Cis-Lunar Autonomous Navigation via Implementation of Optical Asteroid Angle-Only Measurements

Mark B. Hinga, Ph.D., P.E. *

Autonomous cis- or trans-Lunar spacecraft navigation is critical to mission success as communication to ground stations and access to GPS signals could be lost. However, if the satellite has a camera of sufficient quality, line of sight (unit vector) measurements can be made to known solar system bodies to provide observations which enable autonomous estimation of position and velocity of the spacecraft, that can be telemetered to those interested space based or ground based consumers. An improved Gaussian-Initial Orbit Determination (IOD) algorithm, based on the exact values of the f and g series (free of the 8th order polynomial and range guessing), for spacecraft state estimation, is presented here and exercised in the inertial coordinate frame (2-Body Problem) to provide an initial guess for the Batch IOD that is performed in the Circular Restricted Three Body Problem (CRTBP) reference frame, which ultimately serves to initialize a CRTBP Extended Kalman Filter (EKF) navigator that collects angle only measurements to a known Asteroid 2014 EC (flying by the Earth) to sequentially estimate position and velocity of an observer spacecraft flying on an APOLLO-like trajectory to the Moon. With the addition of simulating/expressing the accelerations that would be sensed in the IMU platform frame due to delta velocities caused by either perturbations or corrective guidance maneuvers, this three phase algorithm is able to autonomously track the spacecraft state on its journey to the Moon while observing the motion of the Asteroid.

This three phase algorithm enables highly-accurate autonomous spacecraft orbit estimation and continued navigation in the CRTBP frame for a single space based observer that is initialized with a limited set of measurements (at least 10) to the Asteroid 2014 EC, in both the inertial and CRTBP frames.

*Principal Investigator Researcher, AFRL/RDST, U.S. Air Force Research Laboratory, Kirtland, AFB, Albuquerque, NM 87117

I. Nomenclature

i	=	inclination
ω	=	periapsis
Ω	=	right ascension of the ascending node
ρ_k	=	kth range to target satellite
$\hat{\boldsymbol{\rho}}_k$	=	kth unit vector "line of sight" to target satellite
$\overline{\mathbf{r}}_k$	=	kth inertial position vector to target satellite
$\overline{\mathbf{v}}_k$	=	kth inertial velocity vector of target satellite
$\overline{\mathbf{R}}_k$	=	kth inertial position vector to observer satellite
f, g, c_k, d_k	=	Lagrange coefficients
μ	=	Earth's gravitational parameter, $3.986004418 \times 10^{14} \frac{m^3}{sec^2}$
μ^*	=	Earth-Moon non dimensional mass ratio, 0.01215059
ξ_k	=	right hand side of Gaussian triplet of observations
H	=	"design" matrix of unit vectors in Least Squares estimation
t_f	=	fixed time of arrival
t_0	=	time at which impulse is applied
δx	=	small perturbation in state vector
δr	=	small perturbation in position
δv	=	small perturbation in velocity
δt	=	small perturbation to fixed time of arrival
$\Phi(t_f, t_0)$	=	state transition matrix, from t_0 to arbitrary final time t_f
$A(t_f, t_0)$	=	Jacobian partials matrix, from t_0 to arbitrary final time t_f

II. Introduction

II.A. Initial Orbit Determination (IOD) - Inertial Reference Frame (2-Body Problem - Phase One)

THE problem of determining the orbit of an unknown object began with the advent of celestial mechanics seen in the works of Laplace[1] in 1780 and Gauss in 1801.[2] Their angle-only techniques utilized three observations to compute a position of a celestial object without the knowledge of range, which had to be guessed with the help of the roots of an eighth order polynomial. Later in 1889, Gibbs[3] developed his own technique enhancing the Gauss method of position estimation to include the determination of the velocity, which thus defines an orbit in space. Later, Herrick[4] improved on Gibbs' technique (for short arcs) with the use of a Taylor series to compute the velocity at the middle position vector. Clearly, these techniques were developed for celestial applications well before the beginning of the space age and availability of the computer. However astronomer Paul Herget[5] introduced an algorithm in 1964, that uses more than three angle-only measurements to estimate an orbit in which an iterative approach is applied through the variation of guessed geocentric distances to minimize a set of residuals in a least squares approach, using as many observations as are available, performed on an IBM 1620 computer.

Over the last several decades, many iterative methods to estimate the orbit of an unknown object (natural or artificial) using angle-only measurements have been developed. The Double r-iteration technique by Escobal[6] (1965), iterates on an initial guess of the range between the observer and a target object via the numerical partial derivatives and a Newton-Raphson iteration to converge on the true range. The Gooding[7] method (1993), using a minimum of three measurements, requires an initial "good" guess of the first and third ranges and whether the orbit is pro or retrograde. Common to all of these methods are the assumptions about a target satellite they make in order to converge to a solution for its orbit.

II.B. Development of the Coplanar System of Equations in the Inertial Frame (2-Body Problem - Phase One)

We define the *kth* inertial position vector of a target satellite for n observation times, $\mathbf{t}_1, \mathbf{t}_2, ..., \mathbf{t}_n$ as the following

$$\overline{\mathbf{r}}_k = \rho_k \hat{\boldsymbol{\rho}}_k + \overline{\mathbf{R}}_k, \; k = 1, 2, ..., n \tag{1}$$

where $\overline{\mathbf{R}}_k$ is the known observer position, ρ_k the scalar range from the observer to the target satellite and $\hat{\rho}_k$ is the corresponding "line of sight" unit vector. The observer can be on the Earth surface or on an orbiting satellite somewhere in space above the Earth.

For three observations, the following standard Gaussian equation expresses the relation of three vectors in inertial space as a linear combination summing to zero with three distinct coefficients, c_1, c_2 and c_3 as

$$c_1\overline{\mathbf{r}}_1 + c_2\overline{\mathbf{r}}_2 + c_3\overline{\mathbf{r}}_3 = 0. \tag{2}$$

This was the equation used by Gauss when he predicted the position of the first minor planet after its conjunction with the Sun (Herget).[5] If we substitute Equation 1 into Equation 2 and separate all known quantities on the right side, we end up with

$$c_1\rho_1\hat{\boldsymbol{\rho}}_1 + c_2\rho_2\hat{\boldsymbol{\rho}}_2 + c_3\rho_3\hat{\boldsymbol{\rho}}_3 = -c_1\overline{\mathbf{R}}_1 - c_2\overline{\mathbf{R}}_2 - c_3\overline{\mathbf{R}}_3. \tag{3}$$

After setting $c_2 = -1$ and after many algebraic manipulations, it is found that

$$c_1 = \frac{g_3}{f_1 g_3 - f_3 g_1} \tag{4}$$

$$c_3 = \frac{-g_1}{f_1 g_3 - f_3 g_1}, \tag{5}$$

where f_1, f_3, g_1, g_3, are the so-called "Lagrange f and g coefficients" (Curtis).[8] For this three observations example, the approximation (without the velocity term) of the f and g coefficients are $f_1 \approx 1 - \frac{1}{2}\frac{\mu}{r_2^3}\tau_1^2$, $f_3 \approx 1 - \frac{1}{2}\frac{\mu}{r_2^3}\tau_3^2$, $g_1 \approx \tau_1 - \frac{1}{6}\frac{\mu}{r_2^3}\tau_1^3$, $g_3 \approx \tau_3 - \frac{1}{6}\frac{\mu}{r_2^3}\tau_3^3$ where τ_1, τ_3 are the time intervals between successive measurements of $\hat{\boldsymbol{\rho}}_1, \hat{\boldsymbol{\rho}}_2$ and $\hat{\boldsymbol{\rho}}_3$.

Just like Gauss, we put Equation 3 into matrix format. Because the unknown ranges appear on both sides of the equation no closed form solution exists, forcing us to solve the system of coplanar equations through an iterative procedure. By starting with an initial guess of zero for the three unknown ranges we avoid forming an eighth order polynomial and instead, iteratively solve for the scalar ranges of ρ_1, ρ_2 and ρ_3.

$$\begin{bmatrix} \hat{\boldsymbol{\rho}}_1 & \hat{\boldsymbol{\rho}}_2 & \hat{\boldsymbol{\rho}}_3 \end{bmatrix} \begin{bmatrix} c_1\rho_1 \\ c_2\rho_2 \\ c_3\rho_3 \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{R}}_1 \\ \overline{\mathbf{R}}_2 \\ \overline{\mathbf{R}}_3 \end{bmatrix} \begin{bmatrix} -c_1 \\ -c_2 \\ -c_3 \end{bmatrix}. \tag{6}$$

We do this by inverting the 3x3 matrix of known $\hat{\boldsymbol{\rho}}_i$ unit vectors, pre-multiplying against the right hand side by this inverse, avoid forming and solving the traditional eighth order polynomial for the scalar magnitude of the middle inertial position vector r_2 (Vallado)[9] and solve for ρ_1, ρ_2 and ρ_3.

$$\begin{bmatrix} c_1\rho_1 \\ c_2\rho_2 \\ c_3\rho_3 \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\rho}}_1 & \hat{\boldsymbol{\rho}}_2 & \hat{\boldsymbol{\rho}}_3 \end{bmatrix}^{-1} \begin{bmatrix} \overline{\mathbf{R}}_1 \\ \overline{\mathbf{R}}_2 \\ \overline{\mathbf{R}}_3 \end{bmatrix} \begin{bmatrix} -c_1 \\ -c_2 \\ -c_3 \end{bmatrix}, \tag{7}$$

Where

$$\begin{bmatrix} c_1\rho_1 \\ c_2\rho_2 \\ c_3\rho_3 \end{bmatrix} \tag{8}$$

is defined as the state vector to be estimated.

At each iteration step, both the scalar ranges, ρ_i, to the target and the corresponding magnitudes, r_i, of the inertial position vectors, $\overline{\mathbf{r}}_i$, are also computed. Both are required at each iteration step (after inversion for the state vector) for computing the f and g coefficients. ρ_i is given by the relation which describes the simple geometry of the measurement scenario.

$$r_i = \left[{\rho_i}^2 + 2\rho_i\hat{\boldsymbol{\rho}}_i \cdot \overline{\mathbf{R}}_i + {R_i}^2\right]^{\frac{1}{2}} \tag{9}$$

Carrying this out to *n* observations, we still group the observations into sets of three where the *kth* relation is defined as

$$\overline{\mathbf{r}}_k = c_k\overline{\mathbf{r}}_{k-1} + d_k\overline{\mathbf{r}}_{k+1},\ k = 2, 3, ..., n-1 \tag{10}$$

Here the coefficients of c_k and d_k are obtained similarly above (in the three observation example) by expressing the vectors $\overline{\mathbf{r}}_{k-1}$ and $\overline{\mathbf{r}}_{k+1}$ in terms of position and (including) velocity vectors at time t_k, r_k and v_k using the Lagrange coefficients f and g in the following (Curtis)[8] / (Karimi)[10] format.

$$\overline{\mathbf{r}}_{k-1} = f_{k-1}\overline{\mathbf{r}}_k + g_{k-1}\overline{\mathbf{v}}_k \tag{11}$$

$$\overline{\mathbf{r}}_{k+1} = f_{k+1}\overline{\mathbf{r}}_k + g_{k+1}\overline{\mathbf{v}}_k \tag{12}$$

By inserting these two expressions into Equation 10, vector v_k can be eliminated and a relationship between $\overline{\mathbf{r}}_{k-1}$, $\overline{\mathbf{r}}_k$, and $\overline{\mathbf{r}}_{k+1}$ is completely defined giving the expression for c_k and d_k (Karimi)[10] as

$$c_k = \frac{g_{k+1}}{f_{k-1}g_{k+1} - f_{k+1}g_{k-1}} \quad \text{and} \quad d_k = -\frac{g_{k-1}}{f_{k-1}g_{k+1} - f_{k+1}g_{k-1}}. \tag{13}$$

Notice that by eliminating the velocity term in the Lagrange equations above we are constrained to a "triplet" of three position vectors to compute the f and g coefficients. We can still increase the number of observations above three, however we will group the measurements into sets of three as we take on more observations to compute an orbit. This is discussed below in Section II.C. The Lagrange coefficients f_k and g_k, appearing in Equation 13 can be expanded in series with a time difference $\Delta t_k = t_k - t_{k-1}$ up to a fourth order series expansion approximation as (Curtis 2005)[8]

$$f_{k-1} \approx 1 - \frac{\mu}{2r_k^3}\Delta t_k^2 - \frac{\mu(\overline{\mathbf{r}}_k \cdot \overline{\mathbf{v}}_k)}{2r_k^5}\Delta t_k^3 + \frac{\mu}{24}\left[-2\frac{\mu}{r_k^6} + 3\frac{v_k^2}{r_k^5} - 15\frac{(\overline{\mathbf{r}}_k \cdot \overline{\mathbf{v}}_k)^2}{r_k^7}\right]\Delta t_k^4 \tag{14}$$

$$f_{k+1} \approx 1 - \frac{\mu}{2r_k^3}\Delta t_{k+1}^2 + \frac{\mu(\overline{\mathbf{r}}_k \cdot \overline{\mathbf{v}}_k)}{2r_k^5}\Delta t_{k+1}^3 + \frac{\mu}{24}\left[-2\frac{\mu}{r_k^6} + 3\frac{v_k^2}{r_k^5} - 15\frac{(\overline{\mathbf{r}}_k \cdot \overline{\mathbf{v}}_k)^2}{r_k^7}\right]\Delta t_{k+1}^4 \tag{15}$$

$$g_{k-1} \approx -\Delta t_k + \frac{\mu}{6r_k^3}\Delta t_k^3 + \frac{\mu(\overline{\mathbf{r}}_k \cdot \overline{\mathbf{v}}_k)}{4r_k^5}\Delta t_k^4 \tag{16}$$

$$g_{k+1} \approx \Delta t_{k+1} - \frac{\mu}{6r_k^3}\Delta t_{k+1}^3 + \frac{\mu(\overline{\mathbf{r}}_k \cdot \overline{\mathbf{v}}_k)}{4r_k^5}\Delta t_{k+1}^4 \tag{17}$$

where $\mu = 3.986004418 \times 10^{14}\,\frac{m^3}{sec^2}$ is the Earth's gravitational parameter.

For time intervals Δt_k that are small in comparison with the orbital period, these coefficients f and g can be well approximated using the first two terms of the series expansion, which yields the approximate expressions (Curtis 2005).[8] These expressions for c_k and d_k will actually be used in experimentation because the velocity magnitude at the middle position vector, v_k is unknown.

$$c_k \approx \frac{\Delta t_{k+1}}{\Delta t_k + \Delta t_{k+1}}\left[1 + \mu\frac{(\Delta t_k + \Delta t_{k+1})^2 - \Delta t_{k+1}^2}{6r_k^3}\right] \tag{18}$$

$$d_k \approx \frac{\Delta t_k}{\Delta t_k + \Delta t_{k+1}}\left[1 + \mu\frac{(\Delta t_k + \Delta t_{k+1})^2 - \Delta t_k^2}{6r_k^3}\right] \tag{19}$$

where $k = 2, ..., n-1$. For equally space measured times($\Delta t = constant$) it is very easy to see that

$$c_k = d_k = \frac{1}{2}\left[1 + \frac{\mu}{2r_k^3}\Delta t^2\right] \tag{20}$$

II.C. Multiple Observations in the Inertial Frame (2-Body Problem - Phase One)

To extend the number of measurements to four or more using the Lagrange f and g coefficients in the coplanar system of equations, we must arrange them in groups of three so that the coefficients are still based on f and g being expressed as a combination of one initial position and one initial velocity vector. Since velocity was solved for in Equation 11 and inserted into Equation 12, we ended up with three position vectors, resulting in the relation in which the middle

position vector is a linear combination of the first and third, seen in Equation 10. Repeating this relation into a series of triplet measurements for a number $n > 3$ observations, the corresponding indices are seen as:

$$\begin{aligned}
\bar{\mathbf{r}}_2 &= c_2\bar{\mathbf{r}}_1 + d_2\bar{\mathbf{r}}_3 \\
\bar{\mathbf{r}}_3 &= c_3\bar{\mathbf{r}}_2 + d_3\bar{\mathbf{r}}_4 \\
\bar{\mathbf{r}}_4 &= c_4\bar{\mathbf{r}}_3 + d_4\bar{\mathbf{r}}_5 \\
&\vdots \\
\bar{\mathbf{r}}_k &= c_k\bar{\mathbf{r}}_{k-1} + d_k\bar{\mathbf{r}}_{k+1}
\end{aligned} \tag{21}$$

As we increase the number of observations, from $k = 2, 3, ..., n-1$ each triplet relation is preserved by considering the additional line of sight unit vector as the "third" observation to form r_{k+1}, complementing the two previous. As this additional position vector is included, the corresponding *kth* "right hand side" similar to that seen in Equation 3, called the"residual" ξ_k, is formed. The following logic illustrates this principle.

In Equation 10, we substitute for each of the three inertial position vectors with the position vector given by Equation 1, $\bar{\mathbf{r}}_k = \rho_k\hat{\boldsymbol{\rho}}_k + \overline{\mathbf{R}}_k$, expand the terms and organize the line of sight unit vectors on the left hand side and the position vectors of the observer on the right. (Note: the observer could be a terrestrial or spacebourne platform.)

$$\begin{aligned}
\bar{\mathbf{r}}_k &= c_k\bar{\mathbf{r}}_{k-1} + d_k\bar{\mathbf{r}}_{k+1} \\
\overline{\mathbf{R}}_k + \rho_k\hat{\boldsymbol{\rho}}_k &= c_k(\overline{\mathbf{R}}_{k-1} + \rho_{k-1}\hat{\boldsymbol{\rho}}_{k-1}) + d_k(\overline{\mathbf{R}}_{k+1} + \rho_{k+1}\hat{\boldsymbol{\rho}}_{k+1}) \\
c_k(\overline{\mathbf{R}}_{k-1} + \rho_{k-1}\hat{\boldsymbol{\rho}}_{k-1}) + d_k(\overline{\mathbf{R}}_{k+1} + \rho_{k+1}\hat{\boldsymbol{\rho}}_{k+1}) &= \overline{\mathbf{R}}_k + \rho_k\hat{\boldsymbol{\rho}}_k \\
c_k\overline{\mathbf{R}}_{k-1} + c_k\rho_{k-1}\hat{\boldsymbol{\rho}}_{k-1} + d_k\overline{\mathbf{R}}_{k+1} + d_k\rho_{k+1}\hat{\boldsymbol{\rho}}_{k+1} &= \overline{\mathbf{R}}_k + \rho_k\hat{\boldsymbol{\rho}}_k \\
c_k\rho_{k-1}\hat{\boldsymbol{\rho}}_{k-1} - \rho_k\hat{\boldsymbol{\rho}}_k + d_k\rho_{k+1}\hat{\boldsymbol{\rho}}_{k+1} &= \overline{\mathbf{R}}_k - c_k\overline{\mathbf{R}}_{k-1} - d_k\overline{\mathbf{R}}_{k+1} = \xi_k
\end{aligned} \tag{22}$$

On the left side Equation 22 the measurement "triplet" is shown as three known unit vectors pointing to the target satellite multiplied by unknown scalar ranges and Lagrange coefficients. The right hand side (labeled as ξ_k) contains the corresponding three known inertial position vectors of the observing (satellite) platform (or the known position of the Asteroid in the case of autonomous navigation) with the same unknown Lagrange coefficients. (The Lagrange coefficients are unknown because they are a function of the scalar magnitude of the inertial middle position vector of the target body.) Notice that the Lagrange coefficient of the "middle" line of sight unit vector remains as negative one for subsequent sets of triplet observations. Starting with the very first set of n = 3 observations, k = 2, we have

$$c_2\rho_1\hat{\boldsymbol{\rho}}_1 - \rho_2\hat{\boldsymbol{\rho}}_2 + d_2\rho_3\hat{\boldsymbol{\rho}}_3 = \overline{\mathbf{R}}_2 - c_2\overline{\mathbf{R}}_2 - d_2\overline{\mathbf{R}}_3 = \xi_2. \tag{23}$$

Putting Equation 23 into matrix format leads to

$$\begin{bmatrix} c_2\hat{\boldsymbol{\rho}}_1 & -\hat{\boldsymbol{\rho}}_2 & d_2\hat{\boldsymbol{\rho}}_3 \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \end{bmatrix} = \xi_2. \tag{24}$$

Adding a fourth observation (n = 4, k= 3), we group the measurements into sets of three yielding two equations,

$$\begin{aligned}
c_2\rho_1\hat{\boldsymbol{\rho}}_1 - \rho_2\hat{\boldsymbol{\rho}}_2 + d_2\rho_3\hat{\boldsymbol{\rho}}_3 &= \overline{\mathbf{R}}_2 - c_2\overline{\mathbf{R}}_1 - d_2\overline{\mathbf{R}}_3 = \xi_2 \\
c_3\rho_2\hat{\boldsymbol{\rho}}_2 - \rho_3\hat{\boldsymbol{\rho}}_3 + d_3\rho_4\hat{\boldsymbol{\rho}}_4 &= \overline{\mathbf{R}}_3 - c_3\overline{\mathbf{R}}_2 - d_3\overline{\mathbf{R}}_4 = \xi_3
\end{aligned} \tag{25}$$

To prepare this set of equations for a least squares solution (Herget)[5] we leave them set equal to their own residual and "stagger" them into matrix form in Equation 26

$$\begin{bmatrix} c_2\hat{\boldsymbol{\rho}}_1 & -\hat{\boldsymbol{\rho}}_2 & d_2\hat{\boldsymbol{\rho}}_3 & \mathbf{0} \\ \mathbf{0} & c_3\hat{\boldsymbol{\rho}}_2 & -\hat{\boldsymbol{\rho}}_3 & d_3\hat{\boldsymbol{\rho}}_4 \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \rho_4 \end{bmatrix} = \begin{bmatrix} \xi_2 \\ \xi_3 \end{bmatrix}, \tag{26}$$

where $\mathbf{0}$ represents a 3×1 vector of zeros.

