

Finite-Dimensional Density Representation for Aerocapture Uncertainty Quantification

Samuel W. Albert*, Alireza Doostan[†], and Hanspeter Schaub[‡]
University of Colorado Boulder, Boulder, CO, 80303

Aerocapture is a problem of high interest for future interplanetary missions. It is dominated by a high sensitivity to a number of uncertainties, especially atmospheric density variability. The common state of practice for uncertainty quantification in this context is the Monte Carlo simulation methodology. A number of methods exist which may be superior to Monte Carlo methods but require a lower finite-dimensional representation of the random inputs. A Karhunen–Loève expansion approximation of perturbed atmospheric density is presented, which by reducing the dimensionality of the atmosphere model could enable the application of more advanced techniques such as polynomial chaos expansion. This representation is then demonstrated by implementation in a Monte Carlo simulation of entry dynamics for aerocapture. Early results of applying compressed sensing polynomial chaos expansion to this problem are also presented, and compared with the Monte Carlo baseline. Compressed sensing-based polynomial chaos expansion is shown to converge faster than Monte Carlo for this problem, and filtering out trajectories that impact the surface is shown to further improve convergence, but further validation is needed.

I. Introduction

PLANETARY entry is a unique domain characterized by nonlinear dynamics and high sensitivity to error and uncertainty. The most important source of this uncertainty is the inherent variability and limited knowledge of the atmosphere. As a vehicle transitions from deep-space flight to hypersonic atmospheric flight, the uncertainties inherent in both regimes conspire to make accurate prediction and navigation impossible without appropriate quantification and propagation of those uncertainties. Aerocapture is a subset of planetary entry problems and serves as the example problem of this paper. By harnessing aerodynamic forces from the atmosphere, a spacecraft performing aerocapture decreases its energy enough to capture into orbit without a large propulsive orbit insertion burn. As shown in Figure 1, the vehicle goes from an incoming hyperbolic orbit, passes through the upper atmosphere reducing energy by the desired amount, then exits the atmosphere and coasts to apoapsis where it performs a periapsis raise maneuver to prevent re-entry. The vehicle then performs apoapsis and out-of-plane correction maneuvers as necessary.

The most relevant uncertainties to aerocapture are grouped into three general categories: the vehicle (drag/lift coefficients, mass, center of gravity location, etc.), the navigation state (entry flight path angle, entry velocity, etc.), and the atmosphere (density, winds, etc. along the trajectory). Every category must be accounted for and each has its own challenges, but the atmosphere poses some unique difficulties. The vehicle and navigation state uncertainties are mostly represented by random variables, i.e. the mass of the vehicle is a single random scalar and the position vector a random vector. The atmosphere profiles, however, are represented by random fields, i.e. they map a random outcome to a continuous function in space. The key difference is that because a random field is defined by a set of n th-order probability density functions (pdfs) for all positive integers n , the representation is not finite-dimensional. For this reason, the approach to atmosphere modeling is especially relevant to consider here before further discussing uncertainty quantification (UQ).

Many planetary entry studies model the atmospheric density and its perturbations in one of two ways: using an exponential model [1–3] or relying on an existing empirical atmosphere model [4–6], such as a Global Reference Atmospheric Model (GRAM) from NASA or the Mass Spectrometer and Incoherent Scatter radar (MSIS) model from

*Graduate Student, Ann and H.J. Smead Aerospace Engineering Sciences, Colorado Center for Astrodynamics Research, 3775 Discovery Drive, 429 UCB - CCAR, Boulder, CO 80303. AIAA Student Member.

[†]H. Joseph Smead Faculty Fellow, Ann and H.J. Smead Aerospace Engineering Sciences, Aerospace Mechanics Research Center, University of Colorado, Engineering Center, ECAE 189, Campus Box 429, Boulder, CO 80309. AIAA Member.

[‡]Professor, Glenn L. Murphy Chair in Engineering, Ann and H.J. Smead Aerospace Engineering Sciences, Colorado Center for Astrodynamics Research, 3775 Discovery Drive, 429 UCB - CCAR, Boulder, CO 80303. AIAA and AAS Fellow.

