Optical Initial Orbit Determination Using Polynomial Chaos Surrogate Functions

Daniel P. Lubey and Hemanshu Patel
The Aerospace Corporation, El Segundo, California, United States

ABSTRACT

Initial Orbit Determination (IOD) is a critical capability in the field of Space Situational Awareness (SSA); however, IOD methods are not always reliable. Under certain conditions, angles-only IOD algorithms can either fail to produce a solution or produce a solution too inaccurate to use with subsequent estimation methods. This paper develops a new optical IOD algorithm that removes constraints and assumptions from the common optical IOD processes in an attempt to create a more robust framework for IOD. The algorithm uses a polynomial chaos expansion to minimize a Maximum Likelihood cost function (assuming Gaussian errors on the measurements). This new method does not assume any measurements are perfect; rather, it uses all available measurements (including non-angle measurements), and it does not require Keplerian dynamics. Sample applications of this algorithm are provided to demonstrate its capabilities when tracking orbiting targets in various scenarios.

1. INTRODUCTION

Initial Orbit Determination (IOD) is a critical capability within the field of Space Situational Awareness (SSA). Anytime a new detection is made, IOD can help initiate tracking of the object. If a previously tracked object is lost across a long observation gap or due to an unknown maneuver, then IOD may be needed to regain tracking of the target. However, IOD algorithms can fail to produce a solution, and even when a solution is obtained it may be too inaccurate to use with subsequent estimation techniques [1].

Many of the problems with common IOD algorithms stem from the assumptions and constraints that are added to the problem. Typical methods assume that certain measurements are perfect (i.e., do not include any error), that the target's motion is governed by Keplerian dynamics, and many methods ignore some available measurements within the observation arc. Certain IOD algorithms also require numerical derivatives, which may be unreliable in certain cases like short-arc IOD and IOD on highly elliptical orbits. To improve the IOD process we seek a method that does not assume any measurement is perfect, that allows for non-Keplerian dynamics, that uses all available information, and uses analytical derivatives to compute a Maximum Likelihood and Bayesian optimal state estimate. We will specifically address the optical IOD problem, which focuses on using angle data from sensors to produce an estimate of a target's orbit; however, other measurement types can also be included to improve the observability of the scenario.

To achieve this capability in the optical IOD process, we developed an optical IOD algorithm that leverages a method called polynomial chaos. A polynomial chaos expansion (PCE) is an analytical mapping that approximates the relationship between a function's stochastic inputs and its resulting outputs. By varying the error in the angle measurements along with the associated range values, a PCE can be developed that maps these inputs to an optical IOD cost function that includes the information from all available measurements (including non-angle measurements). The PCE can then be used to efficiently evaluate the cost function of interest, and its derivatives can be computed analytically in order to find the inputs that minimize the cost function.

It should be noted that Henderson, Mortari, and Davis [2] addressed a similar topic in previous research. They developed a modified Gooding IOD algorithm that allowed for $N \geq 3$ angle measurement pairs, and their method searched the area around the nominal solution for a better estimate. This removed the constraint that two of the angle measurement pairs are required to have a zero-residual, as discussed previously. The method developed and presented in this paper follows a similar motivation; however, our method explicitly develops an estimator that satisfies the Maximum Likelihood conditions, allows for non-Keplerian dynamics, allows for multiple stations and non-angle measurements in the observation arc, and provides a formalized method for removing the zero measurement residual constraints from the IOD process. Our method also does not require the user to provide a guess for the initial and final

range values. Finally, our method provides a framework that can be expanded to address more than just optical IOD problems.

The remainder of this paper defines, develops, and demonstrates this new PCE-imbued optical IOD algorithm. Section 2 provides background information including information on typical optical IOD methods and how a PCE is generated. Section 3 develops the central algorithm of this paper - an optical IOD method that uses a PCE to find an optimal, measurement residual-unconstrained solution. Section 4 demonstrates the capabilities of this new algorithm through sample applications. Finally, section 5 concludes this paper with a summary of the developments along with suggestions for future expansions of this work.

2. BACKGROUND

The focus of this paper is infusing polynomial chaos into the optical IOD problem. This section provides background on these two areas in order to establish a foundation on which to build this new IOD method.

2.1. Optical Initial Orbit Determination

Optical IOD is the process of taking angle measurements of an orbiting object and producing an estimate of the underlying orbit, which is close enough to truth to use with subsequent estimation algorithms like batch processors or Kalman filters. There is a rich history of work in this area that starts with the development of algorithms to track the motion of various celestial bodies with respect to Earth to the current day where engineers and scientists use similar algorithms to track artificial satellites in orbit as well as newly detected natural bodies.

The first notable methods were developed by Laplace (1780) and Gauss (1809) [1, 3]. Laplace's method takes three angle measurement pairs from a single site at three different times, and then computes the position and velocity vectors of the target at the time of the middle observation via a process that involves computing the roots of an 8th-order polynomial. The method assumes that the dynamics are Keplerian, and it utilizes the Lagrange interpolation scheme to provide the station-to-target direction vector as a function of time (using the three measurements as nodes in the interpolation). In general, the method has deficiencies for Earth orbiting applications but can be quite effective for heliocentric applications. Gauss's method behaves quite differently than Laplace's, but it requires the same inputs (three angle observations at three different times). Rather than estimating the position and velocity at one time, the Gauss method estimates the position at all three measurement times. It assumes the three position vectors are coplanar (a requirement for a Keplerian orbit), and then uses an f and g function formulation that also requires the user to solve for the roots of an 8th-order polynomial. Once the position vectors are estimated, a Gibbs or Herrick-Gibbs method may be used to estimate the full state at a single time [4]. The Gibbs method is a classical approach to this problem, and the Herrick-Gibbs method [5] is an updated version that uses a Taylor expansion in time to make it more applicable to short arcs - which is a shortcoming of Gibbs' method.

The Laplace and Gauss methods are not entirely robust when applied to Earth orbiting problems [6]. Many methods have been devised to adapt these methods to improve their robustness, including adapting the Laplace method to allow for more than three measurements. However, the lack of stability in these methods motivated the development of more modern and computationally robust methods, which use newer numerical techniques. Escobal's Double-R method was developed in 1965 to address a weakness of the existing algorithms - optical IOD with long observation gaps [7]. The method uses a numerical Newton-Raphson root finding method to solve for the radii associated with the initial and final measurements when provided with three angle measurement pairs. This requires the user to provide an initial guess, though. This method also assumes Keplerian dynamics, which requires all of the position vectors to be coplanar. Gooding devised an independent method in 1993, which relied upon his solution to Lambert's problem [8]. This method varies the range associated with the initial and final measurements, and uses a Newton-Raphson method to find a solution that minimizes the interior measurement residual. The resulting trajectories are based upon a Lambert solver, so this method also assumes that the dynamics are Keplerian. As designed, the method only uses three measurements, and it assumes that first and final measurements are perfect. The Constrained Admissible Region (CAR) has been used more recently as a means for short-arc initial orbit determination [9–11]. The method combines physical constraints with optical measurement information in order to produce short-arc optical IOD state estimate. This algorithm is used as a basis for the admissible region approach that is discussed later in this paper.

The new optical IOD method described in this paper was developed to remove the constraints and limitations that exist within the classical methods. These include: 1) assuming that the first and final measurement residuals are zero (i.e., perfect measurements), 2) under-informing the IOD estimate (in the presence of noise) by only using three pairs

of angle measurements, 3) assuming the dynamics are purely Keplerian, and 4) the usage of numerical derivatives to converge to a solution (as in Gooding's method). The under-informed limitation and the Keplerian assumption are removed by developing a Maximum Likelihood cost function that uses all pieces of information to estimate the state, and by solving a two-point boundary value problem (TPBVP) that allows for the inclusion of perturbations. The zero-residual constraint and the limitation of using numerical derivatives are removed by combining the filter with a polynomial chaos algorithm that allows for variations in the range as well as error variations in the first and last measurements. The PCE acts as an analytical surrogate to a fully informed cost function, so that it can be minimized via a Newton-Raphson algorithm (using analytical derivatives) to obtain an optimal optical IOD state estimate. The process of generating a PCE is explored in the following section.