Writing Equation 26 in compact matrix notation , we have

$$H\overline{\boldsymbol{\rho}} = \overline{\boldsymbol{\xi}}. \tag{27}$$

Expanding this to n observations, the system of equations is given as

$$\begin{bmatrix} c_2\hat{\boldsymbol{\rho}}_1 & -\hat{\boldsymbol{\rho}}_2 & d_2\hat{\boldsymbol{\rho}}_3 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & c_3\hat{\boldsymbol{\rho}}_2 & -\hat{\boldsymbol{\rho}}_3 & d_3\hat{\boldsymbol{\rho}}_4 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & c_4\hat{\boldsymbol{\rho}}_3 & -\hat{\boldsymbol{\rho}}_4 & d_4\hat{\boldsymbol{\rho}}_5 & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & d_{n-1}\hat{\boldsymbol{\rho}}_n \end{bmatrix} \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \vdots \\ \rho_n \end{bmatrix} = \begin{bmatrix} \xi_2 \\ \xi_3 \\ \xi_4 \\ \vdots \\ \xi_{n-1} \end{bmatrix}. \tag{28}$$

Notice that H is a $3(n-2) \times n$ matrix with $\overline{\boldsymbol{\rho}}$ and $\overline{\boldsymbol{\xi}}$ having the dimensions of $n \times 1$ and $3(n-2) \times 1$, respectively.

Instead of inverting matrix H on the left side and iteratively solving for the state vector of unknown scalar ranges $\overline{\boldsymbol{\rho}}$, as seen in the similar example of Equation 7, here we are able form the normal equation, Equation 29, treating $\overline{\boldsymbol{\xi}}$ as the residual with equal weights for all measurements in matrix H, and solve for $\overline{\boldsymbol{\rho}}$ in Equation 30. This iterative approach is similar to that of Herget.[5] But instead of trying to guess "good" starting distances for scalar ranges, ρ_k, initializing all of them to zero provides for excellent starting values and have always converged to the non-trivial solution (in this study) for orbiting platform observations. This result was also found to be true by Karimi.[10]

$$H^T H\overline{\boldsymbol{\rho}} = H^T\overline{\boldsymbol{\xi}}. \tag{29}$$

$$\overline{\boldsymbol{\rho}} = [H^T H]^{-1} H^T\overline{\boldsymbol{\xi}}. \tag{30}$$

By forming $[H^T H]$ we created a matrix that is $n \times n$ and is **possibly** of full rank n, where its column space completely spans the space of $\overline{\boldsymbol{\rho}}$ and is invertible when its nullity is the empty set $\emptyset$ (not including the null space containing precisely one zero vector). However, if the observations (line of sight unit vectors $\hat{\boldsymbol{\rho}}_k$) suffer the condition when successive measurements contain no or very small changing values of either relative right ascension α, or declination δ angles, then these measurements represent nothing more than a linear combination of the previous two unit vectors and do not span a complete dimensional space of n. This condition of "observational coplanarity" causes the corresponding rows of H to be the same or very similar to previous rows and leads to rank deficiency in H, where the number of elements in the basis of its null space is not zero. This means that the matrix $[H^T H]$ is not full rank n and any attempt to invert it yields singularities in the solution for $\overline{\boldsymbol{\rho}}$. Such a stressful scenario may occur when the observational coplanarity sinks down to angles of about $5°$ or less. At angles smaller than $1°$ the condition is generally pronounced, giving a solution of infinity, or other nonsense, for scalar ranges to the target satellite.

The investigation by Hinga[11] introduces a method which "salvages" the scenarios of difficult observational coplanarities by stabilizing the inversion of matrix $[H^T H]$, via the utilization of its eigenvectors and eigenvalues,providing for useful solutions.

II.D. Exact f and g coefficients in the Inertial Frame (2-Body Problem - Phase One)

The accuracy of the original Gauss method can be improved by replacing the approximation of the Lagrange f and g coefficients with the exact values of the coefficients in terms of universal anomaly χ which is solved for in the Universal Kepler's Equation (Prussing).[12] The coefficients are written in terms of the universal anomaly as

$$\begin{aligned} f &= 1 - \frac{\chi^2}{r_0} C(\alpha\chi^2) \\ g &= \Delta t - \frac{1}{\sqrt{\mu}} \chi^3 S(\alpha\chi^2). \end{aligned} \tag{31}$$

C and S are the Stumpff (Prussing)[12] functions and are defined as

$$
\begin{aligned}
S(\alpha\chi^2) &= \frac{\sqrt{\alpha\chi^2} - sin(\sqrt{\alpha\chi^2})}{(\sqrt{\alpha\chi^2})^3} \\
C(\alpha\chi^2) &= \frac{1 - cos(\sqrt{\alpha\chi^2})}{\alpha\chi^2}.
\end{aligned} \tag{32}
$$

The quantity α is defined as

$$
\alpha = \frac{2}{r_0} - \frac{v_0^2}{\mu}, \tag{33}
$$

where r_0 and v_0 are the magnitudes of the initial position and velocity respectively. The Universal Kepler's Equation in terms of the Universal anomaly χ is solved for in Equation 34

$$
\sqrt{\mu}\Delta t = \frac{r_0 v_{r0}}{\sqrt{\mu}}\chi^2 C(\alpha\chi^2) + (1 - \alpha r_0)\chi^3 S(\alpha\chi^2) + r_0\chi. \tag{34}
$$

v_{r0} is the magnitude of the tangential component of the velocity vector. The solution for χ in Equation 34 is achieved via the "Laguerre-Newton-Raphson" method described in (Prussing 1993).[12] This value of χ is inserted into Equation 32 to solve for both Stumpff functions which are then used in Equations 31 to compute exact values of f and g for any type of orbit. Convergence for χ is always guaranteed (Prussing 1993).[12] Equation 31 is used to compute the values of f and g in this study.

II.E. Solving for Velocity: Lambert (Phase One only) and Fixed Time of Arrival Solution (Phase One and Two)

After the vector of unknown ranges have been estimated, the series of inertial position vectors of the target spacecraft are calculated. To evaluate the spacecraft's velocity, a Lambert solver is applied to the first and last inertial positions with the known fixed time of flight between them. The solver used in this study is that developed by the European Space Agency (ESA) for their ten year "Rosetta" mission.[13] However robust and dependable this Lambert solver is, an improvement to its solution is obtained by implementing a Two Point Boundary Value Problem (TPBVP) shooting method to "fine tune" the departure and arrival velocity at initial and final time, repsectively. The guidance or "shooting" algorithm is based on the Linear Peturbation Theory (Battin)[14] developed for the **A**merica's **P**rogram for **O**rbiting **L**unar and **L**anding **O**perations (APOLLO) program during the 1960s. This TPBVP guidance algorithm can be applied equally well in either the rotating or inertial frames, where the Lambert method serves as a guess for the initial/final conditions in the inertial frame.

Using the notation of Battin,[14] the relation of the final state (position and velocity) error of the target ballistic spacecraft to the deviation of the current state from the nominal at time t is given by Equation 36. The term $\Phi(t_f, t_0)$ is known as the state transition matrix and is formed by taking the partial of the time rate of change of the state with respect to the state, Gelb.[15] This derivative is a matrix of partials, commonly known as the "Jacobian", shown as matrix A in Equation 35. In this study the Earth's J2 gravitation model is used to define the inerial accelerations in this derivative and for the accelerations in the rotating frame derivative, the CR3BP accelerations are used . Integrating this equation produces the state transition matrix, which relates the change in state from some time t_0 to another time t_f. The homogenous solution, for a fixed time of integration, is given in Equation 36, and yields a fixed time of arrival solution.

$$
\dot{\Phi}(t_f, t_0) = A(t_f, t_0)\Phi(t_f, t_0) \tag{35}
$$

$$
\delta x(t_f) = \Phi(t_f, t_0)\delta x(t_0) \tag{36}
$$

$$
\begin{bmatrix} \delta r(t_f) \\ \delta v(t_f) \end{bmatrix} = \begin{bmatrix} \tilde{R} & R \\ \tilde{V} & V \end{bmatrix} \begin{bmatrix} \delta r(t_0) \\ \delta v(t_0) \end{bmatrix}. \tag{37}
$$

Let us expand Equation 36 in terms of position "*r*" and velocity "*v*" components and assign convenient labels to the portions of the Φ matrix. Because we know the position of where the spacecraft starts and do not want to vary it, we

set $\delta r(t_0) = 0$. Then the expression for the perturbation to the position and velocity at final time t_f is,

$$\delta r(t_f) = R\delta v(t_0) \tag{38}$$
$$\delta v(t_f) = V\delta v(t_0) \tag{39}$$

Thus, the equation that defines how to vary (or perturb) the spacecraft velocity at time t_0, based on the missed distance at the target impact, is

$$\delta v(t_0) = R^{-1}\delta r(t_f), \tag{40}$$

where R is the upper right $3x3$ matrix of the state transition matrix $\Phi(t_f, t_0)$.

The velocity correction defined in Equation 40 is used as the iterative correction term at initial time t_0 in the search for the optimal improvement of the initial velocity to minimize the miss distance at t_f. Upon convergence of the this shooting/guidance method, the Lambert velocity solution has been improved allowing for more accurate orbital elements to be evaluated in the inertial frame and also allows for the improvement of the the initial state velocity in the synodic frame of the CRTBP problem.

II.F. Initial Orbit Determination (IOD) - Circular Restricted Three Body Problem (CR3BP) Reference Frame - Phase Two

Solving the Initial Orbit Determination (IOD) problem in the Circular Restricted Three Body Problem (CR3BP) reference frame introduces the issue of trying to compute an orbit in a synodic frame where conic sections do not exist and can not be applied. For example the f and g series are not applicable for the CR3BP because they are formulated in the inertial frame where the 2-body problem is defined. In this study, an initial guess for the state of an unknown orbit in the CRBP basis space is first computed (using an exact f and g series Gaussian-Batch IOD) in the inertial frame and then is transformed into the rotating frame. (Note: in some cases this transformation may not be necessary)

This transformed intial state is then used to initialize a numerically integrated batch filter expressed in the CRTBP synodic frame. The initialized non-dimensional CRTBP equations of motion (state dynamic derivatives) are numerically integrated/propagated through the observations taken in the CRTBP frame, in such a way as to minimize the residuals between the model of the motion of the perceived target body and the perceived (angle-only) measurements to that target body. Linearized analytic functions describing motion in the CR3BP reference frame, Grebow,[16] may possibly be used in a batch filter, but their radius of convergence (i.e. their distance to truth caused by linearization) is worrisome and is not examined.

This "three phase" algorithm will provide an immediate position and velocity report for real-time analyses of spacecraft trajectories with the use of at least ten observations (ten "line of sight" unit vectors pointing from the observing spacecraft to the target Asteroid) enabling a real-time (autonmous on board platform) orbit estimate of the spacecraft position and velocity. This spacecraft solution, along with the covariance, can be reported quickly to other ground or space-based sensors.

The significance of the algorithm in this study is that there are no assumptions or guesses made about the spacecraft's orbit or about its scalar range from the observed Asteroid. A value of zero is an excellent start for the unknown ranges contained in the state vector, between the Spacecraft and the Asteroid of this least squares algorithm. This initialization of zero is made possible due to the geometry between the observed Asteroid and the Spacecraft, see Equation 9. There is no guessing of range or whether motion is pro/retro grade nor is there a required search for the real roots of an eighth order polynomial as mentioned above in the original Gauss method. By substituting the expression for r_2 (Herget)[5] during the iteration of the least squares solution (first phase - inertial), ρ can be optimally varied until convergence to a real root. If the problem is not ill-conditioned the convergence to truth is "guaranteed", consistently throughout the scope of this investigation. If the scenario is ill-conditioned, i.e. a severe observational coplanarity exists, a divergent nonsensical answer is inevitable. However, using the stabilization algorithm proposed by Hinga,[11] the so-called Eigenvalue Descent Control (EDC) method, provides for good orbit estimates when the above mentioned traditional algorithms all fail. However, in this study, we are not interested in investigating observational coplanar issues. An Asteroid is chosen that flies well above (at high angle to) the orbit-trajectory-plane of the APOLLO-like Spacecraft.

Further significance of this algorithm is that the inertial phase of the Gaussian-Initial Orbit Determination (IOD) method builds the system of co-planar-motion conditions that form the normal equation and is constructed from and based on the analytical exact functions of the f and g coefficient series expansion of universal variables (Prussing).[12] The co-planar-motion is not to be confused with "observational coplanarity" but rather, is the important assumption that a series of satellite position vectors in inertial space form a plane in which its motion is constrained. The latter

is the relative geometry between an observer and target body and connotes the angular height that body has above the observer's orbital plane of motion.

Once these inertial initial conditions of position and velocity are determined they are transformed into the CRTBP synodic frame and used to initialize a Batch Filter (Differential Corrector (DC)) IOD algorithm formulated in the CRTBP synodic frame. Upon converge of the CRTBP Batch Filter, the state solution is provided to a CRTBP Extended Kalman Filter (EKF) orbit estimator for continued tracking using subsequent line of sight unit vectors to the observed Asteroid in the CRTBP reference frame.

To evaluate the CRTBP orbit error computed by the algorithm in this study, it is important to remember that there are no classical orbital elements. Evaluation of estimated CRTBP orbits is provided by the covariance estimates (which are merely "opinions" of uncertainty given by the EKF) and the distance of the state positions and velocities from the known truth as defined by the simulation. The so called Jacobi Constant (the only constant integral of motion in the CRTBP frame) is also computed by the CRTBP EKF as a helpful indicator of it's opinion of the quality of its numerical integration.

For purposes of curiousity, the 2-Body inertial EKF is computed in parallel throughout these experiments to illustrate how the 2-body solution performs in the CRTBP synodic frame. Comments are made to that effect and the results are not surprising that the closer the spacecraft approaches the Moon, the worse the 2-Body intertial solution becomes.

Finally, it is important to mention that all satellite orbit and measurement simulations, algorithm development and verification were carried out inside the self-developed and validated Hinga Orbit Simulator (HOrbitSIM).[17]

III. Equations of Motion in the CRTBP reference frame - Phase Two and Three

The formulation of the Circular Restricted Three Body Problem (CRTBP) is based on the following assumption Vallado[9] and is illustrated in Figure 1:

1. The primary and secondary bodies move in circular orbits about the center of mass, which lies between the two objects.

2. The mass of the third body (satellite/spacecraft) is negligible compared to that of the major bodies.

Furthermore, the reference frame of this problem is rotating along with the circular motion of both the Earth and Moon orbiting about the common barycenter of total mass in this binary system. This frame is commonly called a *synodic coordinate frame* and is rotating with angular velocity ω_s. The total acceleration experienced by the negligible 3rd body mass is a sum total of the gravitational pull from the Earth and Moon, centripetal and Coriolis effects. Using the notation seen in Vallado,[9] Figure 1 illustrates also the geometry in this CRTBP system.

The $\hat{x}_s$ axis (of the rotating frame) points in the direction of the primary body (the Earth). The $\hat{y}_s$ axis lies at a right angle in the plane of the Earth-Moon rotating motion. The $\hat{z}_s$ axis is normal to the $\hat{x}_s - \hat{y}_s$ plane and is aligned with the fixed barycentric frame $(\hat{x}_B, \hat{y}_B, \hat{z}_B)$. The distances from the barycenter to each object (r_{B1}, r_{B2}) appear in the below equations of motion. The mass ratio μ^* allows for the normalization of the problem, Vallado.[9]

The equation which contains all terms of the acceleration seen in this snynodic reference frame is expressed as follows Vallado:[9]

$$\ddot{\overline{\mathbf{r}}}_{Bsat} = \overline{a}_s + \dot{\overline{\omega}}_s \times \overline{r}_s + \overline{\omega}_s \times (\overline{\omega}_s \times \overline{r}_s) + 2\overline{\omega}_s \times \overline{v}_s + \overline{a}_{org}. \tag{41}$$

Because the rotation rate of the reference is constant, $\dot{\overline{\omega}}_s = 0$ and the origin is not accelerating, $\overline{a}_{org} = 0$, Equation 41 can be reduced. But as we do that we express this relation in cartesian coordinates to simplify it for better interpretation of integration within the synodic frame.

$$\ddot{\overline{\mathbf{r}}}_{Bsat} = \ddot{\overline{r}}_s + {\omega_s}^2(x\hat{x}_s + y\hat{y}_s) + 2\omega_s(\dot{y}\hat{x}_s + \dot{x}\hat{y}_s). \tag{42}$$

At this point we define the total potential R for this system and take the gradient of it and set it equal to the acceleration in Equation 42. Therefore,

$$R = \frac{\mu_1}{r_1} + \frac{\mu_2}{r_2}, \tag{43}$$

where μ_1 and μ_2 are the gravitational parameters of the primary and secondary body, repsectively and r_1 and r_2 are the distances between the third body to the first and second attracting bodies, respectively.

If we take the gradient of the potential R

$$\nabla R = \frac{\partial R}{\partial x}\hat{x}_s + \frac{\partial R}{\partial y}\hat{y}_s + \frac{\partial R}{\partial z}\hat{z}_s \tag{44}$$

and set it equal to the acceleration $\ddot{\bar{\mathbf{r}}}_{Bsat}$,

$$\ddot{\bar{\mathbf{r}}}_{Bsat} = \nabla\left(\frac{\mu_1}{r_1} + \frac{\mu_2}{r_2}\right), \tag{45}$$

and then simplify, we end up with the following system of (dimensional) equations:

$$\begin{aligned}
\ddot{x} - 2\omega_s\dot{y} - {\omega_s}^2 x &= \frac{\partial}{\partial x}\left(\frac{\mu_1}{r_1} + \frac{\mu_2}{r_2}\right) = -\frac{\mu_1(x - r_{B1})}{{r_1}^3} - \frac{\mu_2(x + r_{B2})}{{r_2}^3} \\
\ddot{y} + 2\omega_s\dot{x} - {\omega_s}^2 y &= \frac{\partial}{\partial y}\left(\frac{\mu_1}{r_1} + \frac{\mu_2}{r_2}\right) = -\frac{\mu_1 y}{{r_1}^3} - \frac{\mu_2 y}{{r_2}^3} \\
\ddot{z} &= \frac{\partial}{\partial z}\left(\frac{\mu_1}{r_1} + \frac{\mu_2}{r_2}\right) = -\frac{\mu_1 z}{{r_1}^3} - \frac{\mu_2 z}{{r_2}^3}
\end{aligned} \tag{46}$$

where

$$\begin{aligned}
r_1 &= \sqrt{(x - r_{B1})^2 + y^2 + z^2} \\
r_2 &= \sqrt{(x + r_{B1})^2 + y^2 + z^2}.
\end{aligned} \tag{47}$$

To finalize the equations of motion that are used in this study we must make them "nondimensional". We chose to set the mass of the primary attracting body, $m_1 = 1 - u^*$ and the secondary body, $m_2 = u^*$. u^* is known as the mass ratio of the restricted three body problem. Choosing the value of unity for the distance between the two bodies, the larger and smaller masses are at distances from the origin u^* and $1 - u^*$, respectively Vallado.[9] We can now redefine r_1 and r_2 as

$$\begin{aligned}
r_1 &= \sqrt{(x - u^*)^2 + y^2 + z^2} \\
r_2 &= \sqrt{(x - u^* + 1)^2 + y^2 + z^2}.
\end{aligned} \tag{48}$$

With $\omega_s = 1$, we can rewrite the equations of motion into their nondimensional form as:

$$\begin{aligned}
\ddot{x} - 2\dot{y} - x &= -\frac{(1 - u^*)(x - u^*)}{{r_1}^3} - \frac{u^*(x + 1 - u^*)}{{r_2}^3} \\
\ddot{y} + 2\dot{x} - y &= -\frac{(1 - u^*)y}{{r_1}^3} - \frac{u^* y}{{r_2}^3} \\
\ddot{z} &= -\frac{(1 - u^*)z}{{r_1}^3} - \frac{u^* z}{{r_2}^3}
\end{aligned} \tag{49}$$

where the characteristic length "l" is the Earth to Moon distance and the characteristic velocity/speed is l times the synodic rate of the Moon angular velocity.