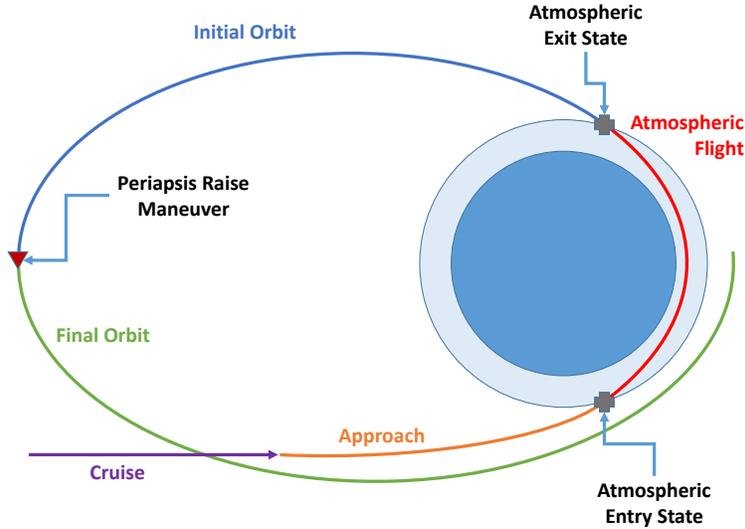


Fig. 1 Diagram of the basic aerocapture process

the US Naval Research Laboratory. While sufficient in a number of cases, both of these approaches have their limitations. The exponential model is useful because it provides a reasonably accurate approximation of how density varies with altitude while reducing the density profile to a function of two scalar parameters, surface level density and atmospheric scale height [7]. A common approach to creating dispersed density profiles is to model these two parameters as random (usually Gaussian) variables, selecting a new pair of values for each simulation. However, this approach is primarily useful as a simplified approximation enabling analytical approaches. As shown in Figure 2 even a comparison of only nominal scenarios reveals significant disagreement between the exponential profile and a model such as Earth-GRAM. Furthermore, dispersing only the surface density and scale height will always retain the same exponential shape of the density profile and simply scales the result in either direction. These two limitations highlight why a more realistic modeling approach is necessary for many scenarios.

The other common approach is using one of several existing models which can return density and other parameters of interest (pressure, temperature, etc.) based on user inputs, GRAM and MSIS being two common choices. While these models are still limited, in general they are more realistic and more flexible than a simple exponential model. The GRAM tools, for example, can take solar activity parameters as inputs in addition to location, date, time, and others. Importantly, they also have a built-in capability to randomly generate profiles with physically realistic perturbations. In the context of uncertainty quantification, the major limitation of GRAM and models like it is that they are often treated as a black box analysis component. That is, the analyst generates a large number of random perturbed profiles and then randomly selects a profile to use for each simulation in a Monte Carlo simulation setting. By relying on random sampling techniques like Monte Carlo and simply selecting full pre-generated profiles, the analyst has implicitly forgone the implementation of other UQ techniques which, in some cases, may have outperformed Monte Carlo.

The issue is that many UQ techniques, such as stochastic collocation [8, 9] and polynomial chaos expansion (PCE) [10, 11], require low stochastic dimensionality (i.e., a relatively small number of dispersed input parameters). Any computer model of a continuous atmosphere will discretize the output profile and this may be considered a random vector with a dimensionality equal to the number of discretization steps, e.g. an altitude profile with data every 500 meters from 0 to 125 km would have a dimensionality of 251 in this sense. Thus, in general a model with reasonable resolution quickly exceeds the low stochastic dimensionality requirement when simply relying on discretization for finite-dimensionality, precluding the use of PCE or similar techniques. Therefore, the high dimensionality of this black box modeling approach effectively limits the analyst to a Monte Carlo approach to uncertainty quantification even for cases where a different UQ approach might otherwise be more appropriate.

The goal of this paper is to implement a lower-dimensional yet realistic representation of perturbed density profiles by representing density using a Karhunen–Loève expansion. This enables the implementation of other UQ techniques like PCE without sacrificing the realism of the atmosphere model. This density model is applied in a baseline Monte Carlo simulation of an entry dynamics problem, dispersing inputs around a nominal aerocapture scenario. Finally, preliminary results of applying compressed sensing polynomial chaos expansion to this aerocapture problem are presented.

