Article

Angles-Only Initial Orbit Determination via Multivariate Gaussian Process Regression

David Schwab [1,*] **, Puneet Singla** [1] **and Sean O'Rourke** [2]

1 Department of Aerospace Engineering, The Pennsylvania State University, University Park, PA 16802, USA; psingla@psu.edu
2 Sensors Directorate, Air Force Research Laboratory, Wright-Patterson AFB, OH 45433, USA; sean.orourke.3@us.af.mil
* Correspondence: dvs5558@psu.edu

Abstract: Vital for Space Situational Awareness, Initial Orbit Determination (IOD) may be used to initialize object tracking and associate observations with a tracked satellite. Classical IOD algorithms provide only a point solution and are sensitive to noisy measurements and to certain target-observer geometry. This work examines the ability of a Multivariate GPR (MV-GPR) to accurately perform IOD and quantify the associated uncertainty. Given perfect test inputs, MV-GPR performs comparably to a simpler multitask learning GPR algorithm and the classical Gauss–Gibbs IOD in terms of prediction accuracy. It significantly outperforms the multitask learning GPR algorithm in uncertainty quantification due to the direct handling of output dimension correlations. A moment-matching algorithm provides an analytic solution to the input noise problem under certain assumptions. The algorithm is adapted to the MV-GPR formulation and shown to be an effective tool to accurately quantify the added input uncertainty. This work shows that the MV-GPR can provide a viable solution with quantified uncertainty which is robust to observation noise and traditionally challenging orbit-observer geometries.

Keywords: Space Situational Awareness; Initial Orbit Determination; Gaussian Process Regression; machine learning

1. Introduction

As access to space becomes more readily available, the need for robust Space Situational Awareness (SSA) increases. Space-object tracking, an integral aspect of SSA, is performed by a sequential filtering algorithm which must be initialized. To initialize these orbits, tools collectively called Initial Orbit Determination (IOD) algorithms are utilized to provide an estimate of a previously unknown orbit. Furthermore, these tracks (orbits) initialized by IOD algorithms can be used for data association purposes as shown in [1]. Although IOD algorithms have been developed to initiate orbit-tracks from radar as well as optical sensors, angles-only IOD algorithms represent an area of particular interest due to the limited availability of deep space radars.

The seminal angles-only IOD algorithms include Laplace's method, Gauss' method, Gooding's method, and Double-R iteration (all detailed in [2]). Laplace's method is often intractable for Earth-based satellites, but does provide a viable solution for heliocentric orbits. Gauss' method may be used for Earth-based space objects, but at a limited angular separation between observations. Double-R iteration and Gooding's method allow for longer time periods between observations, but require a guess to initialize the iteration. These classical angles-only IOD algorithms are sensitive to noise and to certain observer-target geometrys as shown in both our own previous work [3–5] and also various other IOD studies [6–11]. All angles-only IOD algorithms mentioned utilize the observer-target geometry and assumptions defined from the Kepler problem to define a point solution, thus lacking any uncertainty information. The addition of high levels of noise essentially

violates these relationships and causes severe degradation in accuracy. Additionally, a well-known failure case for these algorithms is known as a coplanar orbit. This occurs when the observation site lies close to the target's orbital plane causing a singularity in many of the underlying algorithms. Though each of the IOD algorithms have a different formulation, all leverage the observer-target geometry; therefore, accuracy may severely degrade near coplanar orbits. In summary, many existing IOD algorithms do not inherently provide uncertainty information, are sensitive to sensor noise, and degrade in coplanar observer orbit geometries.

There exists an avenue to address the shortcomings of the classical IOD algorithms using a supervised learning method known as Gaussian Process Regression (GPR) as first proposed in our previous works [3,5]. The motivation of this paper is to examine the efficacy of GPR in providing an accurate solution with quantified uncertainty to the angles-only IOD problem both in general and in traditionally challenging observer-target sensing geometries.

While learning methods have been proposed to address the orbit determination problem, most methods focus on the range and range-rate orbit determination problem as in [12] or as a correction term to a sequential filter. GPR itself has even been introduced in the orbit determination space by Peng and Bai [13] who fuse the model estimate of a Kalman filter with the a GP model to essentially correct the orbit determination output of the sequential filter.

GPR methods outperform many regression techniques [14], and Rasmussen and Williams [15] demonstrate GPR's ability to learn the complex, nonlinear inverse dynamics of a robotic arm with seven degrees-of-freedom. In its basic formulation, GPR assumes either a scalar output or independence between output dimensions. Many formulations exist to expand this method to vector-valued processes, as surveyed by Liu et al. [16], but the method most pertinent to this work vectorizes the process in both the data and output dimension [17–19]. Chen et al. [20] shows that this method is equivalent to defining a matrix-variate distribution. Additionally, GPR assumes deterministic input in the original formulation. In this work, training is performed via simulations, so perfect inputs may be assumed for training. In a real-world scenario, the inputs for inference are measurement-based with a defined uncertainty. Given GPR in inherently nonlinear, input noise cannot be directly mapped to the regression output. The work of Candela et al. [21] shows how one may exactly define the first and second statistical moments of the unknown output distribution. This defines a way to simultaneously define an initial orbit with accurate uncertainty quantification given uncertain inputs. This work builds upon our previous works [3,5,22] in showcasing the ability of Multivariate GPR (MV-GPR) to provide a solution to the IOD problem while providing accurate uncertainty characterization given uncertainty inputs at inference. The objective is to train the MV-GPR from perfect simulations and then exploit the work of Candela et al. [21] to accurately characterize the errors associated with IOD estimates while taking into account for measurement errors.

The structure of the paper is as follows: first, an overview of classical IOD algorithms is presented, followed by the review of conventional GPR process. Section 4 presents the MV-GPR formulation, followed by definition of model hyperparameters in Section 5 and testing with noisy data in Section 6. Section 7 provides results from various numerical experiments performed to assess the performance of MV-GPR for the IOD problem, and Section 8 provides the concluding remarks.

2. Classical Initial Orbit Determination

While many algorithms exist which provide a solution to the angle-only IOD problem, this work will focus on one of the oldest and well documented methods known as *Gauss' method* to provide a benchmark for the GPR-based IOD method. Like most IOD algorithms, Gauss' method leverages the geometry of the observer and target at three measurement times (t_1, t_2, t_3), defined by

$$\mathbf{r}_i = \rho_i \hat{\mathbf{L}}_i + \mathbf{r}_{site,i}, \qquad i = 1, 2, 3, \tag{1}$$

where $\mathbf{r}$ denotes the spacecraft position vector, ρ denotes the range and $\hat{\mathbf{L}}$ denotes the unit vector from observer to target, $\mathbf{r}_{site}$ denotes the observer's position vector, and i indexes the time of measurement. These lines-of-sight may be determined by topocentric measurements of the right-ascension (RA) and declination (DEC) such that

$$\hat{\mathbf{L}}_i = \begin{bmatrix} \cos(DEC_i)\cos(RA_i) \\ \cos(DEC_i)\sin(RA_i) \\ \sin(DEC_i) \end{bmatrix}. \tag{2}$$

The geometry of defining Equations (1) and (2) is shown in Figure 1. With $\hat{\mathbf{L}}_i$ and $\mathbf{r}_{site,i}$ known at each measurement, the problem becomes estimating the range in order to determine the radius vector at each measurement. Given the observer's position is well defined, three linearly independent line-of-sight measurements, and the times of each observation, Gauss' method provides an estimate for the position at the three observations by posing an 8-th order polynomial of $||\mathbf{r}_2||$. The algorithm takes advantage of the planar nature of orbits and the method of Lagrange coefficients for propagating orbits as a linear combination of position and velocity, commonly referred to as the f and g solution. Given an estimate for the three position vectors, a prediction of the instantaneous $\mathbf{v}_2$ can be defined utilizing the approximate the f and g solution. With $\mathbf{r}_2$ and $\mathbf{v}_2$ now estimated, the target's Cartesian state is fully defined.
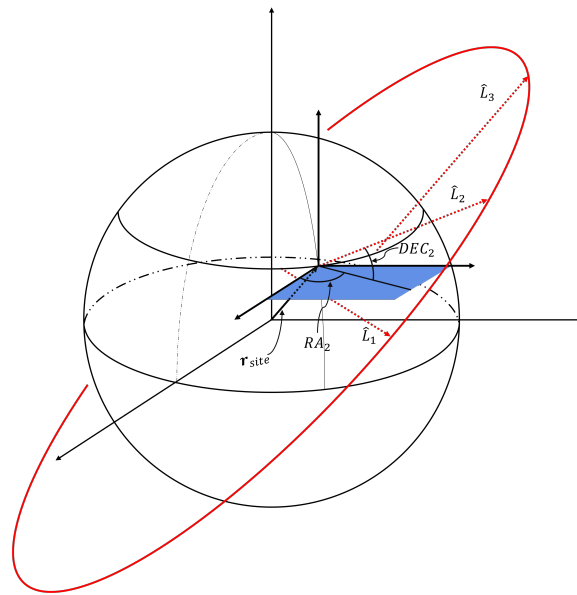
Figure 1. Geometry of angles-only Initial Orbit Determination.

In more recent history, methods to increase the accuracy of the method have been defined. Most notably, Gauss' method is combined with Gibbs' or the Herrick–Gibbs methods to provide a more accurate $\mathbf{v}_2$ measure [2]. Both Gibbs' and the Herrick–Gibbs methods are IOD algorithms that start with the premise of three known position vectors and times of measurement then solve for the middle velocity. Gibbs' method is a geometric approach while Herrick–Gibbs essentially represents a series expansion. Therefore, the Herrick–Gibbs algorithm is the best for measurement separations around 1° or less, while Gibbs' method is used for larger angular separations as stated in [2]. This work uses the combined Gauss–Gibbs method as the IOD benchmark. Additionally, Curtis in [23] defines an iterative scheme using universal variables which refine the "initial pass" estimated by Gauss' method to determine the exact solution assuming perfect measurements. This will be referred to as an iterative Gauss–Gibbs method throughout the rest of this paper.

3. Standard Gaussian Process Regression (GPR)

As defined in [15], *a Gaussian Process is a collection of random variables, any finite number of which have a joint Gaussian distribution.* Therefore, analogous to the Gaussian distribution for random variables, the GP is completely defined by its mean and covariance functions:

$$f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^m$. The mean and covariance functions are defined using the expectation, such that

$$\mu(\mathbf{x}) = E[f(\mathbf{x})]$$
$$k(\mathbf{x}, \mathbf{x}') = cov(f(\mathbf{x}), f(\mathbf{x}')) = E[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(x'))]. \tag{4}$$

By defining functions as simply an infinite-dimensional vector, the GP can be compared to a Gaussian distribution with an infinitely-sized mean vector and covariance matrix. Therefore, every subset of the distribution is defined by the full distribution due to the marginalization property of Gaussian processes [15]. Now, let the model be defined as

$$y(\mathbf{x}) = f(\mathbf{x}) + \epsilon \tag{5}$$

where $y(\mathbf{x}) \in \mathbb{R}$ and $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$ represents model noise. Let there be a finite data set $\mathcal{D} = \{(\mathbf{x}_i, y_i) | i = 1, ..., N\}$ in the domain and range of Equation (5), then the distribution of $\mathbf{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix}^T$ is defined as a joint Gaussian distribution,

$$\mathbf{y} \sim \mathcal{N}_N(\boldsymbol{\mu}(\boldsymbol{\mathcal{X}}), \mathbf{K}(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}}) + \sigma_n^2 \mathbf{I}_N). \tag{6}$$

$\mathbf{K}$ is the relevant submatrix defined from the covariance function of the GP in Equation (5) such that

$$\mathbf{K}(\mathbf{X}, \mathbf{X}') = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1') & k(\mathbf{x}_1, \mathbf{x}_2') & \cdots & k(\mathbf{x}_1, \mathbf{x}_q') \\ k(\mathbf{x}_2, \mathbf{x}_1') & k(\mathbf{x}_2, \mathbf{x}_2') & \cdots & k(\mathbf{x}_2, \mathbf{x}_q') \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}_p, \mathbf{x}_1') & k(\mathbf{x}_p, \mathbf{x}_2') & \cdots & k(\mathbf{x}_p, \mathbf{x}_q') \end{bmatrix}, \tag{7}$$

where $\mathbf{X}$ is any set of p vectors $\mathbf{x}$ and $\mathbf{X}'$ is any set of q vectors $\mathbf{x}'$.

While various kernel functions exist, we will focus on the commonly used *Squared Exponential* (SE) covariance function:

$$k_{SE}(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^T \mathbf{P}^{-1}(\mathbf{x} - \mathbf{x}')\right). \tag{8}$$

The matrix $\mathbf{P}$ is often a diagonal matrix which encodes what is known as Automatic Relevance Determination (ARD). Together, $\mathbf{P}$ and σ_f^2 make up the hyperparameters of the SE kernel. Given that a specific form of the covariance kernel is specified, the next step involves determining the optimal hyperparameters, γ, to define the model, which may also include the model noise, σ_n^2.

