\boldsymbol{q}$-method to compute a quaternion	$\tilde{\boldsymbol{b}}_c$, $\hat{\boldsymbol{r}}_i$	$\boldsymbol{q}_{c,i}$

Table 3.1: Summary of Geoid Correction Functions

3.2 Derivation of the Reference Unit Gravity Vector

A reference unit gravity vector can be derived from the geoid. In particular, the North-South (ξ) and East-West (η) deflections of the vertical plumb-line are used to form the first two components of $\hat{\mathbf{g}}_\ell$. In the local reference frame, the unit gravity vector is

$$\hat{\mathbf{g}}_\ell = \left\{ \begin{array}{c} \sin\eta \\ \sin\xi \\ -\sqrt{1-\sin^2\eta-\sin^2\xi} \end{array} \right\}. \tag{3.1}$$

In order to apply a multivariate Newton update, the Jacobian matrix of the reference unit gravity vector must be derived. Here, we have

$$\hat{\mathbb{J}} = \left[\frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \varphi}, \quad \frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \lambda} \right] = \left[\frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \xi}\frac{\partial \xi}{\partial \varphi} + \frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \eta}\frac{\partial \eta}{\partial \varphi}, \quad \frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \xi}\frac{\partial \xi}{\partial \lambda} + \frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \eta}\frac{\partial \eta}{\partial \lambda} \right].$$

These partial derivatives are easily computed from Eq. (3.1) as,

$$\frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \xi} = \left\{ \begin{array}{c} 0 \\ \cos\xi \\ \dfrac{\sin\xi\ \cos\xi}{\sqrt{1-\sin^2\eta-\sin^2\xi}} \end{array} \right\}$$

and

$$\frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \eta} = \left\{ \begin{array}{c} \cos\eta \\ 0 \\ \dfrac{\sin\eta\ \cos\eta}{\sqrt{1-\sin^2\eta-\sin^2\xi}} \end{array} \right\}.$$

Furthermore, Eq. (2.8) yields the remaining partial derivatives necessary to define the Jacobian matrix. Namely,

$$\begin{cases} \dfrac{\partial \xi}{\partial \varphi} = \dfrac{\partial}{\partial \varphi} f(\varphi, \lambda) = \displaystyle\sum_{k=1}^{N} \alpha_k \frac{\partial S_k(\varphi, \lambda)}{\partial \varphi} \\ \\ \dfrac{\partial \xi}{\partial \lambda} = \dfrac{\partial}{\partial \lambda} f(\varphi, \lambda) = \displaystyle\sum_{k=1}^{N} \alpha_k \frac{\partial S_k(\varphi, \lambda)}{\partial \lambda} \end{cases} \quad \begin{cases} \dfrac{\partial \eta}{\partial \varphi} = \dfrac{\partial}{\partial \varphi} g(\varphi, \lambda) = \displaystyle\sum_{k=1}^{N} \beta_k \frac{\partial S_k(\varphi, \lambda)}{\partial \varphi} \\ \\ \dfrac{\partial \eta}{\partial \lambda} = \dfrac{\partial}{\partial \lambda} g(\varphi, \lambda) = \displaystyle\sum_{k=1}^{N} \beta_k \frac{\partial S_k(\varphi, \lambda)}{\partial \lambda} \end{cases}$$

3.3 Derivation of the Measurement Unit Gravity Vector

A pair of inclination sensors can also be used to measure the orientation of the local unit gravity vector. This vector is observed in the body reference frame of the tilt sensors; therefore, a series of transformations are required in order to map the geoid description of gravity, as described in the previous subsection, to its measured counterpart. In the body reference frame, we have

$$\tilde{\boldsymbol{g}}_b = \begin{Bmatrix} \tilde{g}_x \\ \tilde{g}_y \\ -\sqrt{1 - \tilde{g}_x^2 - \tilde{g}_y^2} \end{Bmatrix}_b = \begin{Bmatrix} \cos \vartheta_x \\ \dfrac{\cos \vartheta_y - \cos \vartheta_x \sin \varepsilon}{\cos \varepsilon} \\ -\sqrt{1 - \tilde{g}_x^2 - \tilde{g}_y^2} \end{Bmatrix}_b , \tag{3.2}$$

where ϑ_x and ϑ_y represent the angular deviation of the local unit gravity vector with respect to the two body reference axes. The measured unit gravity vector can then be expressed in the ECEF such that independence from latitude and longitude is preserved. Here, we have

$$\tilde{\boldsymbol{g}}_f = C_{f,i}(t)\; C_{c,i}^{\mathrm{T}}(\boldsymbol{q})\; C_{c,b}(\varepsilon)\; \tilde{\boldsymbol{g}}_b.$$

A local description of the measurement unit gravity vector, $\tilde{\boldsymbol{g}}_\ell$, requires us to consider $C_{\ell,f}$, which depends on the sensitivity parameters, φ and λ. In doing so, we can write $\tilde{\boldsymbol{g}}_\ell$ as follows,

$$\begin{aligned}\tilde{\boldsymbol{g}}_\ell = C_{\ell,f}(\varphi,\lambda)\tilde{\boldsymbol{g}}_f &= \begin{bmatrix} -\sin\lambda & \cos\lambda & 0 \\ -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ \cos\varphi\cos\lambda & \cos\varphi\sin\lambda & \sin\varphi \end{bmatrix} \begin{Bmatrix} g_x \\ g_y \\ g_z \end{Bmatrix}_f \\ &= \begin{Bmatrix} -g_{x,f}\sin\lambda + g_{y,f}\cos\lambda \\ -g_{x,f}\sin\varphi\cos\lambda - g_{y,f}\sin\varphi\sin\lambda + g_{z,f}\cos\varphi \\ g_{x,f}\cos\varphi\cos\lambda + g_{y,f}\cos\varphi\sin\lambda + g_{z,f}\sin\varphi \end{Bmatrix}_\ell .\end{aligned} \tag{3.3}$$

Taking the partial derivatives of Eq. (3.3), we have

$$\frac{\partial\tilde{\boldsymbol{g}}_\ell}{\partial\varphi} = \begin{Bmatrix} 0 \\ -g_{x,f}\cos\varphi\cos\lambda - g_{y,f}\cos\varphi\sin\lambda - g_{z,f}\sin\varphi \\ -g_{x,f}\sin\varphi\cos\lambda - g_{y,f}\sin\varphi\sin\lambda + g_{z,f}\cos\varphi \end{Bmatrix}$$

and

$$\frac{\partial\tilde{\boldsymbol{g}}_\ell}{\partial\lambda} = \begin{Bmatrix} -g_{x,f}\cos\lambda - g_{y,f}\sin\lambda \\ g_{x,f}\sin\varphi\sin\lambda - g_{y,f}\sin\varphi\cos\lambda \\ -g_{x,f}\cos\varphi\sin\lambda + g_{y,f}\cos\varphi\cos\lambda \end{Bmatrix}.$$

Finally, we can derive a measurement Jacobian matrix as follows,

$$\begin{aligned}\tilde{\mathbb{J}} &= \left[\frac{\partial\tilde{\boldsymbol{g}}_\ell}{\partial\varphi}, \quad \frac{\partial\tilde{\boldsymbol{g}}_\ell}{\partial\lambda}\right] \\ &= \begin{bmatrix} 0 & -g_{x,f}\cos\lambda - g_{y,f}\sin\lambda \\ -g_{x,f}\cos\varphi\cos\lambda - g_{y,f}\cos\varphi\sin\lambda - g_{z,f}\sin\varphi & g_{x,f}\sin\varphi\sin\lambda - g_{y,f}\sin\varphi\cos\lambda \\ -g_{x,f}\sin\varphi\cos\lambda - g_{y,f}\sin\varphi\sin\lambda + g_{z,f}\cos\varphi & -g_{x,f}\cos\varphi\sin\lambda + g_{y,f}\cos\varphi\cos\lambda \end{bmatrix}.\end{aligned}$$

3.4 Nonlinear Iterative Estimation of Geographic Coordinates

With the reference and measurement unit gravity vector definitions in mind, let us define a loss vector, of size (3×1), as the difference between their components,

$$\mathbb{L} = \hat{\boldsymbol{g}}_\ell(\eta, \xi) - \underbrace{C_{\ell,f}(\varphi, \lambda)\, C_{f,i}(t)\, C^{\mathrm{T}}_{c,i}(\boldsymbol{q})\, C_{c,b}(\varepsilon)\, \tilde{\boldsymbol{g}}_b(\vartheta_x, \vartheta_y)}_{\tilde{\boldsymbol{g}}_\ell(\varphi,\lambda,t,\boldsymbol{q},\varepsilon,\vartheta_x,\vartheta_y)}. \tag{3.4}$$

Regional convergence can be somewhat predictive based on the loss function's behavior over a data set. For both test cases (namely, the Himalayan mountain range provided by EGM2008 and the region around Olympus Mons provided by MGM2011), the L_2-norm of the loss function was computed over the entirety of the sub-grids with respect to their midpoints. Fig 3.2 shows this result, in which the darkest section of the surface represents the coordinates that minimize the Euclidean norm of the loss function.

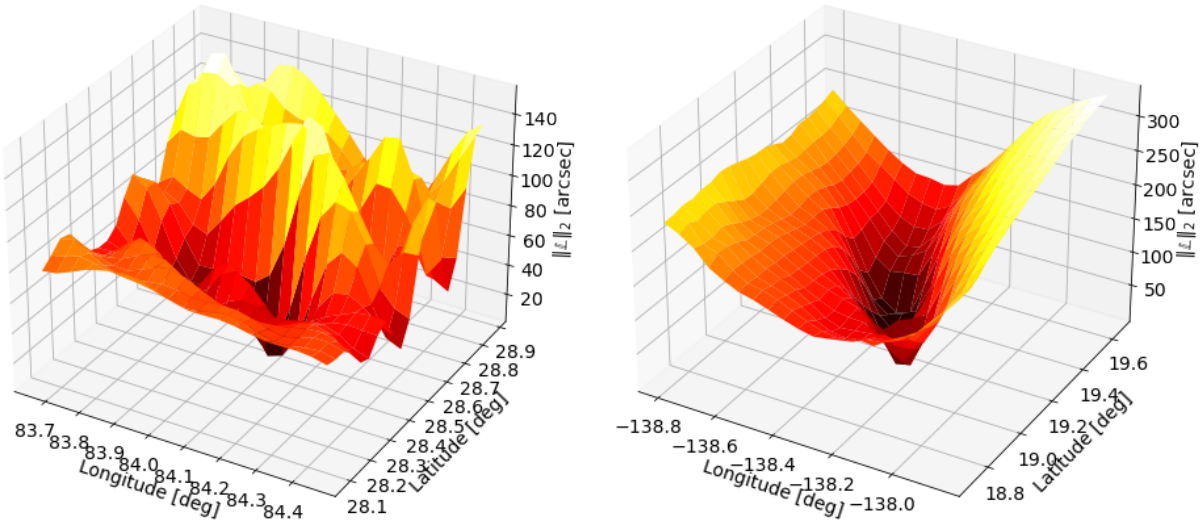

Figure 3.2: L_2-norm of Loss Function for the Himalayan Mountain Range (left) and Olympus Mons (right)

Let us now define a Jacobian difference matrix, of size (3×2), whose values depend on the reference and measurement Jacobian matrices described in previous sections,

$$\mathbb{J} = \hat{\mathbb{J}} - \tilde{\mathbb{J}} = \left[\frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \varphi}, \quad \frac{\partial \hat{\boldsymbol{g}}_\ell}{\partial \lambda} \right] - \left[\frac{\partial \tilde{\boldsymbol{g}}_\ell}{\partial \varphi}, \quad \frac{\partial \tilde{\boldsymbol{g}}_\ell}{\partial \lambda} \right].$$

The goal in applying the Newton method is to minimize Eq. (3.4), or more specifically, the difference between the unit gravity vector provided by the geoid vertical deflection data and the unit gravity vector measured by a pair of inclination sensors. Since we seek $\mathbb{L} = \mathbf{0}$, a first-order Taylor series expansion can be used to derive the following Newton update equation,

$$\begin{Bmatrix} \varphi \\ \lambda \end{Bmatrix}_{k+1} = \begin{Bmatrix} \varphi \\ \lambda \end{Bmatrix}_{k} - \left(\mathbb{J}_k^{\mathrm{T}} \, \mathbb{J}_k \right)^{-1} \mathbb{J}_k^{\mathrm{T}} \, \mathbb{L}_k.$$

