

CONJUNCTION ASSESSMENT USING POLYNOMIAL CHAOS EXPANSIONS

Brandon A. Jones⁽¹⁾, Alireza Doostan⁽²⁾, and George Born⁽³⁾

⁽¹⁾⁽³⁾*Colorado Center for Astrodynamics Research, University of Colorado Boulder, UCB 431, Boulder, Colorado, 80309, 303-735-4490, brandon.jones@colorado.edu, george.born@colorado.edu*

⁽²⁾*Department of Aerospace Engineering Sciences, University of Colorado Boulder, UCB 429, Boulder, Colorado, 80309, 303-492-7572, alireza.doostan@colorado.edu*

Abstract: This paper presents the application of polynomial chaos expansions (PCEs) to estimate the probability of collision between two spacecraft. Common methods of collision risk assessment use either Monte Carlo analyses or assume a Gaussian probability distribution function (pdf). PCEs provide a means for approximating the solution to a large set of stochastic ordinary differential equations, which includes orbit propagation. When compared to Monte Carlo methods, sampling-based, non-intrusive PCE techniques greatly reduce the number of orbit propagations required to approximate the *a posteriori* pdf. Additionally, PCEs make no assumption that the propagated distribution will be Gaussian or that the state pdfs are uncorrelated for two spacecraft. This paper considers two cases where these assumptions are no longer valid. Results demonstrate that more than 100,000 Monte Carlo trials are required to estimate the collision probability, but accurate estimates may be achieved with less than 300 orbit propagations by performing most of the analysis via PCE evaluations.

Keywords: conjunction assessment, uncertainty propagation, polynomial chaos

1. Introduction

As the space in low-Earth and geosynchronous orbit becomes increasingly congested, rigorous methods of estimating the risk of collision between any two objects becomes increasingly important. Such methods must be highly accurate to both identify possible collisions and minimize the number of false alarms that waste fuel and disrupt satellite operations. This paper considers a new application of polynomial chaos (PC) expansions for the quantification of collision risks in scenarios where already established methods no longer apply.

Several methods of conjunction assessment already exist and are used for satellite operations today. The most commonly discussed methods [1–3] share several assumptions, including, but not limited to:

1. the probability density function (pdf) describing the position state is a multivariate Gaussian distribution,
2. the uncertainties for the two objects are uncorrelated.

This paper considers situations where at least one of these conditions is not satisfied. Recent research demonstrates that assumption (1) is not strictly true in astrodynamics [4–10], with various methods suggested for non-Gaussian uncertainty propagation. Assumption (2) is valid for most scenarios, and almost surely true for orbital debris. However, given the increasing interest in formation flying and fractionated spacecraft, the inclusion of relative measurements in their orbit determination creates correlated uncertainties between the satellites.

Conjunction assessment based on Monte Carlo (MC) methods provides the most accurate, but computationally expensive, estimate of the collision risk. Such methods reduce the assumptions, and are only limited by the accuracy of the propagator, the initial uncertainty, and computation time. However, the computation time requirements pose a major limitation in adoption of such methods [11–13]. Several tools exist to reduce the long computation times associated with Monte Carlo methods, with these tools leveraging off of both improvements in computation capabilities and theories developed to reduce the number of MC trials. Sabol, et al. [11] demonstrate the use of the supercomputer capabilities to perform Monte Carlo analyses within seconds per satellite. Dolado et al. [12] consider importance sampling to reduce the number of orbit propagations required to quantify collision probability. Garmier et al. [13] reduce the overall computation time by filtering out scenarios with a sufficiently low probability of collision, thereby limiting the total number of Monte Carlo trials for a given space object catalog. The PC-based methods discussed in this paper share similarities with Monte Carlo, but use polynomial surrogates to reduce the total number of orbit propagations.

Polynomial chaos expansions (PCEs) allow for the approximation of the solution to a stochastic differential equation that is square-measurable, possibly non-Gaussian, with respect to the input uncertainties [14–25]. PC-based methods use a projection of the stochastic solution onto a basis of orthogonal polynomials in stochastic variables that is dense in the space of finite-variance random variables. Using this polynomial basis, these PC methods provide a response surface in the space of the random inputs, thereby allowing for the generation of realizations at a future point in time when given random inputs at the initial time. In generating these realizations, no additional evaluations of the orbit propagator are required, thereby reducing the computation time. The current authors already demonstrated the use of PC for propagating orbit uncertainty over relatively long durations (10 days), and results demonstrate the ability to accurately approximate a non-Gaussian pdf at the final time [4]. This research considers the use of PC, specifically the generated response surfaces, to conduct conjunction assessment. As discussed later, the collision assessment assumptions listed above do not apply when using PC-based methods.

The rest of this paper is organized as follows. Section 2 provides an overview of polynomial chaos. Section 3 then discusses how methods based on PC may be employed to estimate the probability of collision. The test cases used to demonstrate the benefits of PC are then described in Section 4, with the test results presented in Section 5. Section 6 then concludes the paper.

2. Polynomial Chaos

Methods based on polynomial chaos provide a means for generating an approximation to the solution of a stochastic system by projecting it onto a basis of spectral polynomials. Wiener [26] first proposed this type of approximation, with methods based on Hermite chaos expansions first established [14–17] and later generalized to non-Gaussian input uncertainties [18]. The generated PCE represents the stochastic solution as a linear expansion of multi-variate polynomials that are functions of the input uncertain variables $\xi \in \mathbb{R}^d$ where d is the *stochastic dimension* of the system. Solving for the PCE means estimating the coefficients, or weights, of the expansion. PC-based methods provide a computationally efficient means to represent any square-measurable, possibly non-Gaussian distribution, and have already been demonstrated for orbit propagation [4].

Let $(\mathcal{S}, \mathcal{F}, \mathcal{P})$ be a probability space with the sample space \mathcal{S} and the probability measure \mathcal{P} on σ -algebra \mathcal{F} . The d random inputs to the system are $\xi \in \mathbb{R}^d : \mathcal{S} \rightarrow \Gamma^d \subseteq \mathbb{R}^d$ on $(\mathcal{S}, \mathcal{F}, \mathcal{P})$, where these elements are independent and identically distributed (iid) and characterize the input uncertainties. Additionally, we denote the set of ordinary differential equations (ODEs) that, for the present case case, describe the temporal evolution of a satellite's state, as

$$\mathcal{A}(t, \xi) = 0, \quad (t, \xi) \in [t_0, t_f] \times \Gamma^d, \quad \mathcal{P} - a.s. \text{ in } \mathcal{S} \quad (1)$$

where \mathcal{A} is a stochastic ordinary differential operator and $t \in [t_0, t_f]$ is the temporal variable. This work considers the propagation of orbit state uncertainty using PC with the goal of performing conjunction assessment. Let $\mathbf{X}(t_0)$ be the orbit state at time t_0 with uncertainty that may be represented by the probability space $(\mathcal{S}, \mathcal{F}, \mathcal{P})$. For this discussion, assume $\mathbf{X} \in \mathbb{R}^d$, i.e., the number of state variables equals the number of stochastic inputs to the system. PC-based methods are then used to generate a solution to the stochastic operator \mathcal{A} that describes the space of possible solutions $\mathbf{X}(t_f)$ at any future time t_f .