Now, introducing a single test point, $\mathbf{x}_*$, the marginalization property may again be invoked such that $\mathbf{y}$ and y_* are defined from the same GP and therefore

$$\begin{bmatrix} \mathbf{y} \\ y_* \end{bmatrix} \sim \mathcal{N}_{N+1}\left(\begin{bmatrix} \boldsymbol{\mu}(\boldsymbol{\mathcal{X}}) \\ \mu(\mathbf{x}_*) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}}) + \sigma_n^2 \mathbf{I}_N & \mathbf{K}(\boldsymbol{\mathcal{X}}, \mathbf{x}_*) \\ \mathbf{K}(\mathbf{x}_*, \boldsymbol{\mathcal{X}}) & K(\mathbf{x}_*, \mathbf{x}_*) + \sigma_n^2 \end{bmatrix}\right). \tag{9}$$

The predictive distribution of y_* consists of conditioning the prior distribution on the training observations, which returns a Gaussian defined as

$$y_* \,|\mathcal{D}, \mathbf{x}_* \sim \mathcal{N}\left(\hat{\mu}_{y_*}, \hat{\Sigma}_{y_*}\right)$$

$$\hat{\mu}_{y_*} = \mu(\mathbf{x}_*) + \mathbf{K}(\mathbf{x}_*, \mathcal{X})\left(\mathbf{K}(\mathcal{X}, \mathcal{X}) + \sigma_n^2 \mathbf{I}_N\right)^{-1}[\mathbf{y} - \mu(\mathcal{X})] \tag{10}$$

$$\hat{\sigma}_{y_*}^2 = K(\mathcal{X}_*, \mathcal{X}_*) + \sigma_n^2 - \mathbf{K}(\mathcal{X}_*, \mathcal{X})\left(\mathbf{K}(\mathcal{X}, \mathcal{X}) + \sigma_n^2 \mathbf{I}_N\right)^{-1}\mathbf{K}(\mathcal{X}, \mathcal{X}_*).$$

4. Multivariate Gaussian Process Regression

In the formulation above, it is assumed that the output dimension is a scalar. There are many ways to expand the GPR framework to address multioutput regression problems, such that $\mathbf{y}(\mathbf{x}) \in \mathbb{R}^D$. One can pose the multidimensional output problem as one of multitask learning. By definition, multitask learning is learning one model to represent multiple data sets. We can define the training inputs and the *j*-th dimension as representing their own data sets, $\mathcal{D}_j$, where $j = 1, \ldots, D$. In this context, the multitask learning problem will define a GPR model for each of the output dimensions, but the models share the same hyperparameters. This method is defined as *ind-GPR* throughout the rest of this work. The downside is that ind-GPR assumes mutual independence between the outputs, and thus limits the ability to account for output correlation during training.

Alternatively, a method to address this issue is to vectorize the output [17–19] such that

$$\mathbf{y} = [y_{1,1}, \ldots, y_{1,D}, \ldots, y_{N,1}, \ldots, y_{N,D}]^T. \tag{11}$$

This reformulates the Multi-Output problem as a standard GPR method discussed in Section 3. As shown by Chen et al. [20], this may be accomplished equivocally by generalizing the Gaussian process to a vector-valued function such that

$$\mathbf{f}(\mathbf{x}) \sim \mathcal{MGP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'), \mathbf{B}). \tag{12}$$

Here, $k(\mathbf{x}, \mathbf{x}')$ encodes the covariance along the range of the *D*-vector GP and **B** encodes the covariance in the *m*-dimensions of the vector-valued process. Again, the marginalization property holds true. Therefore if there exists a finite data set, $\mathcal{D}_{MV} = \{(\mathbf{x}_i, \mathbf{y}_i) | i = 1, \ldots, N\}$, then the distribution of $\mathcal{Y} = \begin{bmatrix} \mathbf{y}_1^T & \mathbf{y}_2^T & \cdots & \mathbf{y}_N^T \end{bmatrix}^T \in \mathbb{R}^{N \times D}$ has a joint matrix-variate Gaussian distribution:

$$\mathcal{Y} \sim \mathcal{MN}_{N \times D}(\mathbf{M}, \mathbf{K}, \mathbf{B}), \tag{13}$$

where $\mathbf{M} \in \mathbb{R}^{N \times D}$, $\mathbf{K} \in \mathbb{R}^{N \times N}$, and $\mathbf{B} \in \mathbb{R}^{D \times D}$. As discussed in Chen et al. [20,24], a matrix-variate Gaussian distribution has a density of the form

$$p(\mathcal{Y}|\mathbf{M}, \mathbf{K}, \mathbf{B}) = (2\pi)^{-\frac{DN}{2}} |\mathbf{K}|^{-\frac{D}{2}} |\mathbf{B}|^{-\frac{N}{2}} \exp\left[-\frac{1}{2}\mathrm{tr}\left(\mathbf{B}^{-1}(\mathcal{Y} - \mathbf{M})^T \mathbf{K}^{-1}(\mathcal{Y} - \mathbf{M})\right)\right] \tag{14}$$

Additionally, the $N \times D$-matrix-variate normal distribution is may be mapped to a *ND*-multivariate normal distribution:

$$\mathcal{Y} \sim \mathcal{MN}_{N \times D}(\mathbf{M}, \mathbf{K}, \mathbf{B}) \iff \mathrm{vec}(\mathcal{Y}) \sim \mathcal{N}_{ND}(\mathrm{vec}(\mathbf{M}), \mathbf{K} \otimes \mathbf{B}). \tag{15}$$

Therefore, the work of Chen et al. [20] is similar to previous works [17–19] given that the matrix-variate distribution may be mapped to a multivariate one for normal distributions, as shown in Equation (15).

If a test point, $\mathbf{x}_*$, is introduced into this formulation, the marginalization property used in the standard GPR formulation may also be invoked for matrix-variate normal distributions [20,25] such that

$$\begin{bmatrix} \boldsymbol{\mathcal{Y}} \\ \mathbf{y}_*^T \end{bmatrix} \sim \mathcal{MN}_{(N+1)\times D}\left(\begin{bmatrix} \mathbf{M}(\boldsymbol{\mathcal{X}}) \\ \mathbf{M}(\mathbf{x}_*) \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}})+\sigma_n^2\mathbf{I}_N & \mathbf{K}(\boldsymbol{\mathcal{X}},\mathbf{x}_*) \\ \mathbf{K}(\mathbf{x}_*,\boldsymbol{\mathcal{X}}) & K(\mathbf{x}_*,\mathbf{x}_*)+\sigma_n^2 \end{bmatrix}, \mathbf{B} \right). \tag{16}$$

As a property of matrix normal distributions, the test output conditioned on the training data, $\mathcal{D} = \{\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}\}$, is defined as

$$\mathbf{y}_*^T \,|\, \mathcal{D}, \mathbf{x}_* \sim \mathcal{MN}_{1\times D}\left(\hat{\boldsymbol{\mu}}_{y_*}, \hat{\Sigma}_{y_*}, \mathbf{B} \right), \text{ where}$$

$$\hat{\boldsymbol{\mu}}_{y_*}^T = \mathbf{M}(\mathbf{x}_*) + \mathbf{K}(\mathbf{x}_*,\boldsymbol{\mathcal{X}})\left(\mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}})+\sigma_n^2\mathbf{I}_N \right)^{-1}[\boldsymbol{\mathcal{Y}} - \mathbf{M}(\boldsymbol{\mathcal{X}})], \tag{17}$$

$$\hat{\Sigma} = K(\mathbf{x}_*,\mathbf{x}_*) + \sigma_n^2 - \mathbf{K}(\mathbf{x}_*,\boldsymbol{\mathcal{X}})\left(\mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}})+\sigma_n^2\mathbf{I}_N \right)^{-1}\mathbf{K}(\boldsymbol{\mathcal{X}},\mathbf{x}_*).$$

Using Equation (15), the corresponding multivariate distribution is then defined as

$$\mathbf{y}_* \sim \mathcal{N}_D\left(\hat{\boldsymbol{\mu}}_{y_*}, \hat{\Sigma} \otimes \mathbf{B} \right). \tag{18}$$

This method is referred to as *MV-GPR* throughout the rest of this paper.

5. Defining Model Hyperparameters

Though not necessarily the case, the mean function is often taken to be zero as the mean of the posterior process will not be constrained to be zero [15]. See Chapter 2.7 of [15] for details on mean function formulation and definition. Therefore, we will assume a zero-mean prior for both ind- and MV-GPR methodologies. The posterior estimation will not be confined to be zero [15], and therefore an accurate GPR model may still be learned. Additionally, we use the square-exponential kernel as defined in Equation (8) as our covariance function. Therefore, the hyperparameters associated with both GPR algorithms include $\sigma_f^2 \in \mathcal{R}$, $\sigma_n^2 \in \mathcal{R}$, and $\text{diag}(\mathbf{P}) \in \mathcal{R}^m$. For ind-GPR, this results in a set of hyperparameters to optimize for such as

$$\gamma_{ind} = [\sigma_f^2, \sigma_n^2, \text{diag}(\mathbf{P})] \in \mathcal{R}^{m+2}. \tag{19}$$

By contrast, MV-GPR's hyperparameters also include the elements of the matrix $\mathbf{B}$. By definition, $\mathbf{B}$ must be a positive definite matrix since it defines the covariance between the dimensions of the vector-valued function $\mathbf{f}(\mathbf{x})$. The Cholesky decomposition for positive definite matrices allows $\mathbf{B}$ to be factored such that $\mathbf{B} = \mathbf{L_B}\mathbf{L_B}^T$, where $\mathbf{L_B}$ is a lower triangular matrix. Optimizing for the elements of $\mathbf{L_B}$ not only reduced the number of hyperparameters for MV-GPR, but also ensures $\mathbf{B}$ is positive definite. This adds an additional $\frac{1}{2}D(D+1)$ hyperparameters to optimize, thus giving

$$\gamma_{MV} = [\sigma_f^2, \sigma_n^2, \text{diag}(\mathbf{P}), \text{vec}(\mathbf{L_B})] \in \mathcal{R}^{\frac{1}{2}D(D+1)+m+2}. \tag{20}$$

Under the ind-GPR assumption, the cost function for optimization of the hyperparameters is accomplished using the negative-log-likelihood

$$\mathcal{NLL}_{ind} = \sum_{j=1}^{D}\left(-\frac{1}{2}\mathbf{y}_j^T\mathbf{K}_y^{-1}\mathbf{y}_j - \frac{1}{2}\ln\det(\mathbf{K}_y) - \frac{N}{2}\ln 2\pi \right). \tag{21}$$

The MV-GPR algorithm directly uses its negative-log-likelihood as its cost function such that

$$\mathcal{NLL}_{MV} = \frac{mN}{2}\ln(2\pi) + \frac{m}{2}\ln|\mathbf{K}_y| + \frac{N}{2}\ln|\mathbf{B}| + \frac{1}{2}\text{trace}\left(\mathbf{K}_y^{-1}\boldsymbol{\mathcal{Y}}\mathbf{B}^{-1}\boldsymbol{\mathcal{Y}}^T \right). \tag{22}$$

The complexity in the training for both methods and the prediction is dominated by the inverse $\mathbf{K}_y = (\mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}})+\sigma_n^2\mathbf{I}_N)^{-1}$, assuming $N >> m$. There is no guarantee that

optimization for either formulation will not suffer from local optima. As stated in [15], the local optima are not often devastating for less complex covariance functions and will provide differing interpretations of the data. For instance, let us state that a given problem has two optima: (1) an optimum with high σ_n^2 and a high length-scale and (2) an optimum with the low σ_n^2 and a low length-scale. Case (1) would correspond to a very smooth mean curve with large, smooth uncertainty bounds. Case (2) would result in a mean curve that passes much closer to the individual training points and reflect uncertainty bounds which shrink significantly near those points.

6. Estimate Given Random Test Inputs

As formulated, the GPR algorithm assumes perfect inputs into the regression model. This is not often the case in engineering systems. In this study, perfect inputs, $\mathcal{X}$, and perfect outputs, $\mathcal{Y}$, generated from Equations (1) and (2) are used during training. In real world scenarios, only measurements of the input are available at inference, such that

$$\mathbf{x}_* \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right). \tag{23}$$

Given that GPR is a nonlinear function in $\mathbf{x}_*$, the true predictive distribution of the GPR output will not be Gaussian. However, it may be approximated by a Gaussian with the same mean and covariance as the true distribution, i.e., moment matching. The following moment matching derivation stems from the method defined by Candela et al. [21] and presented in chapter 2 of Deisenroth [26] with amendments due to the use of MV-GPR presented in Section 4.