4.1 Gravity Covariance Matrix

Since the pair of inclination sensors are likely to be corrupted by random noise, a covariance analysis was conducted in order to quantify this effect on the system's ability to maintain accurate measurements of the gravity vector over time. Let the two angles measured by the inclinometer set, ϑ_x and ϑ_y, be perturbed by additive Gaussian noise (implying a local approximation to the solution space),

$$\vartheta_x \sim \mathcal{N}\left(\mu_x, \sigma_\vartheta^2\right) \quad \text{and} \quad \vartheta_y \sim \mathcal{N}\left(\mu_y, \sigma_\vartheta^2\right).$$

where μ_x and μ_y are the mean values of ϑ_x and ϑ_y, respectively, and σ_ϑ^2 is the measurement variance associated with the inclinometers. Assuming the hardware is identical on each of the measurement axes, these angles can be rewritten as a fixed bias with additive zero-mean noise,

$$\begin{cases} \vartheta_x = \mu_x + \delta_\vartheta \\ \vartheta_y = \mu_y + \delta_\vartheta \end{cases} \qquad \text{where} \quad \delta_\vartheta \sim \mathcal{N}\left(0, \sigma_\vartheta^2\right).$$

To estimate the unit gravity vector direction in the inclinometer (body) reference frame, Eq. (3.2) requires that we evaluate the the first moment, or mean value, of $\cos\vartheta_x$ and $\cos\vartheta_y$ using the expectation operator, $E\{\cdot\}$. In this case, we drop the axis-specific subscript, and arrive at

$$E\{\cos\vartheta\} = E\{\cos(\mu + \delta_\vartheta)\} = \cos\mu \cdot E\{\cos\delta_\vartheta\} - \sin\mu \cdot E\{\sin\delta_\vartheta\}. \tag{4.1}$$

Now, because sine is an odd function, we know that the expected value of an odd function of a zero-mean Gaussian random variable is identically zero. Therefore, Eq. (4.1) simplifies as

$$E\{\cos\vartheta\} = \cos\mu \cdot E\{\cos\delta_\vartheta\}.$$

We compute $E\{\cos\delta_\vartheta\}$ by expanding it via a Maclaurin series. Here too, we drop a subscript ϑ, such that

$$E\{\cos\delta\} = E\left\{\sum_{k=0}^{\infty}\frac{(-1)^k}{(2k)!}\delta^{2k}\right\} = \sum_{k=0}^{\infty}\frac{(-1)^k}{(2k)!}E\left\{\delta^{2k}\right\}. \tag{4.2}$$

From [46], we know that a zero-mean Gaussian angle has the following identity,

$$E\left\{\delta^{2k}\right\} = 1\cdot 3\cdot 5\cdot\ldots\cdot(2k-1)\,\sigma^{2k} = \frac{(2k)!}{2^k\,k!}\,\sigma^{2k}. \tag{4.3}$$

Therefore, we may substitute Eq. (4.3) into Eq. (4.2) and obtain the expected value of cosine of a zero-mean Gaussian angle,

$$E\{\cos\delta\} = \sum_{k=0}^{\infty}\frac{(-1)^k}{2^k\,k!}\,\sigma^{2k} = \sum_{k=0}^{\infty}\frac{(-1)^k}{k!}\left(\frac{\sigma^2}{2}\right)^k = e^{-\sigma^2/2}.$$

Now, we reintroduce subscripting in order to write individual definitions for the expectations of $\cos\vartheta_x$ and $\cos\vartheta_y$,

$$\begin{cases} E\{\cos\vartheta_x\} = e^{-\sigma_\vartheta^2/2}\,\cos\mu_x \\ E\{\cos\vartheta_y\} = e^{-\sigma_\vartheta^2/2}\,\cos\mu_y \end{cases}. \tag{4.4}$$

Consequently, the expected values of the first two components of the measured unit gravity vector (see Eq. (3.2)), $\hat{g}_x = E\{g_x\}$ and $\hat{g}_y = E\{g_y\}$, can be written as,

$$\hat{g}_x = e^{-\sigma_\vartheta^2/2}\,\cos\mu_x \tag{4.5}$$

and

$$\hat{g}_y = e^{-\sigma_\vartheta^2/2}\left(\frac{\cos\mu_y}{\cos\varepsilon} - \cos\mu_x\,\tan\varepsilon\right). \tag{4.6}$$

Now, the symmetric gravity direction covariance matrix is defined as,

$$P_b = E\left\{(\boldsymbol{g}_b - \hat{\boldsymbol{g}}_b)(\boldsymbol{g}_b - \hat{\boldsymbol{g}}_b)^{\mathrm{T}}\right\} = \begin{bmatrix} E\{g_x^2\} - \hat{g}_x^2 & E\{g_x g_y\} - \hat{g}_x\,\hat{g}_y & E\{g_x g_z\} - \hat{g}_x\,\hat{g}_z \\ E\{g_x g_y\} - \hat{g}_x\,\hat{g}_y & E\{g_y^2\} - \hat{g}_y^2 & E\{g_y g_z\} - \hat{g}_y\,\hat{g}_z \\ E\{g_x g_z\} - \hat{g}_x\,\hat{g}_z & E\{g_y g_z\} - \hat{g}_y\,\hat{g}_z & E\{g_z^2\} - \hat{g}_z^2 \end{bmatrix}.$$

Computing the remaining terms of this matrix requires us to estimate an expected value containing $E\{\cos^2\delta\}$ and $E\{\sin^2\delta\}$, namely

$$E\{\cos^2\vartheta\} = E\{(\cos\mu\,\cos\delta - \sin\mu\,\sin\delta)^2\} = \cos^2\mu\,E\{\cos^2\delta\} + \sin^2\mu\,E\{\sin^2\delta\}.$$

It turns out that these expectations can also be computed using a Maclaurin series, whereby

$$\begin{aligned} E\{\cos^2\delta\} &= E\left\{\frac{1}{2} + \frac{1}{2}\sum_{k=0}^{\infty}\frac{(-1)^k 2^{2k}}{(2k)!}\,\delta^{2k}\right\} = \frac{1}{2} + \frac{1}{2}\sum_{k=0}^{\infty}\frac{(-1)^k 2^{2k}}{(2k)!}\,E\left\{\delta^{2k}\right\} = \\ &= \frac{1}{2} + \frac{1}{2}\sum_{k=0}^{\infty}\frac{(-1)^k\,2^{2\,k}}{(2\,k)!}\cdot\frac{(2k)!}{2^k\,k!}\cdot\sigma^{2k} = \frac{1}{2} + \frac{1}{2}\sum_{k=0}^{\infty}\frac{(-1)^k\,2^k}{k!}\cdot\sigma^{2k} = \\ &= \frac{1}{2}\left(1 + e^{-2\,\sigma^2}\right) \end{aligned}$$

and, similarly,

$$E\{\sin^2\delta\} = E\left\{\frac{1}{2} - \frac{1}{2}\sum_{k=0}^{\infty}\frac{(-1)^k 2^{2k}}{(2k)!}\,\delta^{2k}\right\} = \cdots = \frac{1}{2}\left(1 - e^{-2\,\sigma^2}\right).$$

Therefore,

$$E\{\cos^2\vartheta\} = \frac{1}{2}\,\cos^2\mu\,\left(1 + e^{-2\,\sigma^2}\right) + \frac{1}{2}\,\sin^2\mu\,\left(1 - e^{-2\,\sigma^2}\right) = \frac{2\cos^2\mu - 1}{2}\,e^{-2\,\sigma^2} + \frac{1}{2}$$

and, specifically,

$$\begin{cases} E\{\cos^2\vartheta_x\} = \dfrac{2\cos^2\mu_x - 1}{2}\,e^{-2\,\sigma_\vartheta^2} + \dfrac{1}{2} \\ E\{\cos^2\vartheta_y\} = \dfrac{2\cos^2\mu_y - 1}{2}\,e^{-2\,\sigma_\vartheta^2} + \dfrac{1}{2} \end{cases}. \tag{4.7}$$

Using these results, the exact (analytical) expression of the $P[1,1]$ term of the covariance matrix is known,

$$P[1,1] = \frac{2\cos^2\mu_x - 1}{2}\, e^{-2\,\sigma^2} + \frac{1}{2} - e^{-\sigma^2}\, \cos^2\mu_x. \tag{4.8}$$

As for the term, $P[2,2]$, we may write it as

$$P[2,2] = E\{g_y^2\} - \hat{g}_y^2 = E\{g_y^2\} - \hat{g}_y^2 = E\left\{\left(\frac{\cos\vartheta_y}{\cos\varepsilon} - \cos\vartheta_x\, \tan\varepsilon\right)^2\right\} - \hat{g}_y^2.$$

That is,

$$P[2,2] = \frac{E\{\cos^2\vartheta_y\}}{\cos^2\varepsilon} - 2\,\frac{\tan\varepsilon}{\cos\varepsilon}\, E\{\cos\vartheta_x\, \cos\vartheta_y\} + E\{\cos^2\vartheta_x\}\, \tan^2\varepsilon - \hat{g}_y^2,$$

where the $E\{\cos^2\vartheta_x\}$ and $E\{\cos^2\vartheta_y\}$ terms are provided by Eq. (4.7), while the $E\{\cos\vartheta_x\, \cos\vartheta_y\}$ term can be expressed as

$$\begin{aligned} E\{\cos\vartheta_x \cos\vartheta_y\} &= E\{\cos\vartheta_x\}\, E\{\cos\vartheta_y\} = \\ &= \cos\mu_x\, \cos\mu_y\, e^{-\sigma^2/2}\, e^{-\sigma^2/2} = \\ &= \cos\mu_x\, \cos\mu_y\, e^{-\sigma^2} \end{aligned}$$

since the terms, $\cos\vartheta_x$ and $\cos\vartheta_y$, are statistically independent. Therefore, we have,

$$\begin{aligned} E\{g_y^2\} \;\; &= \frac{1}{\cos^2\varepsilon}\left[\frac{2\cos^2\mu_y - 1}{2}\, e^{-2\,\sigma^2} + \frac{1}{2}\right] - 2\,\frac{\tan\varepsilon}{\cos\varepsilon}\, \cos\mu_x\, \cos\mu_y\, e^{-\sigma^2} + \\ &\quad + \left[\frac{2\cos^2\mu_x - 1}{2}\, e^{-2\,\sigma^2} + \frac{1}{2}\right]\, \tan^2\varepsilon \end{aligned}.$$

That is,

$$\begin{aligned} E\{g_y^2\} \;\; &= \left(\frac{\cos^2\mu_y}{\cos^2\varepsilon} + \cos^2\mu_x\, \tan^2\varepsilon\right) e^{-2\,\sigma^2} - 2\,\frac{\tan\varepsilon}{\cos\varepsilon}\, \cos\mu_x\, \cos\mu_y\, e^{-\sigma^2} + \\ &\quad + \left(1 - e^{-2\,\sigma^2}\right)\, \frac{2 - \cos^2\varepsilon}{2\,\cos^2\varepsilon} \end{aligned}.$$

Finally, we arrive at an exact analytical expression for $P[2,2]$,

$$P[2,2] = \left(\frac{\cos^2\mu_y}{\cos^2\varepsilon} + \cos^2\mu_x\,\tan^2\varepsilon\right)\left(e^{-2\,\sigma^2} - e^{-\sigma^2}\right) + \left(1 - e^{-2\,\sigma^2}\right)\frac{2-\cos^2\varepsilon}{2\,\cos^2\varepsilon}. \tag{4.9}$$

Here, we note that when $\varepsilon = 0$ (i.e., the inclinometer mount is perfectly orthogonal), Eq. (4.9) becomes formally identical to Eq. (4.8). We now move to the $P[1,2]$ off-diagonal term, which is computed as

$$\begin{aligned} P[1,2] &= \frac{1}{\cos\varepsilon}E\left\{\cos\vartheta_x\left(\cos\vartheta_y - \cos\vartheta_x\sin\varepsilon\right)\right\} - \frac{\cos\mu_x}{\cos\varepsilon}e^{-\sigma^2}\left(\cos\mu_y - \cos\mu_x\sin\varepsilon\right) = \\ &= \frac{1}{\cos\varepsilon}\left[E\{\cos\vartheta_x\}E\{\cos\vartheta_y\} - E\{\cos^2\vartheta_x\}\sin\varepsilon - e^{-\sigma^2}\cos\mu_x\left(\cos\mu_y - \cos\mu_x\sin\varepsilon\right)\right]. \end{aligned}$$

All of the expectation terms appearing in the expression of $P[1,2]$ were previously derived in Eqs. (4.4) and (4.7). Thus,

$$\begin{aligned} P[1,2] &= \frac{1}{\cos\varepsilon}\left[e^{-\sigma^2}\cos\mu_x\cos\mu_y - \left((2\cos^2\mu_x - 1)e^{-2\,\sigma^2} + 1\right)\frac{\sin\varepsilon}{2}\right. \\ &\qquad \left. - e^{-\sigma^2}\cos\mu_x\left(\cos\mu_y - \cos\mu_x\sin\varepsilon\right)\right] = \\ &= \frac{1}{\cos\varepsilon}\left[-\left((2\cos^2\mu_x - 1)e^{-2\,\sigma^2} + 1\right)\frac{\sin\varepsilon}{2} + e^{-\sigma^2}\cos^2\mu_x\sin\varepsilon\right]. \end{aligned}$$