2.2. Polynomial Chaos Expansions

A polynomial chaos expansion (PCE) is a mapping that relates a function's stochastic inputs to the resulting outputs. The mapping is analytical, so it provides a computationally efficient means for evaluating a function many times and its derivatives can be computed analytically. It may also be used to efficiently compute moments of the probability density function (PDF) of the system's output. This section outlines the process of generating a PCE. For a more detailed overview of the subject Refs. [12], [13], and [14] are suggested. These references are used as a template for the following discussion.

The PCE of function $\vec{g}(\vec{\eta}) : \mathbb{R}^d \rightarrow \mathbb{R}^n$ of order p is defined in Eq. 1. It is a weighted sum whose coefficients are signified by $\vec{c}_{\vec{\nu}_i} \in \mathbb{R}^n$, and its basis functions are signified by $\Psi_{\vec{\nu}}(\vec{\eta}) : \mathbb{R}^d \rightarrow \mathbb{R}$. In these definitions $\vec{\nu}_i$ is the ith element of $\Lambda_{p,d}$ where $i \in \{n \in \mathbb{N} |\ n \leq P\}$.

$$
\begin{aligned}
\tilde{g}(\vec{\eta}) &= \sum_{i=1}^{P} \vec{c}_{\vec{\nu}_i} \Psi_{\vec{\nu}_i}(\vec{\eta}) \\
\Lambda_{p,d} &:= \left\{ \vec{\nu} \in \mathbb{N}_0^d |\ ||\vec{\nu}||_1 \leq p, ||\vec{\nu}||_0 \leq d \right\} \\
\Psi_{\vec{\nu}_i}(\vec{\eta}) &= \prod_{j=1}^{d} \Psi_{\nu_{i,j}}(\eta_j) \\
P &= \frac{(p+d)!}{p!d!}
\end{aligned}
\tag{1}
$$

The PCE can be rewritten in vector notation as shown in Eq. 2.

$$
\begin{aligned}
\tilde{g}(\vec{\eta}) &= C^T h(\vec{\eta}) \\
C^T &= \left[\begin{array}{ccc} \vec{c}_{\vec{\nu}_1} & \dots & \vec{c}_{\vec{\nu}_P} \end{array} \right] \\
h(\vec{\eta})^T &= \left[\begin{array}{ccc} \Psi_{\vec{\nu}_1}(\vec{\eta}) & \dots & \Psi_{\vec{\nu}_P}(\vec{\eta}) \end{array} \right]
\end{aligned}
\tag{2}
$$

The basis functions of a PCE are what give it its statistical properties. Based on the type of the statistical input used, different basis functions are used. Polynomials are chosen to ensure that they are orthogonal with respect to one another when weighted by the PDF of the input's statistical type. Inputs drawn from a uniform distribution with supports $[-1, +1]$ use Legendre polynomials, and inputs drawn from a zero-mean unit-variance Gaussian distribution use Hermite polynomials. We specifically normalize the polynomials in order to ensure that the polynomials are orthonormal to each other when weighted by the appropriate PDF. It should be noted that any n-dimension Normal distribution with mean $\vec{\mu}$ and covariance P can be decomposed into a function of n independent zero-mean, unit variance Gaussian random variables (Eq. 3).

$$
\begin{aligned}
\vec{w} &= B\vec{\eta} + \vec{\mu} \\
\vec{w} &\sim \mathcal{N}(\vec{\mu}, P) \\
\vec{\eta} &\sim \mathcal{N}(\vec{0}, I_{n \times n}) \\
P &= B^T B
\end{aligned}
\tag{3}
$$

A uniform random variable on supports $[a, b]$ can be decomposed into a function of a uniform variable on supports $[-1, +1]$ as shown in Eq. 4.

$$\begin{aligned} u &= \left[\frac{b-a}{2}\right](\beta+1)+a \\ u &\sim \mathcal{U}[a,b] \\ \beta &\sim \mathcal{U}[-1,+1] \end{aligned} \tag{4}$$

These decompositions are essential when applying polynomial chaos to arbitrary systems.

A PCE is generated using the results of a small Monte Carlo evaluation (M evaluations). An optimization method is then employed to solve for the PCE coefficients. An optimal PCE minimizes the residuals between the evaluated function and the PCE approximation. The typical method for solving for an optimal PCE is least squares; however, this method requires $M \gg P$ evaluations. For certain problems, evaluating the function this many times is infeasible, thus we will pursue a solution that is not as computationally intense. Compressive sampling is such a method.

2.2.1. Compressive Sampling via Orthogonal Matching Pursuit

The typical least squares solution to generating a PCE suffers from the curse of dimensionality. This means that as the dimension of the input of increases the number of terms in the PCE increases exponentially. Because the least squares method requires $M \gg P$, this can make the PCE method infeasible to apply to systems with large dimension inputs or even small dimension systems with large order solutions; however, generating a PCE via compressive sampling provides a way around this problem. Compressive sampling methods require $M \ll P$, which allows PCEs to be generated for systems where the least squares solution is infeasible. When the function of interest is sufficiently smooth with respect to its inputs the number of dominant basis functions in the PCE is generally small. Compressive sampling iteratively determines the most dominant terms in the PCE, and assumes all remaining terms are zero. The method described in this section is slightly different than that discussed by Jones, Parrish, and Doostan [14]. Our method applies a weighting to each evaluation of the function. This allows the user to ensure that the resulting PCE has greater accuracy in specific portions of the domain. Since we are focusing on minimizing a PCE surrogate for an IOD cost function, we prefer that smaller cost function values are more accurately approximated by the PCE.

Compressive sampling when applied to an n-dimensional function must be applied to each dimension separately. The cost function used to generate the PCE via compressive sampling for the ith dimension (Eq. 5) minimizes an L0-norm under the constraint that the L2 -norm of the weighted residual between the PCE and the function evaluations ($\vec{\delta}_i$) is less than a given threshold (ϵ_i). In this definition, $\vec{C}_i$ is the ith column of the C matrix.

$$\begin{aligned} \mathcal{J}_{PC}(\vec{C}_i) &= \left|\left|\vec{C}_i\right|\right|_0, \ \left|\left|\vec{\delta}_i\right|\right|_2 \leq \epsilon_i \\ \vec{\delta}_i &= WH\vec{C}_i - W\vec{Y}_i \\ W &= diag\left(\left[\begin{array}{ccc} w_1 & \dots & w_M \end{array}\right]\right) \\ H^T &= \left[\begin{array}{ccc} h(\vec{\eta}_1) & \dots & h(\vec{\eta}_M) \end{array}\right] \\ Y^T &= \left[\begin{array}{ccc} \vec{g}(\vec{\eta}_1) & \dots & \vec{g}(\vec{\eta}_M) \end{array}\right] \end{aligned} \tag{5}$$

To start the procedure, we first assume that all coefficients in the PCE are zero (Eq. 6). This includes defining the index set of dominant PCE terms (Λ) as an empty set, which will have a new and unique index added in each iteration.

$$\begin{aligned} \mathbb{U} &= \{n \in \mathbb{N} |\ n \leq P\} \\ \Lambda &= \emptyset \\ \hat{C}_i &= \vec{0} \end{aligned} \tag{6}$$

To determine the most dominant PCE term in each iteration, we use an Orthogonal Matching Pursuit (OMP) scheme. This involves finding the term that is most orthogonal to the residual vector (Eq. 7). In these definitions $\vec{H}_j$ is the jth column of the H matrix.

$$k = \underset{j \in \Lambda^C}{\arg\max} \left[\frac{\vec{H}_j^T W^T \vec{\delta}_i}{\sqrt{\vec{H}_j^T \left(W^T W\right) \vec{H}_j}}\right] \tag{7}$$

Once the dominant term is identified, it is added to Λ and a new set of PCE coefficients are solved for via a modified weighted least squares method (Eq. 8). This modified version includes solving for the PCE coefficients corresponding to terms in Λ ($\hat{C}_{i,\Lambda}$) and setting all other terms equal to zero ($\hat{C}_{i,\Lambda^C}$). In this expression, H_Λ is the sub-matrix of H that only includes the columns of H that correspond to the indices in Λ.

$$\begin{aligned} &\Lambda \leftarrow \Lambda \cup \{k\} \\ &\hat{C}_{i,\Lambda} = \left[H_\Lambda^T \left(W^T W\right) H_\Lambda\right]^{-1} H_\Lambda^T W^T \vec{\delta}_i \\ &\hat{C}_{i,\Lambda^C} = \vec{0} \end{aligned} \tag{8}$$

This process continues until the constraint in Eq. 5 is satisfied. In practice, the user should set a maximum number of non-zero terms to help ensure that $M \ll P$.