In the context of PCEs, the orbit solution $\mathbf{X}(t, \xi)$ may be approximated by the finite series

$$\hat{\mathbf{X}}(t, \xi) = \sum_{\alpha \in \Lambda_{p,d}} c_\alpha(t) \psi_\alpha(\xi) \quad (2)$$

where $\hat{\mathbf{X}}$ is the approximate solution to the true value \mathbf{X} , $\Lambda_{p,d} := \{(\alpha_1, \dots, \alpha_d) : \|\alpha\|_1 \leq p, \|\alpha\|_0 \leq d\}$ is the set of multi-indices of size d defined on the non-negative integers, ψ_α are the multi-variate polynomials of maximum degree p ,

$$\psi_\alpha(\xi) = \psi_{\alpha_1}(\xi_1) \psi_{\alpha_2}(\xi_2) \cdots \psi_{\alpha_d}(\xi_d), \quad (3)$$

serve as the basis functions for the approximation, and the vector of coefficients $c_\alpha(t)$ yields the PCE solution. In other words, the multivariate polynomials (of maximum degree p) are simply the product of univariate polynomials, and the degree of each polynomial is defined by the index α_i . We note that a vector representation for c_α effectively allows for the simultaneous representation of d PCEs, i.e. one for each element of the vector $\mathbf{X} \in \mathbb{R}^d$. The number of terms in Eq. (2) is

$$P := \frac{(p+d)!}{p!d!}. \quad (4)$$

For PC methods to apply, the basis $\{\psi_\alpha\}$ must be orthonormal with respect to the random inputs ξ

$$\int_{\Gamma_d} \psi_\alpha(\xi) \psi_\beta(\xi) \rho(\xi) d\xi = \delta_{\alpha\beta} \quad (5)$$

where $\delta_{\alpha,\beta}$ is the Kronecker delta function. Eq. (5) specifies that the random inputs ξ and their probability function $\rho(\xi)$ determines the polynomials ψ_α in Eq. (3). The current paper only considers $\xi \in \mathcal{N}(0, \mathcal{I}_d)$, i.e., the random inputs are multivariate Gaussian with zero mean and unit variance. Hence, ψ_α are Hermite polynomials [18]. Although ξ are Gaussian distributed, the uncertainty of $\mathbf{X}(t_0)$ is not necessarily Gaussian.

The PCE in Eq. (2) provides a method for generating a solution $\hat{\mathbf{X}}$ when given a random input ξ , i.e., it describes the response of $\mathbf{X}(t_f)$ to the realizations ξ_i . In this way, the PCE yields a solution to a stochastic system, as opposed to propagating the a priori state pdf. Information on the a posteriori pdf may then be inferred from the stochastic solution. Given sufficient accuracy of \mathbf{c}_α , the polynomial surrogates also provide a means for bypassing an ODE solver and approximating the Monte Carlo results using the computationally inexpensive polynomial evaluation. This latter property allows for conjunction assessment using PC, which is described further in Section 3.

2.1. PCE Solution Methods

This study only considers non-intrusive methods to generate a global PCE approximating the stochastic solution of the propagated orbit at time t_f . Such methods treat already existing ODE solvers as a black box, and require no alterations to existing orbit propagation software. This allows us to wrap current software to propagate the state uncertainty and perform CA. The algorithm behind the non-intrusive methods may be summarized by:

1. Generate M_{PC} realizations ξ_i . For the current paper, these samples are based on a random sampling of the distribution $\mathcal{N}(0, \mathcal{I}_d)$.
2. For each of the M_{PC} random vectors, use the initial uncertainty in $\mathbf{X}(t_0, \xi_i)$ to generate a realization based on the random input ξ_i . For example, a Cholesky decomposition may be used with the covariance matrix \mathbf{P} and initial mean $\bar{\mathbf{X}}$.
3. Propagate $\mathbf{X}(t_0, \xi_i)$ to t_f using the existing ODE solver, and perform this process for each of the M_{PC} realizations.
4. Solve for the PC coefficients $\mathbf{c}_\alpha(t_f)$ based on $\mathbf{X}(t_f, \xi_i)$ and the method of choice. This yields a PCE $\hat{\mathbf{X}}(t_f, \xi)$.

Several things should be noted about this procedure. First, solutions at intermediate steps, e.g. integration time steps, may be easily generated using the same samples at those times. More samples are only required if p increases at the intermediate time t_i . Hence, $\mathbf{c}_\alpha(t)$ may be generated at multiple times using the *same* samples. Second, there are many methods of generating a non-intrusive PCE solution, including, but not limited to: regression, pseudo-spectral collocation, and compressive sampling [24, 25, 27]. Each of these methods have several advantages and disadvantages which will not be discussed here. The current study considers solutions based on regression. Ongoing studies at CCAR consider other methods, but those either do not apply to the current cases or are not yet fully developed. The next section provides a brief overview of the regression methods.

2.1.1. PCE Solutions via Regression

The method to solve for \mathbf{c}_α based on regression uses random samples ξ_i from the density function $\rho(\xi)$. Given the M_{PC} propagated states, one solution for the coefficients \mathbf{c}_α minimizes the cost function

$$J(\mathbf{c}_\alpha) = \frac{1}{2} \sum_{i=1}^{M_{\text{PC}}} \boldsymbol{\epsilon}_i^T \boldsymbol{\epsilon}_i \quad (6)$$

where

$$\epsilon_i = \hat{\mathbf{X}}(t, \xi_i; \mathbf{c}_\alpha) - \mathbf{X}(t, \xi_i). \quad (7)$$

Eq. (6) is simply the cost function of the least squares estimator. Upon inspection of Eq. (2), we may represent the PCE as a linear system

$$\begin{bmatrix} \psi_{\alpha_1}(\xi_1) & \psi_{\alpha_2}(\xi_1) & \cdots & \psi_{\alpha_P}(\xi_1) \\ \psi_{\alpha_1}(\xi_2) & \psi_{\alpha_2}(\xi_2) & \cdots & \psi_{\alpha_P}(\xi_2) \\ \vdots & \vdots & \ddots & \vdots \\ \psi_{\alpha_1}(\xi_{M_{\text{PC}}}) & \psi_{\alpha_2}(\xi_{M_{\text{PC}}}) & \cdots & \psi_{\alpha_P}(\xi_{M_{\text{PC}}}) \end{bmatrix} \begin{bmatrix} \mathbf{c}_{\alpha_1}^T \\ \mathbf{c}_{\alpha_2}^T \\ \vdots \\ \mathbf{c}_{\alpha_P}^T \end{bmatrix} = \begin{bmatrix} \mathbf{X}^T(t, \xi_1) \\ \mathbf{X}^T(t, \xi_2) \\ \vdots \\ \mathbf{X}^T(t, \xi_{M_{\text{PC}}}) \end{bmatrix}. \quad (8)$$

Instead, one may write

$$\mathbf{H}\mathbf{C} = \mathbf{Y} \quad (9)$$

where \mathbf{H} is the $M_{\text{PC}} \times P$ sensitivity matrix on the left hand side of Eq. (8), \mathbf{C} is the matrix of PCE coefficients, and \mathbf{Y} is comprised of the propagated states at t_f . Given this formulation,

$$\mathbf{C} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{X}, \quad (10)$$

which is the normal equation solution to the least squares estimator. This method of solving for the PCE coefficients requires $M_{\text{PC}} \geq P$ measurements. Also, assuming the same samples are used at all times, \mathbf{H} does not change as a function of t . Hence, $(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T$ remains constant, and solving for $\mathbf{c}_\alpha(t_i)$ at different t_i only requires assembling \mathbf{Y} and evaluating a single matrix multiplication. Such operations may be parallelized to minimize computation time.