Finally, we arrive at a closed-form expression for the $P[1,2]$ off-diagonal term of the gravity covariance matrix,

$$P[1,2] = \tan\varepsilon\left[\cos^2\mu_x\left(e^{-\sigma^2} - e^{-2\sigma^2}\right) + \frac{1}{2}\left(e^{-2\sigma^2} - 1\right)\right]. \tag{4.10}$$

The expression for the remaining diagonal covariance matrix term, $P[3,3] = E\{g_z^2\} - \hat{g}_z^2$, is obtained using linear error propagation theory, which provides a reasonably accurate estimation

for $P[3,3]$. Since $g_z = -\sqrt{1-g_x^2-g_y^2}$, we can derive an expression for its variance from

$$\sigma_{g_z}^2 \approx \left.\frac{\partial g_z}{\partial g_x}\right|_{\hat{g}_x,\hat{g}_y}^2 \sigma_{g_x}^2 + \left.\frac{\partial g_z}{\partial g_y}\right|_{\hat{g}_x,\hat{g}_y}^2 \sigma_{g_y}^2 = \frac{\hat{g}_x^2\,\sigma_{g_x}^2 + \hat{g}_y^2\,\sigma_{g_y}^2}{1-\hat{g}_x^2-\hat{g}_y^2}.$$

However, we know that $\sigma_{g_x}^2 \equiv P[1,1]$, $\sigma_{g_y}^2 \equiv P[2,2]$, and $\sigma_{g_z}^2 \equiv P[3,3]$. So,

$$P[3,3] \approx \frac{\hat{g}_x^2\,P[1,1] + \hat{g}_y^2\,P[2,2]}{1-\hat{g}_x^2-\hat{g}_y^2}. \tag{4.11}$$

The estimation of the covariance term $P[3,3]$ allows us to derive an expression for the expected value of the third measured unit gravity vector component, $\hat{g}_z = E\{g_z\}$. Since $\boldsymbol{g}_b$ is a unit vector, whereby $g_z^2 = 1-g_x^2-g_y^2$, it follows that

$$E\{g_z^2\} = P[3,3] + \hat{g}_z^2 = 1 - E\{g_x^2\} - E\{g_y^2\} = 1 - P[1,1] - \hat{g}_x^2 - P[2,2] - \hat{g}_y^2.$$

So,

$$\hat{g}_z = -\sqrt{1-P[1,1]-P[2,2]-P[3,3]-\hat{g}_x^2-\hat{g}_y^2}. \tag{4.12}$$

Thus, the expected gravity direction, which can be written as a three-component unit vector, is obtained from the expressions provided by Eqs. (4.5), (4.6), and (4.12),

$$E\{\boldsymbol{g}_b\} = \hat{\boldsymbol{g}}_b = \begin{Bmatrix} \hat{g}_x \\ \hat{g}_y \\ \hat{g}_z \end{Bmatrix}_b .$$

The gravity covariance matrix derivation is completed by approximating the remaining off-diagonal terms, $P[1,3]$ and $P[2,3]$, which are approximated by

$$P[1,3] \approx \hat{g}_x\sqrt{1-\hat{g}_x^2-\hat{g}_y^2} + \hat{g}_x\,\hat{g}_z \tag{4.13}$$

and

$$P[2,3] \approx \hat{g}_y \sqrt{1 - \hat{g}_x^2 - \hat{g}_y^2} + \hat{g}_y \, \hat{g}_z. \tag{4.14}$$

Next, we look to validate the estimation of the unit gravity vector direction, given by Eqs. (4.5), (4.6), (4.12), as well as the estimation of the gravity covariance matrix, whose diagonal and off-diagonal elements are given by Eqs. (4.8), (4.9), (4.11) and Eqs. (4.10), (4.13), (4.14), respectively. To do so, 1,000,000 Monte Carlo tests were performed using the following approach:

- The true unit gravity vector direction, in the ECEF, was randomly set to

$$\boldsymbol{g}_f = \begin{Bmatrix} g_x \\ g_y \\ g_z \end{Bmatrix}_f = \begin{Bmatrix} \cos\lambda \, \cos\varphi \\ \sin\lambda \, \cos\varphi \\ \sin\varphi \end{Bmatrix}, \quad \text{where} \quad \begin{cases} \lambda = 135^\circ \\ \varphi = 20^\circ \end{cases}.$$

- The non-orthogonality condition was set to $\varepsilon = 1^\circ$, which is likely an extreme over-estimate.

- The standard deviation of the measurement Gaussian noise was set to $\sigma_\vartheta = 0.1^\circ$, which is a reasonable assumption for a typical COTS inclinometer.

The Monte Carlo test results are captured in Figs. 4.1 and 4.2. Fig. 4.1 quantifies the accuracy provided by Eqs. (4.5), (4.6), and (4.12) when estimating the mean value of the three gravity direction components. The left-hand plot provides the statistics associated with the g_x gravity component. The dotted green line represents the mean of the Monte Carlo tests ($\bar{g}_x$) while the red markers indicate the expected mean value ($\hat{g}_x$) defined by Eq. (4.5). The percent error between the mean and the expected mean is also provided as the title of the subplot. For the g_x component, the relative percent error was computed as: $100 \cdot \left| 1 - \frac{\hat{g}_x}{\bar{g}_x} \right| \approx 0.0018\%$. Using the same statistical interpretation, the center and right-hand subplots provide the results obtained for the g_y and g_z gravity direction components, respectively. Furthermore, Fig. 4.2 quantifies the results of the covariance analysis conducted from the same Monte Carlo test using a series of histograms. Each subplot includes the same statistical parameters that were mentioned in the description of Fig. 4.1,

i.e., the dotted green lines represent the mean of the Monte Carlo tests and the red markers represent the expected mean value of the associated covariance element. For each subplot, the absolute error between the mean and expected values is provided as the title (e.g., $|\delta P_{11}| = |\bar{P}[1,1] - \hat{P}[1,1]|$ in the upper-left figure).

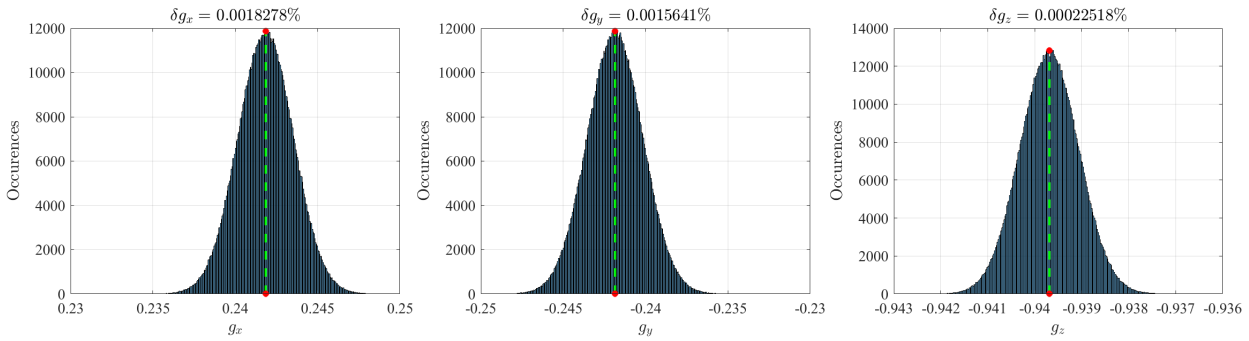

Figure 4.1: Histograms of the Gravity Direction Components Obtained from 1,000,000 Monte Carlo Tests

4.2 Extended Kalman Filter

The observed unit gravity vector, $\tilde{\boldsymbol{g}}_b$, when measured by an inclinometer set, can be fully-defined using two reference angles, ϑ_x and ϑ_y, and a mounting angle, ε (see Eq. (3.2)). The implication of this formulation is that the third component of $\tilde{\boldsymbol{g}}_b$ can be derived from the first two (e.g., $g_z = -\sqrt{1 - g_x^2 - g_y^2}$), which in turn reduces the EKF to a two-state filter. Assuming the vehicle is stationary over a measurement period, the state model dynamics are propagated without an additive uncertainty, meaning the best estimate of the state is the state itself for all time. The inclinometer measurement angles are used to define the filter's observation model based on Eqs. (4.5) and (4.6), wherein additive zero-mean Gaussian noise is used to simulate random fluctuations in the sensor readings due to thermal and electrical cycling. A summary of the EKF's implementation follows:

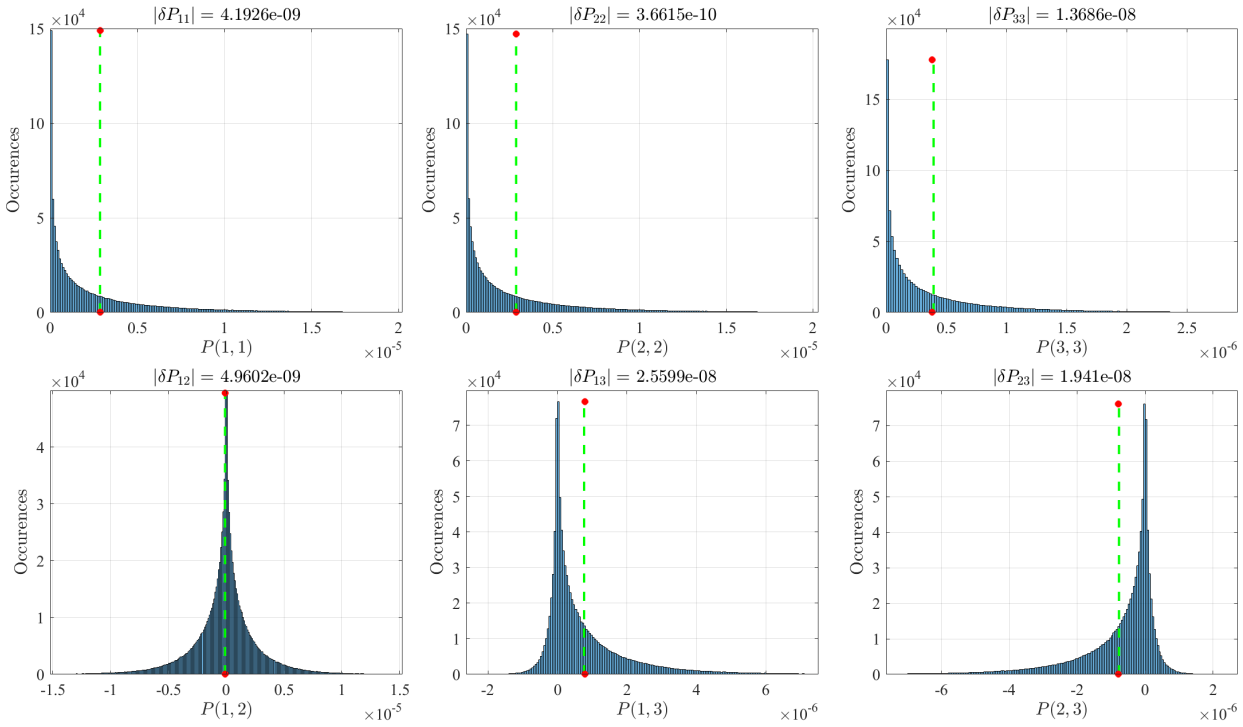

Figure 4.2: Histograms of the Covariance Matrix Elements Obtained from 1,000,000 Monte Carlo Tests

- **State model.**

$$\boldsymbol{x}_k = \begin{Bmatrix} g_x \\ g_y \end{Bmatrix}_k = \begin{Bmatrix} x_k \\ y_k \end{Bmatrix} = \boldsymbol{x}_{k-1} + \boldsymbol{w}_{k-1}.$$

- **Observation model.**

$$\boldsymbol{y}_k = \begin{Bmatrix} \tilde{\vartheta}_x \\ \tilde{\vartheta}_y \end{Bmatrix}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k = \cos^{-1}\left[e^{\sigma^2/2} \begin{Bmatrix} x_k \\ y_k \cos\varepsilon + x_k \sin\varepsilon \end{Bmatrix}_k \right] + \boldsymbol{v}_k.$$

- **Initialization.** Assume we are given an initial state vector, $\boldsymbol{x}_0^-$,

$$\boldsymbol{x}_0^- = \begin{Bmatrix} \cos\tilde{\vartheta}_x \\ \dfrac{\cos\tilde{\vartheta}_y - \cos\tilde{\vartheta}_x \sin\varepsilon}{\cos\varepsilon} \end{Bmatrix} = \begin{Bmatrix} x_0 \\ y_0 \end{Bmatrix},$$

with known covariance matrix, P_0^-,

$$P_0^- = \begin{bmatrix} P_0[1,1] & P_0[1,2] \\ P_0[2,1] & P_0[2,2] \end{bmatrix},$$

whose elements are determined using Eqs. (4.8), (4.10), and (4.9). In terms of the initial state vector, $\boldsymbol{x}_0^-$, we have

$$\begin{cases} P_0[1,1] = \left(e^{-2\,\sigma_\vartheta^2} - e^{-\sigma_\vartheta^2}\right) x_0^2 + \dfrac{1}{2}\left(1 - e^{-2\,\sigma_\vartheta^2}\right) \\ P_0[1,2] \equiv P_0[2,1] = \left[\left(e^{-\sigma_\vartheta^2} - e^{-2\sigma_\vartheta^2}\right) x_0^2 - \dfrac{1}{2}\left(1 - e^{-2\sigma_\vartheta^2}\right)\right] \tan\varepsilon \\ P_0[2,2] = (y_0^2 + 2\,x_0\,y_0\,\tan\varepsilon + 2x_0^2\,\tan^2\varepsilon)\left(e^{-2\,\sigma_\vartheta^2} - e^{-\sigma_\vartheta^2}\right) + \left(\dfrac{2 - \cos^2\varepsilon}{2\cos^2\varepsilon}\right)\left(1 - e^{-2\,\sigma_\vartheta^2}\right) \end{cases}.$$

Now, assume there is no noise on the dynamics, such that $Q_k = E\{\boldsymbol{w}_k \boldsymbol{w}_k^{\mathrm{T}}\} = \mathbf{0}_{2\times 2}$, and that the measurements are corrupted by independently sampled Gaussian white-noise, whose covariance is given by $R_k = E\{\boldsymbol{v}_k \boldsymbol{v}_k^{\mathrm{T}}\} = \sigma_\vartheta^2\,\boldsymbol{I}_{2\times 2}$.