3. OPTICAL IOD VIA POLYNOMIAL CHAOS

Having a foundation in existing optical IOD methods and PCEs, we now proceed with developing the central algorithm of this paper. The algorithm is described in four subsections: 1) a description of the Maximum Likelihood cost function used in this new method, 2) the process of zooming in on the cost function minimum so that an accurate PCE may be generated, 3) the process of minimizing a PCE which varies just range values, and 4) the process of iterating over the range variations with multiple variations in the angle pairs in order to develop a residual unconstrained solution.

3.1. Optical IOD Cost Function

As discussed, typical optical IOD algorithms use only three angle measurement pairs. This artificially under-informs the IOD estimate when more measurements are available. The effect is especially pronounced when multiple stations and non-angle measurement types are ignored. To overcome this issue, we designed this new IOD method to use all available measurement information by having it minimize a weighted least squares cost function with N measurement vectors (Eq. 9). This cost function is also a Maximum Likelihood cost function when the measurements include zero-mean Gaussian error that is distributed according to covaraince R_i.

$$\begin{aligned} \mathcal{J}_{IOD}(\vec{z}_{a,b}) &= \sum_{i=1}^{N} \frac{1}{2} \left[\vec{y}_i - h\left(t_i, \vec{x}(t_i), \vec{x}_s(t_i)\right)\right]^T R_i^{-1} \left[\vec{y}_i - h\left(t_i, \vec{x}(t_i), \vec{x}_s(t_i)\right)\right] \\ \vec{z}_{a,b} &= \left[\begin{array}{cc} \vec{r}(t_a)^T & \vec{r}(t_b)^T \end{array}\right]^T \\ \vec{x}(t) &= \left[\begin{array}{cc} \vec{r}(t)^T & \vec{v}(t)^T \end{array}\right]^T \\ \dot{\vec{x}}(t) &= f(t, \vec{x}(t)) \\ \vec{x}(t) &= \phi(t; t_a, t_b, \vec{z}_{a,b}) \end{aligned} \tag{9}$$

This cost function uses the inverse of the measurement covariance matrix ($R_i \in \mathbb{R}^{m \times m}$) as the weighting matrix for each measurement residual where $\vec{y}_i \in \mathbb{R}^m$ refers to the measurement at time $t_i \in \mathbb{R}$ and $h(t, \vec{x}, \vec{x}_s) : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}^m$ is the function that maps a time, target state, and station state to a corresponding measurement. It should be noted that the measurements may come from different stations, which are not required to be ground-based. It should also be noted that at least two angle measurement pairs must be in the measurement arc at two different times. Beyond this the combined information from all measurements must make the system observable in order to generate a target state estimate. This is a requirement of any estimation method.

The input being optimized for this cost function is the double position vector $\vec{z}_{a,b} \in \mathbb{R}^6$, which includes positions at times t_a and t_b. These times correspond to separate epochs ($t_a \neq t_b$) where angle measurements are taken from at least one station. The positions are generated via Eq. 10, which requires azimuth and elevation measurements (α and $\theta \in \mathbb{R}$, respectively), a range value ($\rho_i \in \mathbb{R}$), and the station position vector ($\vec{r}_s(t_i) : \mathbb{R} \rightarrow \mathbb{R}^3$). Additionally, variations in the azimuth and elevation angles ($\epsilon_{\alpha,i}$ and $\epsilon_{\theta,i} \in \mathbb{R}$, respectively) are considered to obtain a residual-unconstrained solution. The unit vectors defining the topocentric East-North-Up (ENU) frame are used in this definition ($\hat{e}$, $\hat{n}$, and

$\hat{u}$, respectively). A description of how these variations and the range are chosen is provided in Section 3.4.

$$\begin{aligned}\vec{r}(t_i) &= \vec{r}_s(t_i) + \vec{\rho}(t_i) \\ \vec{\rho}(t_i) &= \rho_i \left[\sin\left[\alpha(t_i) + \epsilon_{\alpha,i}\right] \cos\left[\theta(t_i) + \epsilon_{\theta,i}\right] \hat{e}\right. \\ &\quad \left. + \cos\left[\alpha(t_i) + \epsilon_{\alpha,i}\right] \cos\left[\theta(t_i) + \epsilon_{\theta,i}\right] \hat{n} + \sin\left[\theta(t_i) + \epsilon_{\theta,i}\right] \hat{u}\right]\end{aligned} \tag{10}$$

Once computed, these positions are used along with the system dynamics ($f(t, \vec{x}) : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$) to generate the full state as a function of time ($\vec{x}(t) : \mathbb{R} \to \mathbb{R}^6$) via solution of this TPBVP. For Keplerian dynamics, the TPBVP is solved via Lambert's method. For dynamics with additional perturbations, the TPBVP can be solved by using the Lambert solution as an initial guess in a standard Newton-Raphson method. Lambert's problem has multiple solutions, but because we are attempting to minimize this IOD cost function the solution that yields the smallest cost should be used.

With no analytical solution to this TPBVP, there is no analytical solution to this cost function in terms of the two position vectors. This normally would necessitate the use of numerical derivatives to minimize the IOD cost function, which can create computational issues in the IOD process for ill-conditioned systems. However, because we will approximate this cost function with a PCE, analytical derivatives of the PCE may be generated for the minimization procedure.

3.2. Solution Constraints and the Cost Function Zoom Method

The inputs into the cost function are essentially two varied angle pairs ($\tilde{y}_a$ and $\tilde{y}_b$) and associated range values (ρ_a and ρ_b). They correspond to times t_a and t_b, which are separate times in the measurement arc at which angle measurements were obtained. Combined, these values can be turned into two target positions via Eq. 10, which can be turned into a full trajectory by solving the resulting boundary value problem. The two angle pairs are Gaussian distributed based on their accompanying measurements ($\vec{y}_a$ and $\vec{y}_b$). The range values, however, are not based on any available information. Therefore, they need to be drawn from a uniform distribution, which requires finite supports. To obtain these supports the user needs to place physical constraints on the system. This is similar to the constrained admissible region (CAR) method currently used for object correlation and other applications (including IOD) [9–11]; however, we will not enforce the full CAR framework on this initial version of this algorithm. Rather, we will require the user to input minimum perigee and maximum apogee constraints, which will be used to generate an approximated rectangular admissible region for the range values.