2.2. PCE Coordinate System Selection Considerations

As demonstrated in the literature, orbit elements provide a better coordinate system for propagating orbit uncertainty since the pdf remains Gaussian for longer time periods [4–6, 9]. Hence, to reduce d , the PCEs used here represent the orbit state as an element set, e.g., the equinoctial elements. This reduces the number of samples required to generate a PCE.

However, element sets pose a difficulty in performing CA: the mapping from elements to Cartesian *position* is not injective. Specifically, two distinct element set may describe a satellite state at the same position but with different velocities. This results from the fact that all six elements are required to represent a point in \mathbb{R}^3 , while the Cartesian representation only requires three. Hence, there is an inherent ambiguity in the representation of position. For this reason, when generating the PC-based realizations using Eq. (2), the output is then converted to Cartesian coordinates for calculation of the probability of collision, \mathcal{P}_c .

3. Collision Assessment Using PC

3.1. Estimation of \mathcal{P}_c

The methods employed for estimating the probability of collision \mathcal{P}_c using PC are similar to Monte Carlo techniques, but rely on the polynomial surrogates to reduce the overall computation time. The

following discussion begins with a description of Monte Carlo methods, which then serves as a framework for presenting the PC-based methods.

For the state vector

$$\mathbf{X}_l(t_0) = \begin{bmatrix} \mathbf{r}(t_0) \\ \dot{\mathbf{r}}(t_0) \end{bmatrix} \quad (11)$$

let $\bar{\mathbf{X}}_l(t_0)$ be the mean at time t_0 for satellite l with covariance matrix \mathbf{P}_l . In a Monte Carlo test, M_{MC} trials are generated using, for example,

$$\mathbf{X}_l(t_0, \xi_i) = \bar{\mathbf{X}}_l(t_0) + \mathbf{L}_l \xi_i \quad (12)$$

where \mathbf{L}_l is the lower-triangular Cholesky decomposition of \mathbf{P}_l , $i = 1, \dots, M_{\text{MC}}$, and $\xi_i \sim \mathcal{N}(0, \mathcal{I}_d)$. Each trial $\mathbf{X}_l(t_0, \xi_i)$ is then propagated to some time t_k to yield $\mathbf{X}_l(t_k, \xi_i)$. This procedure is conducted for any number of satellites, each with unique ξ_i . Given two satellites, denoted by subscripts 1 and 2, their separation at time t_k is

$$s(t_k, \xi_i, \xi'_i) = \sqrt{\Delta \mathbf{r}_{12}(t_k, \xi_i, \xi'_i) \cdot \Delta \mathbf{r}_{12}(t_k, \xi_i, \xi'_i)} \quad (13)$$

where

$$\Delta \mathbf{r}_{12}(t_k, \xi_i, \xi'_i) = \mathbf{r}_1(t_k, \xi_i) - \mathbf{r}_2(t_k, \xi'_i), \quad (14)$$

and the prime on ξ'_i is only meant to indicate a different set of samples from ξ_i . The empirical probability of collision is then

$$\mathcal{P}_c = \frac{\text{count}(s(t_k, \xi_i, \xi'_i) \leq \mathcal{R})}{M_{\text{MC}}} \quad (15)$$

where \mathcal{R} is a given collision cross-section radius, and the count() operator indicates the number of true results of the argument over $i = 1, \dots, M_{\text{MC}}$. Given M_{MC} samples for each satellite, there are up to M_{MC}^2 possible combinations available for evaluating Eqs. (13)-(15). However, not every comparison will be independent.

Estimating \mathcal{P}_c via PC differs slightly from the above procedure in that using a polynomial surrogate requires the propagation of significantly fewer samples. Instead, when using regression methods to solve for the coefficients c_α , M_{PC} training samples are generated using Eq. (12). Each of these samples are propagated, and the coefficient $c_\alpha(t_k)$ are then generated using Eq. (10). As mentioned earlier, this yields a response surface to generate solutions at t_k when given ξ_i . Hence, M_{MC} evaluations of Eq. (2) with each PCE generates the needed Monte Carlo trials. Eqs. (13)-(15) are then used with the PCE output to estimate \mathcal{P}_c . In other words, evaluations of the orbit propagator are substituted with the more computationally efficient evaluation of a set of polynomials, thereby greatly reducing the computation time of computing \mathcal{P}_c . In general, orbit propagation may take seconds of computer processing time per evaluation, but these polynomial evaluations are on the scale of thousands of evaluations per second.

Of course, the accuracy of \mathcal{P}_c is then limited by the accuracy of the PCE. Methods exist to account for errors in the PCE when calculating a rare failure probability, e.g., satellite collision, by combining PCE evaluations with additional Monte Carlo orbit propagations [28, 29]. However, we do not yet

add these techniques to the current method. Additionally, this work assumes the propagation tool is accurate, and does not account for modeling errors, etc. These elements have been identified as future work.

3.2. Probability of Collision Confidence Interval

Several techniques exist for generating a 1σ confidence interval $\sigma_{\mathcal{P}_c}$. When using Monte Carlo methods, the most accurate, but computationally expensive, method requires repeating a Monte Carlo test N times. To clarify, a single Monte Carlo *test* consists of M_{MC} *trials*. Each test yields a single \mathcal{P}_c , and the empirical standard deviation of these N estimates provides $\sigma_{\mathcal{P}_c}$. Specifically, given N tests,

$$\sigma_{\mathcal{P}_c}^2 = \sum_{j=1}^N \frac{(\mathcal{P}_c^{(j)} - \bar{\mathcal{P}}_c)^2}{N - 1} \quad (16)$$

where the superscript (j) indicates a value produced via Monte Carlo test j and $\bar{\mathcal{P}}_c$ is the mean of all tests. Of course, this is more computationally expensive and requires $M_{MC} \times N$ orbit propagations. Other methods exist to lower the computation cost while computing a confidence interval, but reduce the accuracy of the solution statistics. For example, bootstrap methods use a number of samples, e.g., $2M_{MC}$, and then randomly select M_{MC} trials (with replacement) to serve as a Monte Carlo test. This process is conducted N times to estimate $\sigma_{\mathcal{P}_c}$ while reducing the computation cost, but may yield optimistic confidence intervals. Additionally, assuming \mathcal{P}_c is small, i.e., it represents a rare failure probability, then

$$\sigma_{\mathcal{P}_c} \approx \frac{\mathcal{P}_c(1 - \mathcal{P}_c)}{M_{MC}} \quad (17)$$

The studies of [12, 13] use this final method to approximate $\sigma_{\mathcal{P}_c}$.