- **Predict step.**

$$\begin{cases} \boldsymbol{x}_k^+ = \boldsymbol{x}_{k-1}^- \\ P_k^+ = P_{k-1}^- \end{cases}.$$

- **Update step.**

$$\begin{cases} K_k = P_k^+\,J_{\boldsymbol{h}}^{\mathrm{T}}(\boldsymbol{x}_k)\left(J_{\boldsymbol{h}}(\boldsymbol{x}_k)\,P_k^+\,J_{\boldsymbol{h}}^{\mathrm{T}}(\boldsymbol{x}_k) + R_k\right)^{-1} \\ \boldsymbol{x}_k^- = \boldsymbol{x}_k^+ + K_k\left(\boldsymbol{y}_k - \boldsymbol{h}\left(\boldsymbol{x}_k^+\right)\right) \\ P_k^- = (I - K_k\,J_{\boldsymbol{h}}(\boldsymbol{x}_k))\;P_k^+ \end{cases}.$$

Here, the Jacobian matrix of the observation model is given by,

$$J_{\boldsymbol{h}}(\boldsymbol{x}_k) = \frac{\partial \boldsymbol{h}\,(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k} = \begin{bmatrix} -\dfrac{1}{\sqrt{1 - x_k^2\,e^{\,\sigma_\vartheta^2}}} & 0 \\ -\dfrac{\sin\varepsilon}{d_k} & -\dfrac{\cos\varepsilon}{d_k} \end{bmatrix} e^{\,\sigma_\vartheta^2/2},$$

where

$$d_k = \sqrt{1 - e^{\sigma_\vartheta^2}(x_k \, \sin\varepsilon + y_k \, \cos\varepsilon)^2}.$$

The filter was verified through a numerical simulation using the following approach:

- The true unit gravity vector direction, in the ECEF, was randomly set to

$$\boldsymbol{g}_f = \begin{Bmatrix} g_x \\ g_y \\ g_z \end{Bmatrix}_f = \begin{Bmatrix} \cos\lambda \, \cos\varphi \\ \sin\lambda \, \cos\varphi \\ \sin\varphi \end{Bmatrix}, \quad \text{where} \quad \begin{cases} \lambda = 135^\circ \\ \varphi = 160^\circ \end{cases}.$$

- The non-orthogonality condition was set to $\varepsilon = 1^\circ$, which is likely an extreme over-estimate.

- The standard deviation of the measurement Gaussian noise was set to $\sigma_\vartheta = 0.1^\circ$, which is a reasonable assumption for a typical COTS inclinometer.

Fig. 4.3 proves the filter's convergence capability for the individual vector components of the unit gravity vector, g_x (upper left) and g_y (lower left). For each of these subplots, the true value is given by a solid black line, noisy measurements are indicated using black markers, and the filtered estimate is shown as a solid blue line. $\pm 3\,\sigma$ bounds were also included as dotted red lines. The right-hand figure shows how the error in the gravity direction estimate (with respect to the true direction) evolves over the measurement index.

4.3 Numerical Sensitivity Analysis to Errors

From a numerical simulation perspective, the effectiveness of the geoid correction algorithm can be quantified through a series of Monte Carlo tests. Ultimately, we are interested in characterizing the sensitivity of the unit gravity vector mapping (between inclinometer and local reference frames) to perturbations associated with error sources that could be attributed to the given sensor suite. The goal for each of these preliminary numerical tests was to isolate an expected error source, define a range of possible values for it, and perturb the localizing transformation matrix, $C_{\ell,f}$, in order to identify a threshold that would prevent the algorithm from maintaining accurate position

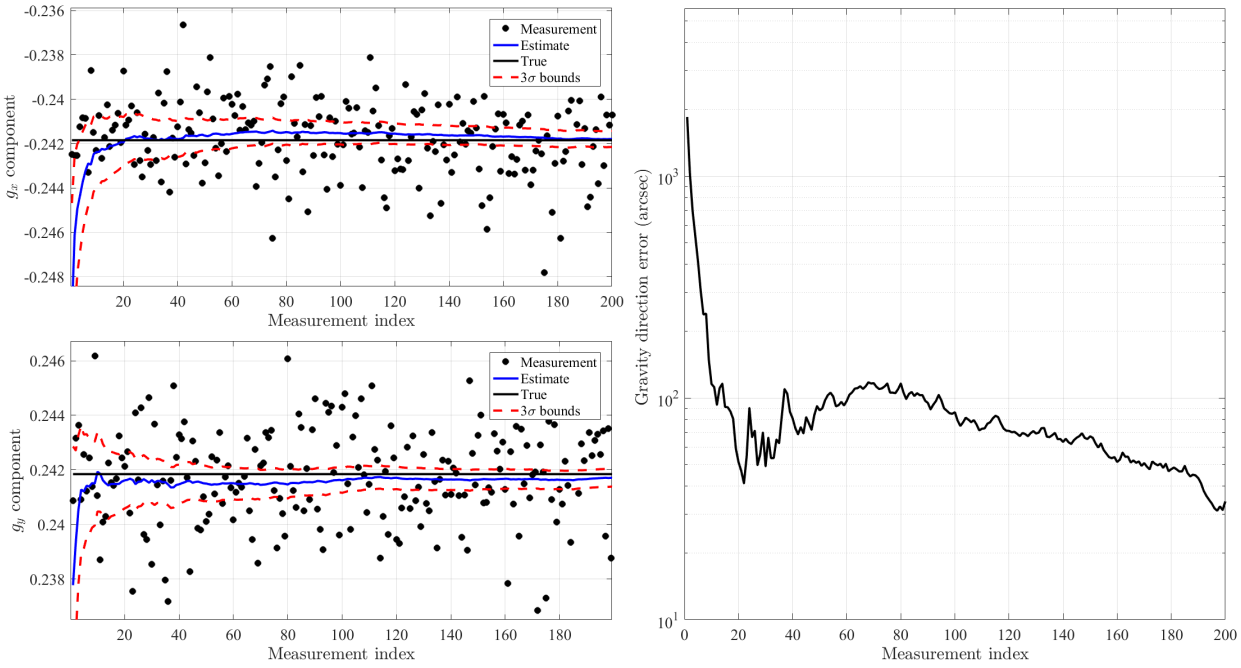

Figure 4.3: Numerical Validation of EKF

estimates. In the ideal case, the geoid data providing the reference unit gravity vector and the inclination sensors providing the measurement unit gravity would be perfect descriptions of each, i.e., no noise or biases are present on either vector, or their mapping. In reality, the relationship between the unit vectors is affected by a number of potential error sources, including:

1. Initial location uncertainty, which adjusts the estimate of $C_{\ell,f}(\varphi, \lambda)$;
2. Time uncertainty, which adjusts the estimate of $C_{f,i}(t)$;
3. Interlock matrix uncertainty, which adjusts the estimate of $C_{c,b}(\varepsilon)$;
4. Attitude uncertainty, which adjusts the estimate of $C_{c,i}(\boldsymbol{q})$;
5. Inclinometer measurement noise, which adjusts the estimate of $\tilde{\boldsymbol{g}}_b(\vartheta_x, \vartheta_y)$.

It is clear that each of these error sources are present in the system when we write the transformation matrix of interest as a function of all other transformation matrices,

$$C_{\ell,b} = C_{\ell,f}(\varphi, \lambda)\; C_{f,i}(t)\; C_{c,i}^{\mathrm{T}}(\boldsymbol{q})\; C_{c,b}(\varepsilon).$$

While parameter ranges vary across numerical tests, the final position error estimate is always computed using the same procedure:

1. Assume the vehicle's true position can be defined in the ECEF as a function of latitude and longitude, such that

$$\hat{\boldsymbol{r}}_f = \begin{Bmatrix} \cos\lambda\ \cos\varphi \\ \sin\lambda\ \cos\varphi \\ \sin\varphi \end{Bmatrix}, \text{ where } \begin{cases} \lambda \in [0,\ 2\pi] \\ \varphi \in \left[-\frac{\pi}{2},\ +\frac{\pi}{2}\right] \end{cases}$$

2. To avoid certain numerical instabilities, the linear (additive) approach to the expected error source analyses is abandoned in favor of an angular (multiplicative) approach [47]. We generate perturbed versions of $\hat{\boldsymbol{r}}_f$ by injecting uncertainty into the system in the following way:

   (a) Sample a random vector from the unit sphere,

$$\hat{\boldsymbol{e}} = \begin{Bmatrix} \cos\alpha\ \cos\beta \\ \sin\alpha\ \cos\beta \\ \sin\beta \end{Bmatrix}, \quad \text{where} \quad \begin{cases} \alpha \sim \mathcal{U}\left[0,\ 2\pi\right] \\ \cos\beta \sim \mathcal{U}\left[-1,\ +1\right] \end{cases}.$$

   (b) Compute the cross product of the reference unit vector and the random unit vector, and normalize by the Euclidean norm of their cross product,

$$\hat{\boldsymbol{n}} = \frac{\hat{\boldsymbol{r}}_f \times \hat{\boldsymbol{e}}}{\|\hat{\boldsymbol{r}}_f \times \hat{\boldsymbol{e}}\|_2}.$$

(c) Rotate the reference unit vector about the orthonormal unit vector by a prescribed angular distance. For a given reference transformation matrix, T, we define a noisy variant using Euler's rotation theorem,

$$\tilde{C} = T \cdot \Delta C(\hat{\boldsymbol{n}}, \gamma),$$

where

$$\Delta C(\hat{\boldsymbol{n}}, \gamma) = \boldsymbol{I}_{3\times 3} \cos\gamma + (1 - \cos\gamma)\, \hat{\boldsymbol{n}} \cdot \hat{\boldsymbol{n}}^{\mathrm{T}} + [\hat{\boldsymbol{n}}\times] \sin\gamma, \quad \gamma \sim \mathcal{N}(0, \sigma_\gamma^2)$$

and $[\times]$ is the skew-symmetric matrix operator.