The mean of the predicted distribution given these approximations is

$$\hat{\boldsymbol{\mu}}_{y_*} = \iint \mathbf{f}_* \mathrm{p}(\mathbf{f}_*, \mathbf{x}_*) \mathrm{d}(\mathbf{f}_*, \mathbf{x}_*) \tag{24}$$

where $\mathbf{f}_* = \mathbf{f}(\mathbf{x}_*)$. Given that $\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_*|\mathbf{x}_*] = \mathcal{Y}^T\left(\mathbf{K}(\mathcal{X}, \mathcal{X}) + \sigma_n^2 \mathbf{I}_N\right)^{-T} \mathbf{K}(\mathcal{X}, \mathbf{x}_*)$ via Equation (17) and utilizing the law of iterated expectations,

$$
\begin{aligned}
\hat{\boldsymbol{\mu}}_{y_*} &= \mathbb{E}_{\mathbf{x}_*}[\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_*|\mathbf{x}_*]] \\
&= \underbrace{\mathcal{Y}^T\left(\mathbf{K}(\mathcal{X}, \mathcal{X}) + \sigma_n^2 \mathbf{I}_N\right)^{-T}}_{\boldsymbol{\Phi}^T} \underbrace{\int \mathbf{K}(\mathcal{X}, \mathbf{x}_*)\mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right)\mathrm{d}\mathbf{x}_*}_{\mathbf{q}}
\end{aligned}
\tag{25}
$$

where $\mathbf{q} = [q_1, \ldots, q_N]^T \in \mathbb{R}^N$. Using the square-exponential covariance function in Equation (8) gives an analytic definition for q_i:

$$q_i = \sigma_f^2 \left| \boldsymbol{\Sigma}_{\mathbf{x}_*}\mathbf{P}^{-1} + \mathbf{I}_m \right|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}_*})^T(\boldsymbol{\Sigma}_{\mathbf{x}_*} + \mathbf{P})^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_{\mathbf{x}_*})\right]. \tag{26}$$

Similarly, the covariance may be defined using the law of total covariance such that

$$\mathrm{cov}\left(\mathbf{y}_*|\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right) = \mathbb{E}_{\mathbf{x}_*}\left[\mathrm{cov}\left(\mathbf{f}_{\mathbf{x}_*}|\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right)\right] + \mathrm{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*]). \tag{27}$$

Using the definition of $\mathrm{cov}\left(\mathbf{f}_{\mathbf{x}_*}|\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right)$ from Equation (17), properties of covariance, and the properties of the matrix trace,

$$
\begin{aligned}
\mathrm{cov}\left(\mathbf{y}_*|\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right) = {} & \left\{K(\mathbf{x}_*, \mathbf{x}_*) + \sigma_n^2 - \mathrm{tr}\left(\left(\mathbf{K}(\mathcal{X}, \mathcal{X}) + \sigma_n^2 \mathbf{I}_N\right)^{-1}\mathbf{Q}\right)\right\} \otimes \mathbf{B} \\
& + \left(\boldsymbol{\Phi}^T\mathbf{Q}\boldsymbol{\Phi} - \left(\boldsymbol{\Phi}^T\mathbf{q}\right)^T\boldsymbol{\Phi}^T\mathbf{q}\right).
\end{aligned}
\tag{28}
$$

The $\mathbf{Q} = \int \mathbf{K}(\mathcal{X}, \mathbf{x}_*)\mathbf{K}(\mathbf{x}_*, \mathcal{X})\mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{x}_*}, \boldsymbol{\Sigma}_{\mathbf{x}_*}\right)\mathrm{d}\mathbf{x}_*$ term may be defined analytically assuming Gaussian input noise and using the square-exponential covariance kernel such that

$$Q_{ij} = \frac{K(\mathbf{x}_i, \boldsymbol{\mu}_{\mathbf{x}_*})K(\mathbf{x}_j, \boldsymbol{\mu}_{\mathbf{x}_*})}{|2\boldsymbol{\Sigma}_{\mathbf{x}_*}\mathbf{P}^{-1} + \mathbf{I}|^{\frac{1}{2}}} \exp\left[(\mathbf{z}_{ij} - \boldsymbol{\mu}_{\mathbf{x}_*})^T (\boldsymbol{\Sigma}_{\mathbf{x}_*} + \frac{1}{2}\mathbf{P})^{-1}\boldsymbol{\Sigma}_{\mathbf{x}_*}\mathbf{P}^{-1}(\mathbf{z}_{ij} - \boldsymbol{\mu}_{\mathbf{x}_*})\right] \qquad (29)$$

with $\mathbf{z}_{ij} = \frac{1}{2}(\mathbf{x}_i + x_j)$. Chapter 2 of Deisenroth [26] shows additional details of this method and a numerically stable way to compute this matrix under a different multioutput GPR formulation. Due to numerical sensitivity of the noise aware MV-GPR method presented in Section 6, the method used in this work to address the numerical sensitivity is shown in Appendix A.

7. Numerical Simulations

7.1. Methodology

Often, statistical learning methods use real data for training purposes. In this work, data is simulated using the Keplerian solution to the two-body problem. The assumption of a Keplerian orbit is accurate for short intervals and is also an assumption present for most other IOD processes. A goal of this work is to study the utility of GPR in learning a solution to the IOD problem and benchmark the solution against the Gauss–Gibbs method. This work will utilize the Gauss–Gibbs method without iteration unless otherwise stated. Iterating on the Gauss–Gibbs solution will return the ground-truth to machine precision if given perfect RA and DEC measurements. The reasoning behind not including iteration is twofold. (1) The GPR method is non-iterative and so it also provides a "first pass" solution, therefore it is more informative to compare the GPR model error it to the well-studied Gauss–Gibbs "first pass" solution's error. (2) Iteration provides very limited increase in accuracy given noisy measurements from a similarly defined noise model as shown in Chapter 3 of [22].

With respect to the GPR algorithms, it is important to inspect their performance across challenging data sets and with respect to various training levels. Orbit samples are generated from ranges of classical orbital elements shown in Table 1, which are tailored directly for the purposes of this study. The semi-major axis (SMA) is the largest semidiameter of the orbital ellipse, and the eccentricity, e, uniquely characterizes the shape. The inclination, longitude of the ascending node, argument of periapsis, and true anomaly—i, Ω, ω, θ, respectively—are geometric relationships defined in Figure 2. For a Keplerian orbit, SMA, e, i, Ω, ω are constant for all time and define the orbit ellipse. θ varies with time and defines the angular position of the spacecraft within it's orbit. The given true anomaly in Table 1, θ_2, corresponds to the 2nd measurement time.

Table 1. Orbital regimes for each case used in GPR study.

Orbit Case	SMA	e	i (°)	Ω (°)	ω (°)	θ_2 (°)
Case A	[10,000,16,000]	[0.1, 0.3]	[35, 45]	[305, 315]	[15, 20]	[50, 58]
Case B	[6800,40,000]	[0.1, 0.3]	[35, 45]	[305, 315]	[15, 20]	[50, 58]
Case C	[6800,40,000]	[0.05, 0.85]	[35, 45]	[305, 315]	[15, 20]	[50, 58]
Case D	[6800,40,000]	[0.05, 0.85]	[20, 60]	[305, 315]	[15, 20]	[50, 58]
Case E	[6800,40,000]	[0.05, 0.85]	[20, 60]	[295, 325]	[15, 20]	[50, 58]
Case F	[6800,40,000]	[0.05, 0.85]	[20, 60]	[295, 325]	[10, 30]	[50, 58]
Case G	[10,000,16,000]	[0.1, 0.3]	[35, 45]	[−5, 5]	[15, 20]	[50, 58]
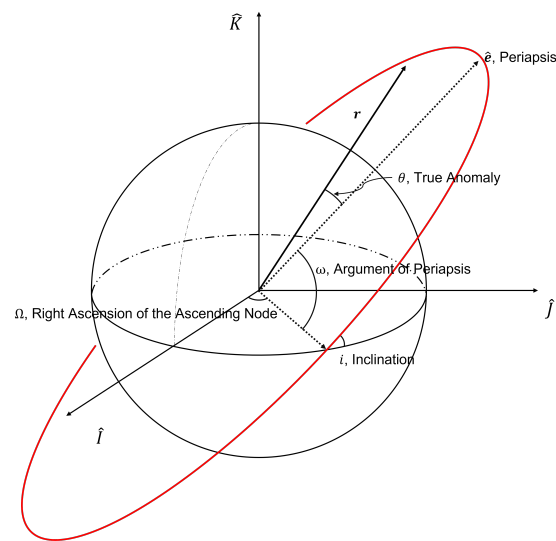
Figure 2. Geometry classical orbital elements.

Cases A-F in Table 1 progress in the level of difficulty for the GPR methods. Case G represents a nearly-coplanar orbit regime, which has been notoriously difficult for classical IOD algorithms. The Cartesian state at the 1st and 3rd measurement time are generated such that θ has regressed and processed, respectively, between 8° and 15°. This ensures that the orbits will stay within the effective angular separation for the Gauss–Gibbs IOD algorithm. The time-of-flight (TOF) measurements are defined using Kepler's equation and the line-of-sight vectors for each measurement are backed out using Equation (1). The observation site is at $\mathbf{r}_{site,1} = [6400, 0, 0]^T$ km in the *IJK*-frame and rotates with Earth's average spin rate. The number of training samples generated vary, but are specified throughout this work. There are 5000 testing orbits generated for each case in Table 1, with the resulting orbital arcs shown in Figure 3.
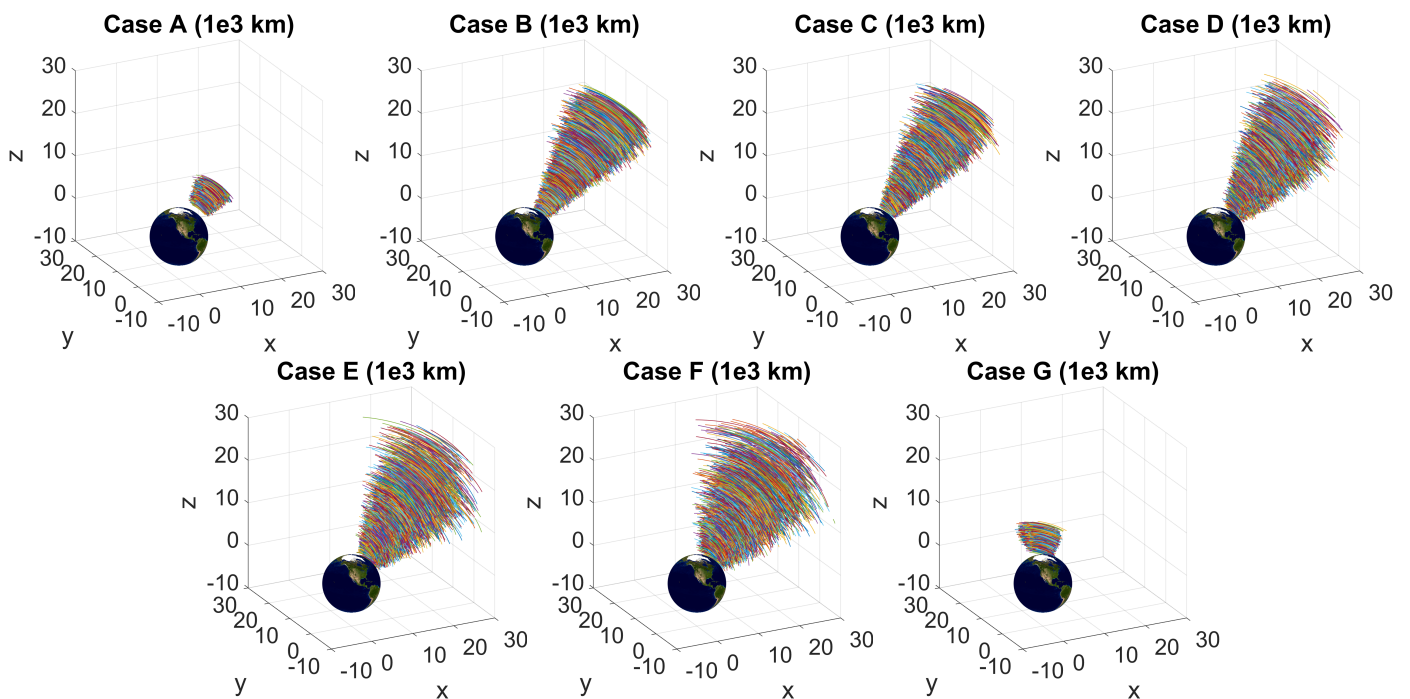
Figure 3. Orbital arcs for 5000 test orbits of the different orbital case regions.

7.1.1. Gpr Specifics

The inputs to the GPR models are defined as the RA and DEC at each measurement time along with the time-of-flight information between the first and second measurement, TOF_1, and the second and third measurement, TOF_3, such that

$$\mathbf{x} = \begin{bmatrix} RA_1 & RA_2 & RA_3 & DEC_1 & DEC_2 & DEC_3 & TOF_1 & TOF_3 \end{bmatrix}^T. \tag{30}$$

The output is the spacecraft state vector at the second measurement time such that

$$\mathbf{y} = \begin{bmatrix} r_1 & r_2 & r_3 & v_1 & v_2 & v_3 \end{bmatrix}^T, \tag{31}$$

where r_i and v_i represent the components of the Cartesian position and velocity, respectively, at the second measurement. As broken down in Table 2, the number of hyperparameters is $\gamma_{ind} \in \mathcal{R}^{10}$ and $\gamma_{ind} \in \mathcal{R}^{31}$.

Table 2. GPR parameters and their dimesnions in the IOD problem.

Input	Output	Hyperparamters				
		Shared				MV-GPR Specific
$\mathbf{x} \in \mathcal{R}^8$	$\mathbf{y} \in \mathcal{R}^6$	$\sigma_f^2 \in \mathcal{R}$	$\sigma_n^2 \in \mathcal{R}$		diag($\mathbf{P}$) $\in \mathcal{R}^8$	vec($\mathbf{L_B}$) $\in \mathcal{R}^{21}$

Normalization of training data is often essential for accurate optimization of machine learning algorithms. For both inputs and targets, *z*-score normalization is used such that the mean of the training data is subtracted and each element is divided by its standard deviation. Additionally, the Cartesian state and time-of-flight information is transformed into canonical units such that 1 $LU = 6400$ km and that $\mu_\oplus = 1$ prior to the z-score normalization, which proved to additionally increase performance. Testing data is normalized utilizing the same method, but with the mean and standard deviations of the training data.

The ind-GPR method is implemented using the API and optimization techniques provided by Rasmussen and Nickisch in [27], while the MV-GPR implemented in this work is based on the work of Chen [20] with modifications to the code provided in [28]. Both methods utilize three restarts in an effort to avoid the local minima problem while balancing the computational cost. Finally, the random input procedure in Section 6 is only implemented for the MV-GPR formulation and is referred to as noise aware (N.A.), while the original formulation shall be referred to as standard (STND).

In summary, training is conducted as follows. (1) The outputs—Cartesian position and velocity—are generated from the classical orbital elements reflected in Table 2. (2) The inputs—RA, DEC, and TOF—are backed out assuming Keplerian orbits and a perfectly spherical rotating earth-based observation site. (3) The outputs are written in canonical units both inputs and outputs are normalized using z-score normalization. (4) Hyperparameters in Table 2 are optimized using Equation (21) for ind-GPR and Equation (22) for MV-GPR as cost functions. The software packages [20,27] are used extensively for ind- and MV-GPR, respectively. Random initial guess and three restarts are used in an attempt to avoid local minimums.