In the following sections, the PC-based $\sigma_{\mathcal{P}_c}$ relies on bootstrap techniques to generate N PCE-based estimates of \mathcal{P}_c . In this process, a collection of training points are propagated forward in time to the point of interest. These propagated realizations serve as a bank of samples, from which M_{PC} trials are randomly selected with replacement. This process is repeated N times to yield a collection of PCEs. Each of these PCEs are individually sampled to generate $\mathcal{P}_c^{(j)}$, and then the confidence interval is approximated using Eq. (16).

4. Simulation Methods

4.1. Test Descriptions

The following sections describe tests for two principle scenarios: (A) two satellites in formation with correlated initial uncertainty, and (B) a potential collision between two satellites in low-Earth orbit (LEO) with a non-Gaussian posteriori pdf. The first case, based on the Magnetospheric Multiscale (MMS) mission [30], considers two satellites in a highly eccentric orbit with a relatively small initial uncertainty. Case B demonstrates the application of PC to a scenario more commensurate with current collision avoidance. Initial conditions for these two cases are described in Tab. 1, and the a priori uncertainties are discussed further in the following sections.

Table 1. Initial Conditions in Classical Orbital Elements

Orbit Element	Case A		Case B	
	Sat 1	Sat 2	Molniya	LEO
a (km)	41545.517	41545.487	26462.0	7441.128
e	0.82043	0.82043	0.741	0.06
ι (deg)	27.60	27.61	63.4	119.34
Ω (deg)	10.182	10.181	90.0	265.20
ω (deg)	81.17	81.18	-90.0	204.79
ν (deg)	147.901	147.902	0.0	86.94

4.1.1. Case A: Correlated Satellites with Long Dwell Time

This case considers two satellites flying in formation with a correlated a priori covariance matrix. This correlation violates the assumption that the uncertainties in the two colliding satellites are independent. A histogram of the correlation coefficients for the 12×12 covariance matrix may be found in Fig. 1. The mean correlation coefficient is 0.19 with a maximum value of approximately 0.85. The position and velocity uncertainties in each Cartesian direction are approximately 1 m and 10^{-4} m/s, respectively, at t_0 . These are commensurate with the expected navigation uncertainties for the MSS mission at a similar point in the orbit¹. For this case, \mathcal{R} is 20 m, which differs from the 120 m used for MMS [31]. The current study considers a smaller collision radius to reduce \mathcal{P}_c to approximately 10^{-4} , i.e., the maneuver decision threshold for MMS.

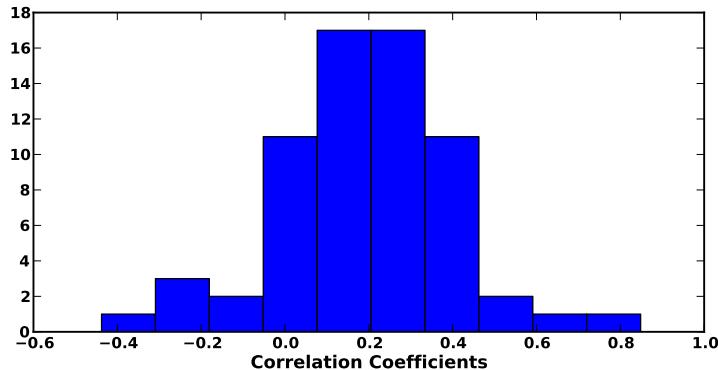


Figure 1. Histogram of correlation coefficients for the Case A satellites.

4.1.2. Case B: Non-Gaussian Conjunction Assessment

This case considers the possible collision of a Molniya and a low-Earth orbit (LEO) satellite. For the Molniya, the case considered in [4] is duplicated. As demonstrated in that work, after a 10-day propagation, the orbit uncertainty in both the Cartesian and Poincaré elements is non-Gaussian. To generate a test case using the already available data, the velocity of the mean Molniya orbit at its perigee at the ten-day mark is rotated about the velocity vector by 180° . This is intended to

¹Personal correspondence with Dan Mattern, a.i. solutions

simulate a nearly head-on collision between satellites represented by the mean states. To constrain the second satellite to LEO, the velocity vector is rescaled to approximately 7.5 km/s (average velocity of a LEO satellite). The position and velocity vector are propagated backwards in time to TCA – 36 hours, which yields the initial mean state. A Gaussian distribution with a RSS 1σ position uncertainty ellipsoid radius of 20 m is then selected for the initial state uncertainty, and matches that of the Molniya orbit [4]. With the mean and covariance now defined for the LEO satellite, the uncertainty is propagated using the PCE methods discussed previously.

4.2. Orbit Propagation Methods

The CU-TurboProp [32] orbit propagation suite serves as the principle orbit simulation tool for this study. It provides a suite of orbit propagation capabilities using both a collection of high- and low-fidelity models of the forces acting on a satellites. Both cases consider the same forces, although with potentially different fidelities appropriate for the given orbit. A description of the force models used may be found in Tab. 2. For Case A, a low-fidelity gravity field is allowed given the high altitude and eccentricity of the orbit. Conversely, the LEO orbit requires a higher degree and order gravity expansion due to the lower altitude. All propagations employ the Dormand-Prince 8(7) [33] integration method implemented in CU-TurboProp with a relative tolerance of 10^{-13} . For all satellites, A/m is $0.01 \text{ m}^2/\text{kg}$.

Table 2. Test Case Force Models

Force	Case A	Case B
Gravity Perturbations	GGM02C, 4×4	GGM02C, 200×200
Sun/Moon Ephemeris		DE405
Atmospheric Density		U.S. Standard 1976, $C_D = 2.2$
Solar Radiation Pressure		Cannonball, $C_R = 1.8$

5. Collision Assessment Results

5.1. Case A Results

This section describes the result for calculating \mathcal{P}_c and $\sigma_{\mathcal{P}_c}$ for Case A. As mentioned previously, this test considers a scenario with two satellites in formation. \mathcal{P}_c is computed via the algorithm described in Section 3.1. To compute $\sigma_{\mathcal{P}_c}$, the bootstrap analysis described in Section 3.2 is conducted to generate 100 PCEs from 300 training samples. To generate each PCE, 26 samples are randomly selected from the training sample pool. The PCEs for this case are expressed in Cartesian coordinates. As discussed below, an orbit element representation is not required for this case. For comparison purposes, $M_{MC} = 10^6$ Monte Carlo samples are also generated. Since the initial uncertainties are coupled and $\xi_i \in \mathbb{R}^{12}$ is used to compute r for both spacecraft, we may not analyze M_{MC}^2 combinations.