III.A. Phase One - Spacecraft Inertial Batch Filter IOD

At just under one day of flight, the Spacecraft "gets lost" and loses knowledge of its position and velocity. It has no idea where it is. Shortly thereafter its onboard camera notices and identifies an Asteroid (2014 EC) flying by the Earth and identifies and verifies the Asteroid's identity and accesses the stored ephemeris in its flight computer. By taking ten angle only measurements (line of sight unit vectors) to the center mass of the Asteroid, the spacecraft is able to estimate with a Batch DC Filter (using the Gaussian method described in Section II.B), its own position and velocity with the following inertial accuracy and confidence:

Note: Because the measurment model XII.E can not observe velocity, the estimated velocity terms of motion are declared to have the same level of uncertainty as that for position at initial (epoch) time.

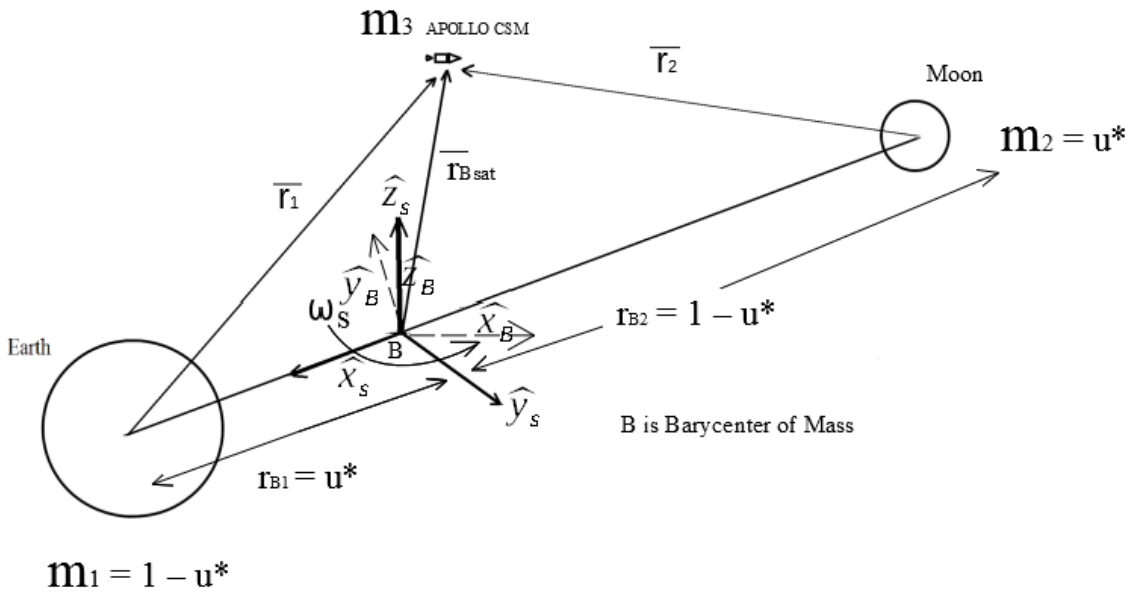

Figure 1. CRTBP Body Diagram

Spacecraft Orbit State: Soln (diff to truth)	Spacecraft Observes Asteroid EC 2014
x (km)	686
y (km)	-107
z (km)	19
$\dot{x}$ (m/s)	1
$\dot{y}$ (m/s)	0.3
$\dot{z}$ (m/s)	-5.3
overall percent error	position 0.07 velocity 0.37

Table 1. Inertial Batch Solution Difference to Truth

Spacecraft Orbit State: Soln (standard deviation)	Spacecraft Observes Asteroid EC 2014
x (km)	1.25
y (km)	1.25
z (km)	1.25
$\dot{x}$ (km/s)	1.25
$\dot{y}$ (km/s)	1.25
$\dot{z}$ (km/s)	1.25
overall rms (km)	3.300

Table 2. Inertial Batch Solution Standard Deviation

III.B. Phase Two - Spacecraft CRTBP Batch Filter IOD

By taking ten angle only measurements (line of sight unit vectors) to the center mass of the Asteroid in the CRTBP synodic frame, the spacecraft is able to estimate with a Batch Filter (using the numerical method described in Section XI.A), its own position and velocity with the following accuracy and confidence in the synodic CRTBP frame:

Spacecraft Orbit State: Soln (diff to truth)	Spacecraft Observes Asteroid EC 2014
x (km)	325
y (km)	142
z (km)	-0.259
$\dot{x}$ (m/s)	-2.85
$\dot{y}$ (m/s)	3.814
$\dot{y}$ (m/s)	0.169
overall percent error	position 0.18 velocity 0.33

Table 3. CRTBP Batch Solution Difference to Truth

Spacecraft Orbit State: Soln (standard deviation)	Spacecraft Observes Asteroid EC 2014
x (km)	1.565
y (km)	2.379
z (km)	0.321
$\dot{x}$ (km/s)	11.3
$\dot{y}$ (km/s)	15.7
$\dot{z}$ (km/s)	42.3
overall rms (km)	3.029

Table 4. CRTBP Batch Solution Standard Deviation

Using the computed starting conditions of this phase, the Spacecraft is able to initialize the CRTBP EKF of Phase Three of the Algorithm and continue taking angle only unit vector line of sight measurements to sequentially compute the position and velocity of the Spacecraft in the CRTBP Navigation frame. Note: The confidence in estimated velocity is low because the measurement model XII.E can not observe velocity. In this case, each standard deviation in the velocity estimate is not the same because the CRTBP EKF does indeed model the velocity. With a better tuned filter , the confidences could be reduced.

IV. Phase Three - Spacecraft CRTBP EKF

Now that the Spacecraft knows its position and velocity, it can maintain that knowledge with an EKF. This Navigator (EKF) is kept running until the spacecraft performs a maneuver to insert itself into a circular orbit about the Moon with a target altitude of 140 Km. Figure 2 illustrates a ”bird’s eye” view of the CRTBP EKF estimated (in red) and the simulated (truth) trajectory along with 50 other CRTBP orbits (for perspective) that are simulated in this investigation. From this perspective it appears that the EKF performance is quite good, see Figures 3 and 4. The red trajectory is coincident with the blue trajectory. Even with a perturbation of 100 m/s at t = 2 days, the EKF is able to sense the delta velocity (with inclusion of the perturbation in the equations of motion that a typical IMU would sense) and maintain knowledge of the state on a target path to the Moon by computing and applying a correction guidance maneuver that is applied two minutes later.

Figures 5 and 6 show the performance of the Kalman Filter in comparison to the Truth (simulation) up until it is “turned-off” just before Lunar orbit insertion. (Figure 7 illustrates the performance of the Filter during and after the 100 m/sec perturbation and guidance correction maneuver, which are separated by 30 seconds.)

Notice that including the 100 m/sec perturbation, the CRTBP EKF is able to keep track of the state. (Note: without implementing sensed accelerations in the CRTBP EKF equations of motion, the EKF would require about two days to re-converge back to a state “close enough” to truth that could be useful to compute a guidance maneuver.) The perturbation is applied in the $\hat{z}_{CRTBP}$ direction. The computed guidance maneuver results in a correction of velocity in the opposite $\hat{z}_{CRTBP}$ direction.

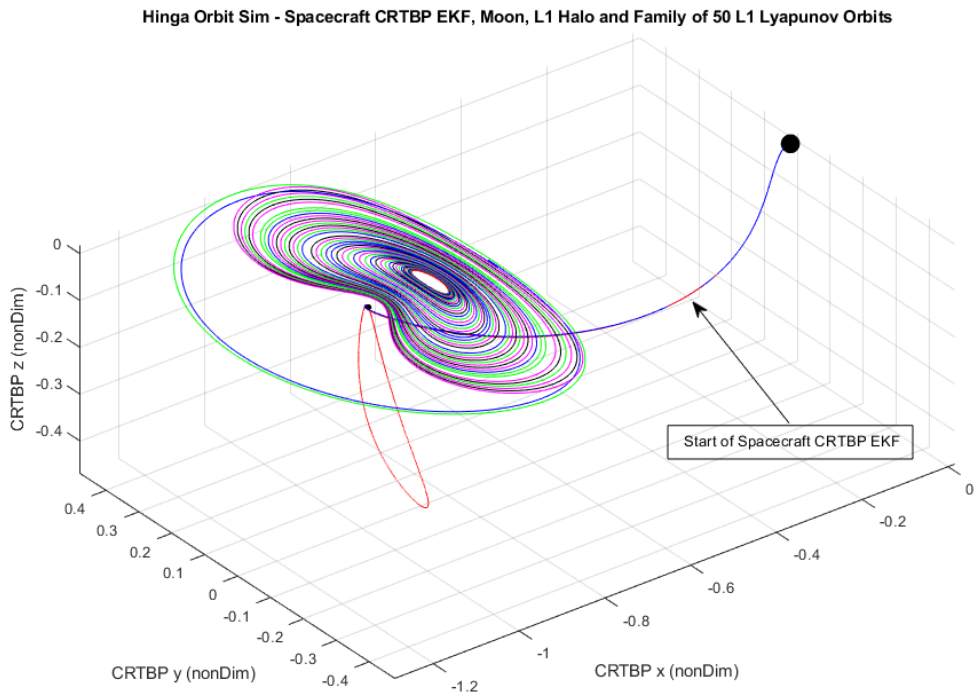

Figure 2. **CRTBP EKF Trajectory to Moon and Halo/Lyupanov/DRO Orbits**

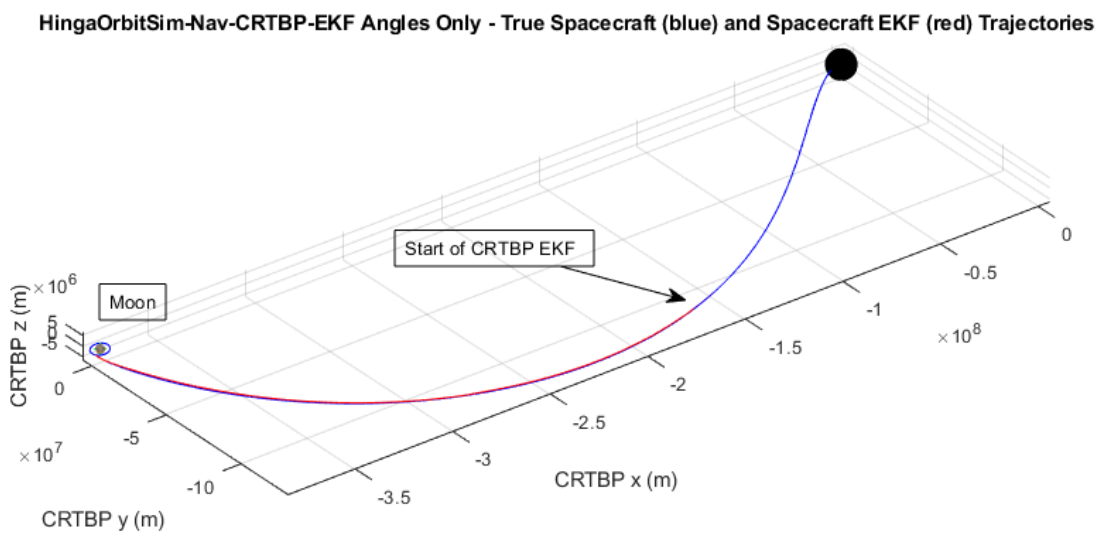

Figure 3. **CRTBP EKF Trajectory to Moon**

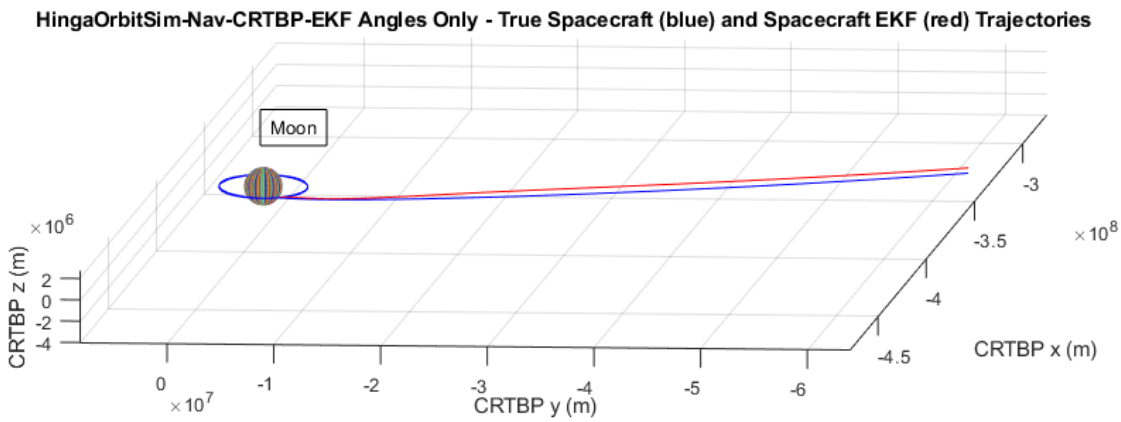

Figure 4. **CRTBP EKF Approach and Insertion into Lunar Orbit**

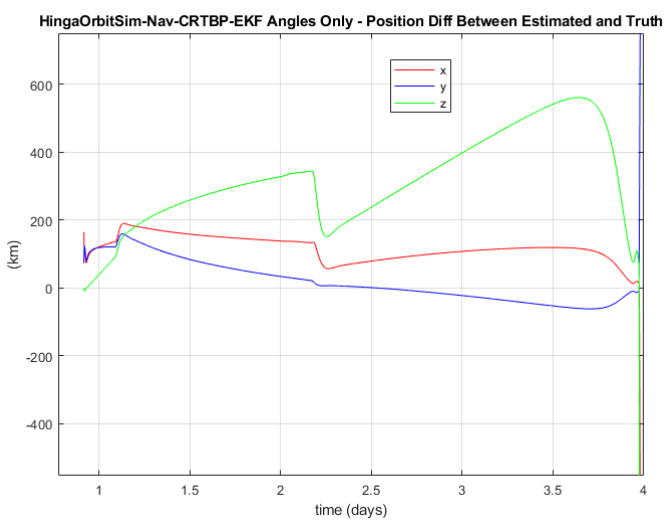

Figure 5. **CRTBP EKF Trajectory Diff to Truth Position**

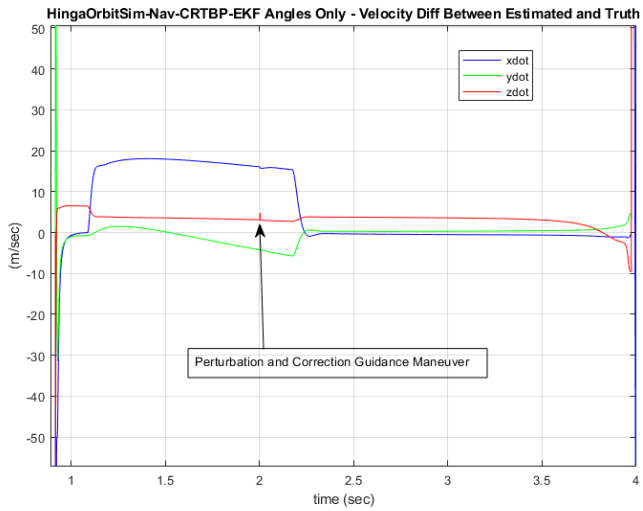

Figure 6. CRTBP EKF Trajectory Diff to Truth Velocity

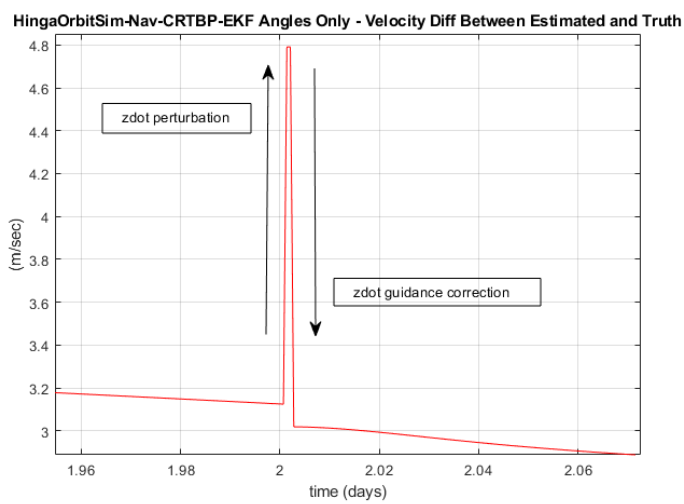

Figure 7. CRTBP EKF Trajectory Diff to Truth Velocity: Perturbation and Guidance Correction

Figures 8 and 9 show the confidence of the CRTBP EKF of its position and velocity estimate, respectively, up until it is "turned-off" just before Lunar orbit insertion.

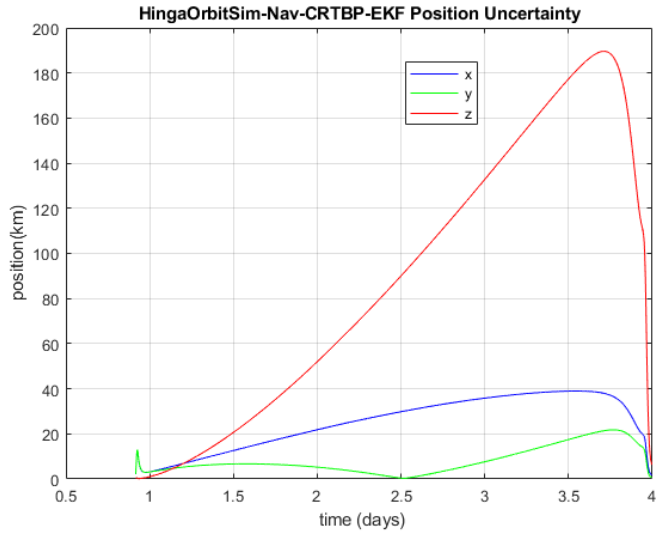

Figure 8. CRTBP EKF Trajectory Standard Deviation Position

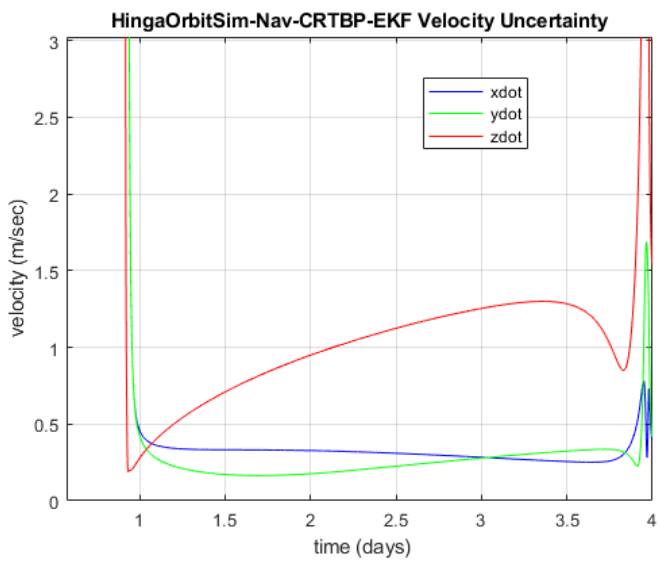

Figure 9. CRTBP EKF Trajectory Standard Deviation Velocity

Figure 10 shows the residuals of the angle measurments formed by the Kalman Filter up until it is also "turned-off" at four days just before Lunar orbit insertion. Because the ephemerides of Asteroid 2014 EC are well known (given by the JPL DE421 binary ephemeris[18]) and with the accuracy of a good onboard camera, these residuals are achievable.