Testing is conducted as follows. Steps (1) and (2) remain the same for training and testing on simulated data. (3) Inputs generated from the test points are transformed using the same shift and scaling generated by the z-score normalization of the training data. (4a) With the hyperparameters defined via optimization, Equations (10) and (17) are used to define the predicted state and state covariances for ind- and MV-GPR, respectively. (4b) Noise in the testing input may be accounted for using the Equation (25) for prediction mean and Equation (28) for estimated state covariance. The formulation in Appendix A overcomes some of the numerical issues that appeared when applying the noise-aware inference formulation presented in Section 6. (5) GPR estimates are transformed back into Cartesian space using the same shift and scaling used on the training outputs.

7.1.2. Noise Model

The noise model used in this work is additive Gaussian noise to the RA and DEC measurements, $\mathbf{x}_* = \mathbf{x}_t + \boldsymbol{\nu}$, where

$$\boldsymbol{\nu} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{x}_*}), \qquad \boldsymbol{\Sigma}_{\mathbf{x}_*} = \begin{bmatrix} \sigma_{x_*}^2 \mathbf{I}_{6\times6} & \mathbf{0}_{6\times2} \\ \mathbf{0}_{2\times6} & \mathbf{0}_{2\times2} \end{bmatrix} \tag{32}$$

and $\mathbf{x}_t$ is the ground truth measurements defined in Equation (30). Under these assumptions, it is assumed that the time of flight measurements are deterministic while the angular measurements have an normal distribution defined by $\sigma_{x_*}^2$. Given the formulation provided in Section 6 does not involve $\boldsymbol{\Sigma}_{x_*}^{-1}$, the singularity of the current covariance matrix does not affect the solution. If one were to use an equivalent algebraic expression involving the inverse matrix, then the deterministic and random elements of the input would have to be separated. For the rest of this work, one may assume $\sigma_{x_*} = 0.05°$ unless otherwise stated and 5 samples are generated for each of the 5000 original testing orbits—giving 25,000 total random noisy samples.

7.1.3. Metrics

Due to the wide range of orbits investigated, the measures used in this work include the absolute percent error (APE),

$$APE = \frac{|Truth - Prediction|}{|Truth|}, \tag{33}$$

and the median absolute percent error ($MdAPE_i$),

$$MdAPE_i = median\left(\left[\frac{|Truth_i - Prediction_1|}{|Truth_i|}, \cdots, \frac{|Truth_i - Prediction_{n_s}|}{|Truth_i|}\right]\right), \tag{34}$$

where n_s defines the number noisy samples and i indexes from which orbit the noisy samples were generated. Both of these measure error for each element in a given state and thus represents a vector of the same size as the state it describes. The average corresponds to the arithmetic mean of the elements in the APE or $MdAPE_i$ error vector.

Our first metric, the Malanobis distance is introduced:

$$d_{mahal} = \sqrt{(\mathbf{y} - \boldsymbol{\mu})^T \hat{\boldsymbol{\Sigma}}_{y_*}(\mathbf{y} - \boldsymbol{\mu})}, \tag{35}$$

where $\mathbf{y}_i$ is the ground-truth state and $\boldsymbol{\mu}_i$ and $\hat{\boldsymbol{\Sigma}}_{y_*,i}$ is the GPR-predicted mean and covariance, respectively. The Mahalanobis distance is introduced to measure the accuracy of the covariance estimate. A large number of high (i.e., greater than 3) Mahalanobis distances would show that the predicted covariance is vastly overestimating the precision of the estimate. A large number of very low (i.e., less than 1) Mahalanobis distances would show that the predicted covariance is "too cautious".

7.2. Accuracy

First, the accuracy of the GPR algorithms is benchmarked against the classical, non-iterative Gauss–Gibbs algorithm. Figure 4 compares box-plots of the APE for ind- and MV-GPR at 200, 400, 800, 1000, 1500 along with the Gauss–Gibbs method. The bars enclose the range of points between the 25th and the 75th quantiles of APE while the point represents the median APE. Figure 4 shows the APE evolving as the algorithms progress through the defined cases and as more training points are utilized. Table 3 compares the median APE of each prescribed method at the most accurate training level, defined by the median APE averaged across output dimensions. There are multiple useful trends

which are important to analyze. First, the GPR models surpass Gauss–Gibbs in prediction accuracy relatively quickly across training levels in cases A and B. This is highlighted in Table 3 as ind-GPR and MV-GPR at 1500 training points provide estimates an order of magnitude more accurate than the non-iterative Gauss–Gibbs method. The GPR methods appear to slowly approach the Gauss–Gibbs accuracy in the more difficult cases E-F due to the increased sparsity of training data relative to the size of the orbit regime.

As shown in Figure 4, the MV-GPR and the ind-GPR perform similarly for most cases, with a slight advantage given to ind-GPR at a fixed training level. This edge is due to the added complexity of the cost function and the 21 additional hyperparameters associated with MV-GPR. More complex models require denser training sets to more accurately estimate their hyperparameters. Additionally, notice that the ind-GPR exhibits a steady increase in prediction accuracy as more training points are added while MV-GPR shows more erratic behavior. The MV-GPR cost function represents a much more complex surface and therefore is likely getting caught in local minimums even with the multiple restarts.

Table 3. Median *APE* for for most accurate model in each orbit case given perfect RA and DEC measurements.

Case	Model	Training Level	Median APE					
Case A	ind-GPR	1500	0.0095	0.0317	0.0303	0.0463	0.0200	0.0850
	MV-GPR	1500	0.0089	0.0314	0.0287	0.0563	0.0250	0.1453
	GG	-	0.0983	0.4646	0.5004	0.2307	0.1718	0.1243
Case B	ind-GPR	1500	0.0534	0.0841	0.0830	0.1255	0.0604	0.1844
	MV-GPR	1000	0.0881	0.1529	0.1472	0.3337	0.1485	0.6143
	GG	-	0.1957	0.3900	0.4321	0.4139	0.2337	0.1174
Case C	ind-GPR	1500	0.0772	0.1603	0.1549	0.3618	0.1371	0.4269
	MV-GPR	1500	0.0972	0.2343	0.2234	0.6334	0.2284	0.8049
	GG	-	0.1576	0.4149	0.4546	0.4413	0.2182	0.2059
Case D	ind-GPR	1500	0.1276	0.2829	0.2371	0.6868	0.2332	0.8654
	MV-GPR	1500	0.1257	0.2814	0.2486	0.7773	0.2758	1.0611
	GG	-	0.1615	0.4083	0.4645	0.4448	0.2267	0.2026
Case E	ind-GPR	1500	0.1637	0.4312	0.3371	1.0399	0.3625	1.1506
	MV-GPR	1500	0.2004	0.5669	0.4477	1.4703	0.5296	1.9463
	GG	-	0.1557	0.3912	0.4458	0.4582	0.2251	0.2112
Case F	ind-GPR	1500	0.2519	0.6297	0.5063	1.2845	0.5387	1.8216
	MV-GPR	1500	0.2734	0.7242	0.5911	1.6985	0.6567	2.4545
	GG	-	0.1495	0.3989	0.4391	0.4091	0.2230	0.2350
Case G	ind-GPR	1500	0.2059	0.2463	0.2513	0.1660	0.4957	0.4790
	MV-GPR	1500	0.2089	0.2539	0.2592	0.1734	0.5351	0.5192
	GG	-	0.4098	0.4964	0.5066	0.2463	0.3569	0.2897

Figure 5 compares box-plots of the *APE* for standard and noise aware MV-GPR at 200, 400, 800, 1000, 1500 along with the Gauss–Gibbs method. The bars enclose the range of points between the 25th and the 75th quantiles of *APE* while the point represents the median APE. Table 4 displays the lowest median *APE* as averaged across the Cartesian states for each method of interest. Referring to Figure 5 and Table 4, the accuracy of the GPR state prediction is nearly identical between the standard and noise aware solutions. In fact, the state estimates provided by the STND. and N.A. are often equal to within four digits of each other as shown in Table 4. This is expected because, as shown in Equation (26), $\mathbf{q}_i$ approaches $\mathbf{K}(\mathcal{X}, \mathbf{x}_*)$ when $\Sigma_{\mathbf{x}_*}$ is as small as it is in this work. The benefits of the formulation presented in Section 6 will become evident in the uncertainty analysis. For Cases A-D the accuracy of the method does not steadily increase as a function of training data. In general, the output becomes more sensitive to disturbances in the inputs as the training set becomes more populated and the regression becomes more "jagged". This cannot be the only explanation as with Case B, there is a significant drop in *APE* between 1000 and 1500 training points. In addition to the density of the training data, smoothness of the regression—how closely the mean curve must approach to the training point—is also

a function of the signal-to-noise ratio, σ_f/σ_n. In Case B, σ_f/σ_n for the models using 800 and 1000 training samples is over two orders of magnitude larger than with 1500 training points. Figure 4 shows a slightly worse prediction for the Case B MV-GPR model defined by 1500 training samples, compared to 800 and 1000 samples, given the general trend of increasing accuracy given denser training sets. However, Figure 5 shows that this model is particularly robust to noisy inputs. Therefore, smaller σ_f/σ_n seems to reduce regression accuracy in the deterministic input case, but provide a model which is more robust to input noise.
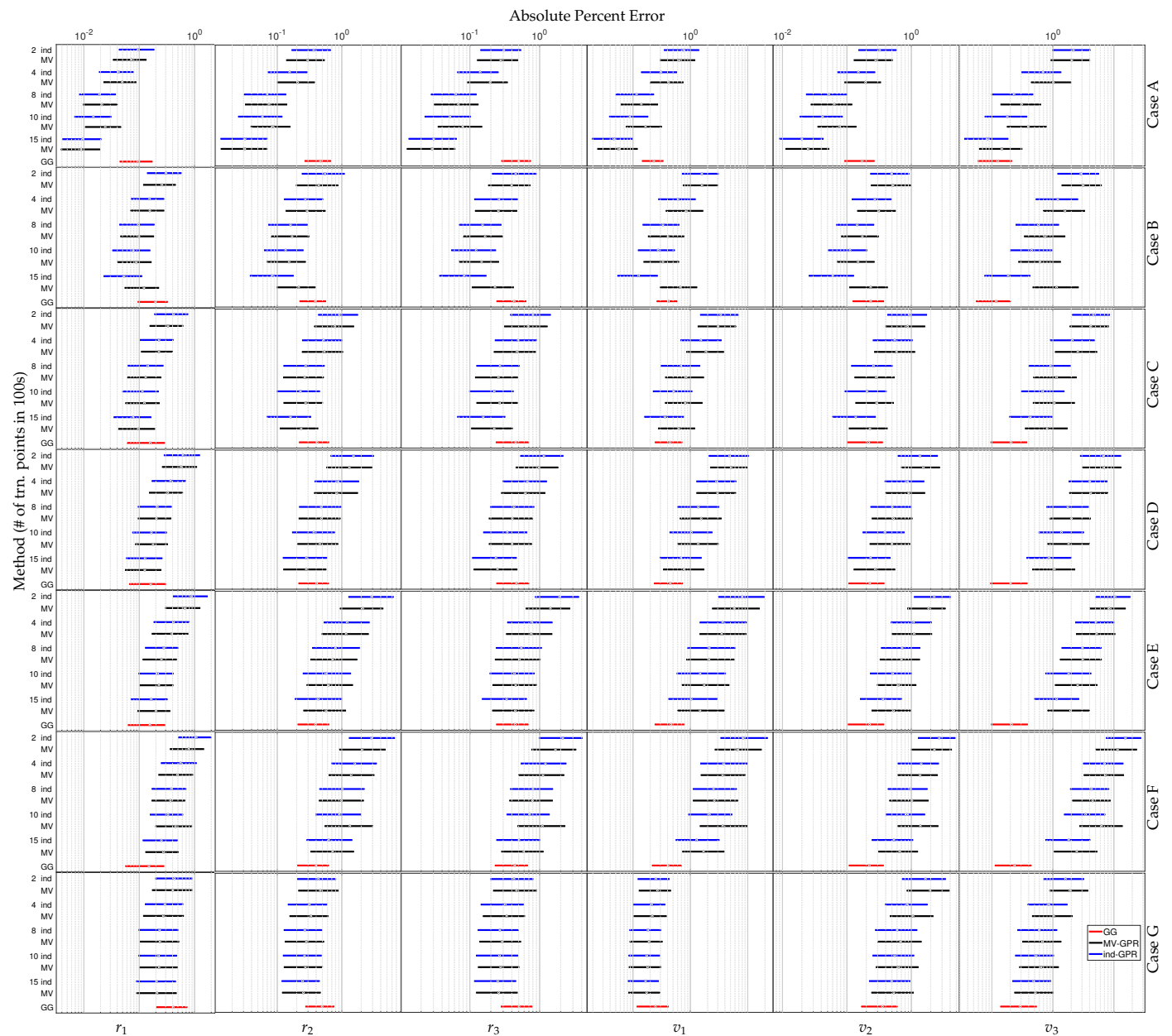
Figure 4. Given the 5000 testing orbits, the point represents the median *APE* while the boxes enclose the range of points between the 25th and the 75th quantiles of *APE* for each Cartesian state given perfect RA and DEC measurements. The GPR methods are listed ind-GPR (blue) then MV-GPR (black) and are ordered by number of training points—200, 400, 800, 1000, and 1500—listed in 100 s. The Gauss–Gibbs results are labeled GG and shown in (red).
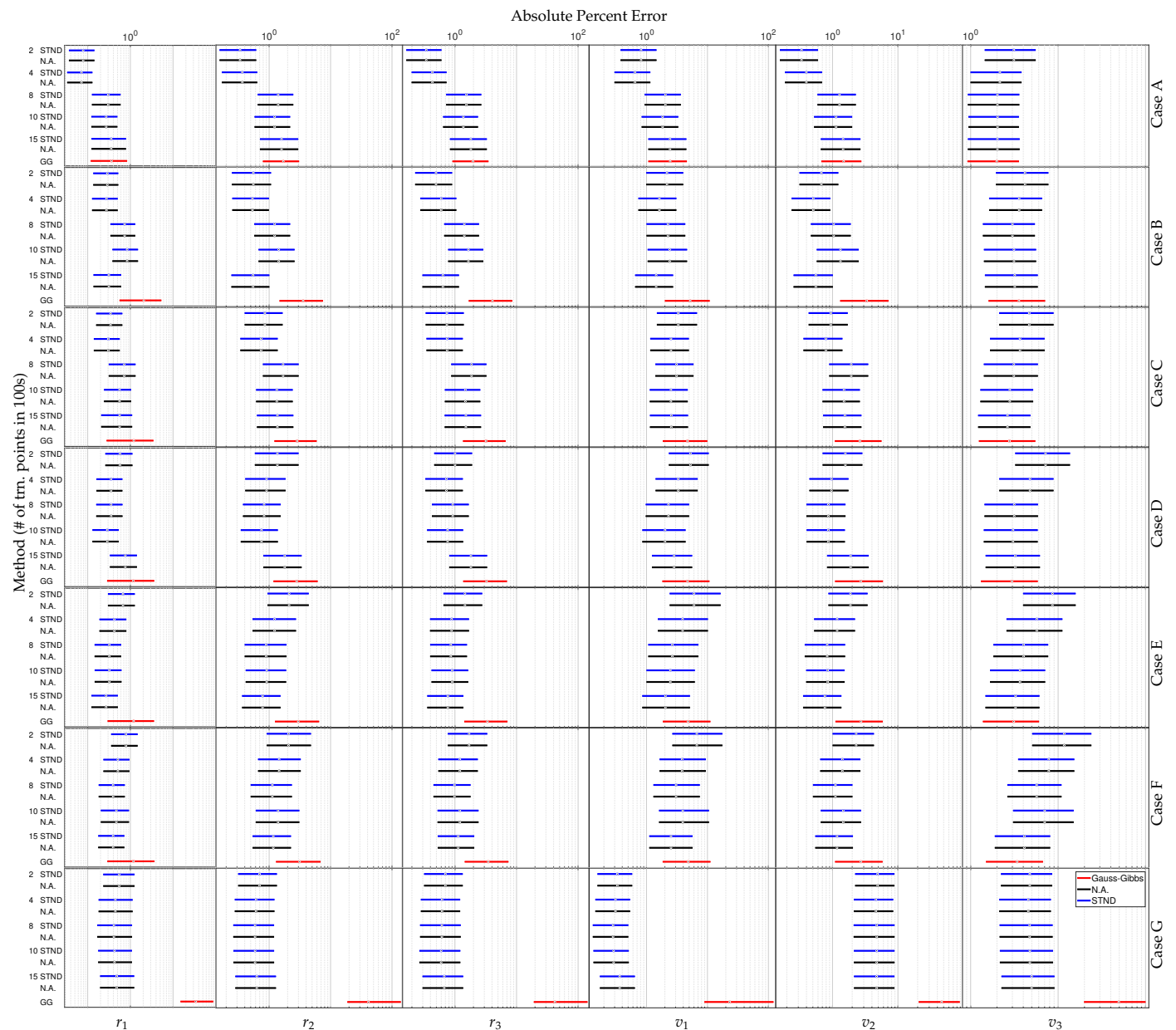
Figure 5. The point represents the median *APE* while the boxes enclose the range of points between the 25th and the 75th quantiles of *APE* for each Cartesian state given uncertain RA and DEC measurements. The GPR methods are listed STND. (blue) Then, N.A. (black) and are ordered by number of training points—200, 400, 800, 1000, and 1500—listed in 100s. The Gauss–Gibbs results are labeled GG and shown in (red).