Figure 2 illustrates the position pdf for the two satellites at TCA. The realizations used to illustrate the pdf are based on 10^5 Monte Carlo propagations. Given the relatively small standard deviations

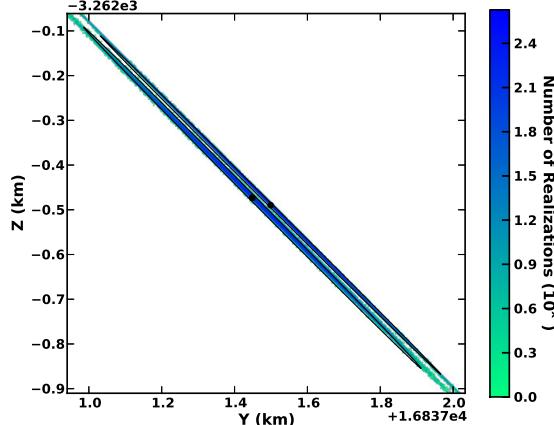


Figure 2. Position and uncertainty for Case A at TCA in the Y - Z plane. The ellipsoids indicate the state uncertainty propagated via the unscented transformation.

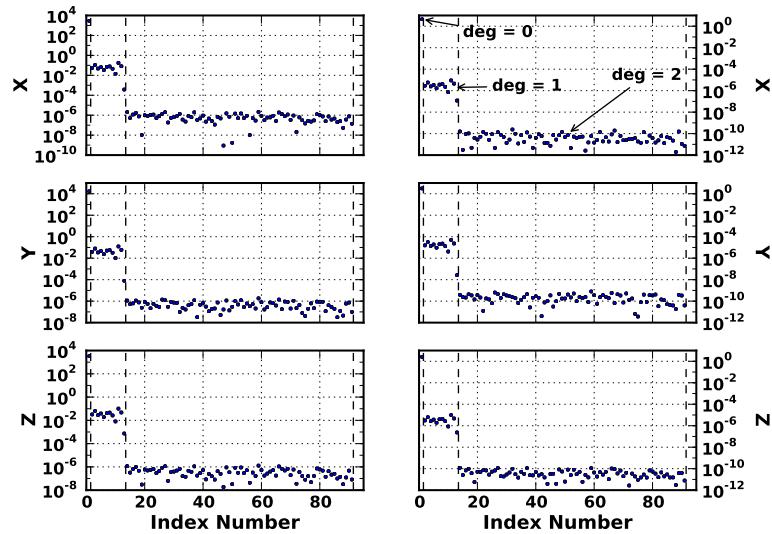


Figure 3. c_α terms for the second satellite in Case A and ordered by polynomial degree.

at t_0 , the propagated pdf at TCA is approximately Gaussian. For this reason, $p = 1$ for the PCEs in Case A will likely be sufficient to calculate \mathcal{P}_c . To confirm this, Fig. 3 demonstrates the rapid decay of the c_α terms for the second satellite. Terms of degree two are 11 orders of magnitude smaller than the degree-zero coefficient. This indicates that they provide little contribution to the stochastic solution. Given their small magnitude, they are likely within the noise of the propagator. Although they are not provided in the interest of brevity, results are similar for the first satellite. For these reasons, the PCEs in the remainder of this section only include terms of degree zero and one. A PCE with $p = 1$ also indicates that the pdf describing the posteriori solution is Gaussian, which is commensurate with Fig. 2.

Figure 4 describes the accuracy of the PCEs generated for Case A. These errors compare the 10^6 Monte Carlo realizations to the values generated using the polynomial surrogates with the

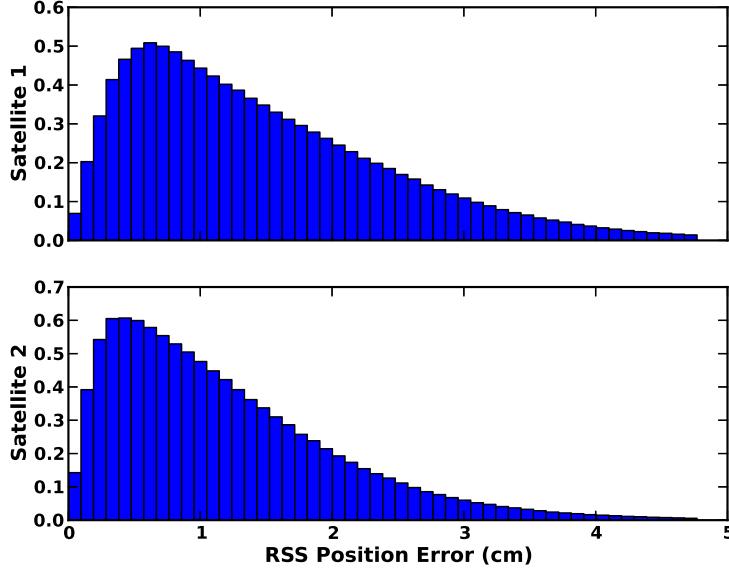


Figure 4. Normalized histogram of evaluation errors for the PCEs of Case A.

same input values ξ_i . The errors are on the order of centimeters, which is almost three orders of magnitude less than \mathcal{R} . The χ^2 distribution is expected for range when the error for each component is approximately Gaussian. Since these errors are small compared to \mathcal{R} , they are not considered a principle factor when assessing the accuracy of \mathcal{P}_c for this case.

The values for \mathcal{P}_c and $\sigma_{\mathcal{P}_c}$ as a function of the number of PCE evaluations and the number of PCE expansions N may be found in Fig. 5. For this test, the number of samples M_{MC} provided to the polynomial surrogates was varied from 10^5 to 10^8 by factors of 10. It is noted that M_{MC} is not the number of training samples for the PCE, but the number of evaluations of the polynomial surrogate. The top left plot describes the convergence of \mathcal{P}_c as a function of N , with a corresponding illustration of $\sigma_{\mathcal{P}_c}$ in the bottom left plot. As described by the bottom left image, the confidence in the solution is driven more by M_{MC} , i.e., the number of evaluations of the PCE. The top right plot compares the values determined via PC to those of 100,000 Monte Carlo trials (independent of the M_{PC} training samples), and provides the computed $\bar{\mathcal{P}}_c$ (solid black line) and 3σ error bars. The 3σ Monte Carlo confidence intervals represent those determined by (a) 10,000 bootstrap samples of the 10^6 available trials, and (b) the low-failure probability approximation given by Eq. (17). Although the performance of the Monte Carlo tests will improve with more samples, these results demonstrate that the PCE-determined statistics lie within 3σ of the current Monte Carlo solution. With 10^8 samples, the PCE-determined value for \mathcal{P}_c , is within the 3σ range $[1.004 \times 10^{-4}, 1.064 \times 10^{-4}]$. Given a requirement to maneuver if $\mathcal{P}_c > 10^{-4}$, this indicates a need for a collision avoidance maneuver. The Monte Carlo solution still exhibits ambiguity in the decision, which is illustrated in the top right image of Fig. 5.