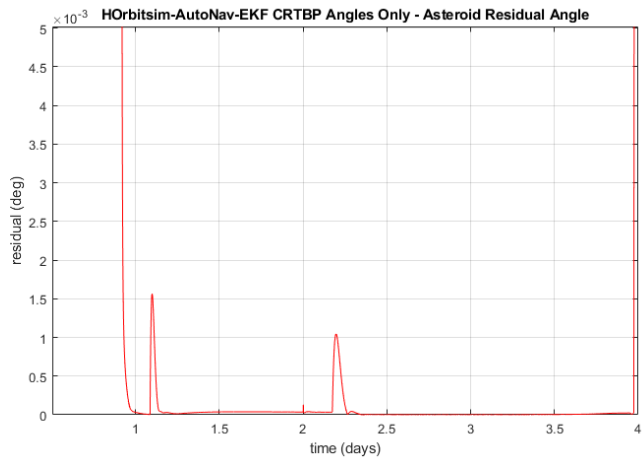

Figure 10. CRTBP EKF Angle Measurement Residuals

Figure 11 shows the inertial view of the Earth to Moon trajectory and a subsequent one day propagation of the Spacecraft Lunar orbit. Notice the very poor performance of the **inertial** EKF estimating its solution in a synodic reference frame. See Figures 20 and 21.

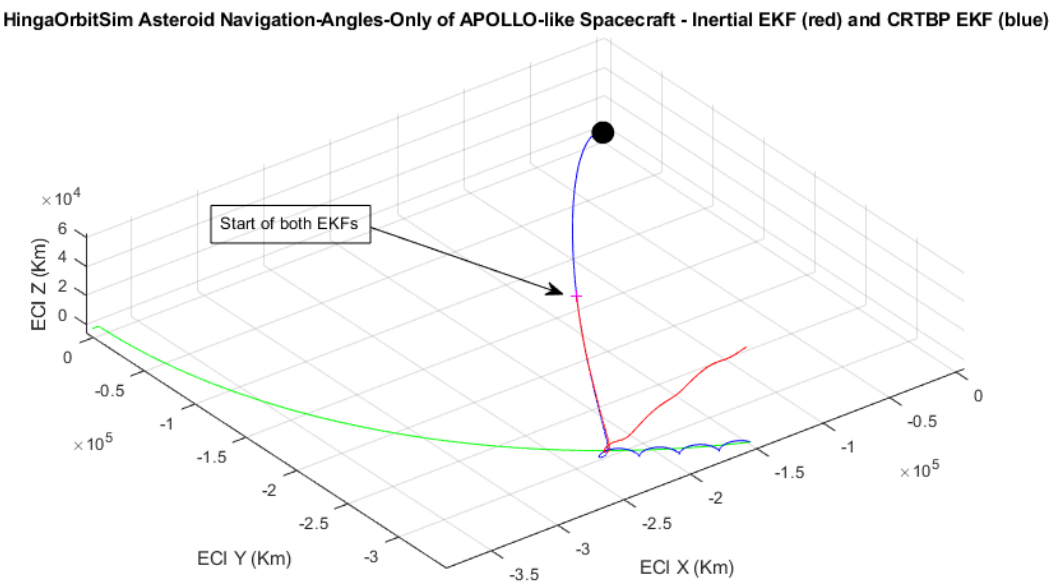

Figure 11. Inertial view of the Lambert and CRTBP Spacecraft Trajectories

V. Conclusion and Future Work

The phase one (inertial frame) Batch Filter IOD algorithm of this study avoids the traditional approach of solving for the roots of an eighth order polynomial for the scalar distance to the middle position vector, originally developed by Gauss. There is no guessing of any scalar ranges to initiate solution, only the simple initialization/guess with zero is required. There is no guessing of whether target satellite motion is pro or retro grade. Convergence to a real root (in the inertial frame) is guaranteed and within the scope of this study, all real roots were converged to correctly. Furthermore, it was found that implementation of the linear perturbation theory proved very helpful in improving the velocity computations from the estimated positions, which directly influenced the final quality of the computed orbits in both the inertial and the synodic CRTBP reference frames. Convergence of the inertial vehicle state solution marks the end of phase one of this overall algorithm. The inertial state is provided to the CRTBP Batch Filter IOD for its initialization and marks beginning of phase two.

In phase two (synodic frame) of the overall algorithm, the CRTBP Batch Filter IOD begins. It uses the (first order ordinary differential) equations of motion to estimate the state error of the vehicle during the process of differential corrections that leads to an initial estimate of the vehicle state of position and velocity. It is this initial vehicle state (in the CRTBP frame) that initializes the guessed state at time zero for the CRTBP EKF (phase three). This three-phase algorithm succeeded and demonstrated proof of concept required for dependable sequential state estimation in the synodic CRTBP frame with no a-priori information.

The CRTBP EKF developed for this study demonstrated that it can be initialized with the state estimate from the CRTBP batch solution and not only track the target satellite state using angle only measurements to an Asteroid, but also can maintain estimate of the spacecraft state quite well during a perturbation event of 100 m/sec with the inclusion of simluated IMU accelerations into the CRTBP EKF derivative equations.

Now that this three phase algorithm has been proven capable of estimating the state of a spacecraft in its cis-lunar trajectory to the Moon and can provide a good enough solution to compute a guidance correction to its trajectory, well enough to reach and insert itself into a circular Lunar orbit, it will be worthwhile to investigate its performance under simulated noise conditions applied to the line of sight unit vectors, as proposed in Section XII.F. Finally, validating its capability and performance using actual measurements (either in real time or postprocessing) taken on board a satellite platform on an Earth-Moon trajectory, or perhaps on a satellite flying in a Halo Orbit for purposes of station-keeping, will prove valuable.

VI. Appendix

VII. Application of Lambert and the TPBVP to Compute a Simulated Trajectory to the Moon in CRTBP Rotating Frame

Figure 12 illustrates the successful best guess of a Lambert transfer trajectory that initializes the starting conditions of a CRTBP Two Point Boundary Value Problem (TPBVP) "shooting method" to converge to a trajectory in the synodic frame that can rendezvous close enough to the Moon (140Km altitude) to enable capture into a Lunar orbit. The delta velocity needed for this Earth to Moon trajectory is $\Delta V = 212.00$ m/sec.

The parking orbit from which the trajectory to the Moon is computed has the following classical orbit elements.

Classical Orbital Elements	Spacecraft Lunar Orbit
a (km)	137661800.9
e	0.949
i (deg)	18.08
ω (deg)	173.11
Ω (deg)	351.82
ν (deg)	175.21
E (deg)	150.87
u (deg)	348.32
Period (hours)	141.19

Table 5. Spacecraft Earth Parking Orbit Classical Orbit Elements

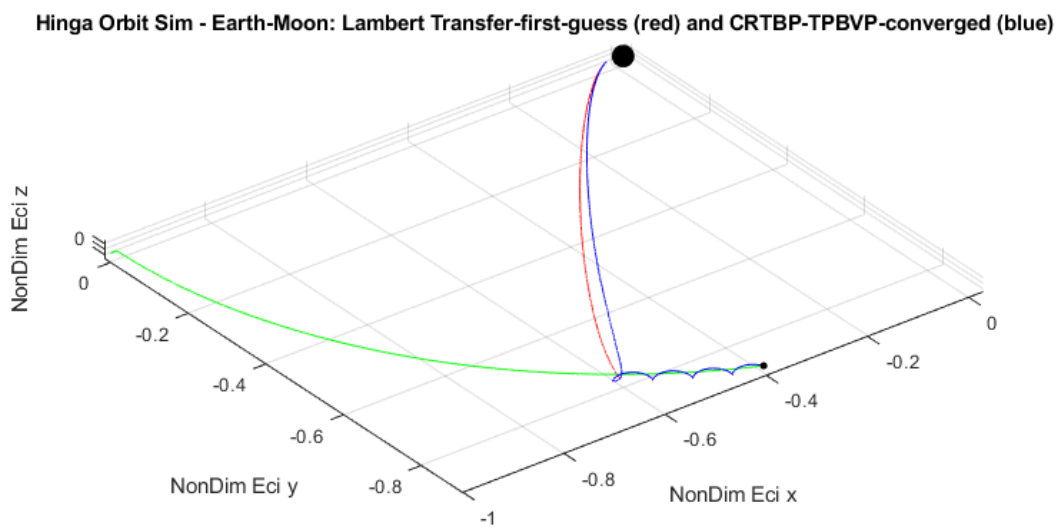

Figure 12. Lambert Best Guess allowing CRTBP Spacecraft Trajectory Lunar Rendezvous Convergence and Orbit Insertion

VIII. Computation of Lunar Orbit Insertion Maneuver - CRTBP Rotating Frame Arrival - Selenocentric Frame

The method of computing the delta velocity maneuver to put the Spacecraft into a Lunar orbit at the chosen altitude was to first satisfy the CRTBP position coordinates constraint in the Newton iteration of the TPBVP shooting method. The moment the Spacecraft arrives at the desired CRTBP coordinates, the required delta velocity maneuver is applied to put it into a circular orbit at the same inclination as the approach trajectory has with respect to the selenocentric frame. It was not necessary to change the inclination of the Spacecraft's orbit about the Moon. The necessary delta velocity burn is computed in the following steps:

First we compute the desired final speed to which the Spacecraft must slow down to, namely,

$$v_{circ} = \left[\frac{\mu_{\text{Moon}}}{r_2}\right]^{\frac{1}{2}}, \tag{50}$$

where $\overline{r_2}$ is that described in Figure 1 and $||r_2|| = R_{Moon}+$Lunar Altitude. Then we compute the desired direction in which a final circular orbit about the Moon exists. This direction is along the "local horizon" direction which is normal to the position vector $\overline{r_2}$. This unit direction and that of the arrival velocity is computed in the following way:

$$\hat{r_{2}}_{unit} = \frac{\overline{r_2}}{||\overline{r_2}||} \tag{51}$$

$$\hat{v_{2}}_{unit} = \frac{\overline{v_2}}{||\overline{v_2}||}. \tag{52}$$

The direction of the selenocentric angular momentum unit vector is defined as

$$\hat{h_{2}}_{unit} = \hat{r_{2}}_{unit} \times \hat{v_{2}}_{unit} \tag{53}$$

Now, the local horizon unit direction can be computed and is defined as:

$$\hat{\rho}_{unit} = \hat{h_{2}}_{unit} \times \hat{r_{2}}_{unit}. \tag{54}$$

Thus, the direction and magnitude of the required velocity needed to enter a circular orbit about the Moon is defined as

$$\overline{V}_{final} = v_{circ} \times \hat{\rho}_{unit}. \tag{55}$$

Therefore the required delta velocity burn, at arrival, is

$$\overline{\Delta v} = \overline{V}_{final} - \overline{v_2}, \tag{56}$$

where both $\overline{V}_{final}$ and $\overline{v_2}$ are vectors w.r.t. to the Moon in Figure 1 expressed in the CRTBP synodic frame.

Note: we can use these terms of position $\overline{r_2}$ and velocity $\overline{v_2}$ in the CRTBP reference frame and ignore the effects of the Earth because the Spacecraft is so very close to the Moon.

Table 6 shows the resulting classical orbital elements of the selenocentric Spacecraft orbit. Note: The initially achieved Lunar altitude is 2322 Km. The remaining descent to 140Km altitude can be reached with subsequent maneuvers. The delta velocity needed for this maneuver is $\Delta V = -716.28$ m/sec. A bit high, but it succeeds. This value could be reduced by having a longer transfer time (more than four days) from the Earth to the Moon.

Classical Orbital Elements	Spacecraft Lunar Orbit
a (km)	4217
e	0.000
i (deg)	170.72
ω (deg)	243.39
Ω (deg)	104.14
ν (deg)	15.11
E (deg)	15.11
u (deg)	258.50
Period (hours)	6.825

Table 6. Spacecraft Lunar Classical Orbit Elements

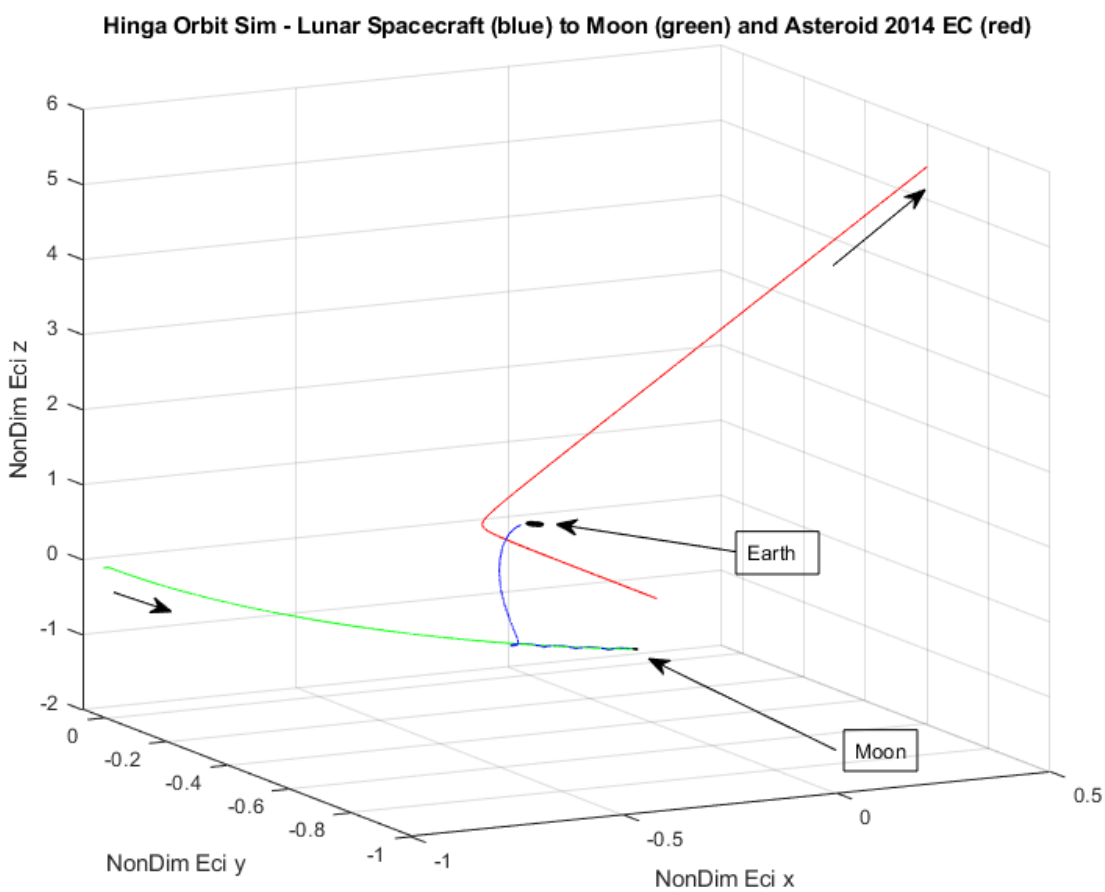

Figure 13. Inertial Asteroid, Moon and Spacecraft Trajectory

IX. Jacobi Constants

As the only constant integral of motion in the CRTBP reference frame, it is important to compute the Jacobi Constant in both the simulation (declared as Truth) and the CRTBP EKF. If this computed constant remains "constant" throughout any numerical propagation, then it increases the confidence of the computed states.

Figures 14 and 15 illustrate the Jacobi constant that the CRTBP EKF computes during its estimation of the vehicle state. Notice that during any perturbation the Jacobi constant is expected to change. However, it does appear to change throughout the estimation and that is because at every time step the CRTBP EKF is applying a small correction to position and velocity to maintain a "track" on to what it believes is the true state of the spacecraft as it flies through cis-Lunar space.

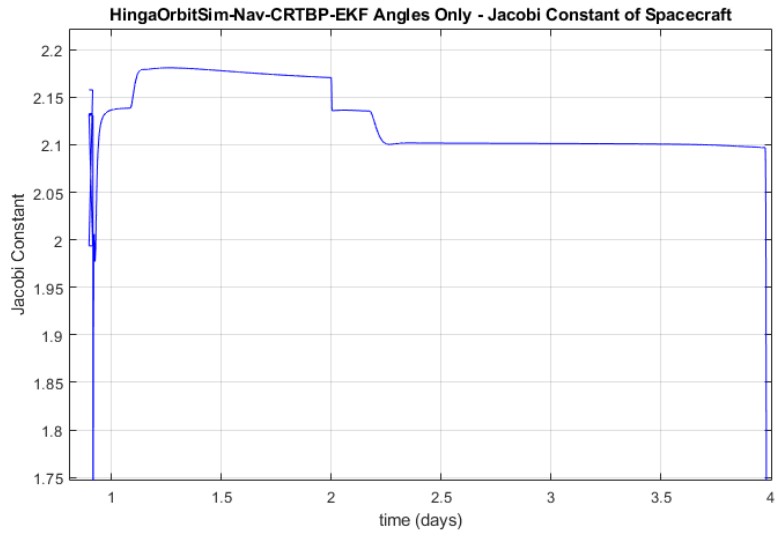

Figure 14. Spacecraft Jacobi Constant as computed by the CRTBP EKF

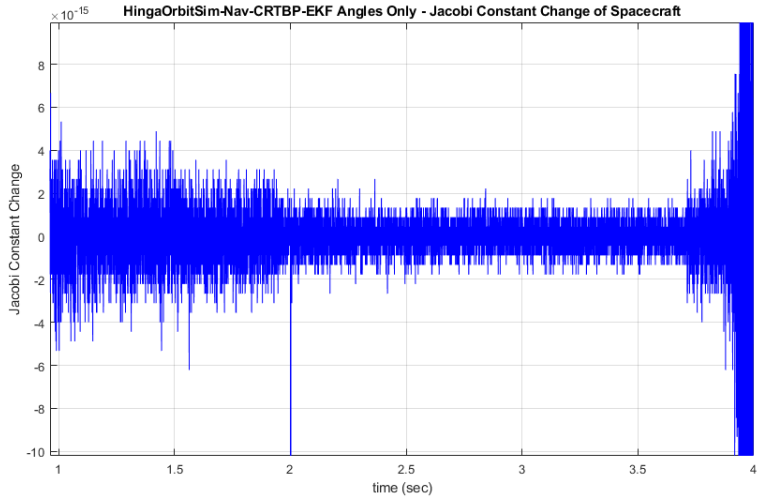

Figure 15. Change in Spacecraft Jacobi Constant as computed by the CRTBP EKF

Figures 16 and 17 illustrate the Jacobi constant that the simulation computes during its numerical propagation of the vehicle state. Notice that during any perturbation the Jacobi constant is expected to change and does so at t = 2 days when there is a perturbation of 100 m/sec applied in the simulation. Another change occurs when a delta velocity (-716.28 m/sec) is applied to enter Lunar orbit.