Table 4. Median *APE* for for most accurate model in each orbit case given noisy RA and DEC measurements.

Case	Model	Training Level	Median APE					
	STND.	400	0.0719	0.3647	0.4297	0.6460	0.3972	2.1586
Case A	N.A.	400	0.0719	0.3647	0.4299	0.6461	0.3973	2.1576
	GG	-	0.3623	1.7049	1.9628	2.4440	1.4774	1.9948
	STND.	1500	0.3134	0.5451	0.6393	1.4412	0.5567	3.1896
Case B	N.A.	1500	0.3135	0.5452	0.6392	1.4416	0.5571	3.1896
	GG	-	2.0710	3.5750	4.0701	5.2170	3.3308	3.5605

Table 4. *Cont.*

Case	Model	Training Level	Median APE					
Case C	STND.	400	0.3048	0.7412	0.7475	2.5160	0.7924	3.6713
	N.A.	400	0.3049	0.7412	0.7476	2.5163	0.7926	3.6715
	GG	-	1.2056	2.8494	3.2037	4.6964	2.6248	2.7794
Case D	STND.	1000	0.2885	0.7497	0.7593	2.0085	0.8639	3.0448
	N.A.	1000	0.2885	0.7498	0.7595	2.0082	0.8639	3.0442
	GG	-	1.1726	2.8190	3.2544	4.7066	2.7084	2.9684
Case E	STND.	1500	0.2687	0.7840	0.7691	2.0489	0.7661	3.2572
	N.A.	1500	0.2687	0.7842	0.7691	2.0486	0.7662	3.2569
	GG	-	1.1912	2.9787	3.3680	4.8038	2.7215	3.1105
Case F	STND.	1500	0.3953	1.1690	1.1260	2.5241	1.1705	4.1178
	N.A.	1500	0.3953	1.1691	1.1256	2.5240	1.1703	4.1165
	GG	-	1.1809	3.1280	3.4665	4.8219	2.7099	3.3952
Case G	STND.	400	0.4437	0.6128	0.6185	0.3103	4.5968	4.5697
	N.A.	400	0.4438	0.6127	0.6185	0.3102	4.5963	4.5689
	GG	-	34.0713	41.3535	42.1487	23.1603	46.5761	49.4980

7.3. Effects of Orbit Shape

Now, the effects of the orbit regime within a given case will be investigated for MV-GPR and compared to Gauss–Gibbs method. Case C with 1000 training points is chosen as it shows an investigation of the whole orbit regime from LEO to high-MEO and provides an accurate MV-GPR model for both perfect and noisy measurements.

Figure 6a is a scatter plot of the Gauss–Gibbs and MV-GPR state estimate's *APE* averaged across the Cartesian states versus the ground truth *SMA* and *e*, which define the orbit shape. Examining the results for the Gauss–Gibbs method shown in the first row of Figure 6, the mean Cartesian *APE* is shown to be relatively agnostic with respect to the orbit shape given the perfect measurements. Similarly, the APEs for MV-GPR shown in the bottom row of Figure 6a are essentially agnostic to the orbit shape with one caveat being that accuracy degrades near the edges of the distribution. For example, investigating the bottom right corner of Figure 6a, there is an obvious degradation in accuracy. The reasoning for this is again related to the sparsity of data as shown in the accuracy investigation. There are no training points with an *SMA* greater than 40,000 km nor eccentricities less than 0.05, leading to the sparsity of neighboring training points to aid in the regression.

Shifting focus to the effects of input noise. Figure 6b examines the state estimate's $MdAPE_i$. There is a noticeable trend towards a degradation in $MdAPE_i$ towards larger, more circular orbits for Gauss–Gibbs method. This follows the similar trend for the scalar triple product of the lines-of-sight $\left(\hat{\mathbf{L}}_1^T \left(\hat{\mathbf{L}}_2 \times \hat{\mathbf{L}}_3\right)\right)$ shown in Figure 6c. The presence of noise together with nearly coplanar geometry—evident by the shrinking scalar triple product—affect the Gauss–Gibbs solution. The MV-GPR study in row 2 of Figure 6b reflects a similar conclusion to that addressed Figure 6a; accuracy is most greatly affected by sparsity of data and not orbital geometry. Unlike the Gauss–Gibbs method, the GPR prediction accuracy should not be directly related to the triple product because it is a regression based method.

7.4. Effects of Coplanarity

The classical Gauss–Gibbs method has been shown to be sensitive to coplanar geometry, as shown in Section 2. Case G is used to further study the effects of coplanarity on the Gauss–Gibbs and MV-GPR solutions. Figure 7a shows the scalar triple product of the lines-of-sight and average of the Cartesian *APE* of both Gauss–Gibbs and MV-GPR versus true Ω given perfect measurements. Figure 7b shows these values with the inclusion of input noise, while Figure 7c examines the median scalar triple product and $MdAPE_i$. As shown in the second row of Figure 7a, there is a large degradation in accuracy around $\Omega \approx -2.5°$. This corresponds to the scalar triple product approaching zero, thus the the lines-of-sight becoming linearly dependent. As shown in Figure 7b,c, this degradation is exacerbated by the addition of noise leading to a magnitude increase in *APE* as far as 5° away from the singularity at $\Omega \approx -2.5°$. Row 3 of Figure 7 reflects the same study but for the MV-GPR method. Again, GPR should be agnostic with respect to coplanarity

of the lines-of-sight. Therefore, a decline in accuracy near the singularity at $\Omega \approx -2.5°$ is not present for MV-GPR. Comparing the second and third rows of Figure 7, the GPR methods provide a better estimate with respect to $MdAPE_i$ by at least a order of magnitude within $\approx 5°$ of the Ω leading to a coplanar singularity.
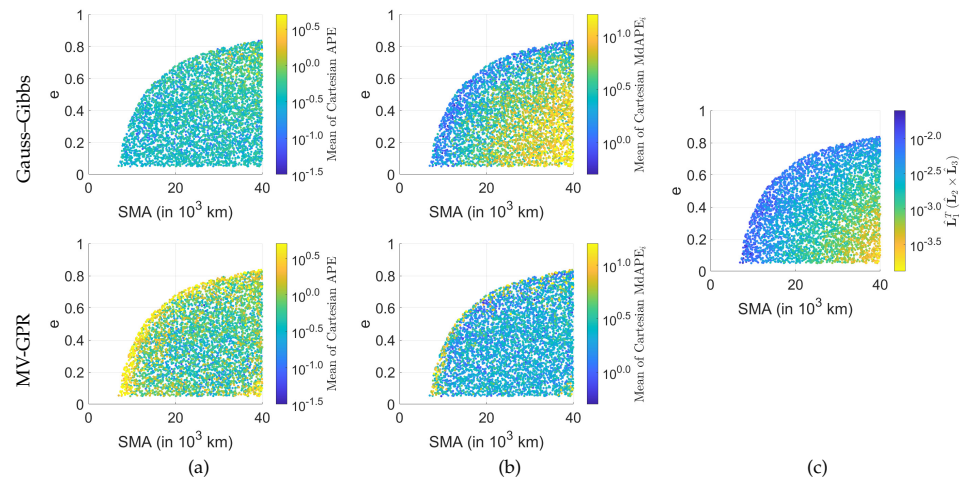
Figure 6. Effects of orbit shape on GPR predictions for Case C: (**a**) average of the Cartesian *APE* of the GPR solution given perfect measurements, (**b**) the average $MdAPE_i$ given noisy measurements, (**c**) the coplanarity of the lines-of-sight. Colorbars represent range for the 99% of points to increase readability in the presence of outliers.

7.5. Bayesian Model Characterization

Unlike the deterministic Gauss–Gibbs IOD method, both ind-GPR and MV-GPR not only provide a state prediction, but also estimate the model uncertainty. This uncertainty characterization is investigated for Case G at training sample supports of 200 and 1000, as it appears to be in the most likely use-case for GPR-based IOD methods and is representative of all cases.
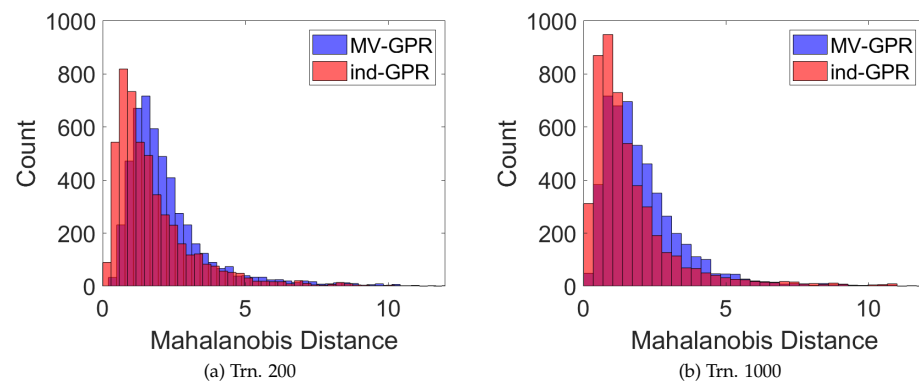
7.5.1. Perfect Measurements