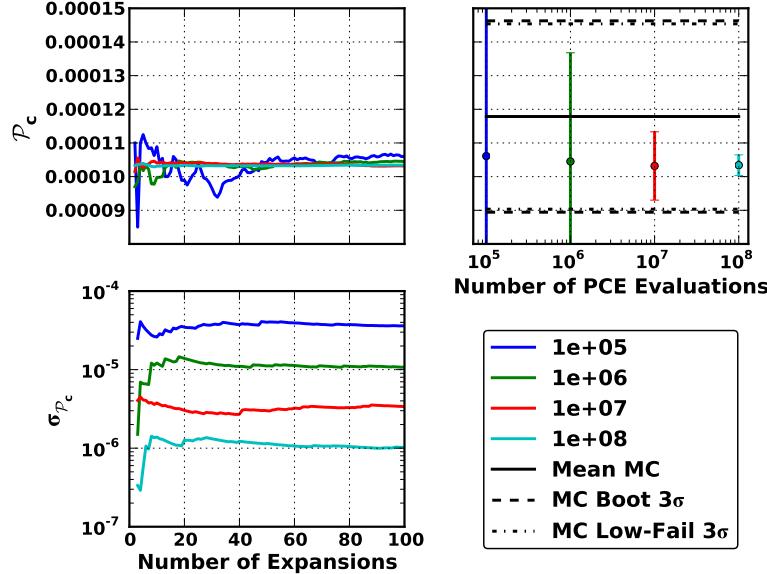


Figure 5. Probability of collision and confidence intervals for Case A.

5.2. Case B Results

This section describes the approximation of \mathcal{P}_c for the second case dealing with non-Gaussian uncertainties. Analysis is not included for $\sigma_{\mathcal{P}_c}$. Figure 6 illustrates the state uncertainties at TCA, with the ellipsoids depicting the propagation of the Gaussian uncertainty via the unscented transformation. Again, the pdfs in this figure use 10^5 realizations of the Monte Carlo test. The satellite with a larger uncertainty volume is the Molniya orbit, with the principle solution variation in the true anomaly component, i.e., the along-track direction. The smaller pdf, which is still non-Gaussian, is for the LEO satellite. The skewness in both solutions, and the poor approximation of the total uncertainty given by the unscented transformation, demonstrates that the Gaussian assumption is not valid for this conjunction.

Table 3. Case B PCE Description

Satellite	p	M_{PC}	M_{MC}
Molniya	3	181	100,000
LEO	3	168	100,000
Monte Carlo	N/A	N/A	100,000 per satellite

Information on the generated PCEs for Case B may be found in Tab. 3. Each PCE requires less than 200 training samples, with 10^5 evaluations of each of the polynomial surrogates used to compute \mathcal{P}_c . The Monte Carlo analysis also uses 10^5 samples of each satellite, but requires that each one be propagated using the ODE solver. Calculation of \mathcal{P}_c below considers all $M_{\text{MC}}^2 = 10^{10}$ possible combinations of the given realizations.

The histograms in Fig. 7 describe the accuracy of the LEO and Molniya PCE evaluations. In both

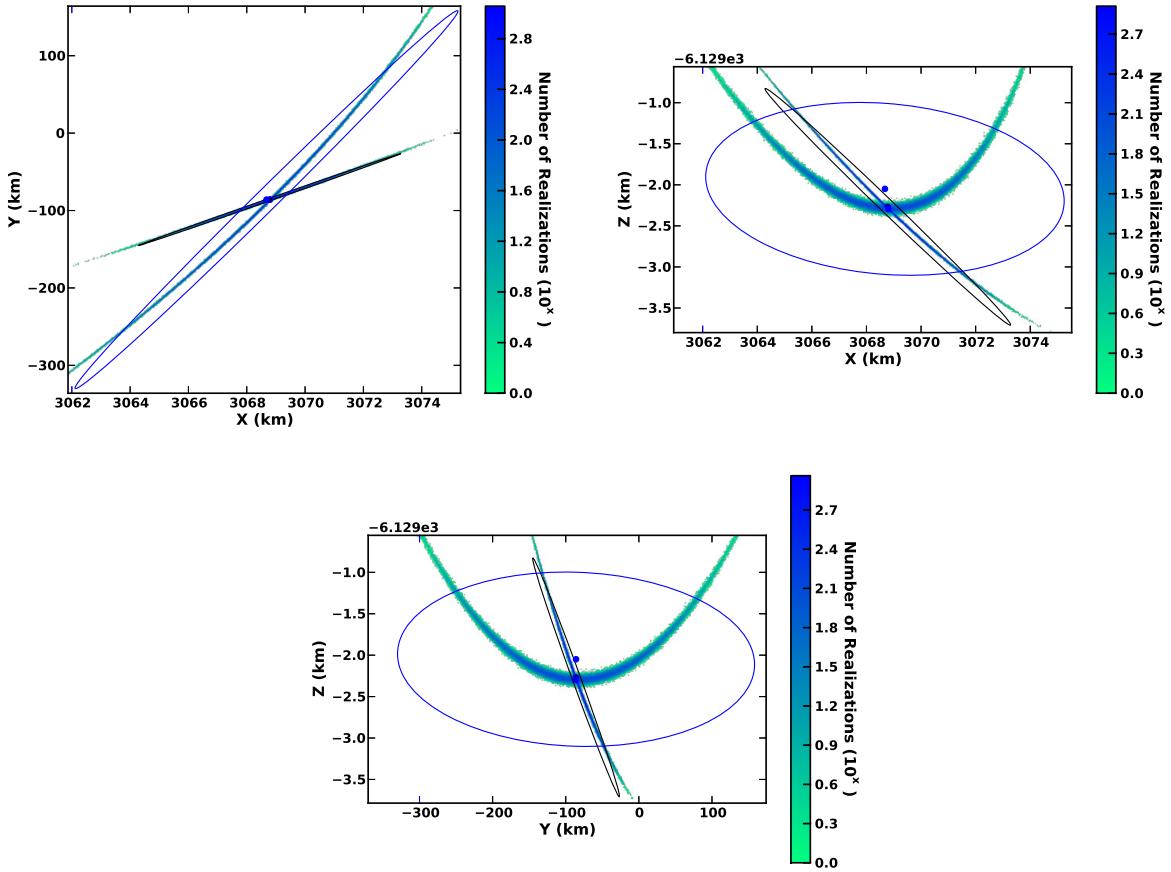


Figure 6. Position pdf for Case B satellites at TCA. The blue and black ellipsoids indicate the propagated Gaussian distribution using the unscented transformation for the Molniya and LEO satellites, respectively.

cases, the largest error may be found in the inertial Y direction. This results from the relatively large coefficient of variation in this component. In such cases, information on the solution is split more evenly among the coefficients c_α , thereby requiring more test samples to generate an accurate solution. When the coefficient of variation is small, then most of the information is contained in the degree zero term, and the accuracy of the solution is primarily, but not solely, determined by this single coefficient. Although not done here, more samples may be used in the generation of the solution to improve accuracy.