(d) Compute a perturbed version of the reference unit gravity vector,

$$\tilde{\boldsymbol{r}}_f = \begin{Bmatrix} \cos\tilde{\lambda}\ \cos\tilde{\varphi} \\ \sin\tilde{\lambda}\ \cos\tilde{\varphi} \\ \sin\tilde{\varphi} \end{Bmatrix}_f = \begin{Bmatrix} \tilde{r}_x \\ \tilde{r}_y \\ \tilde{r}_z \end{Bmatrix}_f = \tilde{C}(\hat{\boldsymbol{n}}, \gamma)\ \hat{\boldsymbol{r}}_f(\varphi, \lambda).$$

3. Extract the perturbed position coordinates of the vehicle,

$$\begin{cases} \tilde{\varphi} = \arcsin\left(\tilde{r}_z\right) \\ \tilde{\lambda} = \arctan2\left(\tilde{r}_y, \tilde{r}_x\right) \end{cases}.$$

4. Finally, evaluate the position error by multiplying the square root of the sum of squared differences in position coordinates by the mean equatorial radius of the planet (e.g., R_E for Earth),

$$\Delta_d = R_E \sqrt{(\varphi - \tilde{\varphi})^2 + \left(\lambda - \tilde{\lambda}\right)^2}.$$

4.3.1 Attitude and Inclinometer Errors

The geoid correction algorithm utilizes an onboard star tracker and inclinometer set to take simultaneous measurements of the body's attitude matrix and unit gravity vector direction. Based on the recommendations of industry experts at NASA's Marshall Spaceflight Center (namely, Dr. Evan Anzalone and Mr. Joel Amert), a COTS CubeSat star tracker from Blue Canyon Technology [48] and a space-grade inclination sensor [49] were selected as the hardware components that would likely be used in a real-world demonstration of the algorithm. Before generating the contour result shown in Fig. 4.5, an initial numerical simulation was conducted to quantify the expected attitude error caused by noise along and about the boresight direction of the BCT star tracker (see Appendix B). The star tracker's single measurement attitude accuracy was estimated based on $N = 10,000$ Monte Carlo tests, wherein a random true attitude matrix, T, was perturbed by about-boresight and cross-boresight uncertainty matrices whose principle angles were determined by sampling from zero-mean Gaussian distributions subject to the standard deviations provided on the hardware specification sheet. Fig. 4.4 proves that the majority of attitude errors fall on the order of 10 arcseconds when taking a single static attitude measurement with the BCT star tracker. Considering the robustness of modern navigation filters, it is reasonable to assume that both the attitude and inclinometer measurement errors could be reduced to the sub-10 arcsecond range. Fig. 4.5 shows the results of this idealized scenario as a contour plot of expected total position error. Both the inclinometer gravity direction and star tracker attitude measurements, which define the inclinometer-to-local and inertial-to-camera mappings, respectively, were purturbed by sweeping over a range of sub-10 arcsecond standard deviations, $0 \leq \sigma_i \leq 5$ arcseconds and $0 \leq \sigma_a \leq 5$ arcseconds. The plotted values were found by taking the mean of $N = 10,000$ Monte Carlo test samples at each error combination within the sample space. In doing so, we can reasonably conclude that measurement errors associated with the critical pieces of onboard hardware, namely, the star tracker and inclinometer set, would likely induce around 100 meters of total position error. In general, a difference of about 0.5 arcseconds on either measurement equates to roughly 10 meters of position error on Earth. Given that this method relies on angular deviations as its reference pa-

rameters, we can also conclude that the position error will monotonically increase with the radius of the celestial body.

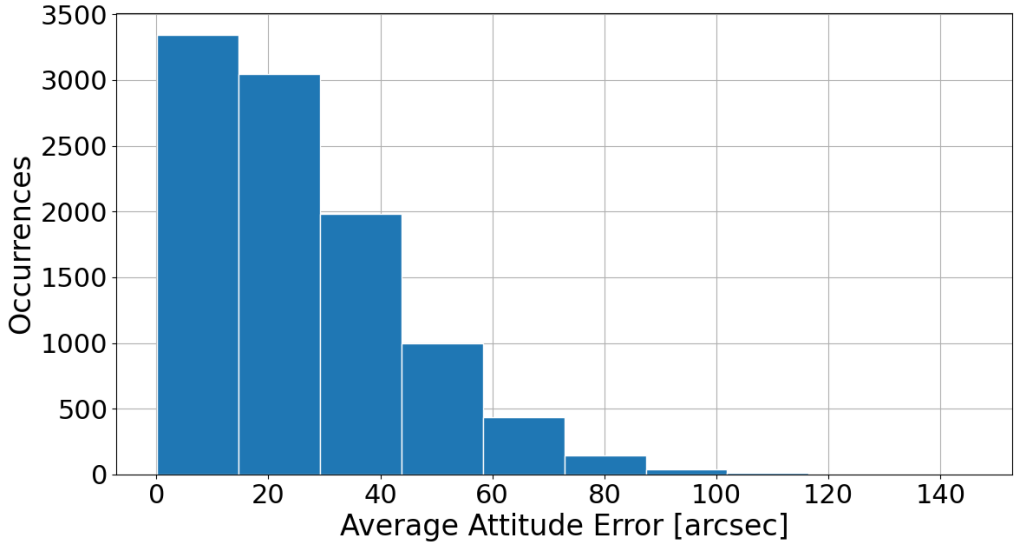

Figure 4.4: Monte Carlo Attitude Error Distribution for BCT Star Tracker [Reprinted from [1]]

4.3.2 Initial Location Error

Through preliminary numerical testing of the geoid correction algorithm, it was quickly determined that the likelihood of achieving a convergent solution was heavily dependent on the amount of error injected into the initialized, or "guessed," location of the vehicle, as well as on characteristics of the regional data set at-hand, including the resolution of the grid and the presence of local minimina around the reference location. In order to gain a better understanding of where the threshold for initial location uncertainty exists, a Monte Carlo analysis was conducted such that annular regions around the vehicle's true location could be iteratively expanded until a minimum threshold for achieving convergent solutions was identified. The initial guesses for latitude and longitude were generated by reframing the problem in terms of polar coordinates, with $\hat{r}$ and $\hat{\theta}$

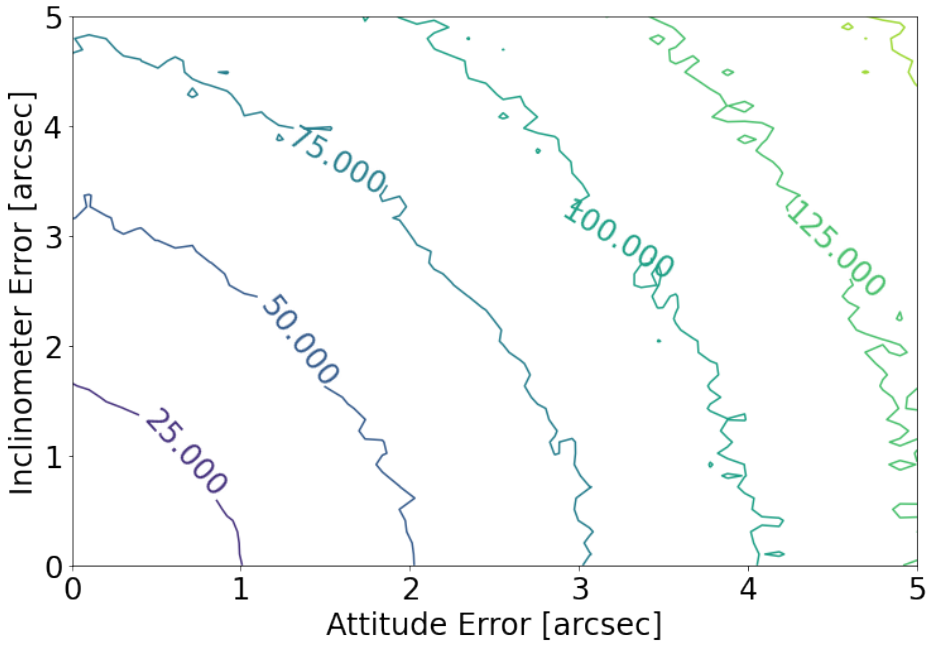

Figure 4.5: Contour Plot of Expected Position Error Due to Attitude Uncertainty and Inclinometer Noise

replacing the two-coordinate axes that were originally described using $\hat{\lambda}$ and $\hat{\varphi}$. For each of the selected sub-grids, initial location uncertainty was implemented by sweeping over a range of annular radii and angular distances. Both of these parameter spaces were randomly sampled from uniform distributions before being added to the reference location defined as the center point of the sub-grids. In other words,

$$\begin{cases} \tilde{\varphi} = \varphi + R\sin\theta \\ \tilde{\lambda} = \lambda + R\cos\theta \end{cases}, \quad \text{where} \quad \begin{cases} R \sim \mathcal{U}[R_{\min}, R_{\max}] \\ \theta \sim \mathcal{U}[0, 2\pi] \end{cases}.$$

Here, the annular radii were defined by a minimum and maximum linear distance from the midpoint, $R_{\min}$ and $R_{\max}$. The parameter combinations were sampled $N = 10,000$ times before taking an average of the position error and plotting the results. Fig. 4.6 includes two subplots associated with this computation, wherein the mean position error due to increasing annular radii up to 1 km is shown for both the Himalayan and (left) and Olympus Mons (right) regions. From

observation of these trend lines, it is clear the algorithm is more sensitive to initial guesses in the Himalayas due to the presence of many local minima in the region. Proving this theory required some additional investigation into the loss vector surfaces for each region. Figs. 4.7 and 4.8 show the individual surfaces of each loss vector component as well as the Euclidean norm of the loss vector for the Himalayan and Olympus Mons data sets, respectively, with reference to the midpoint of these regions. For each of these test cases, the random samples around the true location were binned depending on their eventual convergence or divergence. In Fig. 4.7, the black dashed line represents the trajectory of a random sample that diverged due to a component of the loss vector pulling it away from the reference location. In this case, it appears that the x-component of the loss vector initially forced this initial guess to move toward a local minimum at the right-hand edge of the sub-grid, at which point the nonlinear least-squares could not overcome the local maximum of the z-component of the loss vector and the solution eventually diverged. On the other hand, Fig. 4.8 shows the algorithm quickly moving toward a nearby local minimum present in the x-component of the loss vector, which turned out to be unrecoverable and led to divergence after just a few iterations. While all of the random samples drawn from regions within the 0.25 km (Himalayas) and 1 km (Olympus Mons) bounds derived from Fig. 4.6 converged, nearly all other samples diverged due to the presence of local minima in one or more of the loss vector component surfaces.

4.3.3 Time Error

A simplifying assumption of this sensitivity study is that the relationship between the ECI and ECEF is purely a function of the Earth's rotation rate, ω_E and time, t. Since the Earth rotates at a near-constant rate about its polar axis, errors in time would only perturb the estimate of the longitude, λ. Quantifying the position error due to clock drift was done by initializing the longitude as null, $\lambda_0 = 0^\circ$, and propagating reference values by adding a non-biased and non-noisy rotational velocity parameter that increases linearly with time, $\omega_E \cdot t$. Higher fidelity numerical modeling of atomic clocks considers a drift rate that randomly walks over time. In other words, the bias can appear to self-correct itself when viewed on an abbreviated time scale. For our purposes, a

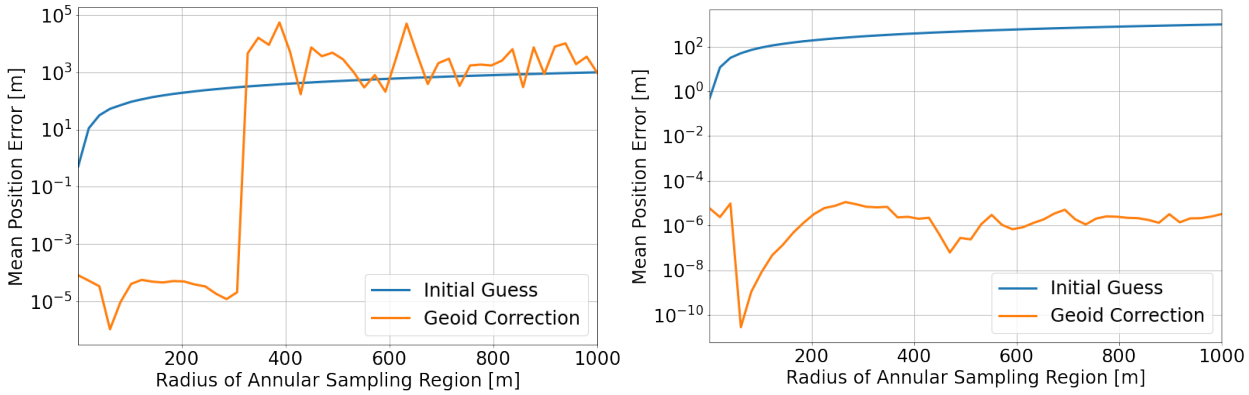

Figure 4.6: Expected Position Error Due to Initial Location Uncertainty Around Himalayas (left) and Olympus Mons (right)