The Mahalanobis distance from the true state to the predicted mean given the predicted covariance is shown in Figure 8. This shows that the majority of Mahalanobis distances for both GPR methods lie between 1 and 3. This is expected for accurate probability modeling of Gaussian variables and means that, in general, the predicted Gaussian model is neither over-confident nor too cautious. Comparing the two methods, ind-GPR carries a slight advantage in the number of points which stay below a Mahalanobis distance of 3, signaling a possible reduction in number of outliers.
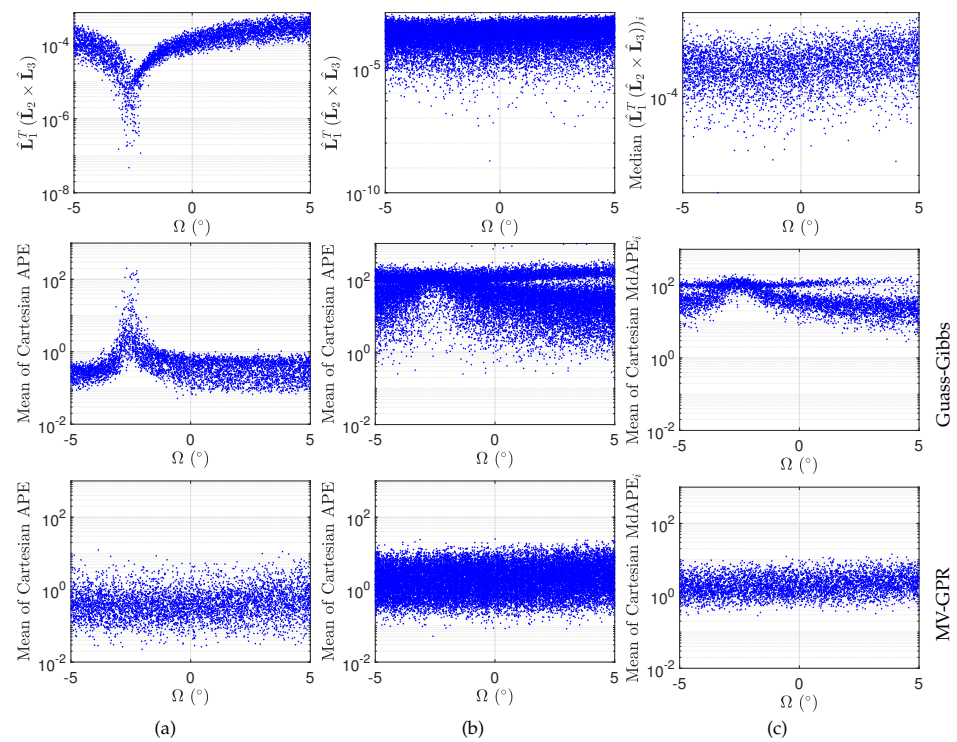
Figure 7. Effects of coplanarity on Gauss–Gibbs and MV-GPR IOD solutions. (**a**) scalar triple product of the lines-of-sight and average of the Cartesian *APE* versus true Ω given perfect measurements. (**b**) scalar triple product and average of the Cartesian *APE* versus true Ω given noisy measurements. (**c**) median scalar triple product and average of the $MdAPE_i$ versus true Ω for each set of five random samples of the 5000 test orbits.

Figure 8. The Mahalanobis distance from the true state to the GPR-predicted mean given the GPR-predicted covariance (**a**) 200 and (**b**) 1000 given perfect RA and DEC measurements.

Figure 9 shows the predicted standard deviation (σ) versus the error between the prediction and the true Cartesian state. Note the threshold for the predicted σ of both methods. This threshold is defined by the hyperparameter σ_n, which in a way acts as the minimal uncertainty in the system, enforcing the fact that one cannot be more certain in predictions of test points than that of the training data. The points for both ind-GPR and MV-GPR follow along the 1-σ line reflecting the fact that higher errors are characterized by higher variances. Furthermore, the majority of points lie below the 3-σ, which further reinforces the characteristic that the model is not overly confident. The trend across training levels is that the method is overly cautious for some of its more accurate uncertainty measures, which is likely due in part to the noise term, σ_n, included in GPR model, which acts as a minimum variance term. Therefore, forcing a small variance for ϵ in the optimization process could help to reduce the over-cautious trend of the prediction covariance. However,

recall this will likely reduce the model's robustness to input noise as posited in the accuracy study. As training levels increase, the GPR estimates appear to more accurately represent that trend line, especially for the MV-GPR algorithm, because the model accuracy increases as training levels increase.
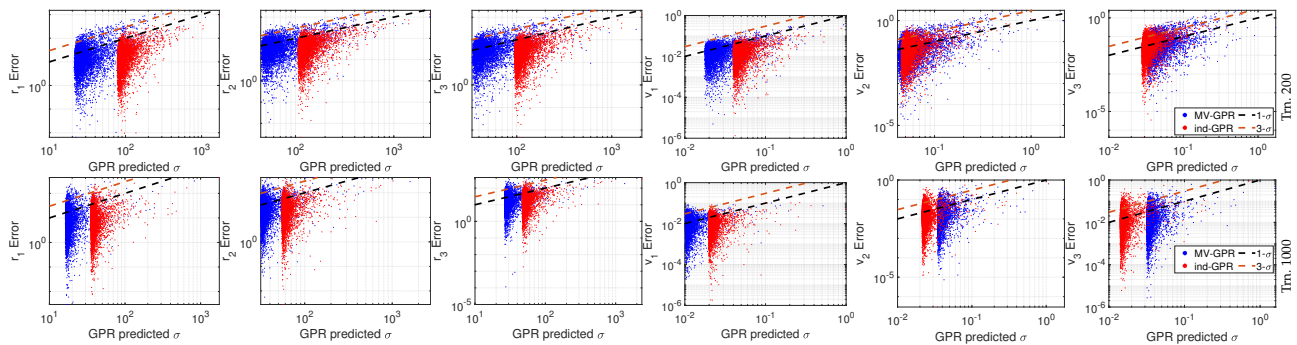

Figure 9. Predicted standard deviation versus the error between the predicted mean and the true Cartesian state given perfect RA and DEC measurements.

Table 5 shows the percentage of points which lie within the predicted $1-\sigma$ and $3-\sigma$ bounds estimated by each GPR model. Under a normal distribution, 68% of all estimates will lie within $1-\sigma$ while 99.7% should lie within $3-\sigma$ bound. At both 200 and 1000 training points, the ind-GPR is vastly over cautious in the variance estimate for r_1, r_2, r_3, and v_1 with the percentage of points within the $1-\sigma$ fluctuating in the high nineties. This highlights the inflexibility of the ind-GPR to provide accurate uncertainty estimation for each output dimension. In contrast, MV-GPR is able to provide more flexible variance estimation such the percentage of points within a given σ bound more accurately reflect the true errors.

Table 5. Percentage of errors within σ–bound for ind- and MV-GPR given perfect input measurements.

| Training Level	Model	σ–Bound	r_1	r_2	r_3	v_1	v_2	v_3
200	ind-GPR	$1-\sigma$	98.1800	96.2200	96.9600	97.4600	66.4000	56.1600
		$3-\sigma$	100.0000	99.9600	99.9800	100.0000	95.7600	91.8400
	MV-GPR	$1-\sigma$	76.5000	73.1200	68.3800	75.0600	64.0600	62.1600
		$3-\sigma$	96.1800	97.6800	96.5600	99.7000	93.0000	92.1800
1000	ind-GPR	$1-\sigma$	95.0800	91.4000	91.4400	85.7800	75.5200	68.5600
		$3-\sigma$	99.7400	99.7400	99.7200	99.9000	95.6600	93.3000
	MV-GPR	$1-\sigma$	78.9200	71.0000	69.8800	56.3000	84.1800	86.2000
		$3-\sigma$	97.1200	97.8000	97.7200	96.6000	97.5800	97.9600

Figure 10 are plots of the true Cartesian state versus the GPR prediction error with accompanying 2-σ bounds for a random 500 point sub-sample of the 5000 test orbits for readability. Here, MV-GPR is able to better track the true error by encoding the covariance between the components of the output state. By definition, the covariance estimate in the ind-GPR algorithm is independent and identically distributed (IID) before re-normalization. From Figure 10 and Table 5, we see that in this case the IID assumption leads to the prediction being over-cautious in the **r** estimates. The covariance bounds for the elements in **r** are much larger than the true error, in general. Therefore, the covariance associated with the ind-GPR only provides a measure of confidence in the output as a whole rather than providing accurate uncertainty information for each of the Cartesian dimensions. MV-GPR tunes the covariance estimation for each output dimension, therefore it is able to more accurately represent model uncertainty in each Cartesian dimension. Finally, the average covariance bounds shrink around four-fold between the case of 200 and 1000 training samples therefore showcasing the effects of sparsity in training data.
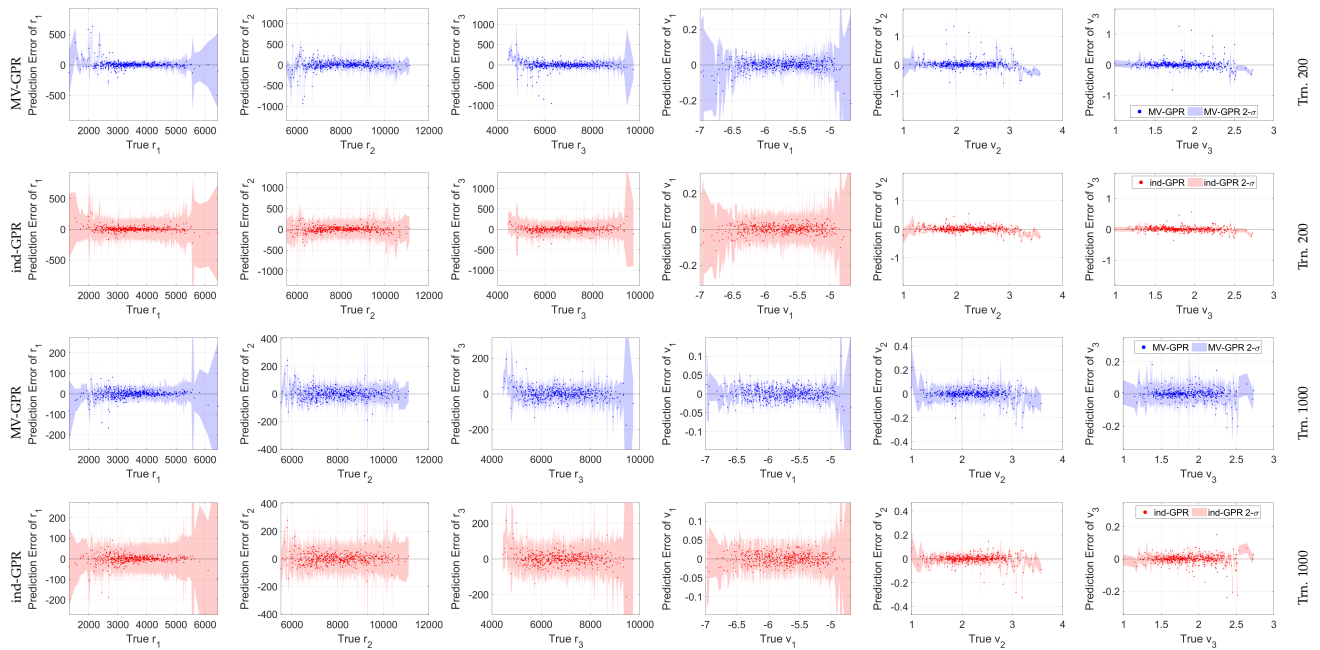
Figure 10. Cartesian state versus the GPR output error with accompanying 3-σ bounds for a 500 point random sub-sample of the 5000 test orbits. Note the differing y-scale between the training levels.
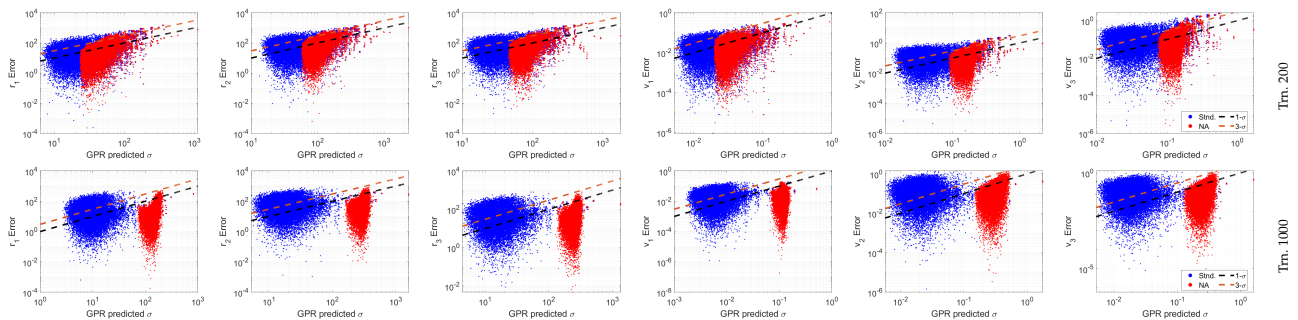
7.5.2. Uncertain Angular Measurements

Recall that the Bayesian model predicted from GPR only characterizes model uncertainty as implemented in this work, therefore the predictive uncertainty characterization will suffer greatly in the presence of input noise. Figure 11 shows the Mahalanobis distance for both the standard MV-GPR formulation and the noise aware formulation presented in Section 6. For the the standard MV-GPR method, the uncertainty is grossly underestimated as the model and inference has zero uncertainty information from the test inputs. The method presented in Section 6 is meant to address this and, as reflected in the second row of Figure 11, does an excellent job at reducing the Mahalanobis distance to within more reasonable bounds. There is a more pronounced effect on the 1000 point training set because model uncertainty decreases as the number of training points increases. Therefore, the effects of uncertain inputs become the dominating term.