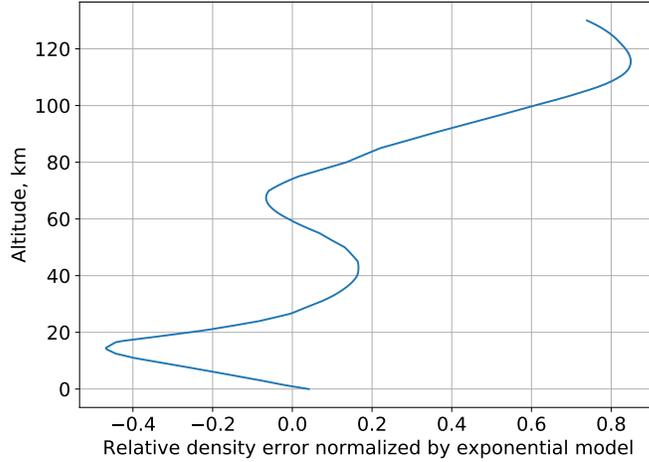


Fig. 2 Relative error between nominal density profiles from the exponential model and Earth-GRAM2016

II. Modeling Atmospheric Density

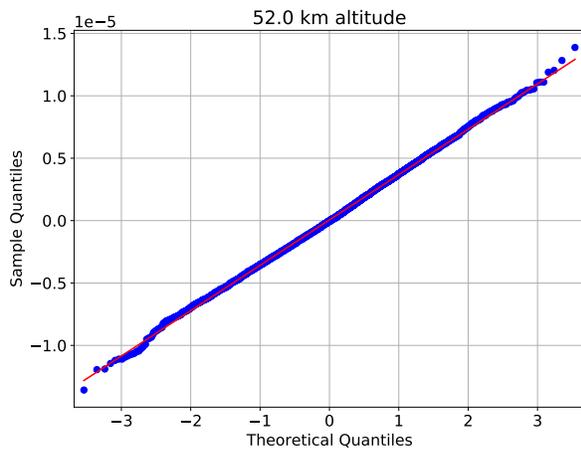
A. Mars-GRAM

For this paper, Mars-GRAM2010 is used as an atmosphere model to generate a dataset of density profiles. Mars-GRAM has the attractive feature of an internal capability to disperse profiles, and has the additional benefit that GRAMs also exist for Earth, Titan, Venus, and Neptune. In generating profiles a columnar atmosphere is assumed and thus latitude and longitude are not incremented in the profiles, only altitude - but note that this assumption is for the sake of simplicity, it is not a requirement for the approach presented here. The data used for this paper are generated for an arbitrary date and location of February 18th 2021 at 18.38°N, 77.58°W, at altitudes from the surface to 130 km in steps of 100 m. The optical depth of background dust setting is 0.3, with dust diameter of 5 μm and particle density of 3000 kg m^{-3} . The 10.7 cm solar flux is set to 68 sfu, and the random density perturbation scale is set to 1. The working assumption for this study is that while changes in these model settings may affect the character of the density profile, they would not significantly impact the applicability of the Karhunen–Loève expansion representation of density, with the possible exception of the random density perturbation scale. 50,000 density profiles are randomly generated with the above settings using Mars-GRAM2010’s built-in Monte Carlo capability.

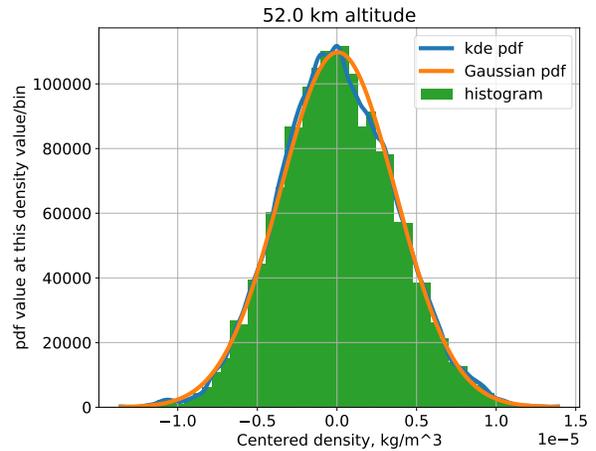
The normality of the data is assessed using quantile-quantile (q-q) plots and by comparing empirical histograms to Gaussian pdfs with the sample mean and variance of the data as shown for two representative altitudes in Figs. 3 and 4. These results show that the density dispersions are fairly Gaussian in nature, although there is some notable skewness that develops for higher altitudes. For this purpose of this paper, the density perturbations in Mars-GRAM2010 are treated as a Gaussian random field, an imperfect but reasonable assumption based on results like those shown in Figs. 3 and 4. Note that this assumption is not a requirement for the approach taken in this study, but does simplify the process significantly. Future work will consider potential improvements to this density modeling step, such as fitting a log-normal distribution to the data instead. Finally, note that there is a theoretical error in describing density as Gaussian: since the support of any Gaussian random variable is $(-\infty, \infty)$, the variable has a nonzero probability of being negative, yet the physical parameter density has a lower-bound of zero. Revising the structure of the density approximation outlined below to implement a truncated Gaussian is another step for future work, but note that no negative density values have actually been generated in this work because they have very low probabilities of occurring.