Figures 18 and 19 illustrate the Jacobi constant that the simulation computes during its numerical propagation of

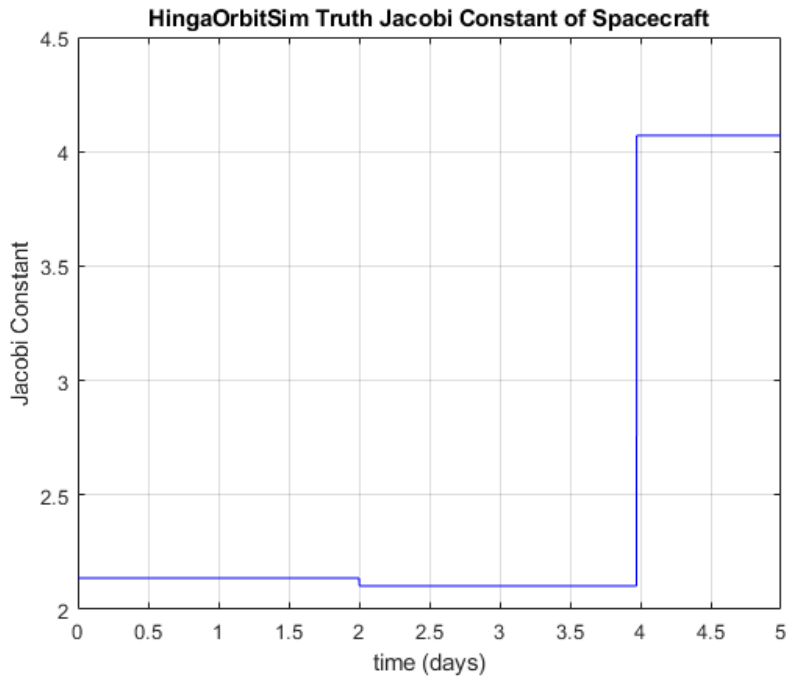

Figure 16. Spacecraft Jacobi Constant as computed by the Simulation

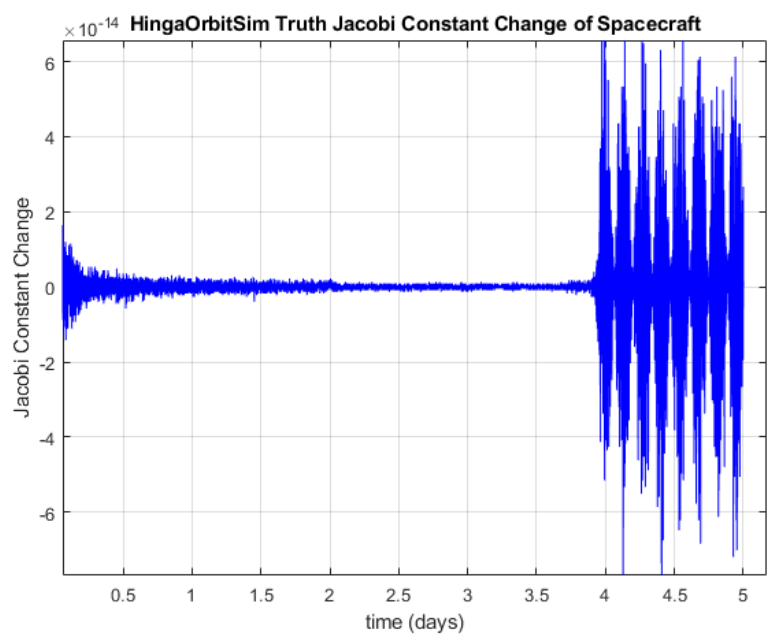

Figure 17. Change in Spacecraft Jacobi Constant as computed by the Simulation

Asteroid 2014 EC. The Jacobi constant is not expected to change, because no perturbations are applied intentionally in the simulation. However, we do see very small changes, but we can believe that they are caused by numerical truncation (during propagation approximations) and round off errors.

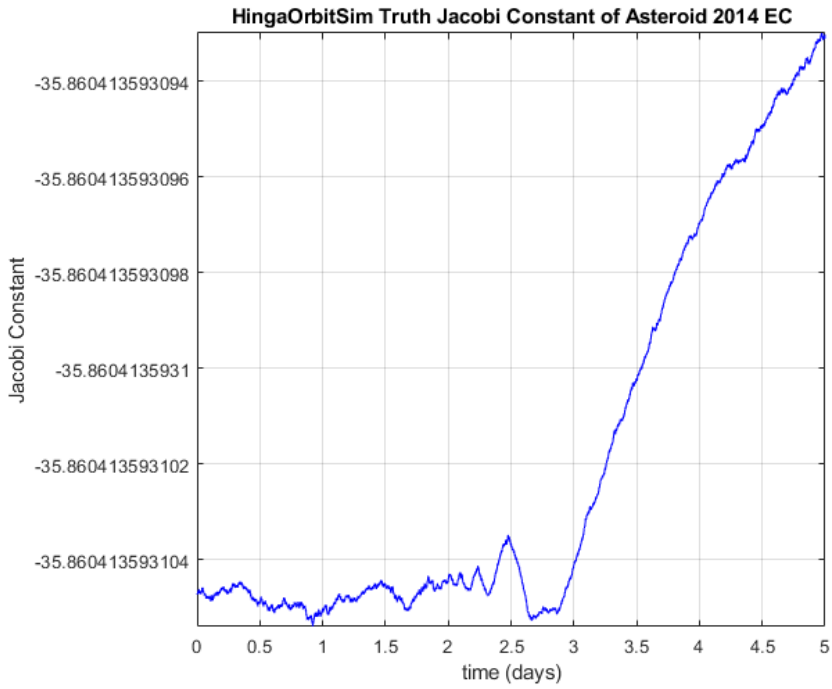

Figure 18. Asteroid 2014 EC Jacobi Constant as computed by the Simulation

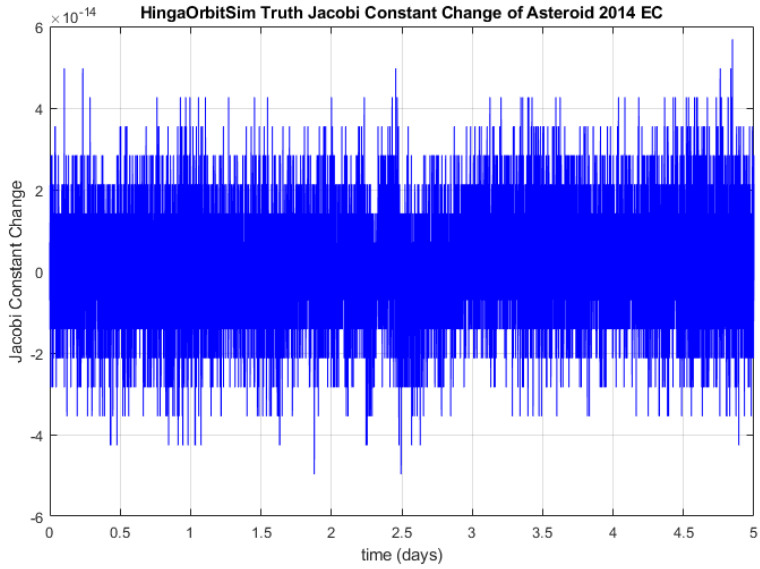

Figure 19. **Change in Asteroid 2014 EC Jacobi Constant as computed by the Simulation**

X. Performance of the Inertial EKF in the CRTBP Synodic Frame

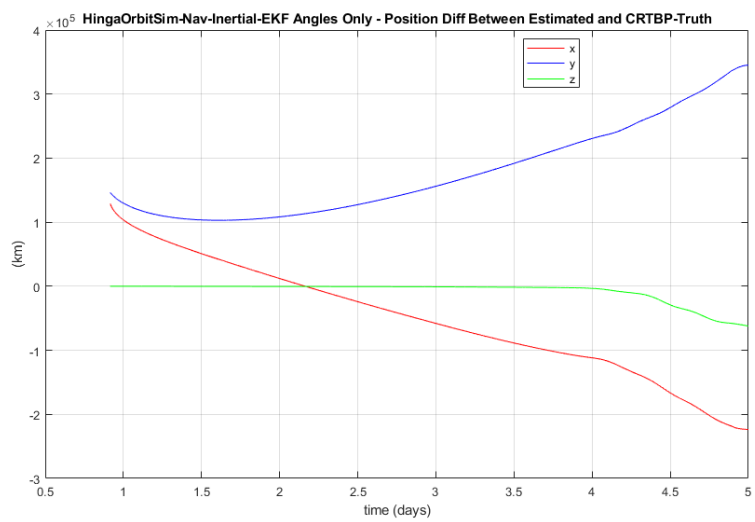

Figure 20. **Inertial EKF Position Performance in the Synodic Frame**

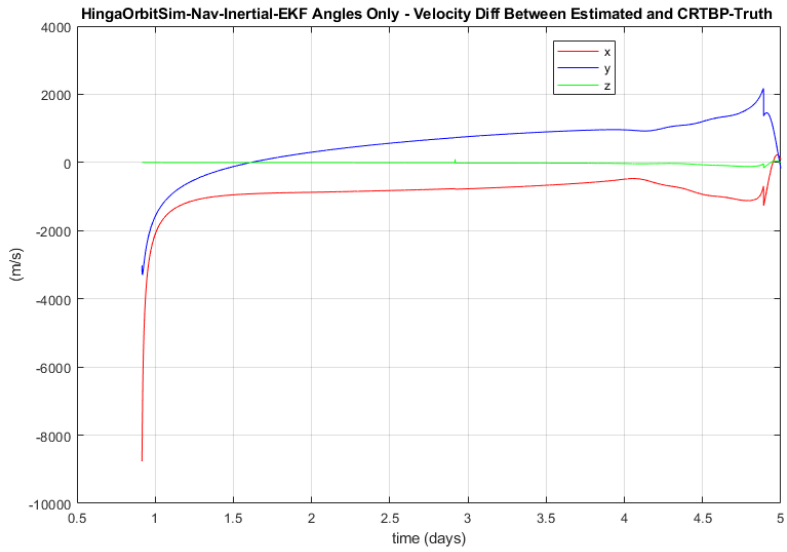

Figure 21. **Inertial EKF Velocity Performance in the Synodic Frame**

XI. Precise Orbit Determination

With the start of the year 1600, the evolution of orbit determination began in middle Europe with the fruitful cooperation between two very significant historical figures, Johannes Kepler and Tycho Brahe. These two men concerned themselves with solving the perplexing problem of the erratic behavior of planet Mars in its heliocentric trajectory. After Kepler abandoned his assumption of perfect circular motion, he was able to match Brahe's exquisite orbital observations to an elliptical path. Thus, Kepler was able to determine the true shape of Mars' orbit.

Much later, in 1795, Karl Friedrich Gauss invented the process of least squares, providing a firm computational basis of orbit prediction (Vallado[9]). Gauss' next remarkable achievement was to accurately predict the reappearance of the asteroid cluster Ceres from behind the Sun on New Year's Day in 1802 (Bate, Mueller, White[3]). The goal of orbit prediction and determination is to obtain accurate ephemeris (positions and velocities) of an orbiting satellite, using temporal sequences of observations. By integrating the equations of motion of a satellite from a reference epoch to the time of a true observation in relation to the model of how the satellite is observed, a *predicted* observation is produced. The difference between a predicted and a true observation is called a residual. Minimizing the residuals for all observations in a Gauss least squares sense, is an estimation process that determines the kinematic and dynamic parameters which describe the satellite's ephemeris and those which designate the participating models.

XI.A. Batch Filter Estimation - CRTBP - Phase Two

The equations of motion of a satellite in orbit are represented in vector form, as a system of linear first order ordinary differential equations, with time t as the independent variable,

$$\dot{X}(t) = F[X(t), t] \quad \text{and} \quad \dot{X}^*(t) = F[X^*(t)], \tag{57}$$

both for the true and nominal state, respectively, where

$$\begin{aligned}
X &= \begin{bmatrix} \vec{r} \\ \vec{v} \\ \vec{\alpha} \end{bmatrix} \text{(n x 1 vector)} \\
\vec{r} &= \text{satellite position (3 x 1 vector)} \\
\vec{v} &= \text{satellite velocity (3 x 1 vector)} \\
\vec{\alpha} &= \text{vector of model parameters} \\
F &= \text{derivatives of the state (n x 1 vector)}
\end{aligned}$$

The term "nominal" refers to the state of the satellite that is computed, using mathematical and physical models, and is given the "*" notation. The initial conditions are $X(t_0)$ and $X^*(t_0)$. Knowing that the true state $X(t)$ is the combination of the nominal and some deviation, $x(t)$

$$X(t) = X^*(t) + x(t), \tag{58}$$

where $x(t)$ is an (nx1) vector of deviations away from the computed nominal value of the state $X^*(t)$, a weak variation equation of state may be constructed. Rearranging terms in Equation 58, making one substitution, and taking the derivative with respect to time, we get

$$\begin{aligned} \dot{x}(t) &= \dot{X}(t) - \dot{X}^*(t) \\ &= F[X(t), t] - F[X^*(t), t] \\ &= F[X^*(t) + x(t), t] - F[X^*(t), t] \end{aligned}$$

Expanding this equation in a Taylor series about the nominal trajectory, and ignoring higher order terms, leads to,

$$\begin{aligned} \dot{x}(t) &= F[X^*(t), t] + \left.\frac{\partial F[X^*(t), t]}{\partial X^*(t)} x(t)\right|_{\text{eval. on nominal}} + \text{higher order terms} - F[X^*(t), t] \\ &= A(t)x(t), \end{aligned}$$

where

$$A(t) = \frac{\partial F[X^*(t), t]}{\partial X^*(t)} = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \cdots & \frac{\partial F_1}{\partial X_n} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \cdots & \frac{\partial F_2}{\partial X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial X_1} & \frac{\partial F_n}{\partial X_2} & \cdots & \frac{\partial F_n}{\partial X_n} \end{bmatrix} \quad \text{(nxn matrix)} \tag{59}$$

The linearized differential Equation 59 has the solution

$$x(t) = \Phi(t, t_0)x(t_0), \tag{60}$$

where $x(t_0)$ is the value of x(t) at epoch t_0 and $\Phi(t, t_0)$ is the state transition matrix, which relates a deviation to the state at some time t to the state at t_0 (Gelb[15]). The matrix $\Phi(t, t_0)$ satisfies the differential equation

$$\dot{\Phi}(t, t_0) = A(t)\Phi(t, t_0), \tag{61}$$

with the initial condition of $\Phi(t_0, t_0) = I$, the identity matrix. By numerically integrating Equation 61 (using either the inertial or the CRTBP equations of motion), the Equation 60 may be obtained. This deviation is then related to the satellite observation residuals through the linearized observation-state relation. With this relation, observation residuals are used to estimate corrections to the nominal state vector. By using the state transition matrix $\Phi(t, t_0)$ to map all observations to some common epoch $x(t_0)$, the corrections are applied at time $t = t_0$. If there are p actual observations taken at time t, they can be represented by the (p x 1) vector $Y(t)$. The observation-state equation is assumed to have the following form

$$Y(t) = G[X(t), t] + \epsilon(t) \tag{62}$$

where $G[X(t), t]$ is a (p x 1) vector representing the mathematical model of satellite observations. The (p x 1) vector ϵ represents the errors of commission and omission in the mathematical models of motion. Using Equation 58 and expanding Equation 62 in a Taylor series and dropping terms higher than first order, we obtain the following equation which relates an observation residual y(t) to x(t),

$$\begin{aligned} Y(t) &= G[X^*(t) + x(t), t] + \epsilon(t) \\ Y(t) &= G[X^*(t)] + \frac{\partial G[X^*(t), t]}{\partial X^*(t)} x(t) + \epsilon(t) \\ Y(t) - G[X^*(t)] &= \frac{\partial G[X^*(t), t]}{\partial X^*(t)} x(t) + \epsilon(t) \\ y(t) &= \tilde{H}(t)x(t) + \epsilon(t), \end{aligned} \tag{63}$$

where $\tilde{H}(t)$ is defined by ,

$$\tilde{H}(t) = \frac{\partial G[X^*(t), t]}{\partial X^*(t)} \tag{64}$$

and $\epsilon(t)$ now contains errors due to linearization in the observation and motion models. Inserting Equation 60 into Equation 63 we find that

$$y(t_i) = \tilde{H}(t_i)\Phi(t_i, t_0)x(t_0) + \epsilon(t_i). \tag{65}$$

If we let $H(t_i) = \tilde{H}(t_i)\Phi(t_i, t_0)$,we then have the following expression which maps an observation taken at time t_i, to the initial time t = t_0

$$y(t_i) = H(t_i)x(t_0) + \epsilon(t_i), \tag{66}$$

where y(t_i) and $\epsilon(t_i)$ are (p x 1) vectors, $x(t_0)$ is (n x 1) and $H(t_i)$ is a (p x n) matrix. If a set of observations, termed a "batch", is taken at times $[t_1, t_2, \cdots t_k]$, all can be represented by one equation in the following matrix form.

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} = \begin{bmatrix} H(t_1) \\ H(t_2) \\ \vdots \\ H(t_k) \end{bmatrix} x_0 + \begin{bmatrix} \epsilon(t_1) \\ \epsilon(t_2) \\ \vdots \\ \epsilon(t_k) \end{bmatrix} \tag{67}$$

Compactly written as

$$y = Hx_0 + \epsilon, \tag{68}$$

where y has the dimensions (kp x 1 = m x 1), H is (kp x n = m x n), x_0 is (n x 1) and ϵ is (kp x 1 = m x 1). (Note: H often has the name "information matrix") Usually the number of observations is much greater than the number of parameters to be solved (for), m $\gg$ n and the observations are assigned weights. Thus the solution for x_0 in Equation 68 can be obtained by using the weighted least squares estimation technique. This can be done by either forming the normal equation to solve for x_0 or by directly performing an orthogonal factorization on the information matrix H. Another approach is to employ the linear unbiased minimum variance estimate method.

XI.A.1. Forming the Normal Equation

The solution to Equation 68 is an estimated correction vector $\hat{x}$ that is added to the nominal state vector $X^*(t)$ at the initial epoch, t_0, namely

$$X^*(t_0) = X^*(t_0) + \hat{x}(t_0). \tag{69}$$

This correction vector can by obtained by minimizing the weighted sum of the square of the observation residuals as defined by the performance index J,

$$J = \epsilon^T W \epsilon. \tag{70}$$

Rearranging Equation 66, and dropping the indices i and 0 for simplicity, we find that at time t,

$$\epsilon(t) = y(t) - H(t)x(t), \tag{71}$$

leads to

$$J = [y(t) - H(t)x(t)]^T W [y(t) - H(t)x(t)], \tag{72}$$

where W is a diagonal matrix containing assigned observations weights, a topic which is discussed in a later section. Setting the first variation of Equation 72 equal to zero, results in the following normal equation of the linear system,

$$(H^T W H)\hat{x} = H^T W y. \tag{73}$$

Of course when inversion of the matrix $(H^T W H)$ is possible, either by direct or indirect means, the solution is written as

$$\hat{x} = (H^T W H)^{-1} H^T W y. \tag{74}$$

(The notation $\hat{x}$ implies that the correction vector which satisfies the first variation condition, occurs at an extremum.) The variance and covariance of the unbiased estimated correction vector $\hat{x}$ is given by (Tapley[19]) as

$$\begin{aligned} P &= E[(\hat{x} - E[\hat{x}])(\hat{x} - E[\hat{x}])^T] \\ &= E[(\hat{x} - x)(\hat{x} - x)^T] \\ &= (H^T W H)^{-1} H^T W E[\epsilon\epsilon^T] W H (H^T W H)^{-1}. \end{aligned} \tag{75}$$

When the weighting matrix W is chosen to equal the inverse of *observation* covariance, i.e. $W = \left\{E[\epsilon\epsilon^T]\right\}^{-1}$, Equation 75 reduces to the following simple form

$$P = (H^T W H)^{-1}. \tag{76}$$

To include a previous estimate $\overline{x}$, known as an *a priori* estimate and its corresponding error covariance $\overline{P}$ into a current estimation, after the proper mapping has been carried out, the performance index must first be redefined at time t as

$$J = [y(t) - H(t)x(t)]^T W[y(t) - H(t)x(t)] + [\overline{x}(t) - x(t)][\overline{P}(t)^{-1}][\overline{x}(t) - x(t)]. \tag{77}$$

Setting the first variation equation equal to zero, and dropping the index t for ease of presentation, gives the adjusted weighted least squares estimate, for the inclusion of *a priori* information, as

$$\hat{x} = [(H^T W H) + \overline{P}^{-1}]^{-1}[H^T W y + \overline{P}^{-1}\overline{x}], \tag{78}$$

Again, by letting $W = \left\{E[\epsilon\epsilon^T]\right\}^{-1}$, the resultant error covariance is

$$P = \left[(H^T W H) + \overline{P}^{-1}\right]^{-1}. \tag{79}$$

XI.A.2. Orthogonal Factorization

Another name for orthogonal factorization is QR decomposition. It is an important solution technique, because the disadvantage of the normal-equation approach is that it is sometimes less accurate than the QR approach. In fact, critical information can be lost when $H^T H$ is formed (Watkins,[20] Tapley[21]). A simple example taken from (Watkins[20]), illustrates an issue that can occur for any size problem. Let

$$H = \begin{bmatrix} 1 & 1 \\ \tau & 0 \\ 0 & \tau \end{bmatrix}, \tag{80}$$

where $\tau > 0$, is small. Clearly H has full rank, and

$$H^T H = \begin{bmatrix} 1+\tau^2 & 1 \\ 1 & 1+\tau^2 \end{bmatrix} \tag{81}$$

which is positive definite. However if τ is small enough such that τ^2 is below machine precision, i.e. numerically zero, then the computed $H^T H$ will be $\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$, which is singular. Of course using double precision arithmetic will, in many cases of the least square approach, be an adequate remedy. Despite this numerical property and because H is often sufficiently well conditioned, (and despite the fact that the condition number of $H^T H$ is the square of the condition number of matrix H) the normal equation approach is frequently used. For further discussion on the sensitivity of the least-squares problem, the text by Watkins[20] is suggested to the curious reader. However, by using the QR approach, the matrix $H^T H$ is not formed, rather the matrix H is used directly in the solution process. Let us introduce the definition of Q here. The concept of orthogonality between two arbitrary vectors $u, v \in \Re^3$, is commonly defined by the dot product $\langle u, v\rangle$ = 0, namely that the angle between them is $\frac{\pi}{2}$. Extending the orthogonality concept to matrices, a matrix $Q \in \Re^{nxn}$ is said to be orthogonal if $QQ^T = I$. This equation also says that Q has an inverse, and $Q^{-1} = Q^T$ (Watkins[20]). The following theorem and proof (Watkins[20]), justify why/how orthogonal factorization can be used as a solution technique.