(**a**) Trn. 200

(**b**) Trn. 1000
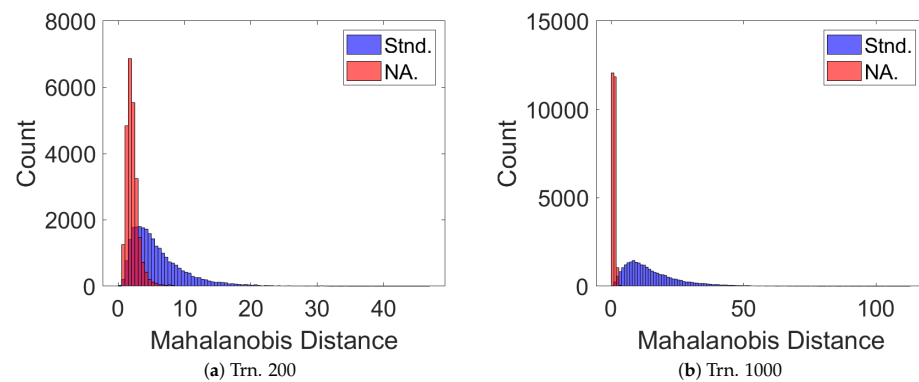
Figure 11. The Mahalanobis distance from the true state to the GPR-predicted mean given the GPR-predicted covariance for Case G at training samples of (**a**) 200 and (**b**) 1000 given uncertain RA and DEC measurements for the standard MV-GPR (blue) and the noise-aware MV-GPR (red).

Figure 12 shows the predicted σ versus the estimate error for each Cartesian state for models with 200 training points in the first row and 1000 training points in the second. The

ability of the noise aware MV-GPR method to accurately quantify the true error is evident. At 200 training points, the model uncertainty is very much present, so the noise aware estimation adds often less than an order of magnitude to the estimated σ. Since the model uncertainty is reduced at increased training levels, the noise aware MV-GPR method has a greater effect on predicted σ values. This results in an almost order of magnitude increase between the standard and noise aware methods.


Figure 12. Predicted standard deviation versus the error between the predicted mean and the true Cartesian state given uncertain angular measurements.

The percentage of points which lie within the GPR estimated 1–σ and 3–σ bounds are shown in Table 6. Over 60% of points lie outside the 1–σ bounds predicted by standard MV-GPR at 1000 training points. This is only marginally improved examining the standard model at 200 training points, but that is largely attributed to the model error of the regression model accounting for a still significant portion of the true error. The percentage of points which lie within σ bounds for the noise aware MV-GPR at 200 training points hovers close to the 68% and 99.7 which define a perfect normal distribution. At 200 training points, the noise aware model over-estimates the uncertainty contribution of the input noise as evident from the high ninety percentile of points within the 1–σ.

Table 6. Percentage of errors within σ–bound for standard and noise aware MV-GPR.

Training Level	Model	σ–Bound	r_1	r_2	r_3	v_1	v_2	v_3
200	STND.	$1 - \sigma$	53.5880	43.6200	39.8320	50.3240	25.0120	24.1080
		$3 - \sigma$	86.0760	82.4120	78.9120	89.7000	57.8480	55.9160
	N.A.	$1 - \sigma$	76.7160	72.4440	69.2400	74.6800	66.7240	66.0720
		$3 - \sigma$	98.2960	99.2000	98.6640	99.7680	98.7160	98.4600
1000	STND.	$1 - \sigma$	36.6560	24.4280	24.0440	22.3760	13.6880	14.6960
		$3 - \sigma$	68.1120	55.2760	54.4840	54.7680	35.0440	37.1280
	N.A.	$1 - \sigma$	98.3080	98.4440	98.5120	99.4320	92.1240	92.1600
		$3 - \sigma$	100.0000	100.0000	100.0000	100.0000	99.9960	99.9960

Figure 13 shows the true Cartesian elements versus the prediction error for a 500 point sub-sample of the 25,000 random noisy measurements. At 200 training points, the difference between the standard and noise aware MV-GPR inference implementations is limited as the regression uncertainty is dominated by model uncertainty. The benefits of the noise aware method is much more evident at 1000 training points. The standard inference method vastly underestimates uncertainty for the majority of samples while the noise aware method better captures the true error though being slightly over-cautious.
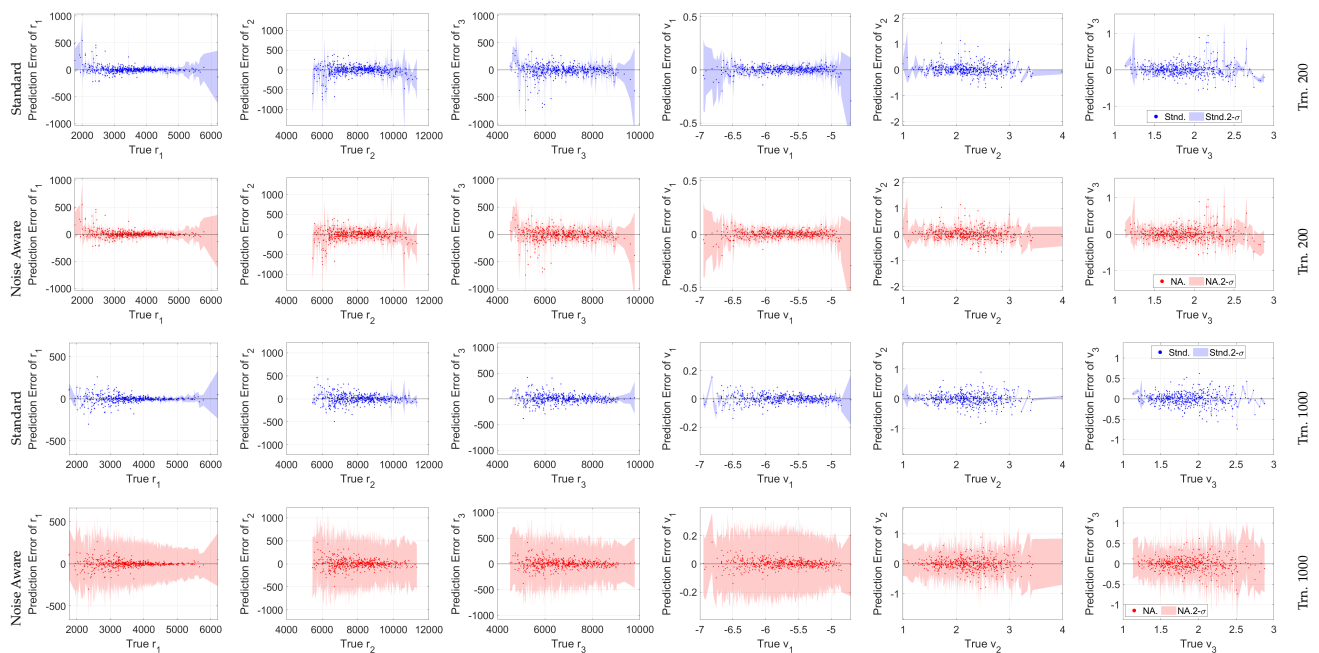
Figure 13. Cartesian state versus the GPR output error with accompanying 3-σ bounds for a 500 point random sub-sample of the 25,000 total samples. Note the different *y*-axis scales between the training levels.
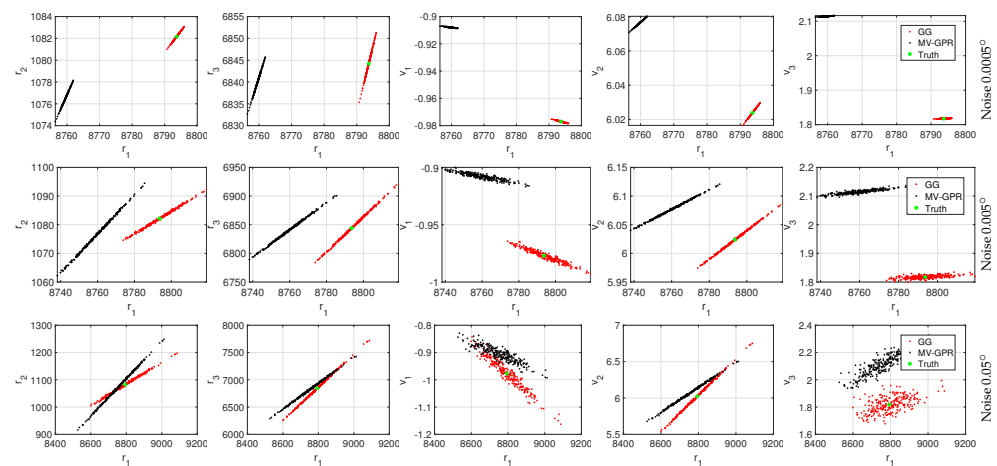
7.6. Qualitative Study

Now, with general trends established, it is important to perform a qualitative study on the proposed techniques using exemplar orbits shown in Table 7. In this section, the iterative Gauss–Gibbs method is utilized to better show the effect of GPR model bias. Orbit A represents an orbit in a regime where both Gauss–Gibbs method and the proposed GPR methods perform well. Orbit B represents a high MEO orbit in an orbit shape which the Gauss–Gibbs method with iteration is shown to begin to degrade in accuracy. Orbit C represents the failure case of a nearly coplanar orbit. In these studies, the true anomaly at the first and third measurement time are separated 10° from θ_2. For each orbit, 250 uncertain RA and DEC measurements are generated using the noise model defined in Section 7.1.2 at pointing errors of 0.0005°, 0.005°, and 0.05°. The GPR models for Orbits A and B are generated using 1000 training samples from Case C in Table 1, while the model for Orbit C is generated from Case G training data set at 1000 training samples.

Table 7. Exemplar orbits for qualitative study.

| Orbit	SMA	e	i (°)	Ω (°)	ω (°)	θ_2 (°)
Orbit A	13,000	0.2	40	300	17	55
Orbit B	36,000	0.5	40	300	17	55
Orbit C	13,000	0.2	40	3	17	55

Figure 14 shows the distribution of the 250 samples for Orbit A at varying noise levels for both Gauss–Gibbs and noise aware MV-GPR methods. The levels of noise have very similar effects in inflating the resulting distribution between both methods. The distribution of the Gauss–Gibbs iterative method centers around the truth, while that of the GPR method shows a model bias. This bias is especially prevalent for pointing errors of 0.0005° and 0.005° because the prediction uncertainty at these low levels of noise is still dominated by the model error. Therefore, the GPR distributions are centered around their model bias, or their prediction given perfect RA and DEC measurements. Again, this dominating model uncertainty and bias is caused by the sparsity of training data associated with the large orbit distribution of Case C.

Figure 14. Qualitative study of Orbit A at various noise levels. The black points are MV-GPR and the red are Gauss–Gibbs predictions,while the green point represents the truth.

The effects of geometry and noise in the IOD process begin to show in Figure 15, a study of the distribution of the 250 samples for Orbit B at varying noise levels for both Gauss–Gibbs and noise aware MV-GPR methods. Again, model uncertainty is the dominate term for the MV-GPR model at a noise level of $0.0005°$. At a level of $0.005°$, the measurement noise begins to degrade the Gauss–Gibbs prediction accuracy. Here, the GPR methods provide a tighter distribution in all Cartesian states, though still centered around the model bias. Finally at a noise level of $0.05°$, the measurement noise dominates. This leads to a large distribution in the Gauss–Gibbs predictions while the MV-GPR solutions provide a tighter-distribution, closer to the truth, for all Cartesian elements.
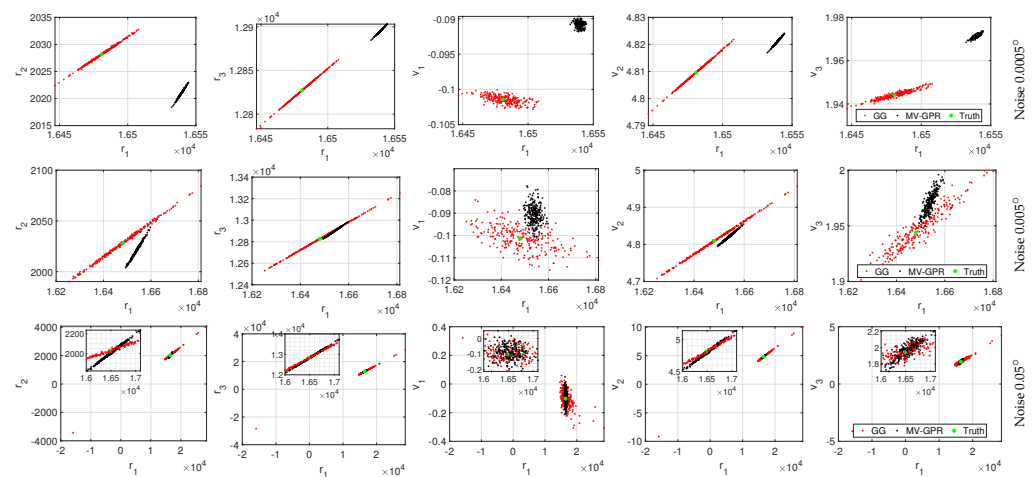
Figure 15. Qualitative study of Orbit B at various noise levels. The black points are MV-GPR and the red are Gauss–Gibbs predictions,while the green point represents the truth.