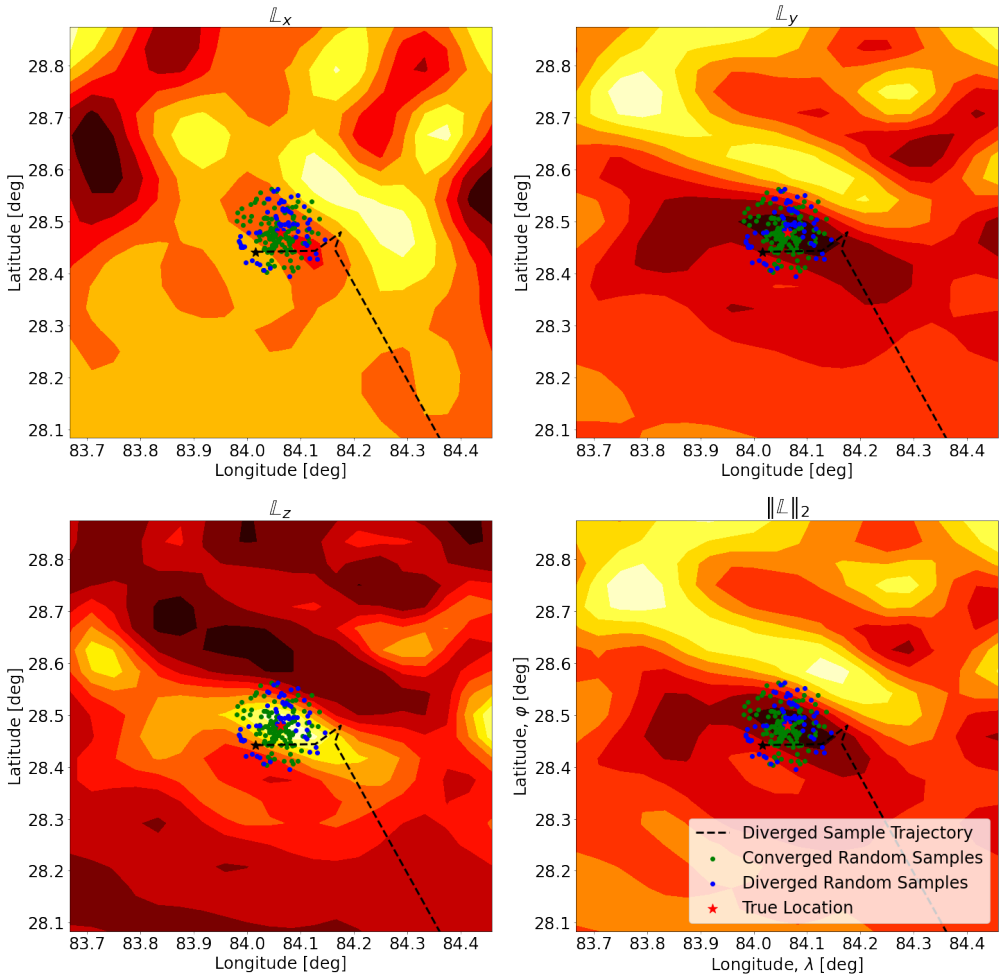

Figure 4.7: Loss Vector Surface Plots with Convergent/Divergent Random Samples for Himalayan Region

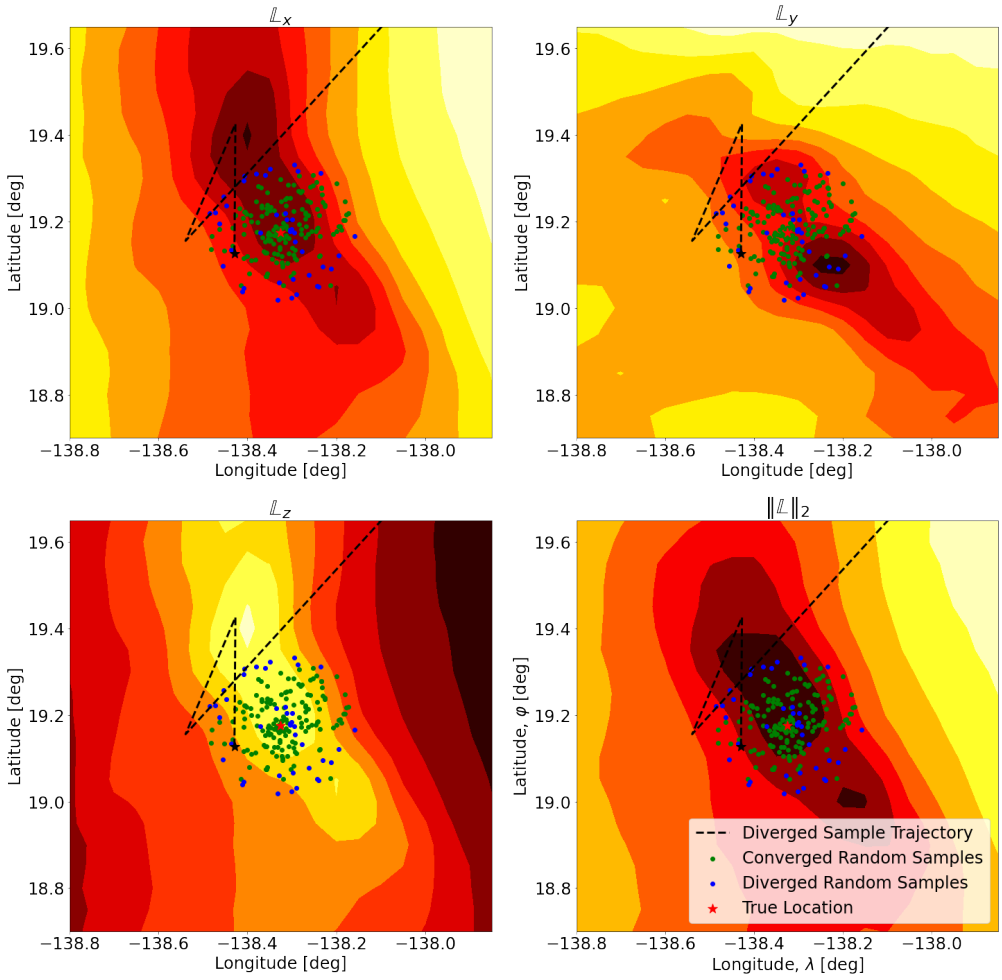

Figure 4.8: Loss Vector Surface Plots with Convergent/Divergent Random Samples for Olympus Mons Region

simplified approach was taken to compute time-corrupted variations of the reference longitudes using a fixed bias and zero-mean white noise, such that

$$\lambda^* = \lambda_0 + \omega_E(t + b_t + \delta_t), \quad \delta_t \sim \mathcal{N}(0, \sigma_t^2).$$

The values for the time error parameters were pulled from a chip-scale atomic clock (CSAC) specification sheet provided by Microsemi [50], wherein the expected drift rate of a high-end Rubidium atomic oscillator was listed as 200 nanoseconds per day, i.e., $b_t \approx 2 \cdot 10^{-7}$ seconds, while the white noise standard deviation was set to a picosecond, i.e., $\sigma_t = 1 \cdot 10^{-12}$ seconds. Finally, the expected position error due to time was computed by multiplying the absolute difference between reference and perturbed longitudes by the mean radius of the Earth. Fig. 4.9 shows the effect of clock drift on the position error over a one year measurement period. At its worst, the position error due to time is still on the centimeter scale, which is negligible when compared to the effects of attitude and inclinometer noise as detailed in the prior subsection.

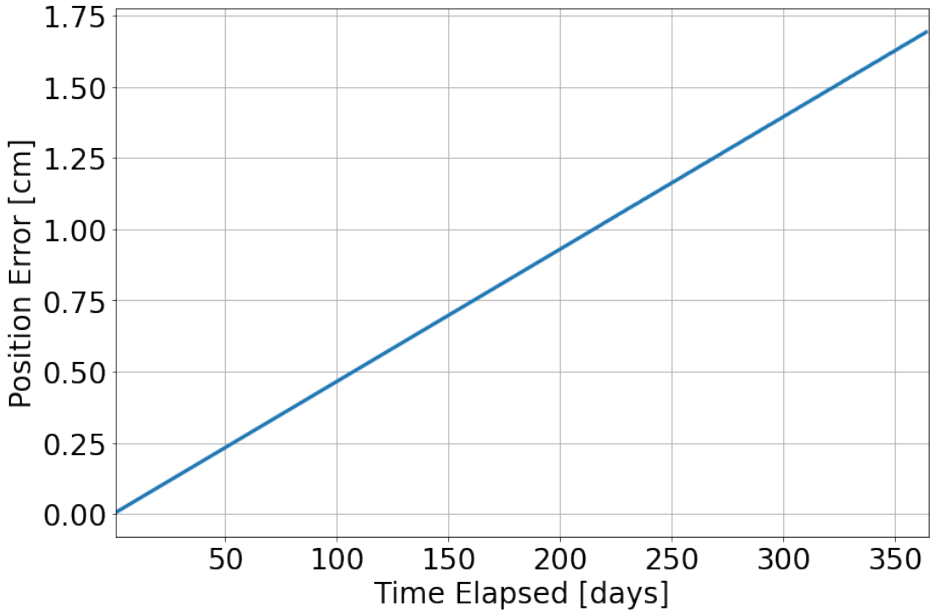

Figure 4.9: Expected Position Error Due to Atomic Clock Drift

4.3.4 Interlock Matrix Error

The interlock matrix is typically quantified in a laboratory setting using a mounting table and precision measurement instrumentation. In general, the deviation between camera and inclinometer mounts is small but not negligible; therefore, a numerical test was required in order to establish an initial guess as to how significant the effect of the interlock matrix could be on position estimates if left uncalibrated. Here again, we take the same approach to quantifying the effect of an expected error source. Namely, we generate variants of a reference transformation matrix, which would simply be the identity matrix if the inclinometer mount was perfectly orthogonal, using a range of expected standard deviations, $0° \leq \sigma_\varepsilon \leq 0.01°$. Fig. 4.10 shows that in a worst case scenario, where the mounting misalignment angle cannot be confidently estimated to a resolution greater than $0.01°$, the position error could be reasonably anticipated to exist on the centimeter scale, similar to what was found in the atomic clock subsection. In the same way, we can conclude that the position error induced by uncertainties in the interlock matrix angle are negligible with respect to inclinometer and attitude errors.

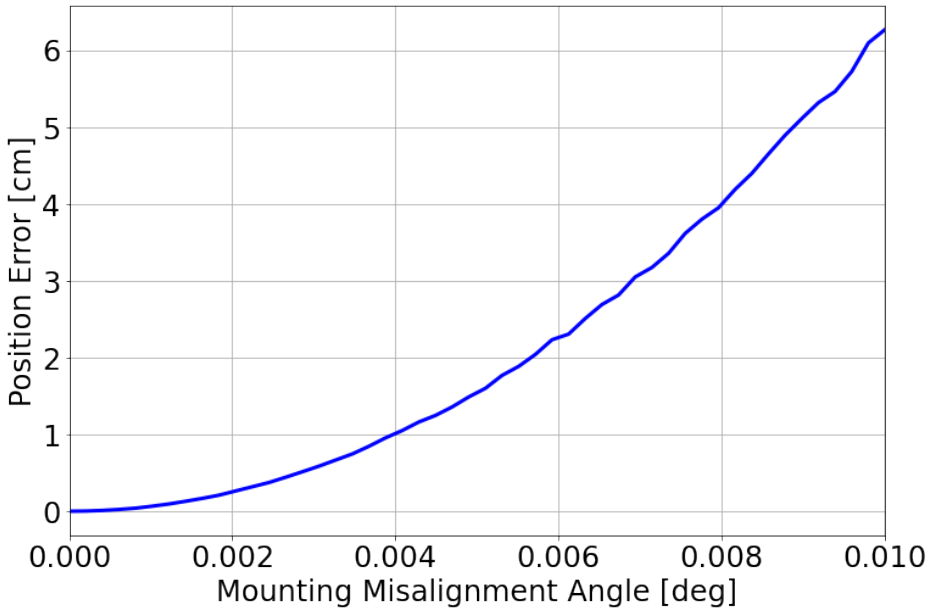

Figure 4.10: Expected Position Error Due to Mounting Misalignment Angle Between Non-orthogonal Inclinometer Set

4.3.5 Full System Sensitivity Test

The final demonstration of the geoid correction algorithm's performance was characterized through a full system sensitivity test, where each of the aforementioned error sources were instantiated and randomly sampled over a set of $N = 1,000$ Monte Carlo tests. Two data sets were simulated over within this subsection, namely, the highly nonlinear sub-grid of the Himalayan mountain range and the smoother sub-grid of Olympus Mons. For each test case, the initial position error was bounded by the results of the sensitivity study presented in Section 4.3.2. In particular, the Earth test case only considered initial guesses that were within 0.25 km of the sub-grid's midpoint, while the Mars test cases only considered initial guesses within 1 km. In both scenarios, the reference transformation matrices were perturbed by zero-mean Gaussian noise whose standard deviations were set based on the specification sheets of the hardware under consideration [48, 49, 50]. The following bulleted list provides a summary of the noise parameters applied to each reference frame:

- The initializing coordinates were derived from the fixed-to-local perturbation matrix, $\Delta C_{\ell,f}$, which was defined by a position error standard deviation of $\sigma_E = 0.25$ km for Earth and $\sigma_M = 1$ km for Mars;
- The inertial-to-fixed perturbation matrix, $\Delta C_{f,i}$, was defined by a time error standard deviation of $\sigma_t = 1 \cdot 10^{-7}$ seconds;
- The inertial-to-camera perturbation matrix, $\Delta C_{c,i}$, was defined by an attitude error standard deviation of $\sigma_a = 10$ arcseconds;
- The inclinometer-to-local perturbation matrix, $\Delta C_{\ell,b}$, was defined by an inclinometer error standard deviation of $\sigma_i = 1^\circ$;
- The inclinometer-to-camera perturbation matrix, $\Delta C_{c,b}$, was defined by a mounting error standard deviation of $\sigma_\varepsilon = 0.01^\circ$.

After applying each of these error sources to the reference coordinates, the "guessed" position of the vehicle (defined by its latitude and longitude) was passed into the geoid correction algo-

rithm, which performed the nonlinear least-squares procedure detailed in Chapter 3. Figs. 4.11 and 4.12 show the algorithm's performance over the Earth and Mars data sets, respectively. For the Himalayan region, the geoid correction, on average, provided a two to three order of magnitude improvement to the total initial position error. For the Olympus Mons region, the algorithm returned even better results, as shown by the tightly bound distribution of latitude and longitude errors.