Given the user-defined minimum perigee ($r_{p,min}$) and maximum apogee ($r_{a,max}$) values, the minimum and maximum range values can be obtained at time t_i for a given elevation measurement (θ_i) from a station with a known distance from the Earth center of mass ($r_s(t_i)$) by solving Eq. 11.

$$\begin{aligned}\rho_{i,min} &= r_s(t_i) \left[-\sin\theta_i + \sqrt{\left(\frac{r_{p,min}}{r_s(t_i)}\right)^2 - \cos^2\theta_i}\right] \\ \rho_{i,max} &= r_s(t_i) \left[-\sin\theta_i + \sqrt{\left(\frac{r_{a,max}}{r_s(t_i)}\right)^2 - \cos^2\theta_i}\right]\end{aligned} \tag{11}$$

With supports established for the range values, we are to generate a PCE on the optical IOD cost function; however, if the cost function is not sufficiently smooth with respect to the inputs on the domain of interest, then the resulting PCE will be an inaccurate approximation [13]. If the user starts with a large search area in their perigee and apogee constraints, then it is unlikely that the cost function will be sufficiently smooth with respect to range variations. This problem can be solved by zooming in on the solution area.

To zoom in on the solution area, the user needs to iteratively evaluate the cost function by varying range for fixed angle measurements. In each iteration, the user randomly generates M_0 realizations of range pairs from the computed limits. For the given angle data, the user then evaluates the cost function M_0 times. The algorithm then selects the smallest evaluations of the cost function and assigns their indices to the set $\Theta^{(i)}$ where i is the iteration number. We define the smallest evaluations as all evaluations that are c times greater than the smallest evaluation or smaller. We also require a minimum N_0 evaluations in this set. Extra values may need to be added if there are not enough that meet the following condition. In this case the smallest one should be added to the set until the set has N_0 members. It

is a requirement that $N_0 < M_0$. The algorithm then finds a bounding rectangle around these evaluations to shrink the search area. The process is initialized with iteration $i = 0$ as described in Eq. 12.

$$
\begin{aligned}
l_p^{(0)} &= \frac{\rho_{a,max} - \rho_{a,min}}{2},\ \hat{p}^{(0)} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
l_q^{(0)} &= \frac{\rho_{b,max} - \rho_{b,min}}{2},\ \hat{q}^{(0)} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
\bar{\rho}^{(0)} &= \begin{bmatrix} \frac{\rho_{a,max}+\rho_{a,min}}{2} \\ \frac{\rho_{b,max}+\rho_{b,min}}{2} \end{bmatrix}
\end{aligned}
\tag{12}
$$

After initializing the process, the next step is to randomly sample range pairs as described in Eq. 13. The optical IOD cost function should then be evaluated for each of these range pairs. The indices of the smallest evaluations (as described above) are then use to create the set $\Theta^{(i)}$.

$$
\begin{aligned}
\bar{\rho}_j^{(i)} &= \begin{bmatrix} \rho_{a,j}^{(i)} \\ \rho_{b,j}^{(i)} \end{bmatrix} = \bar{\rho}^{(i)} + \gamma_{1,j}^{(i)} l_p^{(i)} \hat{p}^{(i)} + \gamma_{2,j}^{(i)} l_q^{(i)} \hat{q}^{(i)} \\
\gamma &\sim \mathcal{U}[-1, +1]
\end{aligned}
\tag{13}
$$

Once the smallest evaluations have been identified, the updated range domain should be computed as described in Eq. 14 where $N_{\Theta^{(i)}}$ is the number of values in the set $\Theta^{(i)}$. This process involves finding the mean range pair, finding the range pair that has the largest L2-norm with respect to the mean, and then sizing the remaining perpendicular component of the bounding rectangle. Once this update is completed, the process continues by repeating the steps after the initialization until the maximum number of predetermined iterations is reached. Alternatively, the user could choose to end this process once a convergence criteria is obtained (e.g., maximum range search area, difference in successive iterations is below some threshold, etc.).

$$
\begin{aligned}
\bar{\rho}^{(i+1)} &= \frac{1}{N_{\Theta^{(i)}}} \sum_{j \in \Theta^{(i)}} \bar{\rho}_j^{(i)} \\
k &= \underset{j \in \Theta^{(i)}}{\arg\max} \left[\left\| \bar{\rho}_j^{(i)} - \bar{\rho}^{(i+1)} \right\|_2 \right] \\
l_p^{(i+1)} &= \left\| \bar{\rho}_k^{(i)} - \bar{\rho}^{(i+1)} \right\|_2,\ \hat{p}^{(i+1)} = \frac{\bar{\rho}_k^{(i)} - \bar{\rho}^{(i+1)}}{l_p^{(i+1)}} \\
m &= \underset{j \in \Theta^{(i)}}{\arg\max} \left[\left\| \left(\bar{\rho}_j^{(i)} - \bar{\rho}^{(i+1)} \right) - \left[\left(\bar{\rho}_j^{(i)} - \bar{\rho}^{(i+1)} \right)^T \hat{p}^{(i+1)} \right] \hat{p}^{(i+1)} \right\|_2 \right] \\
l_q^{(i+1)} &= \left\| \left(\bar{\rho}_m^{(i)} - \bar{\rho}^{(i+1)} \right) - \left[\left(\bar{\rho}_m^{(i)} - \bar{\rho}^{(i+1)} \right)^T \hat{p}^{(i+1)} \right] \hat{p}^{(i+1)} \right\|_2 \\
\hat{q}^{(i+1)} &= \frac{\left(\bar{\rho}_m^{(i)} - \bar{\rho}^{(i+1)} \right) - \left[\left(\bar{\rho}_m^{(i)} - \bar{\rho}^{(i+1)} \right)^T \hat{p}^{(i+1)} \right] \hat{p}^{(i+1)}}{l_q^{(i+1)}}
\end{aligned}
\tag{14}
$$

Once the area of the solution has been identified, the user may start the minimization procedure via a PCE of the cost function with the range pair as inputs. It should be noted that this method is heuristic, and is vulnerable to undersampling problems. It is important that the user selects a large enough value for M_0 to ensure that the cost function is evaluated a sufficient number of times to identify the area where the cost function minimum exists.

3.3. Residual-Constrained Optical IOD with PCEs

This subsection describes an application of this algorithm that is similar to typical IOD methods in that it constrains the residuals of two angle measurements to be zero. However, this method uses all available information, and it does not require the target to be guided by Keplerian dynamics. This method will also use analytical derivatives to find the range pair that minimizes the optical IOD cost function.

This algorithm finds the optimal vector $\vec{\gamma}^T = [\gamma_1\ \gamma_2]$, which minimizes the optical IOD cost function given angle information at two times: $(t_a, \tilde{y}_a, \vec{r}_s(t_a))$ and $(t_b, \tilde{y}_b, \vec{r}_s(t_b))$. γ_1 and γ_2 are uniform random variables, which can be used to compute a range pair (Eq. 13). Using M_γ realizations of $\vec{\gamma}$, we next obtain M_γ evaluations of the optical IOD cost function. A PCE is then generated according the process outlined in Section 2.2. For this procedure, we prefer to generate the PCE on the natural logarithm of the cost function evaluations, which tends to produce more reliable results than when using the raw cost function value. We also use the inverse of the raw cost function value as the weighting ($w_i = 1/\mathcal{J}_{IOD}(\vec{\gamma}_i)$) in the PCE generation in order to ensure that the smallest values are fit best since we are most interested in this part of the mapping. A sample residual-constrained PCE mapping is shown in Fig. 1. It can be seen in these results that the most peculiar behavior occurs with the largest cost function values. This is a symptom of weighting the PCE to prefer minimizing the residuals of the small cost function evaluations.

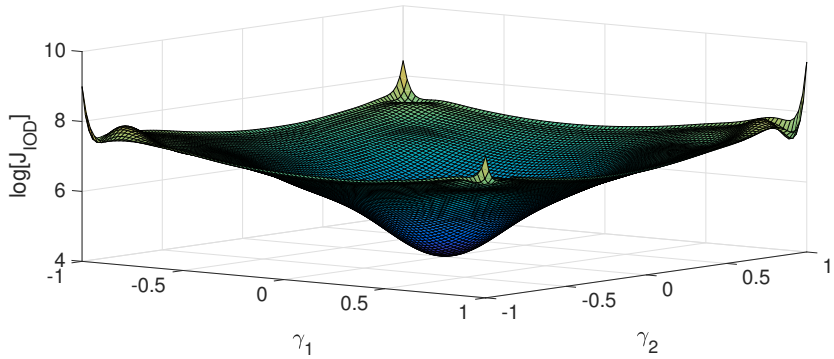

Figure 1: Residual-constrained PCE of a satellite in GEO observed by a single optical-only ground station.