Theorem XI.1. *If $Q \in \Re^{nxn}$ is orthogonal, then for all x,y $\in \Re^n$,*

$$\begin{aligned} (a) \quad \langle Qx, Qy\rangle &= \langle x, y\rangle \\ (b) \quad \|Qx\|_2 &= \|x\|_2 \end{aligned}$$

Proof.

$$\begin{aligned} (a) \quad \langle Qx, Qy\rangle &= (Qy)^T Qx = y^T Q^T Qx = y^T x = \langle x, y\rangle \\ (b) \quad \| \langle Qx, Qx\rangle \|_2 &= (Qx)^T Qx = x^T Q^T Qx = x^T x = \| \langle x, x\rangle \|_2 \end{aligned}$$

□

Because part (b) of the Theorem XI.1 says that Qx and x have the same length, *orthogonal transformations preserve lengths*. By combining parts (a) and (b), and using the Law of Cosines (which permits the computation of the cosine of an angle by the dot product of two vectors) it is clear that

$$\arccos\frac{\langle Qx, Qy\rangle}{\|Qx\|_2\|Qy\|_2} = \arccos\frac{\langle x, y\rangle}{\|x\|_2\|y\|_2}. \tag{82}$$

Because the angle between Qx and Qy is the same as the angle between x and y, *orthogonal transformations preserve angles*. Therefore the application of Q in the solution of a least squares problem is permitted because when A and b are replaced by QA and Qb, respectively, all lengths and angles are preserved. Rewriting the performance index seen in Equation 72 in the form of the Euclidean norm, to illustrate how Q is directly applied to H, and dropping the index t for simplicity,

$$\begin{aligned} J &= [y - Hx]^T W^{\frac{T}{2}} W^{\frac{1}{2}} [y - Hx] \\ &= \left\{W^{\frac{1}{2}}[y - Hx]\right\}^T \left\{W^{\frac{1}{2}}[y - Hx]\right\} = \|W^{\frac{1}{2}}[y - Hx]\|^2 \\ &= \|QW^{\frac{1}{2}}[y - Hx]\|^2. \end{aligned} \tag{83}$$

By choosing the orthogonal transformation matrix Q such that

$$QW^{\frac{1}{2}}H = \begin{bmatrix} R \\ 0 \end{bmatrix}, \tag{84}$$

where R is an upper triangular n x n matrix in the top portion of the m x n transformed H. Applying the same Q to the right hand side of $Hx = y$, where y is an (m x 1) vector, we get

$$QW^{\frac{1}{2}}y = \begin{bmatrix} b \\ e \end{bmatrix}, \tag{85}$$

where b and e are vectors of dimension n and (m-n), respectively. Because the transformation produces the upper triangular matrix R, this orthogonal factorization is commonly called the QR decomposition. The performance index is now written as

$$\begin{aligned} J &= \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} x \quad - \quad \begin{bmatrix} b \\ e \end{bmatrix} \right\|^2 \\ J &= \|Rx - b\|^2 + \|e\|^2. \end{aligned} \tag{86}$$

By inspection, the solution which minimizes J is $R\hat{x} = b$, which is easily obtained by simple backwards substitution, a less costly endeavor than the inversion of the normal matrix $H^T H$. Thus, J will equal the scalar $\|e\|^2$ and represents its minimum value. Since $\|e\|^2$ is an approximation of the root mean square (RMS) of the observation residuals for the estimated solution, it is also termed the linear predicted RMS. By letting $W = \left\{E[\epsilon\epsilon^T]\right\}^{-1}$, as was the case in the formation of the normal equation, and assuming that the errors in ϵ are random only, i.e. E[ϵ] = 0, the covariance matrix can be written as

$$P = (H^T W H)^{-1} = (R^T R)^{-1} \tag{87}$$

By using the matrix R produced by the QR decomposition for example, covariance computation can be performed "in place", meaning that no memory beyond that required to store the (n x n) matrix R is needed. Also, in cases where R is ill-conditioned, the Singular Value Decomposition (SVD) may be applied and used to generate stabilized solutions, by manipulating the singular values.

XI.A.3. Description of the H Matrix

The observation partials matrix H, often called the *sensitivity* or *information* matrix, relates how observations are affected by changes in state. As seen in Equation 65, H is the product of $\tilde{H}(t_i)$ and the state transition matrix $\Phi(t, t_0)$, where the former is derived from G, the observation model. A simple example of this model is that of the angle between line of sight unit vectors pointing from the observing spacecraft to the target Asteroid, shown in (green)

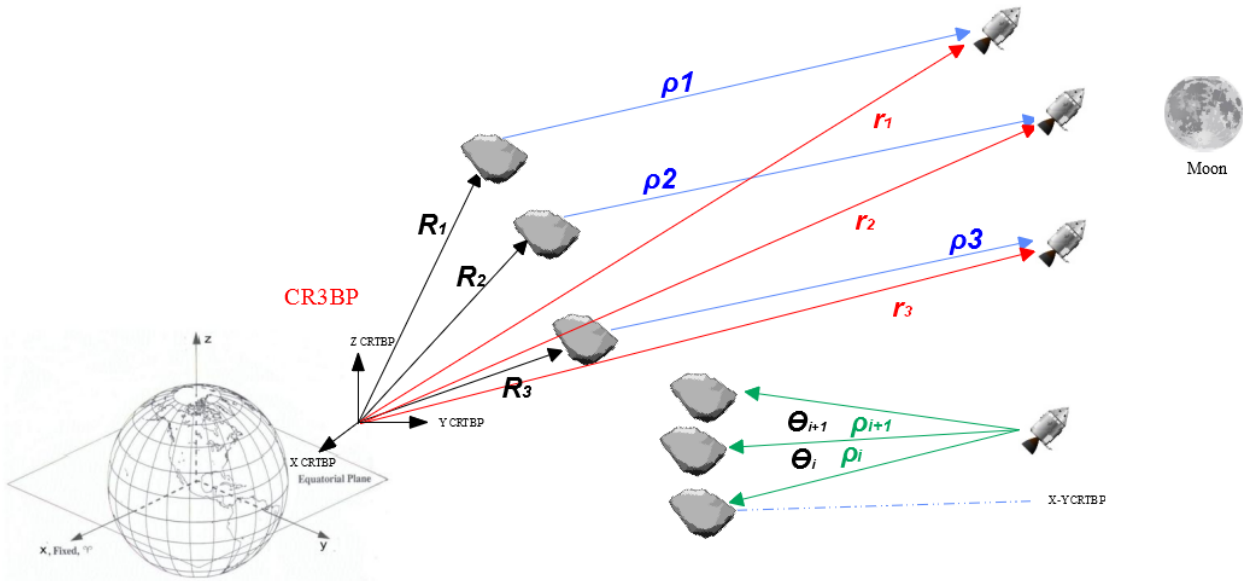

Figure 22. Observation Model for both the Batch and EKF filters.

Figure 22 where small caps "r" (red) equals the (unknown) synodic coordinates of the spacecraft. **R** (in black) is the known position vector of the Asteroid in the synodic coordinate frame (known from JPL DE421), and ρ (in blue) is the unknown relative range vector between the Asteroid and the Spacecraft. Note: the sense of the vectors in this diagram is very important. If the arrow points from the Spacecraft to the Asteroid this means that an actual line of sight measurement is being made. In the computation of the Spacecraft position, this unit vector is "flipped" and points from the Asteroid to the Spacecraft. This means that we are computing the position of the Spacecraft with respect to the Asteroid. This mathematically enables the estimation of the spacecraft position and velocity. (If we knew the position of the Spacecraft and were interested in computing the position and velocity of some unknown Asteroid, then the system of "arrows" would be rearranged as seen in the study by Hinga.[11]) Now, G = $G(X(t), t) = \|\overline{\rho}_k\|$, is the magnitude of the relative position vector between the Spacecraft and the kth "observing" Asteroid line of sight unit vector. Thus the line of sight unit vector G is expressed as,

$$\begin{aligned}
\mathbf{G}[\overline{\mathbf{X}}_k, k] &= \|\overline{\mathbf{r}}_k - \overline{\mathbf{R}}_k\| \\
&= (x_{\mathbf{T}}\hat{i} + y_{\mathbf{T}}\hat{j} + z_{\mathbf{T}}\hat{k}) - (x_{\mathbf{P}}\hat{i} + y_{\mathbf{P}}\hat{j} + z_{\mathbf{P}}\hat{k}) \\
&= (x_{\mathbf{T}} - x_{\mathbf{P}})\hat{i} + (y_{\mathbf{T}} - y_{\mathbf{P}})\hat{j} + (z_{\mathbf{T}} - z_{\mathbf{P}})\hat{k} \\
&= L_x\hat{i} + L_y\hat{j} + L_z\hat{k}.
\end{aligned} \tag{88}$$

and is used to evaluate the partials for row vector $\tilde{H}(t)$,

$$\tilde{H}(t) = \frac{\partial G[X^*(t)]}{\partial X^*(t)} = \left[\begin{array}{cccc} \frac{\partial G[X^*(t)]}{\partial X^*(t)_1} & \frac{\partial G[X^*(t)]}{\partial X^*(t)_2} & \cdots & \frac{\partial G[X^*(t)]}{\partial X^*(t)_n} \end{array} \right], \tag{89}$$

where n is the number of unknown parameters to be estimated. Multiplication by the state transition matrix is then carried out to map dynamic parameters to the required epoch, resulting in the matrix H. It can been seen here, and in Equation 65, that each row of H contains the relation between one observation and a change in state, mapped to the time of epoch. Therefore all rows of H relate how a change in the initial state (at epoch) affect all measurements in a given batch of observations. Of course more than one observation model may be included into a batch of observations which make up all of the rows of the sensitivity matrix H.

XII. Extended Kalman Filter (EKF) Formulation - Phase Three

The extended Kalman filter (EKF), a workhorse of real time spacecraft attitude estimation,[22] is a recursive non linear estimator (perturbed by Gaussian noise) that is descretized in the time domain by linearizing the physical dynamics

about the current best estimate of the parameters of interest. It is not a requirement that the model of the state dynamics $\dot{X} = F[X(t), t]$ and observation model $G[X(t), t]$ be linear, only that they are differentiable. This means that the formulation of the state transition matrix $\Phi(t, t_0)$ and the matrix $H[X(t), t]$, which is the partials matrix needed to compute the predicted measurement from the predicted state (discussed below), is defined. In a non-extended Kalman filter, the same is true, but the linearization occurs about some pre-computed nominal trajectory of the state. In the EKF, the current best estimate comes from an optimal combination of the state from the previous time step (that propagated to the current time) and the current measurement. The definition of this combination is determined by the so-called Kalman gain $\mathbf{K}$, which is defined below.

Kalman filters are unusual in that most filters (i.e. Butterworth filter) are formulated in the frequency domain, then transformed back into the time domain for application. The EKF can be considered to be an adaptive low-pass infinite impulse response (IIR) digital filter, meaning that its response to an impulsive input is non-zero for infinite time.[15] The frequency response of the EKF in this study is of no interest.

Ideally, if the model of the state and measurements are complete and accurate and perpetrate no acts of error omission or commission, then the covariance $\mathbf{P}(\mathbf{t})$ of the estimate state, will accurately reflect the confidence of the estimated state vector, and those parameters will have a mean error of zero. Stated a different way, the variance and covariance of the estimated state parameters will have a distribution about the true state. Invoking the expectation operator $E[f(\xi)] = \int_{-\infty}^{\infty} f(\xi) d\xi$, where $f(\xi)$ is the function of interest, if there are no biases in the estimate $\hat{\mathbf{X}}$ and residual $\tilde{\mathbf{y}} = (\mathbf{Y}_{k truth} - \mathbf{G}[\hat{\mathbf{X}}_k, k])$ then:

$$\begin{aligned} E[\mathbf{X}_{k truth} - \hat{\mathbf{X}}_k] &= 0 \\ E[\mathbf{Y}_{k truth} - \mathbf{G}[\hat{\mathbf{X}}_k, k]] &= 0, \end{aligned} \tag{90}$$

and the covariance matrices for the state estimate and residual, defined as,

$$\begin{aligned} \mathbf{P}_{k,k} &= E[(\hat{\mathbf{X}}_k - E[\hat{\mathbf{X}}_k])(\hat{\mathbf{X}}_k - E[\hat{\mathbf{X}}_k])^T] \\ \mathbf{S}_{k,k} &= E[(\tilde{\mathbf{y}} - E[\tilde{\mathbf{y}}])(\tilde{\mathbf{y}} - E[\tilde{\mathbf{y}}])^T] \end{aligned} \tag{91}$$

will have zero bias. However, since the filter of this study is intentionally mechanized as a suboptimal filter, small biases will be present and the Equations of 90 will be close to zero.

The state of the filter is represented by $\hat{\mathbf{X}}_k$, the estimated state at time k, and the error covariance matrix $\mathbf{P}_k$, which is a measure of the confidence in that state estimate. The EKF has two separate phases, which are called prediction and update. In the prediction phase, the estimate from the previous time step (k-1), both the state and covariance matrix, are propagated forward to the current time step (k). Then during the update phase, the state is refined with measurement information from the current time step. It is intended that after this refinement, the new estimate of the state is more accurate, i.e. closer to the truth. In this study (note: there is neither a control model nor a control input vector), the equations for these two phases are as follows (Kalman).[23]

Predict Phase

$$\hat{\mathbf{X}}_k = \mathbf{\Phi}_{k,k-1}\hat{\mathbf{X}}_{k-1} \quad \text{(predicted state estimate)} \tag{92}$$

$$\mathbf{P}_{k,k-1} = \mathbf{\Phi}_{k,k-1}\mathbf{P}_{k-1,k-1}{\mathbf{\Phi}_{k,k-1}}^T + \mathbf{Q}_k \quad \text{(predicted covariance of estimate)} \tag{93}$$

where $\mathbf{\Phi}_{k,k-1}$ and $\mathbf{Q}_k$ are the state transition and process noise matrices respectively. In this investigation a Runge-Kutta 4 integration scheme (and when needed, a variable step size integrator with an appropriate tolerance) is used to propagate both $\Phi_{k,k-1}$ and $\hat{\mathbf{X}}_{k-1}$ forward one time step to give $\overline{\mathbf{X}}_k$, using the Jacobian matrix (a matrix of partial derivatives) $\mathbf{A}$ defined in Equation XII.B in Section XII.C.

Update Phase

$$\tilde{\mathbf{y}}_k = \mathbf{Y}_k - \mathbf{G}[\overline{\mathbf{X}}_k, k] \quad \text{(form measurement residual or innovation)} \tag{94}$$

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k,k-1}{\mathbf{H}_k}^T + \mathbf{R}_k \quad \text{(covariance of residual or innovation)} \tag{95}$$

$$\mathbf{K}_k = \mathbf{P}_{k,k-1}{\mathbf{H}_k}^T[\mathbf{H}_k\mathbf{P}_{k,k-1}{\mathbf{H}_k}^T + \mathbf{R}_k]^{-1} \quad \text{(optimal Kalman gain)} \tag{96}$$

$$\hat{\mathbf{x}}_k = \mathbf{K}_k\mathbf{y}_k \quad \text{(optimal update for the state estimate)} \tag{97}$$

$$\hat{\mathbf{X}}_k = \overline{\mathbf{X}}_k + \hat{\mathbf{x}}_k \quad \text{(update the state estimate)} \tag{98}$$

$$\mathbf{P}_{k,k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k,k-1}(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{R}_k{\mathbf{K}_k}^T \quad \text{(update the estimate covariance)} \tag{99}$$

The terms $\mathbf{G}[\overline{\mathbf{X}}_k, k]$ and $\mathbf{H}_k$ are the measurement model and the partials of the measurement model with respect to the state, and are defined in Section XII.E. $\mathbf{Y}_k$ is the actual measurement taken by a camera to produce a line of sight unit vector and is discussed in Section II.B.

XII.A. F Vector - The Error State Equations

The state estimate of a navigation filter is an error state. That is, we are estimating the error committed by a navigator. We are not estimating the state of a target (vehicle) itself, directly, i.e. position, velocity, attitude, gyro bias, etc. We are estimating error in these parameters, then updating the parameters themselves with the error, to improve the knowledge of them. In other words, this error state is used to update the knowledge of the state vector.

This error state is represented by $\delta\mathbf{X}_k$, at time *k*, and the error state covariance matrix $\mathbf{P}_k$, which is a measure of the confidence in that error state estimate. By applying the estimate of the state error to the target vehicle state itself, the knowledge (indicated by an IMU, for example) of the state is improved and reconstructed at every time step of the trajectory.

Let's begin with the expression of the "true" target vehicle state followed by that for the error state. Dropping the subscript *k* for clarity we have

$$\begin{aligned} \mathbf{X}(\mathbf{t})_{\text{TRUE}} &= \mathbf{X}(\mathbf{t})_{\text{INS}} + \delta\mathbf{X}(\mathbf{t})_{\text{ERROR}} \\ \delta\mathbf{X}(\mathbf{t})_{\text{ERROR}} &= \mathbf{X}(\mathbf{t})_{\text{TRUE}} - \mathbf{X}(\mathbf{t})_{\text{INS}} \end{aligned} \tag{100}$$

By differentiating this equation with respect to time, we obtain the differential equation for the error state,

$$\delta\dot{\mathbf{X}}(\mathbf{t})_{\text{ERROR}} = \dot{\mathbf{X}}(\mathbf{t})_{\text{TRUE}} - \dot{\mathbf{X}}(\mathbf{t})_{\text{INS}} \tag{101}$$

Because we do not know what truth is, we approximate it with the Taylor series expansion

$$f(y+\delta y) = f(y) + f'(y)\delta y + \frac{1}{2!}f''(y)\delta y^2 + \text{H.O.T.} \tag{102}$$

and ignore second order terms and higher. Thus,

$$\begin{aligned} f(\dot{\mathbf{X}}(\mathbf{t})_{\text{TRUE}}) &= f(\dot{\mathbf{X}}(\mathbf{t})_{\text{INS}} + \delta\dot{\mathbf{X}}(\mathbf{t})_{\text{ERROR}}) \\ \dot{\mathbf{X}}(\mathbf{t})_{\text{INS}} + \delta\dot{\mathbf{X}}(\mathbf{t})_{\text{ERROR}} &= \dot{\mathbf{X}}(\mathbf{t})_{\text{TRUE}} \\ &= \dot{\mathbf{X}}(\mathbf{t})_{\text{INS}} + \frac{\partial\dot{\mathbf{X}}(\mathbf{t})_{\text{INS}}}{\partial\mathbf{X}(\mathbf{t})}\partial\mathbf{X}(\mathbf{t}) \end{aligned} \tag{103}$$

If, for example, we were to include a measurement state U, of an inertial navigation system (INS), defined as

$$\mathbf{U}(\mathbf{t})_{\text{TRUE}} = \mathbf{U}(\mathbf{t})_{\text{INS}} + \delta\mathbf{U}(\mathbf{t})_{\text{ERROR}}, \tag{104}$$

where,

$$\mathbf{U}(\mathbf{t})_{\text{INS}} = \text{funct}(w_{\text{gyro}}, f_{\text{accel}}), \tag{105}$$

as a costate into the error state equation we re-define error state equation as

$$\delta\dot{\mathbf{X}}(\mathbf{t})_{\text{ERROR}} = \frac{\partial\dot{\mathbf{X}}(\mathbf{t})_{\text{INS}}}{\partial\mathbf{X}(\mathbf{t})}\partial\mathbf{X}(\mathbf{t}) + \frac{\partial\dot{\mathbf{X}}(\mathbf{t})_{\text{INS}}}{\partial\mathbf{U}(\mathbf{t})}\partial\mathbf{U}(\mathbf{t}) + \text{H.O.T.}, \tag{106}$$

(notice that the INS term $\dot{\mathbf{X}}(\mathbf{t})_{\text{INS}}$ cancels out as it appears on both sides of Equation 103).