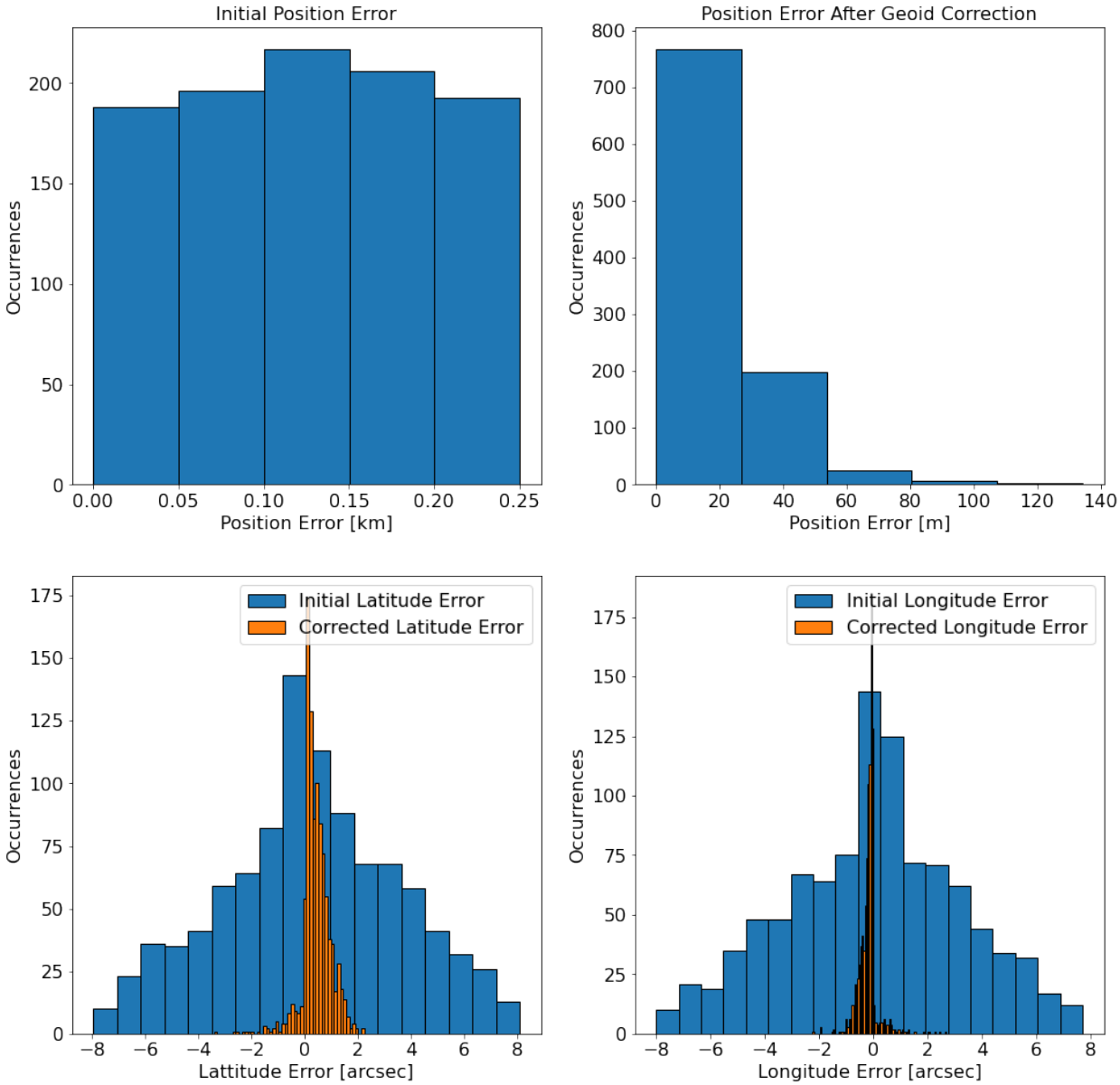

Figure 4.11: Demonstration of Geoid Correction Performance on Earth

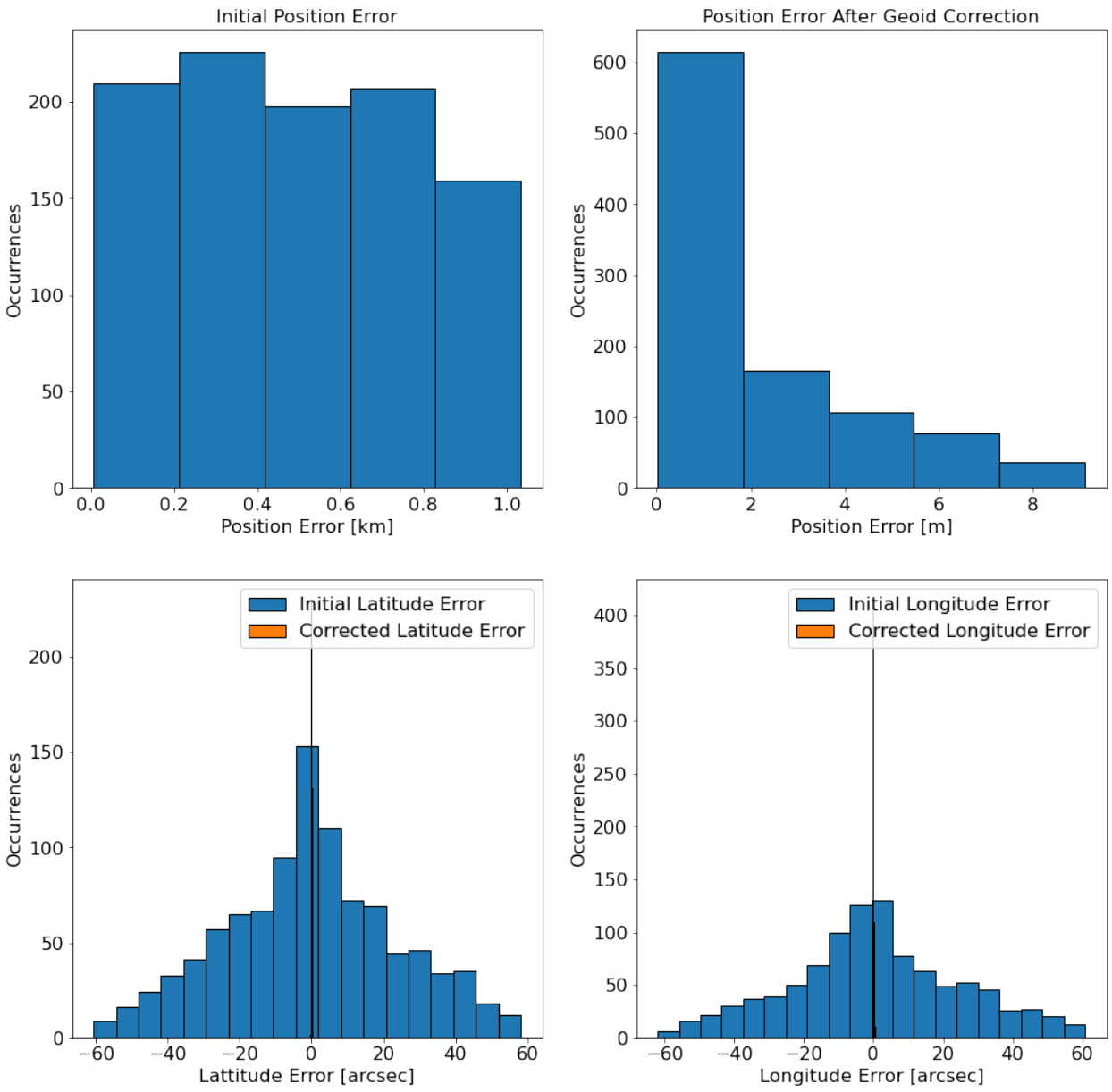

Figure 4.12: Demonstration of Geoid Correction Performance on Mars

5. SUMMARY AND CONCLUSIONS

5.1 Conclusions

This work presented a novel approach to vehicle localization using celestial observations and a direct measurement of the unit gravity vector. While the modern GPS solution to position estimation uses a reference axial-symmetric ellipsoid as its shape model of Earth, this thesis proved that a higher-fidelity approximation of a planet's shape, such as the geoid, can improve position errors by two or three orders of magnitude based on the precision of local gravity measurements. In addition to demonstrating the utility of the geoid as the best-fit shape model of Earth and Mars, several algorithms were derived in order to numerically test this theory, including an iterative non-linear least-squares technique and a gravity covariance analysis with extended Kalman filtering. Finally, a full-system sensitivity study was conducted using simulated models of COTS space-grade hardware, which proved the viability of the geoid correction algorithm. From these tests, it was concluded that the predictability of a convergent solution depends, at least in part, on the convexity of the surface that characterizes the loss vector defined in Chapter 3.

5.2 Future Work

Finding a definitive solution for "convexifying" the loss vector persists as future work for an interested reader, though it is likely that a new optimization approach is required, such as the stochastic sampling techniques used in particle swarms [51] and genetic algorithms [52], in order to properly discern a global minimum in the presence of many local minima. A short-term fix to this problem is highlighted by the subplots found in Fig. 5.1, wherein the loss vector could potentially be swapped for a scalar value, using the Euclidean norm, for instance, in situations where the scalar loss surface happens to be smoother than the individual loss vector component surfaces. Additional future work includes deriving a more comprehensive navigation filter, with a state vector that considers measurement biases and multi-sensor data fusion, and validating the results obtained through numerical simulation with a real-world hardware demonstration.

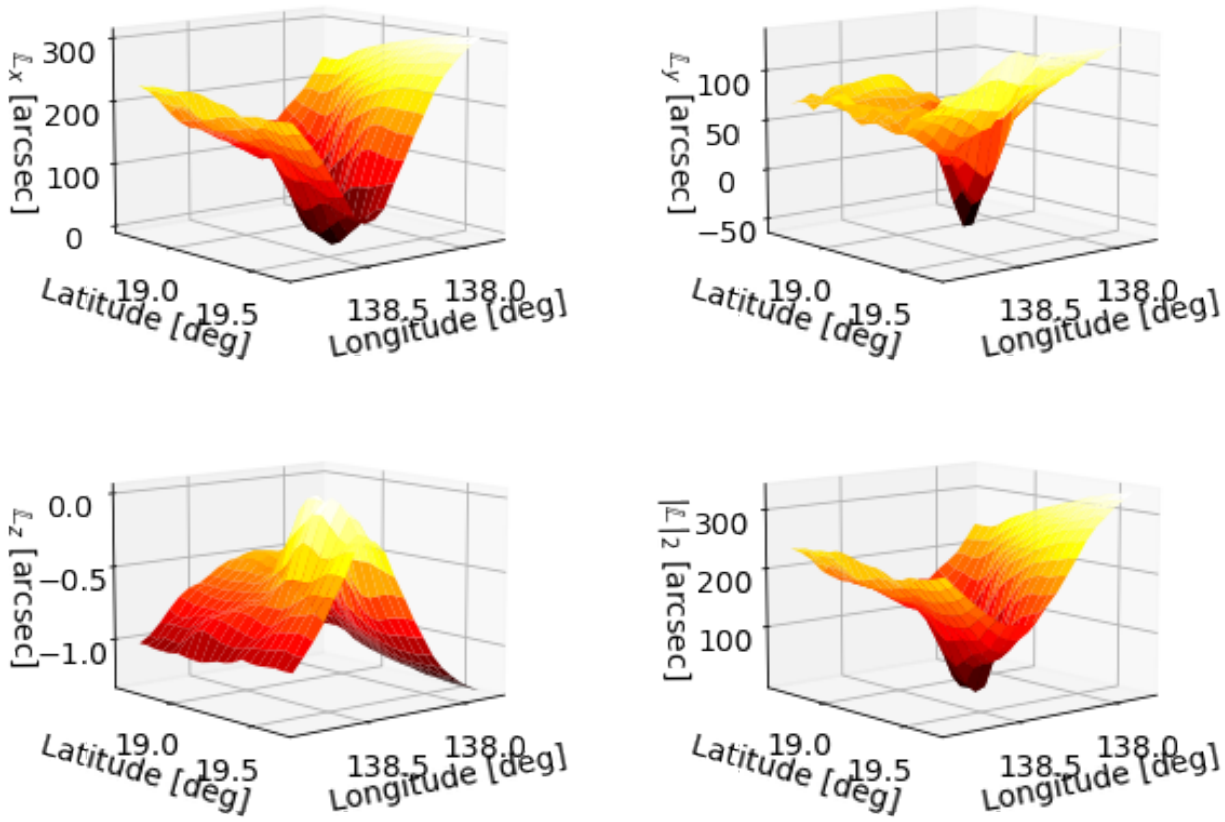

Figure 5.1: Loss Surfaces of Olympus Mons with Respect to its Midpoint

REFERENCES

[1] A. Gardner, H. Johnston, C. Leake, and D. Mortari, "Star Tracker Based Inertial State Estimation on Planetary Bodies: An Update on the Stellar Positioning System," *AAS/AIAA Space Flight Mechanics Meeting*, 2020. AAS 20-548.