The next step in the process is finding the $\vec{\gamma} \in \{[-1,+1] \times [-1,+1]\}$ that minimizes the PCE approximation of the optical IOD cost function. This is obtained via a Newton-Raphson root finder method that uses the analytical derivative of the PCE approximation (Eq. 15). This requires taking first and second derivatives of normalized Legendre polynomials, which can be computed analytically.

$$
\begin{aligned}
&\vec{\gamma}^{(i+1)} = \vec{\gamma}^{(i)} - \left[\left. \frac{d^2 \tilde{\mathcal{J}}_{IOD}}{d\vec{\gamma}^2} \right|_{\vec{\gamma}^{(i)}} \right]^{-1} \left. \frac{d\tilde{\mathcal{J}}_{IOD}}{d\vec{\gamma}}^T \right|_{\vec{\gamma}^{(i)}} \\
&\frac{d\tilde{\mathcal{J}}_{IOD}}{d\vec{\gamma}} = \sum_{i=1}^{P} c_{\vec{\nu}_i} \frac{d\Psi_{\vec{\nu}_i}(\vec{\gamma})}{d\vec{\gamma}}, \quad \frac{d^2\tilde{\mathcal{J}}_{IOD}}{d\vec{\gamma}^2} = \sum_{i=1}^{P} c_{\vec{\nu}_i} \frac{d^2\Psi_{\vec{\nu}_i}(\vec{\gamma})}{d\vec{\gamma}^2} \\
&\frac{d\Psi_{\vec{\nu}_i}(\vec{\gamma})}{d\gamma_j} = \left[\prod_{k=1,k\neq j}^{d} \Psi_{\nu_{i,k}}(\gamma_k) \right] \frac{d\Psi_{\nu_{i,j}}(\gamma_j)}{d\gamma_j} \\
&\frac{d^2\Psi_{\vec{\nu}_i}(\vec{\gamma})}{d\gamma_j d\gamma_k} = \begin{cases} \left[\prod_{m=1,m\neq j,k}^{d} \Psi_{\nu_{i,m}}(\gamma_m) \right] \frac{d\Psi_{\nu_{i,j}}(\gamma_j)}{d\gamma_j} \frac{d\Psi_{\nu_{i,k}}(\gamma_k)}{d\gamma_k} & ;\ k \neq j \\ \left[\prod_{m=1,m\neq j}^{d} \Psi_{\nu_{i,m}}(\gamma_m) \right] \frac{d^2\Psi_{\nu_{i,j}}(\gamma_j)}{d\gamma_j^2} & ;\ k = j \end{cases}
\end{aligned}
\tag{15}
$$

It should be noted that this is a constrained optimization problem, so if the solution falls outside the domain of $\vec{\gamma}$, then the $\vec{\gamma}$ on the border of this domain that results in the smallest cost is the optimal value. This is motivated by the Karush-Kuhn-Tucker (KKT) conditions for optimization with inequality constraints [15].

Once an optimal $\vec{\gamma}$ value has been obtained, the result can be mapped into range values (Eq. 13). This result can be combined with the angle information in order to compute positions at both times (Eq. 10). The trajectory that solves this boundary value problem is the residual constrained solution to the optical IOD problem. The result may then be used with a subsequent estimation method, or the user may proceed to obtain a residual-unconstrained solution.

3.4. Residual-Unconstrained Optical IOD with PCEs

While a residual-constrained IOD solution may be sufficient for certain applications, this new optical IOD method is meant to offer a means to remove this constraint. There are two ways that this constraint may be removed. The first method involves simultaneously varying range and angles. The second method involves a nested PCE approach where for each variation in the angle information a residual constrained solution is determined as described in Section 3.3. Another PCE is generated over these solutions with angles as the varied parameter. Each of these methods are described in Sections 3.4.1 and 3.4.2.

3.4.1. Simultaneous Variation of Range and Angles

This solution method can be more efficient depending on user-defined sampling rates, but typically it requires many evaluations and a high order PCE in order to generate an accurate approximation. In the optical IOD problem, the cost function is far more sensitive to range than it is to the angle variations. Thus, in order to capture the effect of the measurement variations, many evaluations are necessary.

The state vector in this optimization procedure is shown in Eq. 16. The γ values are realizations of a uniform distribution and are used to compute range in the same manner as the residual-constrained version of the algorithm (Eq. 13). The β values are realizations of a zero-mean unit-variance normal distribution, and they are used to vary the angle information based on the angle measurements provided at times t_a and t_b.

$$\begin{aligned}
&\vec{\eta}^T = \left[\begin{array}{cccccc} \gamma_1 & \gamma_2 & \beta_{a,1} & \beta_{a,2} & \beta_{b,1} & \beta_{b,2} \end{array}\right] \\
&\tilde{y}_i = \vec{y}_i - S_i \left[\begin{array}{c} \beta_{i,1} \\ \beta_{i,2} \end{array}\right] \\
&\beta_{i,j} \sim \mathcal{N}(0,1) \\
&\vec{\zeta}^T = \left[\begin{array}{cccc} \rho_a & \tilde{y}_a^T & \rho_b & \tilde{y}_b^T \end{array}\right]
\end{aligned} \tag{16}$$

When zooming in on the solution area, only the nominal measurements should be used; however, care should be exercised to ensure that solution is not too zoomed in such that it would eliminate possible solutions when the angles are varied.

Beyond this adjustment to the zoom-in method, the approach is quite similar to the residual-constrained method except that all parameters are varied between evaluations. Once all M evaluations have been made, a weighted PCE is generated (using the same weightings as described previously). This PCE is then minimized using the same Newton-Raphson procedure except that the state vector is augmented to include the β terms too. It should be noted that the β values are unconstrained in the optimization procedure, but the γ values are still constrained to their finite domain. The optimal $\vec{\eta}$ value may then be transformed into the optimal state by successively applying Eqs. 16, 13, and 10 and then solving the boundary value problem.

3.4.2. Nested Variation of Range and Angles

The simultaneous variation method can be more computationally efficient (depending on how many evaluations are needed), but a nested method tends to be more robust. In this method, M_β variations of the angle information are generated. For each variation, an optimal range pair and cost function value are obtained by solving the residual-constrained version of this algorithm. A final PCE is then generated using the residual-constrained optimal cost function values as inputs and the corresponding β values as inputs. This PCE is then minimized via the Newton-Raphson method (Eq. 15) where the state vector only includes the β values. The optimal $\vec{\beta}$ that results from this procedure may then be used in the residual-constrained optimization in order to obtain corresponding optimal range values. The combined optimal range and angle values define the residual-unconstrained solution, which can be mapped into the optimal trajectory estimate as described in Section 3.4.1.

This method tends to be more robust because it independently captures the effects of angle and range variations. In an optical only problem, range variations tend to have a much larger effect on the cost function as compared to angle variations (when the variations are small). Given that user has at least some information about the angles and no information on the range, the range variations need to be much larger. By independently assessing these effects, the range variables do not dominate the PCE. It is suggested as future work to assess how the range and angle variations may be accurately captured reliably in a single PCE. For the time being, the nested version of this algorithm provides an effective solution to generating a residual-unconstrained IOD state estimate.

4. SIMULATIONS AND RESULTS

Having defined and developed this new PCE-based optical IOD algorithm, we now pursue demonstrating its perfomance through numerical simulation. These simulations cover three orbit types: 1) Geosynchronous Earth Orbit [GEO], 2) Highly Elliptical Orbit [HEO], and 3) Low Earth Orbit [LEO]. These orbits are shown in Fig. 2 and their true initial conditions defining the true orbits are defined in Table 1 with the initial epoch being defined as 2016-01-01 00:00:00.000 UTC.