Therefore, the INS "state" now includes measurement vector $\mathbf{U}$. The time rate of change of this state is

$$\dot{\mathbf{X}}(\mathbf{t})_{\text{INS}} = \dot{\mathbf{X}}(\mathbf{t})_{\text{INS}}(X, U, t) = \mathbf{F}(X, U, t). \tag{107}$$

This is the "state" that refers to the state of the vehicle (position, velocity, attitude) and the state of the sensors, that is the biases, scale factors and misalignments of both the accelerometers and gyros. This is not the (estimated) state vector of the EKF. The state vector of the EKF is an "error state" that is based upon the partial of vehicle dynamics

and sensor dynamics in Equation 107 with respect to the elements of the INS state and sensor states. The time rate of change of the EKF state vector is the following.

$$\begin{bmatrix} \delta\dot{\mathbf{X}}(\mathbf{t})_{\text{ERROR}} \\ \delta\dot{\mathbf{U}}(\mathbf{t})_{\text{ERROR}} \end{bmatrix} = \frac{\partial \mathbf{F}(X,U,t)_{\text{INS}}}{\partial \mathbf{X}(\mathbf{t})} \begin{bmatrix} \delta\mathbf{X}(\mathbf{t}) \\ \delta\mathbf{U}(\mathbf{t}) \end{bmatrix}. \tag{108}$$

Clearly Equation 107 requires defining the vehicle state vector and state dynamics of an INS before we can take the partial of **F** with respect to every element included in that "state", to eventually obtain the "error state" equations. Equation 112 illustrates an INS EKF State vector which includes the error in accelerometer and gyro measurements:

$$\mathbf{X}(t)_{EKF} = \begin{bmatrix} \delta\mathbf{X}(\mathbf{t})_{\text{ERROR}} \\ \delta\mathbf{U}(\mathbf{t})_{\text{ERROR}} \end{bmatrix} = \begin{bmatrix} \partial\mathbf{r}^e \\ \partial\mathbf{v}^e \\ \partial\boldsymbol{\omega}_b^e \\ \partial\mathbf{f}_B^b \\ \partial\mathbf{f}_{SF}^b \\ \partial\mathbf{f}_{MA}^b \\ \partial\mathbf{G}_B^b \\ \partial\mathbf{G}_{SF}^b \\ \partial\mathbf{G}_{MA}^b \end{bmatrix} \tag{109}$$

Therefore $\mathbf{F}(t)$ is defined as (in the following equations let's denote a vector with boldface, i.e. $\mathbf{v}^{\mathbf{e}}$.)

$$\mathbf{F}(X,U,t) = \begin{bmatrix} \dot{\mathbf{r}^e} \\ \dot{\mathbf{v}^e} \\ \dot{\boldsymbol{\phi}_b^e} \\ \dot{\mathbf{f}}_B^b \\ \dot{\mathbf{f}}_{SF}^b \\ \dot{\mathbf{f}}_{MA}^b \\ \dot{\mathbf{G}}_B^b \\ \dot{\mathbf{G}}_{SF}^b \\ \dot{\mathbf{G}}_{MA}^b \end{bmatrix} = \begin{bmatrix} \mathbf{v}^{\mathbf{e}} \\ C_b^e\mathbf{f}^{\mathbf{b}} + \mathbf{g}^{\mathbf{e}}(\mathbf{r}^{\mathbf{e}}) - 2(\boldsymbol{\omega}^{\mathbf{e}} \times \mathbf{v}^{\mathbf{e}}) - \boldsymbol{\omega}^{\mathbf{e}} \times \boldsymbol{\omega}^{\mathbf{e}} \times \mathbf{r}^{\mathbf{e}} \\ \boldsymbol{\omega}^{\mathbf{b}'} + \frac{1}{2}(\boldsymbol{\phi} \times \boldsymbol{\omega}^{\mathbf{b}'}) + \times\frac{1}{12}\boldsymbol{\phi} \times (\boldsymbol{\phi} \times \boldsymbol{\omega}^{\mathbf{b}'}) \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{110}$$

We are not estimating INS error parameters in this investigation, so there is no INS state measurement vector. However we are simulating the acceleration sensed (in the form of a "specific force") in by the Spacecraft due to δV perturbations and guidance maneuvers, so the above equations are simplified down to the following:

$$\begin{bmatrix} \delta\dot{\mathbf{X}}(\mathbf{t})_{\text{ERROR}} \end{bmatrix} = \frac{\partial \mathbf{F}(X,t)}{\partial \mathbf{X}(\mathbf{t})} \begin{bmatrix} \delta\mathbf{X}(\mathbf{t}) \end{bmatrix}. \tag{111}$$

$$\mathbf{X}(t)_{EKF} = \begin{bmatrix} \delta\mathbf{X}(\mathbf{t})_{\text{ERROR}} \end{bmatrix} = \begin{bmatrix} \partial\mathbf{r}^e \\ \partial\mathbf{v}^e \end{bmatrix} \tag{112}$$

$$\mathbf{F}(X,t) = \begin{bmatrix} \dot{\mathbf{r}^i} \\ \dot{\mathbf{v}^i} \end{bmatrix} = \begin{bmatrix} \mathbf{v}^{\mathbf{i}} \\ \mathbf{g}^{\mathbf{i}}(\mathbf{r}^{\mathbf{i}}) + C_b^i\mathbf{f}^{\mathbf{b}} \end{bmatrix}, \tag{113}$$

where $C_b^e\mathbf{f}^{\mathbf{b}}$ is the acceleration sensed by the INS in the sensor frame transformed into the navigation frame. The superscript "i" represents the inertial frame. Note: this same equation is written also in the same form for the CRTBP synodic "c" frame, where $\mathbf{g}^{\mathbf{c}}(\mathbf{r}^{\mathbf{c}})$ now represents the corrsponding combined gravitational, centripetal and Coriolis potentials, namely

$$\mathbf{F}(X,t) = \begin{bmatrix} \dot{\mathbf{r}^c} \\ \dot{\mathbf{v}^c} \end{bmatrix} = \begin{bmatrix} \mathbf{v}^{\mathbf{c}} \\ \mathbf{g}^{\mathbf{c}}(\mathbf{r}^{\mathbf{c}}) + C_b^c\mathbf{f}^{\mathbf{b}} \end{bmatrix}, \tag{114}$$

XII.B. The A Matrix - The Jacobian of the System Dynamics - Inertial Frame

The A matrix, which is the Jacobian $\frac{\partial \mathbf{F}(X,t)}{\partial \mathbf{X(t)}}$ from Equation 111, is a 2x2 matrix, where each element is 3x3. Thus we display a 6x6 matrix where each element A_{ij} is 3x3. The letter e refers to the Earth Centered Earth Fixed (ECEF) Frame, which will be reduced to the Inertial Frame, Earth Centered Inertial (ECI) frame.

$$A(t) = \frac{\partial \mathbf{F}(X,t)}{\partial \mathbf{X(t)}} = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \cdots & \frac{\partial F_1}{\partial X_n} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \cdots & \frac{\partial F_2}{\partial X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial X_1} & \frac{\partial F_n}{\partial X_2} & \cdots & \frac{\partial F_n}{\partial X_n} \end{bmatrix} \quad (nxn \text{ matrix}) \tag{115}$$

$$A(t) = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{116}$$

Let us define each 3x3 element.

$$A_{11} = \frac{\delta \dot{\mathbf{r}}^e}{\delta \mathbf{r}^e} = \begin{bmatrix} \frac{\delta \dot{\mathbf{r}}_x^e}{\delta \mathbf{r}_x^e} & \frac{\delta \dot{\mathbf{r}}_x^e}{\delta \mathbf{r}_y^e} & \frac{\delta \dot{\mathbf{r}}_x^e}{\delta \mathbf{r}_z^e} \\ \frac{\delta \dot{\mathbf{r}}_y^e}{\delta \mathbf{r}_x^e} & \frac{\delta \dot{\mathbf{r}}_y^e}{\delta \mathbf{r}_y^e} & \frac{\delta \dot{\mathbf{r}}_y^e}{\delta \mathbf{r}_z^e} \\ \frac{\delta \dot{\mathbf{r}}_z^e}{\delta \mathbf{r}_x^e} & \frac{\delta \dot{\mathbf{r}}_z^e}{\delta \mathbf{r}_y^e} & \frac{\delta \dot{\mathbf{r}}_z^e}{\delta \mathbf{r}_z^e} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{117}$$

$$A_{12} = \frac{\delta \dot{\mathbf{r}}^e}{\delta \mathbf{v}^e} = \begin{bmatrix} \frac{\delta \dot{\mathbf{r}}_x^e}{\delta \mathbf{v}_x^e} & \frac{\delta \dot{\mathbf{r}}_x^e}{\delta \mathbf{v}_y^e} & \frac{\delta \dot{\mathbf{r}}_x^e}{\delta \mathbf{v}_z^e} \\ \frac{\delta \dot{\mathbf{r}}_y^e}{\delta \mathbf{v}_x^e} & \frac{\delta \dot{\mathbf{r}}_y^e}{\delta \mathbf{v}_y^e} & \frac{\delta \dot{\mathbf{r}}_y^e}{\delta \mathbf{v}_z^e} \\ \frac{\delta \dot{\mathbf{r}}_z^e}{\delta \mathbf{v}_x^e} & \frac{\delta \dot{\mathbf{r}}_z^e}{\delta \mathbf{v}_y^e} & \frac{\delta \dot{\mathbf{r}}_z^e}{\delta \mathbf{v}_z^e} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{118}$$

$$A_{21} = \frac{\delta \dot{\mathbf{v}}^e}{\delta \mathbf{r}^e} = \frac{\delta \mathbf{g}(\mathbf{r}^\mathrm{e}) - (\boldsymbol{\omega}^\mathrm{e} \times \boldsymbol{\omega}^\mathrm{e} \times \mathbf{r}^\mathrm{e})}{\delta \mathbf{r}^\mathrm{e}} \tag{119}$$

$$\frac{\delta \mathbf{g}(r)}{\delta \mathbf{r}^\mathrm{e}} = \begin{bmatrix} \frac{\delta g_x}{\delta x} & \frac{\delta g_x}{\delta y} & \frac{\delta g_x}{\delta z} \\ \frac{\delta g_y}{\delta x} & \frac{\delta g_y}{\delta y} & \frac{\delta g_y}{\delta z} \\ \frac{\delta g_z}{\delta x} & \frac{\delta g_z}{\delta y} & \frac{\delta g_z}{\delta z} \end{bmatrix} \tag{120}$$

$$\frac{\delta(\boldsymbol{\omega}^\mathrm{e} \times \boldsymbol{\omega}^\mathrm{e} \times \mathbf{r}^\mathrm{e})}{\delta \mathbf{r}^\mathrm{e}} = \begin{bmatrix} -\boldsymbol{\omega}_\mathrm{e}{}^2 & 0 & 0 \\ 0 & -\boldsymbol{\omega}_\mathrm{e}{}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{121}$$

Thus,

$$A_{21} = \begin{bmatrix} (\frac{\delta g_x}{\delta x} + \boldsymbol{\omega}_\mathrm{e}{}^2) & \frac{\delta g_x}{\delta y} & \frac{\delta g_x}{\delta z} \\ \frac{\delta g_y}{\delta x} & (\frac{\delta g_y}{\delta y} + \boldsymbol{\omega}_\mathrm{e}{}^2) & \frac{\delta g_y}{\delta z} \\ \frac{\delta g_z}{\delta x} & \frac{\delta g_z}{\delta y} & \frac{\delta g_z}{\delta z} \end{bmatrix}. \tag{122}$$

$$A_{22} = \frac{\delta \dot{\mathbf{v}}^e}{\delta \mathbf{v}^e} = \frac{\delta(-2\boldsymbol{\omega}^\mathrm{e} \times \mathbf{v}^\mathrm{e})}{\delta v} = \begin{bmatrix} 0 & 2\boldsymbol{\omega}_\mathrm{e} & 0 \\ -2\boldsymbol{\omega}_\mathrm{e} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{123}$$

All of these above partials in this section are defined in Section XII.D. For the inertial reference frame, Earth Centered Inertial (ECI), the above expression is simplified. That is, $\boldsymbol{\omega}_\mathrm{e}{}^2 = 0$ which leads to $A_{22} = 0$.

XII.C. The A Matrix - The Jacobian of the System Dynamics - Circular Restricted Three Body Problem (CRTBP) Frame

The A matrix, which is the Jacobian $\frac{\partial \mathbf{F}(X,t)}{\partial \mathbf{X(t)}}$ from Equation 111, is a 2x2 matrix, where each element is 3x3. Thus we display a 6x6 matrix where each element A_{ij} is 3x3. The letter c refers to the CRTBP Frame.

$$A(t) = \frac{\partial \mathbf{F}(X,t)}{\partial \mathbf{X(t)}} = \begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \cdots & \frac{\partial F_1}{\partial X_n} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \cdots & \frac{\partial F_2}{\partial X_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial X_1} & \frac{\partial F_n}{\partial X_2} & \cdots & \frac{\partial F_n}{\partial X_n} \end{bmatrix} \quad (nxn \text{ matrix}) \tag{124}$$

$$A(t) = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{125}$$

Let us define each 3x3 element.

$$A_{11} = \frac{\delta \dot{\mathbf{r}}^c}{\delta \mathbf{r}^c} = \begin{bmatrix} \frac{\delta \dot{\mathbf{r}}_x^c}{\delta \mathbf{r}_x^c} & \frac{\delta \dot{\mathbf{r}}_x^c}{\delta \mathbf{r}_y^c} & \frac{\delta \dot{\mathbf{r}}_x^c}{\delta \mathbf{r}_z^c} \\ \frac{\delta \dot{\mathbf{r}}_y^c}{\delta \mathbf{r}_x^c} & \frac{\delta \dot{\mathbf{r}}_y^c}{\delta \mathbf{r}_y^c} & \frac{\delta \dot{\mathbf{r}}_y^c}{\delta \mathbf{r}_z^c} \\ \frac{\delta \dot{\mathbf{r}}_z^c}{\delta \mathbf{r}_x^c} & \frac{\delta \dot{\mathbf{r}}_z^c}{\delta \mathbf{r}_y^c} & \frac{\delta \dot{\mathbf{r}}_z^c}{\delta \mathbf{r}_z^c} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{126}$$

$$A_{12} = \frac{\delta \dot{\mathbf{r}}^c}{\delta \mathbf{v}^c} = \begin{bmatrix} \frac{\delta \dot{\mathbf{r}}_x^c}{\delta \mathbf{v}_x^c} & \frac{\delta \dot{\mathbf{r}}_x^c}{\delta \mathbf{v}_y^c} & \frac{\delta \dot{\mathbf{r}}_x^c}{\delta \mathbf{v}_z^c} \\ \frac{\delta \dot{\mathbf{r}}_y^c}{\delta \mathbf{v}_x^c} & \frac{\delta \dot{\mathbf{r}}_y^c}{\delta \mathbf{v}_y^c} & \frac{\delta \dot{\mathbf{r}}_y^c}{\delta \mathbf{v}_z^c} \\ \frac{\delta \dot{\mathbf{r}}_z^c}{\delta \mathbf{v}_x^c} & \frac{\delta \dot{\mathbf{r}}_z^c}{\delta \mathbf{v}_y^c} & \frac{\delta \dot{\mathbf{r}}_z^c}{\delta \mathbf{v}_z^c} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{127}$$

$$A_{21} = \frac{\delta \dot{\mathbf{v}}^c}{\delta \mathbf{r}^c} = \frac{\delta \mathbf{g}(\mathbf{r^c}) + (\boldsymbol{\omega}^\mathbf{c} \times \boldsymbol{\omega}^\mathbf{c} \times \mathbf{r^c})}{\delta \mathbf{r^c}} \tag{128}$$

Thus,

$$A_{21} = \begin{bmatrix} \frac{\delta g_x}{\delta x} & \frac{\delta g_x}{\delta y} & \frac{\delta g_x}{\delta z} \\ \frac{\delta g_y}{\delta x} & \frac{\delta g_y}{\delta y} & \frac{\delta g_y}{\delta z} \\ \frac{\delta g_z}{\delta x} & \frac{\delta g_z}{\delta y} & \frac{\delta g_z}{\delta z} \end{bmatrix}. \tag{129}$$

$$\frac{\delta g_x}{\delta x} = 1 + \frac{(1-\mu^*)}{r_1^3}\left[\frac{3(x-\mu^*)^2}{r_1^2} - 1\right] + \frac{\mu^*}{r_2^3}\left[\frac{3(x+1-\mu^*)^2}{r_2^2} - 1\right] \tag{130}$$

$$\frac{\delta g_x}{\delta y} = 3y\left[\frac{(1-\mu^*)(x-\mu^*)}{r_1^5} + \frac{\mu^*(x+1-\mu^*)}{r_2^5}\right] \tag{131}$$

$$\frac{\delta g_x}{\delta z} = 3z\left[\frac{(1-\mu^*)(x-\mu^*)}{r_1^5} + \frac{\mu^*(x+1-\mu^*)}{r_2^5}\right] \tag{132}$$

$$\frac{\delta g_y}{\delta x} = 3y\left[\frac{(1-\mu^*)(x-\mu^*)}{r_1^5} + \frac{\mu^*(x+1-\mu^*)}{r_2^5}\right] \tag{133}$$

$$\frac{\delta g_y}{\delta y} = 1 + \frac{(1-\mu^*)}{r_1^3}\left[\frac{3y^2}{r_1^2} - 1\right] + \frac{\mu^*}{r_2^3}\left[\frac{3y^2}{r_2^2} - 1\right] \tag{134}$$

$$\frac{\delta g_y}{\delta z} = 3yz\left[\frac{(1-\mu^*)}{r_1^5} + \frac{\mu^*}{r_2^5}\right] \tag{135}$$

$$\frac{\delta g_z}{\delta x} = 3z\left[\frac{(1-\mu^*)(x-\mu^*)}{r_1^5} + \frac{\mu^*(x-1+\mu^*)}{r_2^5}\right] \tag{136}$$

$$\frac{\delta g_z}{\delta y} = 3yz\left[\frac{(1-\mu^*)}{r_1^5} + \frac{\mu^*}{r_2^5}\right] \tag{137}$$

$$\frac{\delta g_z}{\delta z} = \frac{(1-\mu^*)}{r_1^3}\left[\frac{3z^2}{r_1^2} - 1\right] + \frac{\mu^*}{r_2^3}\left[\frac{3z^2}{r_2^2} - 1\right] \tag{138}$$

$$A_{22} = \frac{\delta \dot{\mathbf{v}}^c}{\delta \mathbf{v}^c} = \frac{\delta(-2\boldsymbol{\omega}^{\mathbf{c}} \times \mathbf{v}^{\mathbf{c}})}{\delta v} = \begin{bmatrix} \frac{\partial \ddot{x}}{\partial \dot{x}} & \frac{\partial \ddot{x}}{\partial \dot{y}} & \frac{\partial \ddot{x}}{\partial \dot{z}} \\ \frac{\partial \ddot{y}}{\partial \dot{x}} & \frac{\partial \ddot{y}}{\partial \dot{y}} & \frac{\partial \ddot{y}}{\partial \dot{z}} \\ \frac{\partial \ddot{z}}{\partial \dot{x}} & \frac{\partial \ddot{z}}{\partial \dot{y}} & \frac{\partial \ddot{z}}{\partial \dot{z}} \end{bmatrix} = \begin{bmatrix} 0 & 2\boldsymbol{\omega}_{\mathbf{c}} & 0 \\ -2\boldsymbol{\omega}_{\mathbf{c}} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{139}$$

where, $\boldsymbol{\omega}_{\mathbf{c}} = 1$.

XII.D. Earth J2 Gravity Field Model - Inertial Frame

In this investigation, a gravitation field is modelled using the spherical harmonic expansion of Equation 140. This expression describes a three dimensional gravitational potential, U, in the free space (zero density) above the Earth (Tapley, Born, Schutz[21]),

$$\begin{aligned} U &= \frac{GM}{r} + U' \\ U' &= -\frac{GM^*}{r}\sum_{l=1}^{\infty}\left(\frac{a_e}{r}\right)^l P_l(\sin\phi)J_l \\ &+ \frac{GM^*}{r}\sum_{l=1}^{\infty}\sum_{m=1}^{l}\left(\frac{a_e}{r}\right)^l P_{l,m}(\sin\phi)[C_{l,m}\cos m\lambda + S_{l,m}\sin m\lambda], \end{aligned} \tag{140}$$

where mass distribution is expressed in the spherical coordinates (r, ϕ, λ), with ϕ and λ representing geocentric latitude and longitude, respectively. The scale factors $M^\star$ and reference distance a_e nondimensionalize the mass property coefficients $C_{l,m}$ and $S_{l,m}$. The term $P_{l,m}$ is Legendre's Associated Function of degree ℓ and order m. If only the J term is considered with a degree of 2, we call this a J_2 gravitation field. This is of high enough fidelity for the purposes of the first half of a cis-Lunar mission to the Moon. For the Earth, 95 percent of the non-spherical mass distribution can be accounted for by the inclusion of only the J_2 term. Not including higher order terms in the gravitation potential causes about 100 meters of position error over the time length of the flight.

$$U = \frac{GM}{r} + \frac{GM^*}{r}\sum_{l=1}^{\infty}\left(\frac{a_e}{r}\right)^l P_2(\sin\phi)J_2. \tag{141}$$

Expanding the double summation in this equation with $m = 2$ and using it to derive the partials seen in Equations below (along with the observation-state partials of Equation 159), results in a system of equations of 3 unknowns in the representation of the dynamic equations for velocity, i.e. acceleration.