Figure 16 shows the distribution of the 250 samples for the nearly coplanar orbital geometry case, Orbit C, at varying noise levels. Again, the near-coplanar lines-of-sight measurements present for Orbit C lead to a complete degradation in the accuracy of Gauss–Gibbs method. Even at the lowest noise levels, the MV-GPR solution provides a nearly order of magnitude decrease in the distribution of the noisy predictions. At $0.005°$ measurement errors, the Gauss–Gibbs prediction distribution is on the order of 1000 km in position and hundreds of m/s in velocity. In stark contrast, the MV-GPR prediction distribution is on the order of 10 km in position and tens of m/s, providing orders-of-magnitude increase in accuracy. Finally, the Gauss–Gibbs method completely degrades at $0.05°$. MV-GPR's prediction for Orbit C provides accuracy comparable to Gauss–Gibbs method for Orbit A in Figure 14, which represents an orbit where accurate Gauss–Gibbs predictions are expected.
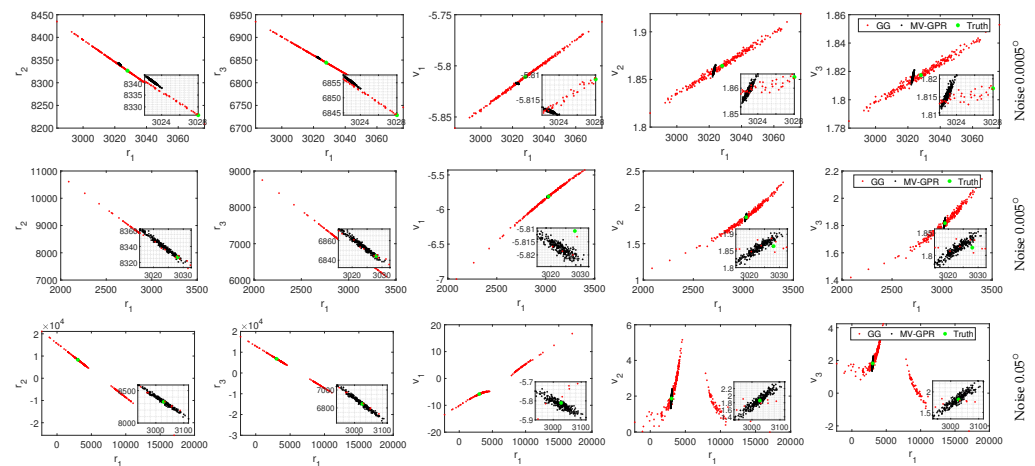
Figure 16. Qualitative study of Orbit C at various noise levels. The black points are MV-GPR and the red are Gauss–Gibbs predictions,while the green point represents the truth.

8. Conclusions

In summary, this work examines the effects of a target's orbit regime on the classical Gauss–Gibbs angles-only IOD method and showcases the sensitivity of the Gauss–Gibbs method to input noise and a coplanar observer-target geometry. Gauss–Gibbs method shows sensitivity to varying levels input noise dependent on the orbit regime of the target. Furthermore, the combination of input noise and nearly-coplanar observation geometry leads to complete degradation of prediction accuracy for Gauss–Gibbs method. The GPR methods implemented in this work provided a relatively accurate model less sensitive to orbital geometry and observation uncertainty. Given perfect inputs, these machine learning methods degrade in accuracy as training sets become more sparse. However, that sparsity actually proves beneficial when high levels of noise are introduced to the right ascension and declination because of the increased GPR mean-function smoothness. The GPR methods not only provide a point solution, but also characterize the model uncertainty—an added benefit over classical methods. In the case of perfect measurements, the multivariate GPR (MV-GPR) method outperforms the multitask learning GPR method (ind-GPR) in this uncertainty characterization due to the inclusion of output correlations. Given noisy RA and DEC measurements, the noise aware inference method inflates the predicted uncertainty to capture not only the model uncertainty, but also the induced uncertainty from random inputs. This results in accurate variance inflation for MV-GPR trained at 200 training points, while predicted variance is over-inflated at 1000 training points for the test case presented in this work.

Given that specific orbital regimes are more frequently encountered in practice, it could prove advantageous to tailor training data to reflect this distribution or even use true observations. Furthermore, although observation frame manipulation may make the GPR method agnostic to observation site longitude, the latitude and altitude will have a significant effect on observer-target geometry. Addressing this observation site dependence will prove beneficial. Proper characterization of the noise ϵ will likely prove extremely beneficial for model robustness with respect to input noise given the GPR signal-to-noise ratio affects on model sensitivity.

Author Contributions: Conceptualization, D.S., P.S. and S.O.; Data curation, D.S.; Formal analysis, D.S.; Funding acquisition, P.S.; Investigation, D.S.; Methodology, D.S.; Project administration, P.S.; Resources, P.S.; Software, D.S.; Supervision, P.S. and S.O.; Validation, D.S.; Visualization, D.S., P.S. and S.O.; Writing—original draft, D.S.; Writing—review & editing, P.S. and S.O. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

IOD	Initial Orbit Determination
GG	Gauss–Gibbs Orbit Determination Algorithm
GPR	Gaussian Process Regression
ind-GPR	Independent Gaussian Process Regression
MV-GPR	Multivariate Gaussian Process Regression
SE	Square Exponential Covariance Function
STND	Standard Multivariate Gaussian Process Regression
NA	Noise Aware Multivariate Gaussian Process Regression
SSA	Space Situational Awareness
RA	Right Ascension
DEC	Declination
SE	Square Exponential
ARD	Automatic Relevance Determination
APE	Absolute percent error
MdAPE	Median absolute percent error
TOF	Time-of-flight
IID	independent and identically distributed

Appendix A

Here, we will give a brief explanation of the numerical difficulties with the method presented in Section 6 and how they were addressed. First, Equation (28) can be rewritten such that

$$\text{cov}\left(\mathbf{y}_*|\boldsymbol{\mu}_{\mathbf{x}_*},\boldsymbol{\Sigma}_{\mathbf{x}_*}\right) = \left\{\hat{\boldsymbol{\Sigma}} + \text{tr}\left(\left(\mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}}) + \sigma_n^2\mathbf{I}_N\right)^{-1}[\mathbf{K}(\boldsymbol{\mathcal{X}},\mathbf{x}_*)\mathbf{K}(\mathbf{x}_*,\boldsymbol{\mathcal{X}}) - \mathbf{Q}]\right)\right\} \otimes \mathbf{B} + \underbrace{\boldsymbol{\Phi}^T\mathbf{Q}\boldsymbol{\Phi} - (\boldsymbol{\Phi}^T\mathbf{q})(\boldsymbol{\Phi}^T\mathbf{q})^T}_{\text{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*])}, \quad \text{(A1)}$$

where $\hat{\boldsymbol{\Sigma}}$ is defined in Equation (17). The numerical issues are most prevalent in the $\text{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*])$ term. Due to the inverse calculation, $\boldsymbol{\Phi} = \left(\mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}}) + \sigma_n^2\mathbf{I}_N\right)^{-1}\boldsymbol{\mathcal{Y}}$ is only accurate to a finite number of digits. Therefore, the term $\text{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*])$ appears to suffer from *catastrophic cancellation*. This is corroborated by inspection by the author. As the condition number of $\mathbf{K}(\boldsymbol{\mathcal{X}},\boldsymbol{\mathcal{X}}) + \sigma_n^2\mathbf{I}_N$ increased, the $\text{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*])$ term became increasingly numerically sensitive as evident by producing a non-symmetric and/or a non-positive semi-definite term. To alleviate this sensitivity, the eigenvalue decomposition is performed on symmetric matrix $\mathbf{Q}$ such that $\mathbf{Q} = \mathbf{P}\boldsymbol{\Lambda}\mathbf{P}^T$, eigenvalues less than 5×10^{-12} were zeroed, and the $\text{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*])$ is calculated such that

$$\text{cov}_{\mathbf{x}_*}(\mathbb{E}_{\mathbf{f}_*}[\mathbf{f}_{\mathbf{x}_*}|\mathbf{x}_*]) = \left(\sqrt{\boldsymbol{\Lambda}}\mathbf{P}^T\boldsymbol{\Phi}\right)^T\left(\sqrt{\boldsymbol{\Lambda}}\mathbf{P}^T\boldsymbol{\Phi}\right) - (\boldsymbol{\Phi}^T\mathbf{q})(\boldsymbol{\Phi}^T\mathbf{q})^T. \quad \text{(A2)}$$

This formulation significantly relieved the numerical difficulties and ensured symmetric positive definite covariance estimates.

References

1. Adurthi, N.; Majji, M.; Singla, P. Quadrature-Based Nonlinear Joint Probabilistic Data Association Filter. *J. Guid. Control. Dyn.* **2019**, *42*, 2369–2381. [CrossRef]
2. Vallado, D.A. *Fundamentals of Astrodynamics and Applications*, 4th ed.; Space Technology Library; Microcosm Press: Hawthorne, CA, USA, 2013.
3. Schwab, D.; Singla, P.; Raquepas, J. Uncertainty Characterization and Surrogate Modeling for Angles-Only Initial Orbit Determination. In Proceedings of the 2019 AAS/AIAA Astrodynamics Specialist Conference, Portland, ME, USA, 11–15 August 2019; American Astronautical Society: Springfield, VA, USA, 2019.
4. Hixon, S.; Schwab, D.; Reiter, J.; Singla, P. Conjugate Unscented Transformation Based Semi-Analytic Approach for Uncertainty Characterization of Angles-Only Initial Orbit Determination Algorithms. In Proceedings of the 70th International Astronautical Congress, Washington, DC, USA, 21–25 October 2019.
5. Schwab, D.; Singla, P.; Huang, D. Multi-Variate Gaussian Process Regression for Angles-Only Initial Orbit Determination. In Proceedings of the 2020 AAS/AIAA Astrodynamics Specialist Conference, Lake Tahoe, CA, USA, 9–12 August 2020.
6. Martínez, F.; Águeda Maté, A.; Grau, J.; Fernández Sánchez, J.; Aivar García, L. Comparison of Angles Only Initial Orbit Determination Algorithms for Space Debris Cataloguing. *J. Aerosp. Eng. Sci. Appl.* **2012**, *4*, 39.
7. Schaeperkoetter, A.V. A Comprehensive Comparison between Angles-Only Initial Orbit Determination Techniques. Master's Thesis, Texas A&M University, College Station, TX, USA, 2011.
8. Vallado, D.A. *Evaluating Gooding Angles-Only Orbit Determination of Space Based Space Surveillance Measurements*; Maui Economic Development Board: Maui, HI, USA, 2010.
9. Weisman, R.M.; Majji, M.; Alfriend, K.T. Analytic Characterization of Measurement Uncertainty and Initial Orbit Determination on Orbital Element Representations. *Celest. Mech. Dyn. Astron.* **2014**, *118*, 165–195. [CrossRef]
10. Worthy, J.L.; Holzinger, M.J. Incorporating Uncertainty in Admissible Regions for Uncorrelated Detections. *J. Guid. Control. Dyn.* **2015**, *38*, 1673–1689. [CrossRef]
11. Binz, C.R.; Healy, L.M. Direct Uncertainty Estimates from Angles-Only Initial Orbit Determination. *J. Guid. Control. Dyn.* **2018**, *41*, 34–46. [CrossRef]
12. Lee, B.S.; Kim, W.G.; Lee, J.; Hwang, Y. Machine Learning Approach to Initial Orbit Determination of Unknown LEO Satellites. In Proceedings of the 2018 SpaceOps Conference, Marseille, France, 28 May–1 June 2018; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2018. [CrossRef]
13. Peng, H.; Bai, X. Covariance Fusion Strategy of Gaussian Processes Covariance and Orbital Prediction Uncertainty. In Proceedings of the 2019 AAS/AIAA Astrodynamics Specialist Conference, Portland, ME, USA, 11–15 August 2019; American Astronautical Society: Springfield, VA, USA, 2019.
14. Rasmussen, C.E. Evaluation of Gaussian Processes and Other Methods for Non-Linear Regression. Ph.D Thesis, University of Toronto, Toronto, ON, Canada, 1997.
15. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; Adaptive Computation and Machine Learning; MIT Press: Cambridge, UK, 2006.
16. Liu, H.; Cai, J.; Ong, Y.S. Remarks on Multi-Output Gaussian Process Regression. *Knowl.-Based Syst.* **2018**, *144*, 102–121. [CrossRef]
17. Boyle, P.; Frean, M. Dependent Gaussian Processes. In *NIPS'04: Proceedings of the 17th International Conference on Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2005; Volume 17, pp. 217–224.
18. Alvarez, M.A.; Rosasco, L.; Lawrence, N.D. Kernels for Vector-Valued Functions: A Review. *arXiv* **2012**, arXiv:1106.6251.
19. Wang, B.; Chen, T. Gaussian Process Regression with Multiple Response Variables. *Chemom. Intell. Lab. Syst.* **2015**, *142*, 159–165. [CrossRef]
20. Chen, Z.; Wang, B.; Gorban, A.N. Multivariate Gaussian and Student-t Process Regression for Multi-Output Prediction. *Neural Comput. Appl.* **2020**, *32*, 3005–3028. [CrossRef]
21. Candela, J.; Girard, A.; Larsen, J.; Rasmussen, C. Propagation of Uncertainty in Bayesian Kernel Models—Application to Multiple-Step Ahead Forecasting. In Proceedings of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, Hong Kong, China, 21 May 2003; Volume 2.
22. Schwab, D. Efficacy of Gaussian Process Regression for Angles-Only Initial Orbit Determination. Master's Thesis, The Pennsylvania State University, University Park, USA, 2020.
23. Curtis, H.D. *Orbital Mechanics for Engineering Students*, 1st ed.; Elsevier: Amsterdam, The Netherlands, 2008.
24. Pocuca, N.; Gallaugher, M.P.B.; Clark, K.M.; McNicholas, P.D. Assessing and Visualizing Matrix Variate Normality. *arXiv* **2019**, arXiv:1910.02859.
25. Dawid, A.P. Some Matrix-Variate Distribution Theory: Notational Considerations and a Bayesian Application. *Biometrika* **1981**, *68*, 265–274. [CrossRef]
26. Deisenroth, M.P. *Efficient Reinforcement Learning Using Gaussian Processes*; Karlsruhe Institute of Technology: Karlsruhe, Germany, 2010.
27. Rasmussen, C.E.; Nickisch, H. The GPML Toolbox Version 4.2. Available online: http://www.gaussianprocess.org/gpml/code/matlab/doc/ (accessed on 6 June 2019).
28. Chen, Z. Multivariate Gaussian and Student-t Process Regression. Available online: https://github.com/Magica-Chen/gptp_multi_output (accessed on 8 April 2020).