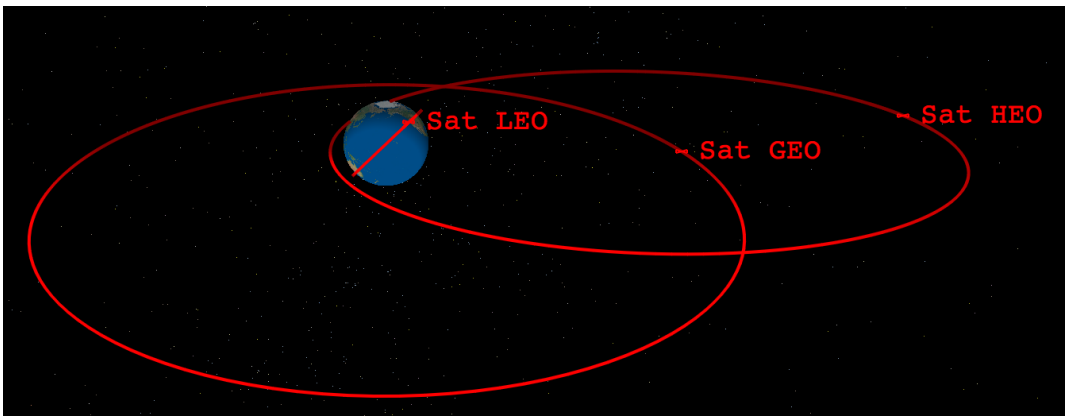

Figure 2: Orientation of considered orbits in numerical simualtions.

Table 1: True Keplerian Initial Conditions for Test Cases at Initial Epoch

Orbit Type	a [km]	e [-]	i [deg]	Ω [deg]	ω [deg]	M [deg]
GEO	42166.26	0.0005	0.20	270.00	15.00	90.00
HEO	42000.00	0.8000	10.00	0.00	200.00	180.00
LEO	7200.00	0.0500	47.00	270.00	0.00	60.00

For each orbit, we first evaluate how well this algorithm estimates the epoch state given an adequately sized observation arc. The GEO, HEO, and LEO cases have arc lengths of 1 hour, 2 hours, and 10 minutes, respectively. Next, the GEO case is tested with a short-arc scenario, which uses a 30-minute observation window. This short-arc scenario is analyzed to identify how this PCE-based optical IOD method improves performance when multiple stations and/or multiple data types are included in the observation arc. In each scenario, measurements are generated from each station once every two minutes. Angle data from each station is corrupted with 2 arcsecond (1-σ) Gaussian error. Range data includes 10 meter (1-σ) Gaussian error. The stations used in these scenarios are defined in Table 2. Each scenario uses the El Segundo, CA location with angle data. The other stations and range data are only used when explicitly stated. Each case also only uses the nested version of the algorithm, and for this study we will only focus on Keplerian dynamics.

Table 2: Available Station Locations

Location	Latitude [deg]	Longitude [deg]	Altitude [m]
El Segundo, CA	33.919182	-118.416466	0.00
Portland, OR	45.5231	-122.6765	0.00
Boulder, CO	40.0150	-105.2705	0.00

For each of the following cases the PCE and zoom parameters defined in Table 3 are used. By reducing the tolerances the user can obtain a more converged result. By increasing the PCE order, the number of cost function evaluations, and the maximum number of non-zero terms in the PCE, the user can obtain a result that is a closer approximation to the truth. The values used in these scenarios were chosen to be computationally intensive, intentionally to first verify that this algorithm works when the cost function is well-sampled, the PCEs are of high order,

and the convergence tolerances are relatively small. Other sample scenarios have been run with smaller a number of cost function evaluations, and the results do not change significantly. It is left as future work to do a full analysis to determine how these values should be properly chosen to achieve a given accuracy while ensuring the algorithm is not too computationally intensive.

In terms of zoom parameters, the minimum perigee and maximum apogee are selected to search a wide volume of space starting at 200 km altitude all the way to a radius that is three times larger than GEO orbit. The number of zoom iterations includes two values. The first is for an initial zoom before varying the angle measurements. The second is a specific zoom for each angle variation. By splitting the zoom method in this way, the user can make the algorithm more efficient because each evaluation of an angle variation does not have to iterate as many times when zooming in on the solution.

Table 3: PCE and Zoom Parameters Used in Numerical Simulation

Parameter Description	**Symbol**	**Value**	**Units**
PCE Order (Range Variations)	p_ρ	15	-
Number of Cost Evaluations (Range Variations)	M_ρ	5000	-
Convergence Tolerance (Range Variations)	Δ_ρ	0.001	km
Compressive Sampling Tolerance (Range Variations)	$\Delta_{cs,\rho}$	1.0×10^{-6}	-
Maximum Terms in PCE (Range Variations)	$P_{max,\rho}$	150	-
PCE Order (Angle Variations)	p_y	10	-
Number of Cost Evaluations (Angle Variations)	M_y	1000	-
Convergence Tolerance (Angle Variations)	Δ_y	0.001	arcsec
Compressive Sampling Tolerance (Angle Variations)	$\Delta_{cs,y}$	1.0×10^{-6}	-
Maximum Terms in PCE (Angle Variations)	$P_{max,y}$	200	-
Maximum Apogee	$r_{a,max}$	126492.5	km
Minimum Perigee	$r_{p,min}$	6578.137	km
Zoom Evaluations	M_0	500	-
Zoom Small Set Minimum Size	N_0	50	-
Zoom Iterations	n_{zoom}	10, 3	-
Zoom Scaling Factor	c	10	-

4.1. Nominal Arc Performance

To first evaluate how this algorithm performs, we will quantify its performance when applied to three nominal optical observation scenarios: 1) a GEO target observed for 1 hour, 2) a HEO target observed for 2 hours, and 3) a LEO target observed for 10 minutes. In each case, the El Segundo station provides optical measurements. The algorithm is given no information about the true orbit, and the object is to estimate the orbit as accurately as possible.

Table 4: Nominal Arc Optical IOD Performance

Metric	**GEO**	**HEO**	**LEO**
Truth Cost	24.9076	59.6165	7.7873
Constrained Estimate Cost	29.1779	62.7525	75.6079
Unconstrained Estimate Cost	23.9769	57.8321	2.3996
Batch 1-σ Position Uncertainty [km]	57.893	243.721	0.103
Constrained Position Error [km]	132.741	348.453	0.196
Unconstrained Position Error [km]	24.138	18.921	0.039
Batch 1-σ Velocity Uncertainty [m/s]	4.575	3.241	0.384
Constrained Velocity Error [m/s]	10.364	3.623	0.327
Unconstrained Velocity Error [m/s]	1.898	2.418	0.076

The results of applying this optical IOD method to these scenarios for a single trial with random error are described in Table 4. These results provide the cost function values on the truth trajectory in addition to the estimates that are

obtained through the constrained and unconstrained versions of this algorithm. In each case, the cost function value of the constrained solution is larger than truth and unconstrained solution is smaller than truth. This is expected behavior for the residual-unconstrained case because this state estimate was designed to minimize the cost function with no residual constraints. As a result, its cost function value should be even smaller than the value associated with the truth trajectory. This implies that the residual-unconstrained state estimate produces a trajectory that fits the measurements better than truth. The residual-constrained solution will always produce a cost function value greater than or equal to the residual-unconstrained cost. This is because the process minimizes the same cost function except with constraints, so it can never produce a smaller cost. In other words, the residual-constrained state estimate is guaranteed to never fit the measurements better than the residual-unconstrained solution. This is highlighted by the LEO case, which has a constrained cost function evaluation that is almost ten times larger than truth while the unconstrained solution has a cost less than a third of that on the truth trajectory. In the zero-noise case all three cost functions would nominally evaluate to zero and the state estimates would be equal to truth, though the PCE approximations would create some amount of numerical error.