[2] WAAS T&E Team, "Global Positioning System (GPS) Standard Positioning Service (SPS) Performance Analysis Report," tech. rep., William J. Hughes Technical Center, Atlantic City International Airport, NJ, July 2017.

[3] F. van Diggelen and P. Enge, "The World's first GPS MOOC and Worldwide Laboratory using Smartphones," in *Proceedings of the 28th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2015)*, (Tampa, FL), pp. 361–369, September 2015.

[4] CCSDS, "Proximity-1 Space Link Protocol–Rationale, Architecture, and Scenarios. Issue 2. Report Concerning Space Data System Standards (Green Book)," Tech. Rep. 210.0-G-2, National Imagery and Mapping Agency, Dec. 2013.

[5] T. P. Jeff Berner, Alaudin M. Bhanji and C. Scott, "Deep Space Network Services Catalog," tech. rep., Jet Propulsion Laboratory, California Institute of Technology, 02 2015.

[6] J. Lauf, M. Calhoun, W. Diener, J. Gonzalez, A. Kirk, P. Kuhnle, B. Tucker, C. Kirby, and R. Tjoelker, "Clocks and Timing in the NASA Deep Space Network," in *Proceedings of the 2005 IEEE International Frequency Control Symposium and Exposition, 2005.*, pp. 830–835, 2005.

[7] S. Chiodini, M. Pertile, S. Debei, L. Bramante, E. Ferrentino, A. G. Villa, I. Musso, and M. Barrera, "Mars rovers localization by matching local horizon to surface digital elevation models," in *2017 IEEE International Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pp. 374–379, 2017.

[8] R. Li, S. He, Y. Chen, M. Tang, P. Tang, K. Di, L. Matthies, R. E. Arvidson, S. W. Squyres, L. S. Crumpler, T. Parker, and M. Sims, "MER Spirit rover localization: Comparison of ground image– and orbital image–based methods and science applications," *Journal of Geophysical Research: Planets*, vol. 116, no. E7, 2011.

[9] Y. Tao, J.-P. Muller, and W. Poole, "Automated localisation of mars rovers using co-registered hirise-ctx-hrsc orthorectified images and wide baseline navcam orthorectified mosaics," *Icarus*, vol. 280, pp. 139 – 157, 2016. MicroMars to MegaMars.

[10] K. Di, F. Xu, R. Li, L. H. Matthies, and C. F. Olson, "High Precision Landing Site Mapping and Rover Localization by Integrated Bundle Adjustment of MPF Surface Images," in *Symposium on Geospatial Theory, Processing and Applications*, 2002.

[11] A. H. Mishkin, J. C. Morrison, T. T. Nguyen, H. W. Stone, B. K. Cooper, and B. H. Wilcox, "Experiences with operations and autonomy of the Mars Pathfinder Microrover," in *1998 IEEE Aerospace Conference Proceedings (Cat. No.98TH8339)*, vol. 2, pp. 337–351 vol.2, 1998.

[12] G. Pyrzak, M. McCurdy, K. J. Rabe, J. S. Norris, J. C. Joswig, J. M. Fox, T. Crockett, and M. W. Powell, "Targeting and Localization for Mars Rover Operations," in *2006 IEEE International Conference on Information Reuse Integration*, pp. 23–27, 2006.

[13] K. S. Ali, C. A. Vanelli, J. J. Biesiadecki, M. W. Maimone, Yang Cheng, A. M. San Martin, and J. W. Alexander, "Attitude and position estimation on the mars exploration rovers," in *2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp. 20–27 Vol. 1, 2005.

[14] R. Volpe, T. Estlin, S. Laubach, C. Olson, and J. Balaram, "Enhanced mars rover navigation techniques," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, vol. 1, pp. 926–931 vol.1, 2000.

[15] R. Li, F. Ma, F. Xu, L. H. Matthies, C. F. Olson, and R. E. Arvidson, "Localization of Mars rovers using descent and surface-based image data," *Journal of Geophysical Research: Planets*, vol. 107, no. E11, pp. FIDO 4–1–FIDO 4–8, 2002.

[16] Yang Cheng, Mark Maimone, and Larry Matthies, "Visual odometry on the Mars Exploration Rovers," in *2005 IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp. 903–910 Vol. 1, 2005.

[17] D. A. Sigel and D. Wettergreen, "Star tracker celestial localization system for a lunar rover," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2851–2856, 2007.

[18] C. Pingyuan, Y. Fuzhan, and C. Hutao, "Attitude and position determination scheme of lunar rovers basing on the celestial vectors observation," in *2007 IEEE International Conference on Integration Technology*, pp. 538–543, 2007.

[19] P. Furgale, J. Enright, and T. Barfoot, "Sun Sensor Navigation for Planetary Rovers: Theory and Field Testing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, pp. 1631–1647, 2011.

[20] J. Enright, T. Barfoot, and M. Soto, "Star tracking for planetary rovers," in *2012 IEEE Aerospace Conference*, pp. 1–13, 2012.

[21] X. Ning and L. Liu, "A Two-Mode INS/CNS Navigation Method for Lunar Rovers," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 9, pp. 2170–2179, 2014.

[22] J.J. Parish, A.S. Parish, M. Swanzy, D. Woodbury, D. Mortari, J.L. Junkins, "Stellar Positioning System (Part I): An Autonomous Position Determination Solution," *Navigation*, vol. 57, pp. 1–12, 2010.

[23] D.P. Woodbury, J.J. Parish, A.S. Parish, M. Swanzy, R. Denton, D. Mortari, J.L. Junkins, "Stellar Positioning System (Part II): Improving Accuracy During Implementation," *Navigation*, vol. 57, pp. 13–24, 2010.

[24] G. Wahba, "A Least-Squares Estimate of Spacecraft Attitude," *SIAM Review*, vol. 7, p. 409, 1965.

[25] P. Davenport, "A Vector Approach to the Algebra of Rotations with Applications," *NASA X-546-65-437*, 1965.

[26] C.H. Acton, "Ancillary Data Services of NASA's Navigation and Ancillary Information Facility," *Planetary and Space Science*, vol. 44, no. 1, pp. 65–70, 1996.

[27] Annex et al., "SpiceyPy: a Pythonic Wrapper for the SPICE Toolkit," *Journal of Open Source Software*, vol. 5, no. 46, p. 2050, 2020.

[28] F. Markley and J. Crassidis, *Fundamentals of Spacecraft Attitude Determination and Control*. New York, NY, USA: Springer, 2014.

[29] D. Mortari, C. Bruccoleri, and J. L. Junkins, "The *Pyramid* Star Identification Technique," *ION Navigation*, vol. 51, no. 3, pp. 171–183, 2004.

[30] M. Shuster and S. Oh, "Three-axis attitude determination from vector observations," *Journal of Guidance and Control, Vol. 4., No. 1, pp. 70-77*, 1981.

[31] D. Mortari, "ESOQ: A Closed-Form Solution to the Wahba Problem," *Journal of the Astronautical Sciences, Vol. 45, No. 2, pp. 195-204*, 1997.

[32] D. Mortari, "ESOQ2: Single-Point Algoirthm for Fast Optimal Attitude Determination," *AAS/AIAA Space Flight Mechanics Meeting*, Huntsville, AL, 1997.

[33] A. Mallama, B. Krobusek, and H. Pavlov, "Comprehensive wide-band magnitudes and albedos for the planets, with applications to exo-planets and Planet Nine," *Icarus*, vol. 282, p. 19–33, Jan 2017.

[34] C. Fabricius, "Paris 1676: The discovery of the velocity of light and roles of Rømer and Cassini," *Journal for the History of Astronomy*, vol. 50, no. 4, pp. 456–475, 2019.

[35] J. Bradley, "A Letter from the Reverend Mr. James Bradley Savilian Professor of Astronomy at Oxford, and F.R.S. to Dr. Edmond Halley Astronom. Reg. &c. Giving an Account of a

New Discovered Motion of the Fix'd Stars.," *Philosophical Transactions of the Royal Society of London Series I*, vol. 35, pp. 637–661, Jan. 1727.

[36] R. R. Karimi and D. Mortari, "Interplanetary Autonomous Navigation using Visible Planets," *Journal of Guidance Control and Dynamics*, vol. 38, pp. 1151–1156, Apr. 2015.

[37] B. Tapley, S. Bettadpur, M. Watkins, and C. Reigber, "The Gravity Recovery and Climate Experiment: Mission overview and early results," *Geophysical Research Letters*, vol. 31, p. pp. 4, 05 2004.

[38] R. Kornfeld, B. Arnold, M. Gross, N. Dahya, W. Klipstein, P. Gath, and S. Bettadpur, "GRACE-FO: The Gravity Recovery and Climate Experiment Follow-On Mission," *Journal of Spacecraft and Rockets*, vol. 56, pp. pp. 931–951, 05 2019.

[39] I. Newton, "Philosophiae Naturalis Principia Mathematica," 1687.

[40] NIMA/Geodesy and Geophysics Department, "Department of Defense World Geodetic System 1984: Its Definition and Relationships with Local Geodetic Systems," Tech. Rep. 8350.2, National Imagery and Mapping Agency, Jan 2000.

[41] J. P. Snyder, *Map Projections – A Working Manual*. U.S. G.P.O., Washington, D.C., Nov 1987.

[42] C.F. Gauss, "Determination of the difference in latitude between the Goettingen and Altona observatories by observing the Ramsden Zenithsector.," *Carl Friedrich Gauss Werke*, vol. 8, p. 9, Jan. 1903.

[43] N.K. Pavlis, S.A. Holmes, S.C. Kenyon, and J.K. Factor, "The development and evaluation of the Earth Gravitational Model 2008 (EGM2008)," *Journal of Geophysical Research: Solid Earth (1978-2012)*, vol. 117, Apr. 2012.

[44] C. Hirt, S. Claessens, M. Kuhn, and W. Featherstone, "Kilometer-resolution gravity field of mars: Mgm2011," *Planetary and Space Science*, pp. 147–154, 2017.

[45] C. Hirt and W. Featherstone, "A 1.5km-resolution gravity field model of the moon," *Earth and Planetary Science Letters (EPSL)*, pp. 329–330, 22–30, 2012.

[46] H. Stark and J. W. Woods, *Probability and Random Processes with Applications to Signal Processing*. Upper Saddle-River, NJ 07458: Prentice-Hall, 2002.

[47] D. Mortari and M. Majji, "Multiplicative Measurement Model," *The Journal of the Astronautical Sciences*, vol. 57, pp. 47—-60, Jan.—June 2009.

[48] Blue Canyon Technlogies, "Star Trackers Specification Sheet." `https://www.bluecanyontech.com`.

[49] F. Guzman, L. Kumanchik, G. M. Harry, J. Pratt, and J. M. Taylor, "Compact fully monolithic optomechanical accelerometer," *Applied Physics Letters, vol. 104*, 2014.

[50] Microsemi, "Chip-Scale Atomic Clock (CSAC) Performance During Rapid Temperature Change." `https://www.microsemi.com`.

[51] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948 vol.4, 1995.

[52] M. Kumar, M. Husian, N. Upreti, and D. Gupta, "Genetic Algorithm: Review and Application," *International Journal of Information Technology and Knowledge Management*, vol. 2, no. 2, pp. 451–454, 2010.

APPENDIX A

CELESTIAL BODY PARAMETERS

Celestial Body	Rotational Velocity Magnitude (rad/s), $\|\boldsymbol{\omega}\|$
Earth	7.292115053925690e-05
Moon	2.661699538941653e-06
Mars	7.088218127178316e-05

Table A.1: Rotation Rates of Earth, Moon, and Mars

Planet	Albedo, a_p	Radius (km), r_p
Mercury	0.106-0.138	2439.7
Venus	0.65-0.67	6052.0
Mars	0.15	3397.0
Jupiter	0.52	71492.0
Saturn	0.47	60268.0

Table A.2: Albedo and Equatorial Radius of the Visible Planets

Planet	Orbital Velocity Magnitude (km/s), $\|\boldsymbol{v}_p\|_2$
Mercury	47.4
Venus	35.0
Mars	24.1
Jupiter	13.1
Saturn	9.7

Table A.3: Orbital Velocity Magnitude of the Visible Planets

Semi-major Axis (km), a	Semi-minor Axis (km), c	Inverse Flattening Factor, $1/f$
6378.1370	6356.7523142	298.2572229328749

Table A.4: WGS84 Reference Ellipsoid Parameters

APPENDIX B

HARDWARE SPECIFICATIONS

Blue Canyon Technologies Standard Star Tracker	Attitude Knowledge
Cross-boresight Accuracy (1σ)	6 arcsec
About-boresight Accuracy (1σ)	40 arcsec

Table B.1: Star Tracker Measurement Uncertainties