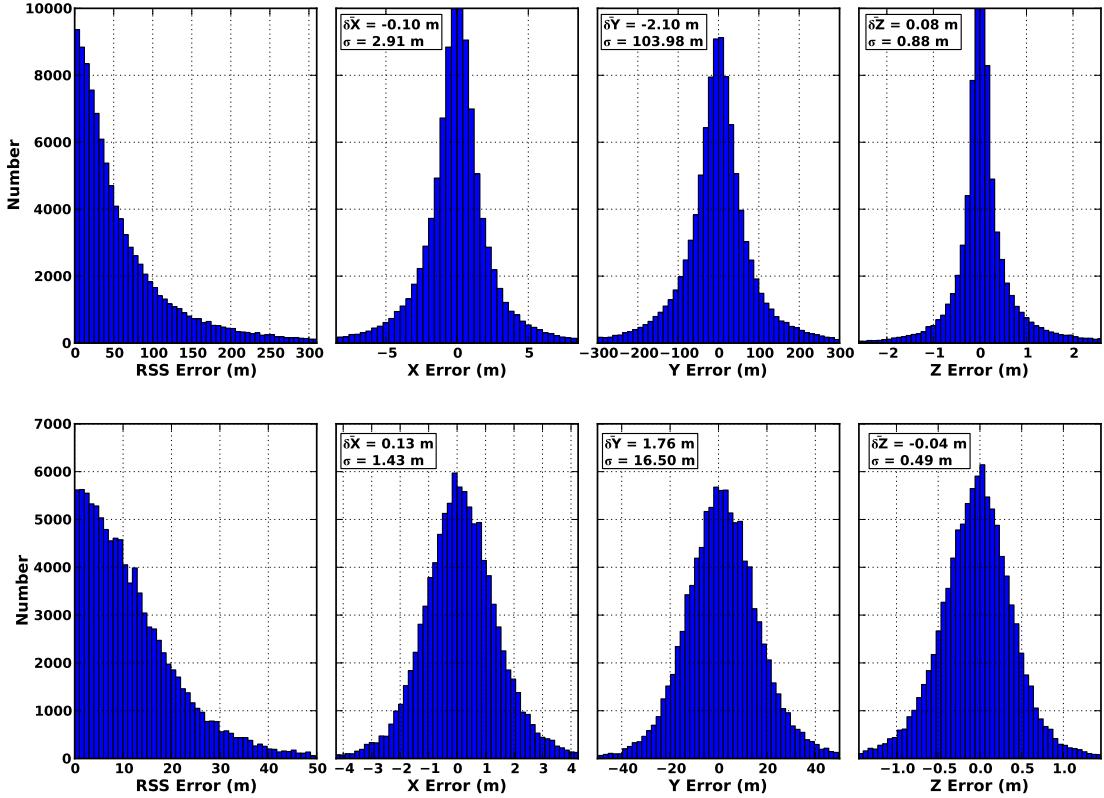


Figure 7. PCE evaluation errors for the Molniya orbit (top) and the LEO orbit (bottom).

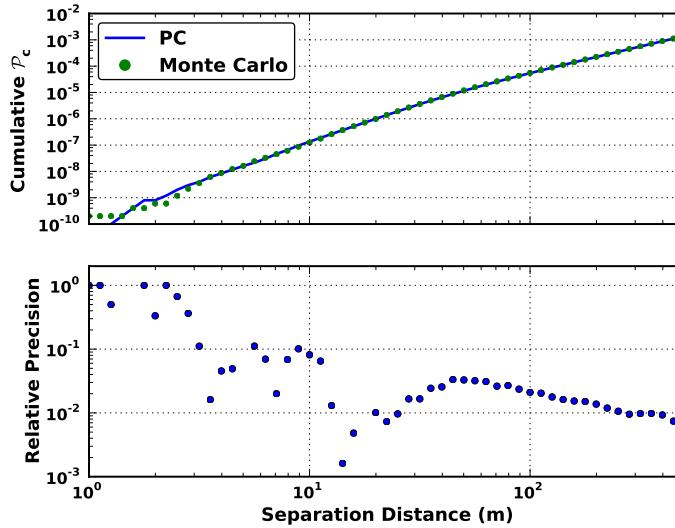


Figure 8. \mathcal{P}_c (top) and agreement between PC and MC solutions (bottom) for Case B.

Figure 8 provides both the PCE- and Monte-Carlo-determined values of \mathcal{P}_c as a function of \mathcal{R} . Even though the Y coordinate error in the PCE is on the order of hundreds of meters, the PCE-based \mathcal{P}_c provides at least one digit of accuracy or better for $\mathcal{R} \geq 10 \text{ m}$. However, more accurate PCEs may be required to resolve the smaller values of \mathcal{P}_c . Additionally, since $1/M_{MC}^2 = 10^{-10}$, the Monte Carlo

solution for $\mathcal{P}_c \sim O(10^{-10})$ is likely also inaccurate. The accuracy of the PCE may be improved by using additional training samples. However, methods exist to improve the accuracy of rare failure probability calculation using a hybrid of polynomial surrogates and additional evaluations of the ODE solver [28, 29], and such techniques may be employed to limit the number of additional orbit propagations required.

6. Conclusions

This paper demonstrated the use of polynomial chaos expansions (PCEs) for conjunction assessment. The PCE provides a polynomial surrogate to map random inputs at the initial time to generate a realized state at some future time. This allows for Monte-Carlo-like conjunction assessment using polynomial evaluations instead of computationally expensive orbit propagations. This work considered two test cases designed to demonstrate the use of PCEs in situations where semi-analytic methods of estimating the probability of collision are not applicable. For both of these test cases, the PCEs provided a probability of collision with enough accuracy to make an informed maneuver avoidance decision. Additionally, each individual PCE required less than 300 evaluations of the orbit propagation tool, which is at least a factor of 330 less than the Monte Carlo results presented. The accuracy of the collision probability may be improved with a careful selection of additional orbit propagations, with methods identified and designated as future work.

7. Acknowledgements

The authors wish to thank Russell Carpenter (NASA/GSFC), Geoff Wawrzyniak, and Dan Mattern (a.i. solutions) for helping to understand the MMS mission, which served as the basis for Case A. Discussions with them also helped in refining the presentation of polynomial chaos to make it more readily accessible to those not familiar with the subject. This work was partially funded by the NASA GSFC through a.i. solutions.

8. References

- [1] Alfriend, K. T., Akella, M. R., Frisbee, J., Foster, J. L., Lee, D.-J., and Wilkins, M. “Probability of Collision Error Analysis.” *Space Debris*, Vol. 1, No. 1, pp. 21–35, 1999.
- [2] Patera, R. P. “General Method for Calculating Satellite Collision Probability.” *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, pp. 716–722, July-August 2001.
- [3] Alfano, S. “A Numerical Implementation of Spherical Object Collision Probability.” *The Journal of the Astronautical Sciences*, Vol. 53, No. 1, pp. 103–109, January - March 2005.
- [4] Jones, B. A., Doostan, A., and Born, G. H. “Nonlinear Propagation of Orbit Uncertainty Using Non-Intrusive Polynomial Chaos.” *Journal of Guidance, Control, and Dynamics*, accepted, 2012.
- [5] Fujimoto, K., Scheeres, D. J., and Alfriend, K. T. “Analytical Nonlinear Propagation of Uncertainty in the Two-Body Problem.” *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 2, pp. 497–509, March-April 2012.