By letting $M = M^*$ (Earth mass), $sin\phi = \frac{z}{r}$, $r = (x_2 + y_2 + z_2)^{\frac{1}{2}}$ and $\mu = GM$, we can express $P_2(\sin\phi) = \frac{3}{2}sin^2(\phi) - \frac{1}{2}$, and simplifying, we obtain the gravitation potential in Cartesian coordinates as

$$U = \frac{GM}{r} - \frac{GM{a_e}^2 J_2}{2}\left[\frac{3z^2}{r^2} - \frac{1}{r^3}\right] \tag{142}$$

The gravitation gradient is obtained by applying the gradient operator ∇ on the potential U to compute the acceleration due to gravitation in the three cardinal directions in the navigation frame, namely

$$\overrightarrow{\mathbf{g}} = \overrightarrow{\nabla}\mathrm{U} = \frac{\delta U}{\delta x}\overrightarrow{i} + \frac{\delta U}{\delta y}\overrightarrow{j} + \frac{\delta U}{\delta z}\overrightarrow{k}. \tag{143}$$

Thus $g_x = \frac{\delta U}{\delta x}$, $g_y = \frac{\delta U}{\delta y}$, and $g_z = \frac{\delta U}{\delta z}$.

After several steps of derivation we obtain the following expression for the acceleration in a J_2 gravitation field in each of the three cardinal directions as

$$g_x = -\frac{\mu x}{r^3} - \frac{3\mu {a_e}^2 J_2 x}{2r^5}\left[1 - \frac{5z^2}{r^2}\right] \tag{144}$$

$$g_y = -\frac{\mu y}{r^3} - \frac{3\mu {a_e}^2 J_2 y}{2r^5}\left[1 - \frac{5z^2}{r^2}\right] \tag{145}$$

$$g_z = -\frac{\mu z}{r^3} - \frac{3\mu {a_e}^2 J_2 z}{2r^5}\left[3 - \frac{5z^2}{r^2}\right] \tag{146}$$

For purposes for determining the Jacobian partials needed to compute the time rate of change of the velocity due to gravitation effects we need to compute the partials of each of the above terms with respect to, again, all three cardinal directions in the navigation frame, namely,

$$\begin{bmatrix} \frac{\delta g_x}{\delta x} & \frac{\delta g_x}{\delta y} & \frac{\delta g_x}{\delta z} \\ \frac{\delta g_y}{\delta x} & \frac{\delta g_y}{\delta y} & \frac{\delta g_y}{\delta z} \\ \frac{\delta g_z}{\delta x} & \frac{\delta g_z}{\delta y} & \frac{\delta g_z}{\delta z} \end{bmatrix} \tag{147}$$

By letting $K = \frac{3\mu {a_e}^2 J_2}{2}$ and after several steps of derivation we obtain the following expressions for all nine terms in Equation 147 for a J_2 gravitation field.

$$\frac{\delta g_x}{\delta x} = \frac{\mu}{r^3}\left[\frac{3x^2}{r^2} - 1\right] - \frac{K}{r^5}\left\{1 - 5\left(\frac{x^2+z^2}{r^2}\right) + 35\frac{x^2z^2}{r^4}\right\} \tag{148}$$

$$\frac{\delta g_x}{\delta y} = \frac{3\mu xy}{r^5} - \frac{5Kxy}{r^5}\left\{\frac{7z^2}{r^4} - \frac{1}{r^2}\right\} \tag{149}$$

$$\frac{\delta g_x}{\delta z} = \frac{3\mu xz}{r^5} - \frac{5Kxz}{r^5}\left\{\frac{7z^2}{r^4} - \frac{3}{r^2}\right\} \tag{150}$$

$$\frac{\delta g_y}{\delta x} = \frac{3\mu xy}{r^5} - \frac{5Kxy}{r^5}\left\{\frac{7z^2}{r^4} - \frac{1}{r^2}\right\} \tag{151}$$

$$\frac{\delta g_y}{\delta y} = \frac{\mu}{r^3}\left[\frac{3y^2}{r^2} - 1\right] - \frac{K}{r^5}\left\{1 - 5\left(\frac{y^2+z^2}{r^2}\right) + 35\frac{y^2z^2}{r^4}\right\} \tag{152}$$

$$\frac{\delta g_y}{\delta z} = \frac{3\mu yz}{r^5} - \frac{5Kyz}{r^5}\left\{\frac{7z^2}{r^4} - \frac{3}{r^2}\right\} \tag{153}$$

$$\frac{\delta g_z}{\delta x} = \frac{3\mu xz}{r^5} - \frac{5Kxz}{r^5}\left\{\frac{7z^2}{r^4} - \frac{3}{r^2}\right\} \tag{154}$$

$$\frac{\delta g_z}{\delta y} = \frac{3\mu yz}{r^5} - \frac{5Kyz}{r^5}\left\{\frac{7z^2}{r^4} - \frac{3}{r^2}\right\} \tag{155}$$

$$\frac{\delta g_z}{\delta z} = \frac{\mu}{r^3}\left[\frac{3z^2}{r^2} - 1\right] - \frac{K}{r^5}\left\{3 - 30\frac{z^2}{r^2} + 35\frac{z^4}{r^4}\right\} \tag{156}$$

XII.E. Observation Model G and the H Matrix: Angle Only - Inertial Frame and CRTBP Frame

Equivalent in both/either reference frame, the inertial and/or CRTBP, the observation model takes the form $\mathbf{G}[\overline{\mathbf{X}}_k, k]$, a function of right ascension and declination, $f(\alpha, \delta)$, is based on the difference between the position vector to the observing platform $\overline{\mathbf{R}}_k$ (P) and the position vector to the target satellite $\overline{\mathbf{r}}_k$ (T), Eqn. 157,

$$\begin{aligned} \mathbf{G}[\overline{\mathbf{X}}_k, k] &= \|\overline{\mathbf{r}}_k - \overline{\mathbf{R}}_k\| \\ &= (x_{\mathbf{T}}\hat{i} + y_{\mathbf{T}}\hat{j} + z_{\mathbf{T}}\hat{k}) - (x_{\mathbf{P}}\hat{i} + y_{\mathbf{P}}\hat{j} + z_{\mathbf{P}}\hat{k}) \\ &= (x_{\mathbf{T}} - x_{\mathbf{P}})\hat{i} + (y_{\mathbf{T}} - y_{\mathbf{P}})\hat{j} + (z_{\mathbf{T}} - z_{\mathbf{P}})\hat{k} \\ &= L_x\hat{i} + L_y\hat{j} + L_z\hat{k}. \end{aligned} \tag{157}$$

Because the target satellite $\bar{\mathbf{r}}_k$ (T) is unknown, $x_{\mathbf{T}}, y_{\mathbf{T}}, z_{\mathbf{T}}$ is unknown, we must compute the line of sight unit vector of $L = l_x\hat{i} + l_y\hat{j} + l_z\hat{k}$, which is based on the known angles of right ascension α and declination δ, namely

$$\begin{aligned} l_x &= cos(\delta)cos(\alpha) \\ l_y &= cos(\delta)sin(\alpha) \\ l_z &= sin(\delta). \end{aligned} \tag{158}$$

To relate the elements of the state vector to the target satellite $\mathbf{X}(t)_{satellite}$, we take the partial of L with respect to the elements of the state vector

$$\begin{aligned} \frac{\delta l_x}{\delta x_{\mathbf{T}}} &= \frac{-(x_{\mathbf{T}} - x_{\mathbf{P}})^2 + L^2}{L^3} \\ \frac{\delta l_x}{\delta y_{\mathbf{T}}} &= \frac{-(x_{\mathbf{T}} - x_{\mathbf{P}})(y_{\mathbf{T}} - y_{\mathbf{P}})}{L^3} \\ \frac{\delta l_x}{\delta z_{\mathbf{T}}} &= \frac{-(x_{\mathbf{T}} - x_{\mathbf{P}})(z_{\mathbf{T}} - z_{\mathbf{P}})}{L^3} \\ \frac{\delta l_y}{\delta x_{\mathbf{T}}} &= \frac{-(y_{\mathbf{T}} - y_{\mathbf{P}})(x_{\mathbf{T}} - x_{\mathbf{P}})}{L^3} \\ \frac{\delta l_y}{\delta y_{\mathbf{T}}} &= \frac{-(y_{\mathbf{T}} - y_{\mathbf{P}})^2 + L^2}{L^3} \\ \frac{\delta l_y}{\delta z_{\mathbf{T}}} &= \frac{-(y_{\mathbf{T}} - y_{\mathbf{P}})(z_{\mathbf{T}} - z_{\mathbf{P}})}{L^3} \\ \frac{\delta l_z}{\delta x_{\mathbf{T}}} &= \frac{-(z_{\mathbf{T}} - z_{\mathbf{P}})(x_{\mathbf{T}} - x_{\mathbf{P}})}{L^3} \\ \frac{\delta l_z}{\delta y_{\mathbf{T}}} &= \frac{-(z_{\mathbf{T}} - z_{\mathbf{P}})(y_{\mathbf{T}} - y_{\mathbf{P}})}{L^3} \\ \frac{\delta l_z}{\delta z_{\mathbf{T}}} &= \frac{L^2 - (z_{\mathbf{T}} - z_{\mathbf{P}})^2}{L^3} \\ \frac{\delta l_x}{\delta \dot{x}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_x}{\delta \dot{y}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_x}{\delta \dot{z}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_y}{\delta \dot{x}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_y}{\delta \dot{y}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_y}{\delta \dot{z}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_z}{\delta \dot{x}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_z}{\delta \dot{y}_{\mathbf{T}}} &= 0 \\ \frac{\delta l_z}{\delta \dot{z}_{\mathbf{T}}} &= 0 \end{aligned} \tag{159}$$

Now, we can define the H matrix and relate the change in a measurement to a change in the state vector.

$$H = \begin{bmatrix} \frac{\delta l_x}{\delta x_{\mathbf{T}}} & \frac{\delta l_x}{\delta y_{\mathbf{T}}} & \frac{\delta l_x}{\delta z_{\mathbf{T}}} & \frac{\delta l_x}{\delta \dot{x}_{\mathbf{T}}} & \frac{\delta l_x}{\delta \dot{y}_{\mathbf{T}}} & \frac{\delta l_x}{\delta \dot{z}_{\mathbf{T}}} \\ \frac{\delta l_y}{\delta x_{\mathbf{T}}} & \frac{\delta l_y}{\delta y_{\mathbf{T}}} & \frac{\delta l_y}{\delta z_{\mathbf{T}}} & \frac{\delta l_y}{\delta \dot{x}_{\mathbf{T}}} & \frac{\delta l_y}{\delta \dot{y}_{\mathbf{T}}} & \frac{\delta l_y}{\delta \dot{z}_{\mathbf{T}}} \\ \frac{\delta l_z}{\delta x_{\mathbf{T}}} & \frac{\delta l_z}{\delta y_{\mathbf{T}}} & \frac{\delta l_z}{\delta z_{\mathbf{T}}} & \frac{\delta l_z}{\delta \dot{x}_{\mathbf{T}}} & \frac{\delta l_z}{\delta \dot{y}_{\mathbf{T}}} & \frac{\delta l_z}{\delta \dot{z}_{\mathbf{T}}} \end{bmatrix}. \tag{160}$$

Notice that the last three columns are all zero. This shows that velocity is not observable in the line of sight unit vector and there is no relation between the change in the elements of L, l_x, l_y, l_z to the change in velocity.

$$H = \begin{bmatrix} \frac{\delta l_x}{\delta x_{\mathbf{T}}} & \frac{\delta l_x}{\delta y_{\mathbf{T}}} & \frac{\delta l_x}{\delta z_{\mathbf{T}}} & 0 & 0 & 0 \\ \frac{\delta l_y}{\delta x_{\mathbf{T}}} & \frac{\delta l_y}{\delta y_{\mathbf{T}}} & \frac{\delta l_y}{\delta z_{\mathbf{T}}} & 0 & 0 & 0 \\ \frac{\delta l_z}{\delta x_{\mathbf{T}}} & \frac{\delta l_z}{\delta y_{\mathbf{T}}} & \frac{\delta l_z}{\delta z_{\mathbf{T}}} & 0 & 0 & 0 \end{bmatrix}. \tag{161}$$

XII.F. Small Angle Noise Simulation: Small Angle DCM Rotation - future study

Because no measurement is perfect, the measured line of sight obtained from a non-perfect camera is simulated with a small angle rotation about each axis of the sensor frame. This results in a small angle rotation about some resultant axis that is pointed at a small angle away from the ideal.

To define this small angle Direction Cosine Matrix (DCM) rotation, the following expression describes a so-called "3-2-1" or a small angle rotation about the "z-y-x" of the sensor frame. Note: it is irrelevant in which order these three rotations are carried out. The final resulting skew symmetric matrix seen in Equation 163 will always occur.

$$\begin{aligned}\Phi &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\phi_x & sin\phi_x \\ 0 & -sin\phi_x & cos\phi_x \end{bmatrix} \begin{bmatrix} cos\phi_y & 0 & -sin\phi_y \\ 0 & 1 & 0 \\ sin\phi_y & 0 & cos\phi_y \end{bmatrix} \begin{bmatrix} cos\phi_y & sin\phi_y & 0 \\ -sin\phi_y & cos\phi_y & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} cos\phi_y cos\phi_z & cos\phi_y sin\phi_z & -sin\phi_y \\ (sin\phi_x sin\phi_y cos\phi_z - cos\phi_y sin\phi_z) & (sin\phi_x sin\phi_y sin\phi_z + cos\phi_x cos\phi_z) & sin\phi_x cos\phi_y \\ (cos\phi_y sin\phi_y cos\phi_z + sin\phi_x sin\phi_z) & (-sin\phi_x sin\phi_y sin\phi_z - sin\phi_x cos\phi_z) & cos\phi_x cos\phi_y \end{bmatrix}.\end{aligned} \tag{162}$$

Because the angles ϕ_x, ϕ_y and ϕ_z are all "small", we can make the approximation that $cos\phi \approx 1$ and $sin\phi \approx \phi$. This leads to the simplification of Equation 162

$$\begin{aligned}\Phi &= \begin{bmatrix} 1 \cdot 1 & 1 \cdot \phi_z & -\phi_y \\ (\phi_x\phi_y \cdot 1 - 1 \cdot \phi_z) & (\phi_x\phi_y\phi_z + 1 \cdot 1) & \phi_x \cdot 1 \\ (1 \cdot \phi_y \cdot 1 + \phi_x\phi_z) & (-\phi_x\phi_y\phi_z - \phi_x \cdot 1) & 1 \cdot 1 \end{bmatrix} \\ &\text{but } \phi \cdot \phi \approx 0 \text{ and } \phi \cdot \phi \cdot \phi \approx 0 \text{ thus} \\ \Phi &\approx \begin{bmatrix} 1 & \phi_z & -\phi_y \\ -\phi_z & 1 & \phi_x \\ \phi_y & -\phi_x & 1 \end{bmatrix} \\ \Phi &\approx I - \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} \\ E &= \begin{bmatrix} 0 & -\phi_z & \phi_y \\ \phi_z & 0 & -\phi_x \\ -\phi_y & \phi_x & 0 \end{bmatrix} \\ \Phi &= [I - E]\end{aligned} \tag{163}$$

Each small angle in matrix E is sampled from a mean zero Gaussian distribution of small angles. The resulting Φ matrix will be the rotation applied to the ideal line of sight unit vector $\hat{\boldsymbol{\rho}}_{ideal}$ to simulate a real world measurement.

$$\hat{\boldsymbol{\rho}}_{corrupted} = \Phi \hat{\boldsymbol{\rho}}_{ideal} \tag{164}$$

The goal in a future study will be to examine the ability of the CRTBP EKF navigator to provide a state estimate good enough to maintain and correct a Moon Flight trajectory for successful Lunar Rendezvous and Lunar Orbit insertion, in the presence of noisy line of sight unit vector measurements to Asteroid 2014 EC.

References

[1]Laplace, P. S., "Memoires sur la determination des orbites des cometes." *Memoires de lAcademie Royale des Sciences de Paris*, 1780.

[2]Gauss, C. F., *Theoria Motus: Theory of the Motion of the Heavenly Bodies Moving About the Sun in Conic Sections*, Boston: Little Brown, 1857.

[3]Bate, Muller, W., *Fundamentals of Astrodynamics*, Dover, 1971.

[4]HERRICK, S., *Astrodynamics: Orbit Determination, Space Navigation, Celestial Mechanics*, Vol. 1, Van Nostrand Reinhold Co., London, 1971.

[5]Herget, P., *The Computation of Orbits. Edwards Brothers Inc., Ann Arbor, published by the author himself (1948)*, The M.I.T. Press, 1974.

[6]Escobal, P., *Methods of Orbit Determination*, John Wiley and Sons, Inc., New York, 1965.

[7]Gooding, R., *A New Procedure for the Solution of the Classical Problem of Minimal Orbit Determination from Three Lines of Sight*, Vol. 66, No. 4, 1997, pp. 387-423, Celestial Mechanics and Dynamical Astronomy, 1997.

[8]Curtis, H., *Orbital Mechanics for Engineering Students*, Elsevier Butterworth-Heinemann, Amsterdam/Boston, 2005.

[9]Vallado, D. A., *Fundamentals of Astrodynamics and Applications*, Microcosm Press, El Segundo, CA, 2001.

[10]Karimi, R. and Mortari, D., "Initial orbit determination using multiple observations," *Celest Mech Dyn Astr*, Vol. 0018-9235/97/, IEEE, 2011, pp. 109:167–180.

[11]Hinga, M. B., "Spacebourne Orbit Determination of Unknown Satellites Using a Stabilized-Gauss-Method, Linear Perturbation Theory and Angle-Only Measurements." Advanced maui optical and space surveillance technologies (amos) conference, september 12 - 14, 2018, 2018.

[12]Prussing, J. E. and Conway, B. A., *Orbital Mechanics*, Oxford Press, New York, 1993.

[13](ESA), E. S. A., *Advanced Concepts Team (ACT), European Space Agency (ESA)*, The PaGMO development team, 2004-2015.

[14]Battin, R. H., *An Introduction to the Mathematics and Methods of Astrodynamics*, American Insitute of Aeronautics and Astrodynamics, 1987.

[15]Gelb, A., Kasper, J. F., Nash, R. A., Price, C. F., and Sutherland, A. A., *Applied Optimal Estimation*, The M.I.T. Press, 1974.

[16]Grebow, D. J., *Generation Periodic Orbits in the Circular Restricted Three BodyProblem with Applications to Lunar South Pole Coverage*, Master's thesis, Purdue University, West Lafayette, Indiana, 2006.

[17]Hinga, M. B., "Hinga Orbit Simulator (HOrbitSim)," Sandia national laboratories, future sand report no. sand 2018-0720, 2018.

[18]https://ssd.jpl.nasa.gov/horizons.cgi, "NASA: Jet Propulsion Laboratory (JPL)," July 2020.

[19]Tapley, B. D., "Statistical Orbit Determination Theory," *Recent Advances in Dynamical Astronomy*, 1972, pp. 396–425.

[20]Watkins, D. S., *Fundamentals of Matrix Computations*, John Wiley and Sons, Inc., 1991.

[21]Tapley, Born, and Schutz, *Statistical Orbit Determination*, Elsevier Academic Press Inc., 2004.

[22]Markley, F. L., Crassidis, J. L., and Cheng, Y., "Survey of Nonlinear Attitude Estimation Methods," *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2005, AIAA Paper 2005-5927, 15-18 August 2005.

[23]Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME–Journal of Basic Engineering*, Vol. 82, No. Series D, 1960, pp. 35–45.