This difference between the residual-constrained and unconstrained methods is underlined by the position and velocity error results that are provided. The position and velocity errors are generated at epoch by comparing the estimates to truth. The uncertainty values by computing the maximum likelihood covariance matrix using the truth trajectory as the nominal trajectory for computing measurement partials. In each case the position error of the unconstrained method is at least five times more accurate than the constrained solution (almost 20 times better in the HEO case). Velocity results show a marked improvement as well. The most important result is that the unconstrained position and velocity errors are on the order of the computed uncertainty for each scenario. This tends to indicate that the obtained state errors are on the order of the Maximum Likelihood uncertainty, which is the design of this optical IOD algorithm. It is left as future work to conduct a full Monte Carlo simulation to demonstrate that the state errors are distributed as Maximum Likelihood estimate error should be.

The results in Fig. 3 validate that the position error remains bounded across the observation gap as well. This indicates that the estimates are a good fit for all measurements, without being biased toward the measurements at any particular portion of the data arc.

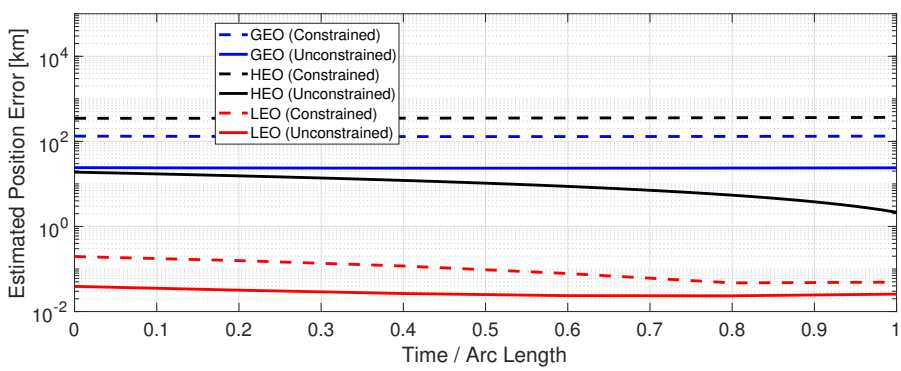

Figure 3: Position error dynamics across observation gap for each nominal arc length IOD case.

The results from these nominal scenarios indicate that this new optical IOD algorithm is working as designed. The residual-unconstrained algorithm is able to converge to cost value that is less than the cost associated with true orbit, which indicates that the algorithm is finding a solution that minimizes the cost function. The residual-unconstrained state estimate error was also on the same order as the Maximum Likelihood uncertainty, which is how the algorithm should behave. Overall, the state errors associated with the residual-constrained solution are larger than the residual-unconstrained solution (significantly larger at times), which indicates the effect that residual constraints can have on the IOD process even when all measurements are included in the estimate. In the next simulations, we analyze how this algorithm performs in short-arc scenarios with varying levels of measurement information.

4.2. GEO Short-Arc Performance with Multiple Stations and Measurement Types

One of the biggest constraints of typical IOD methods is that they are designed to work with only one observation type, and some methods do not even allow for more than one station location. This limits methods from taking advantage of information that can make a scenario far more observable. This is especially an issue in low-observability scenarios like short-arc IOD and optical IOD on highly eccentric orbits. Our method was designed to use multiple measurement types from multiple stations. It just requires at least two measurements in the observation arc (at two different times) that give a line of sight from the station to the target. In the following simulations, we investigate how multiple stations and different measurement types improve the estimate for a short-arc GEO optical IOD case (30 minute observation window). These cases include: 1) the standard method of using optical only observation with a single station (El Segundo), 2) optical only observations with all three stations, 3) optical and radar observations using only the El Segundo location, and 4) optical observations from all three stations with radar observation from the El Segundo station.

Table 5: Multiple Station and Measurement Type Optical IOD Performance for Short-Arc GEO Scenario

Metric	**Standard**	**Multiple Station**	**Range**	**Multiple Station + Range**
Truth Cost	11.2586	43.7598	21.6042	47.3897
Constrained Estimate Cost	14.9921	52.9659	29.6763	108.3172
Unconstrained Estimate Cost	10.5538	41.6042	18.2044	45.6606
Batch 1-σ Position Uncertainty [km]	314.784	3.224	0.200	0.125
Constrained Position Error [km]	189.081	6.0497	0.453	0.718
Unconstrained Position Error [km]	191.415	2.6547	0.176	0.123
Batch 1-σ Velocity Uncertainty [m/s]	23.912	2.101	0.179	0.115
Constrained Velocity Error [m/s]	14.582	1.095	0.474	0.163
Unconstrained Velocity Error [m/s]	14.318	2.267	0.225	0.075

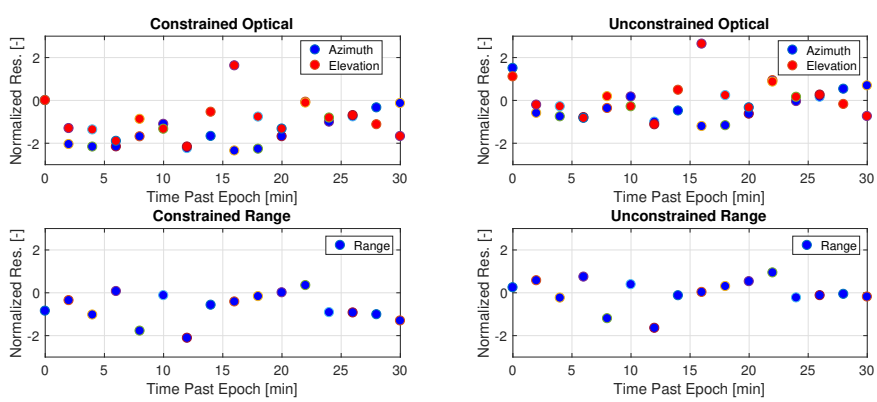

Figure 4: Postfit normalized residuals from the El Segundo site for the Three Optical Stations and 1 Range Station scenario.

The results from sample runs of these four scenarios are summarized in Table 5. In each scenario, the unconstrained solution is able to find an optimal estimate that provides a lower cost than truth. Though the constrained cost values are larger than that of the unconstrained, we still see a significant increase in the accuracy of its estimates when more measurement information is provided. The single station optical only IOD results for the short-arc GEO cases yielded errors of about 200 km. Adding two extra stations reduces that to about 6 km. Using one station with optical and range measurements reduces the error to about 450 m. Three optical stations with one range station yields a result that is of the same order of magnitude, which indicates that range measurements chiefly drive the uncertainty in the estimate, while the optical stations just provide added observability to the scenario. Unconstrained results are even better, with the smallest position errors just over 100 m. What is most important is that the unconstrained errors are of the same order of magnitude as the uncertainty in the problem, which indicates we get batch processor-like

performance from this algorithm without iterating or requiring the user to provide an initial guess of the true orbit. This claim is backed up by the sample scaled residual results in Fig. 4. The constrained residuals clearly are biased - especially the optical ones. However, the unconstrained residuals are well-balanced around zero. This is expected behavior from an unconstrained Maximum Likelihood estimator. Again, it is left as future work to run a full Monte Carlo analysis to verify that the unconstrained state estimate errors are distirbuted like Maximum Likelihood estimate should be when measurement errors are varied.

Decreasing the error from 200 km to 100 m provides a significant estimation improvement, which demonstrates the effect ignored measurements can have. Our algorithm allows for any number of stations and any type of measurement. Currently, it requires optical measurements at two different times. In the following section, we summarize the developments made in this paper, and indicate paths forward in making this algorithm even more robust and adaptable.

5. CONCLUSIONS AND FUTURE WORK

The optical IOD algorithm developed in this paper uses a Gooding-like formulation with a Polynomial Chaos algorithm at its heart to estimate a target's state given observations without needing an initial guess of the system's state. It was devised to remove some constraints that are typical in the optical IOD process. These constraints, how our method removes them, and the expected benefit of each constraint's removal are summarized below.