- [6] Fujimoto, K. and Scheeres, D. J. “Non-Linear Propagation of Uncertainty with Non-Conservative Effects.” “22nd Annual AAS/AIAA Spaceflight Mechanics Meeting,” Charleston, SC, Jan. 29 - Feb. 2 2012.
- [7] DeMars, K. J., Jah, M. K., Giza, D., and Kelecy, T. “Orbit Determination Performance Improvements for High Area-To-Mass Ratio Space Object Tracking Using an Adaptive Gaussian Mixtures Estimation Algorithm.” “21st International Symposium on Space Flight Dynamics,” Toulouse, France, September 28 - October 2 2009.
- [8] Horwood, J. T., Aragon, N. D., and Poore, A. B. “Gaussian Sum Filters for Space Surveillance: Theory and Simulations.” *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 6, pp. 1839–1851, November - December 2011.
- [9] Sabol, C., Sukut, T., Hill, K., Alfriend, K. T., Wright, B., Li, Y., and Schumacher, P. W., Jr. “Linearized Orbit Covariance Generation and Propagation Analysis Via Simple Monte Carlo Simulations.” “20th Annual AAS/AIAA Spaceflight Mechanics Meeting,” San Diego, California, February 15-17 2010.
- [10] Park, R. S. and Scheeres, D. J. “Nonlinear Mapping of Gaussian Statistics: Theory and Applications to Spacecraft Trajectory Design.” *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, pp. 1367–1375, November - December 2006.
- [11] Sabol, C., Binz, C., Segerman, A., Roe, K., and Schumacher, P. W., Jr. “Probability of Collision with Special Perturbations Dynamics Using the Monte Carlo Method.” “AAS/AIAA Astrodynamics Specialist Conference,” Girdwood, AK, July 31 - August 4 2011.
- [12] Dolado, J. C., Legendre, P., Garmier, R., Revelin, B., and Pena, X. “Satellite Collision Probability Computation for Long Term Encounters.” “AAS/AIAA Astrodynamics Specialist Conference,” Girdwood, AK, July 31 - August 4 2011.
- [13] Garmier, R., Dolado, J., Pena, X., Revelin, B., and Legendre, P. “Collision Risk Assessment for Multiple Encounters.” “European Space Surveillance Conference,” Madrid, Spain, June 7-9 2011.
- [14] Ghanem, R. G. and Red-Horse, J. “Propagation of probabilistic uncertainty in complex physical systems using a stochastic finite element approach.” *Physica D: Nonlinear Phenomena*, Vol. 133, No. 1-4, pp. 137–144, 10 September 1999.
- [15] Ghanem, R. G. and Spanos, P. D. *Stochastic Finite Elements: A Spectral Approach*. Springer-Verlag, New York, 1991.
- [16] Ghanem, R. G. and Dham, S. “Stochastic Finite Element Analysis for Multiphase Flow in Heterogeneous Porous Media.” *Transport in Porous Media*, Vol. 32, No. 3, pp. 239–262, 1998.
- [17] Ghanem, R. G. “Ingredients for a General Purpose Stochastic Finite Elements Implementation.” *Computer Methods in Applied Mechanics and Engineering*, Vol. 168, No. 1-4, pp. 19–34, January 1999.

- [18] Xiu, D. and Karniadakis, G. E. “The Wiener-Askey Polynomial Chaos for Stochastic Differential Equations.” *SIAM Journal of Scientific Computing*, Vol. 24, No. 2, pp. 619–644, 2002.
- [19] Xiu, D. and Karniadakis, G. E. “Modeling Uncertainty in Flow Simulations Via Generalized Polynomial Chaos.” *Journal of Computational Physics*, Vol. 187, pp. 137–167, 2003.
- [20] Le Maître, O. P., Najm, H. N., Ghanem, R. G., and Knio, O. M. “Multi-resolution analysis of Wiener-type uncertainty propagation schemes.” *Journal of Computational Physics*, Vol. 197, No. 2, pp. 502–531, 1 July 2004.
- [21] Babuška, I., Tempone, R., and Zouraris, G. E. “Galerkin Finite Element Approximations of Stochastic Elliptic Partial Differential Equations.” *SIAM Journal on Numerical Analysis*, Vol. 42, No. 2, pp. 800–825, 2005.
- [22] Wan, X. and Karniadakis, G. E. “An adaptive multi-element generalized polynomial chaos method for stochastic differential equations.” *Journal of Computational Physics*, Vol. 209, No. 2, pp. 617–642, 1 November 2005.
- [23] Xiu, D. “Fast Numerical Methods for Stochastic Computations: A Review.” *Communications in Computational Physics*, Vol. 5, No. 2-4, pp. 242–272, February 2009.
- [24] Xiu, D. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, Princeton, New Jersey, 2010.
- [25] Le Maître, O. P. and Knio, O. M. *Spectral Methods for Uncertainty Quantification with Applications to Computational Fluid Dynamics*. Springer, 2010.
- [26] Wiener, N. “The Homogeneous Chaos.” *American Journal of Mathematics*, Vol. 60, No. 4, pp. 897–936, October 1938.
- [27] Doostan, A. and Owhadi, H. “A Non-adapted Sparse Approximation of PDEs with Stochastic Inputs.” *Journal of Computational Physics*, Vol. 230, No. 8, pp. 3015–3034, 20 April 2011.
- [28] Li, J. and Xiu, D. “Evaluation of failure probability via surrogate models.” *Journal of Computational Physics*, Vol. 229, No. 23, pp. 8966–8980, November 20 2010.
- [29] Li, J., Li, J., and Xiu, D. “A efficient surrogate-based method for computing rare failure probability.” *Journal of Computational Physics*, Vol. 230, No. 24, pp. 8683–8697, October 1 2011.
- [30] Sharma, A. S. and Curtis, S. A. “Magnetospheric Multiscale Mission.” “Nonequilibrium Phenomena in Plasmas,” Vol. 321 of *Astrophysics and Space Science Library*, pp. 179–195. Springer Netherlands, 2005.
- [31] Carpenter, J. R., Markley, F. L., Alfriend, K. T., Wright, C., and Arcido, J. “Sequential Probability Ratio Test for Collision Avoidance Maneuver Decisions Based on a Bank of Norm-Inequality-Constrained Epoch-State Filters.” “AAS/AIAA Astrodynamics Specialist Conference,” Girdwood, AK, July 31 - August 4 2011.

- [32] Hill, K. and Jones, B. A. TurboProp Version 4.0. Colorado Center for Astrodynamics Research, May 2009.
- [33] Prince, R. J. and Dormand, J. R. “High order embedded Runge-Kutta formulae.” *Journal of Computational and Applied Mathematics*, Vol. 7, No. 1, pp. 67–75, March 1981.