1. Under-Informed Optical IOD: Classical optical IOD methods only use three angle measurement pairs from a given observation arc, and they ignore all other pieces of information.
    - Solution: A measurement-based Maximum Likelihood cost function is minimized rather than just the residual of a single interior measurement. This cost function allows $N \geq 3$ measurements from any number of different stations and non-angle measurements may be included too.
    - Benefit: The final estimate is based on all available pieces of information rather than the minimum number of needed measurements. By allowing for other types of measurements (e.g., range) the problem can be more observable, which results in a more reliable state estimate process.
2. Zero Measurement Residual Constraint: The initial and final angle measurement pairs are assumed to be perfect so that the initial and final measurement residuals are zero.
    - Solution: A PCE-based method is used to evaluate the IOD cost function while varying range values based on an admissible region, and varying initial and final measurement errors so that the zero-residual constraint is removed.
    - Benefit: All realistic measurements inherently contain some amount of error, so this constraint can really hurt the solution in cases where that error is non-negligible. By removing this constraint, we are able to generate a Maximum Likelihood estimate.
3. Keplerian Dynamics Constraint: The orbiting target is assumed to be guided only by Keplerian dynamics.
    - Solution: Trajectories in this optical IOD process are generated by solving a TPBVP (i.e., given two positions with no velocity information compute the full state at the initial time). For Keplerian dynamics Lambert's method solves this TPBVP, and other perturbations can be included by using the solution from Lambert's problem as an initial guess in a single shooter algorithm. For cases where multiple solutions exist, the solution that yields the smallest cost function evaluation is used.
    - Benefit: Assuming Keplerian dynamics artificially constrains an IOD timespan because the assumption is inappropriate over larger timespans. Allowing for non-Keplerian dynamics supports and enables a more general and accurate solution.
4. Usage of Numerical Derivatives: Some classical optical IOD methods use numerical derivatives to generate a solution.
    - Solution: A PCE is an analytical mapping between stochastic inputs and resulting outputs, so its derivatives can be analytically derived.

- Benefit: In low-observability cases derivatives can be quite sensitive to system inputs. A poorly computed numerical derivative can result in the algorithm diverging. Analytical derivatives require no tuning, so they are inherently more stable.

Our method is designed to return a Maximum Likelihood estimate based on only information from measurements and dynamics. This is similar to how a batch processor performs except our method does not require an initial guess to linearize around. The only additional pieces of information that the user needs to provide are bounds on the size of the possible orbit and some tuning parameters for the zoom and PCE methods (if the default parameters are not desired). It was shown that the user could input very large bounds on the orbit size (lowest apogee at the beginning of LEO and highest apogee in translunar space), and still converge near the true orbit when sufficient information is provided (i.e., the system is observable).

Simulation results confirmed that the algorithm works as designed. When applied to GEO, HEO, and LEO orbits the unconstrained estimates were on the order of uncertainty in the system, which is the best that an algorithm could do. Specific analysis of short-arc GEO scenarios demonstrated the impact that extra stations and added measurement types can have. Errors reduced from about 200 km with a single optical station to below 150 m with multiple optical stations and a range station when both scenarios only have 30 minutes of tracking data. These results also show that the unconstrained estimate provides error on the order of the uncertainty of the problem. This is symptomatic of the algorithm's Maximum Likelihood formulation. Like a batch processor, this algorithm takes in just measurements and is able to generate an unconstrained, optimal estimate. Unlike a batch estimator, this algorithm does not require an initial guess of the true orbit.

Moving forward, we intend to further evaluate the existing algorithm by testing it in additional scenarios, identifying appropriate values for all tuning parameters, and testing its robustness in more edge cases (e.g., short observation arcs, highly elliptical orbits, highly perturbed orbits, etc.). We will specifcally focus on a complete study comparing this new method to the existing classical optical IOD methods, especially in scenarios that are difficult for typical optical IOD methods. Furthermore, we will also adapt the algorithm while iteratively striving to meet the following goal.

- Goal: A generalized IOD algorithm that takes in all available pieces of information regardless of measurement type and timing (with no initial guess of the state) and outputs an unconstrained Maximum Likelihood estimate (that may include additional parameters beyond the target's translational state) for all systems that are sufficiently observable.

To accomplish this goal, the IOD framework developed in this paper must be adapted to allow for IOD without optical measurements being required. Additionally, the method must be adapted to allow for estimation of other pertinent parameters such as measurement and dynamic biasing. Either type of biasing can essentially prevent classical IOD methods from ever converging; however, if they are considered as varied parameters in this PCE framework, then their deterministic effects can be effectively removed via estimation.

Beyond generalizing the algorithm, it is also desirable to make the algorithm more efficient and automated. It will be adapted to require less user input and will be optimized to reduce computational load. The current algorithm requires the user to place bounds on the size of the orbit, but it would be preferable if this were not required. Instead an algorithm could be devised which identifies the orbital regime of the target via measurements alone. To improve efficiency a full Constrained Admissible Region (CAR) approach should be employed to ensure the system is not evaluating unrealistic trajectories. Beyond this, the computational efficiency of the algorithm will be analyzed and improved by parallelizing sections of the code where possible.

References

[1] A. V. Shaeperkoetter. *A Comprehensive Comparison Between Angles-Only Initial Orbit Determination Techniques*. PhD thesis, Texas A & M University, College Station, TX, 2011.

[2] T. A. Henderson, D. Mortari, and J. Davis. Modifications to the Gooding algorithm for angles-only initial orbit determination. In *20th AAS/AIAA Space Flight Mechanics Meeting*, 2010.

[3] K. F. Gauss. *Theory of Motion of the Heavenly Bodies Moving About the Sun in Conic Sections: A Translation of Theoria Motus*. Dover Phoenix Editions, 2004 (1809).

[4] D. A. Vallado. *Fundamentals of Astrodynamics and Applications*. Microcosm Press, 3rd edition, 2007.

[5] S. Herrick. *Astrodynamics: Orbit Determination, Space Navigation, Celestial Mechanics - Vol. 1*. Van Nostrand Reinhold Co., 1971.

[6] L. G. Taff. The resurrection of Laplace's method of initial orbit determination. Technical Report 628, Lincoln Laboratory, 1983.

[7] P. R. Escobal. *Methods of Orbit Determination*. John Wiley and Sons Inc., 1965.

[8] R. H. Gooding. A new procedure for orbit determination based on three lines of sight (angles only). Technical Report 93004, Defense Research Agency, 1993.

[9] A. Milani, G. Gronchi, M. Vitturi, and Z. Knezevic. Orbit determination with very short arcs. I: Admissible regions. *Celestial Mechanics and Dynamical Astronomy*, 90:57–85, 2004.

[10] K. J. DeMars, M. K. Jah, and P. W. Schumacher. Initial orbit determination using short-arc angle and angle rate data. *IEEE Transactions on Aerospace and Electronic Systems*, 48:2628–2637, 2012.

[11] K. Fujimoto and D. J. Scheeres. Short-arc correlation and initial orbit determination for space-based observations. In *Advanced Maui Optical and Space Surveillance Technologies Conference*, 2011.

[12] D. Xiu and G. E. Karniadakis. The Wiener-Askey polynomial choas for stochastic differential equations. *SIAM Journal of Scientfic Computing*, 24:619–644, 2002.

[13] B. A. Jones and A. Doostan. Satellite collision probability estimation using polynomial chaos expansions. *Advances in Space Research*, 52:1860–1875, 2013.

[14] B. A. Jones, N. Parrish, and A. Doostan. Post-maneuver collision probability estimation using sparse polynomial chaos expansions. *J. Guidance, Control, and Dynamics*, 85(8):1425–1437, 2015.

[15] H. W. Kuhn and A. W. Tucker. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, 1951.