B. Karhunen–Loève Expansion for Density

A Karhunen–Loève Expansion (KLE) represents a random field through an infinite linear combination of orthogonal basis functions (a Fourier expansion), in such a way that the choice of the basis functions minimizes the mean-square error [12]. This definition is shown by Eq. 1 where Z is the random field, h is the independent variable (not necessarily scalar), $\langle \cdot \rangle$ is the expected value function, and λ_i and $\phi_i(h)$ are the eigenvalues and eigenfunctions of the covariance

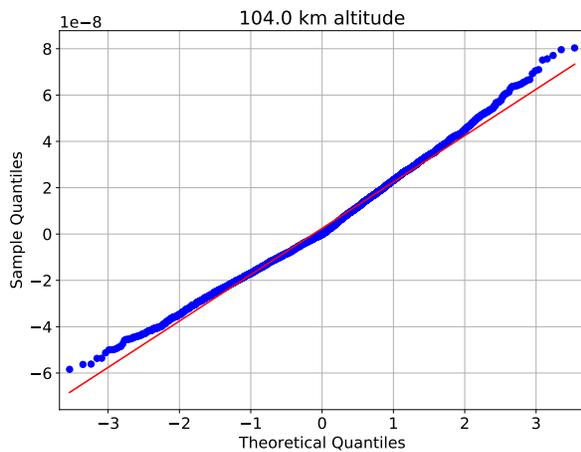


(a) Quantile-quantile plot of centered density, density minus sample mean density

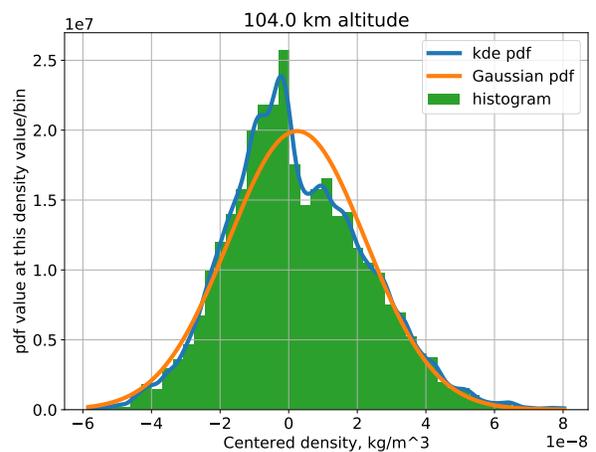


(b) Histogram, kernel density estimated (kde) pdf, and Gaussian pdf using sample mean and variance of centered density

Fig. 3 Centered density at 52 km, showing nicely Gaussian behavior at low/mid altitudes



(a) Quantile-quantile plot of centered density, density minus sample mean density



(b) Histogram, kernel density estimated (kde) pdf, and Gaussian pdf using sample mean and variance of centered density

Fig. 4 Centered density at 104 km, showing some positive skewness at higher altitudes

function of the random field $C_{ZZ}(h_1, h_2)$, respectively, as shown in Eq. 2. Finally, each Y_i is a random variable described by Eq. 3.

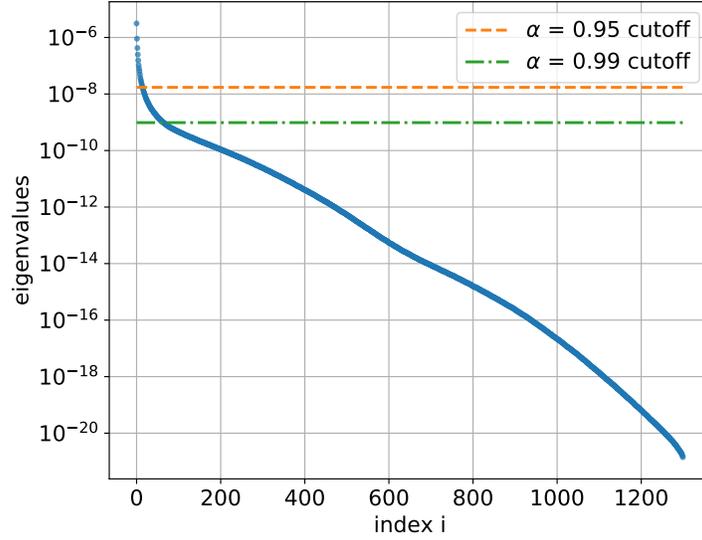


Fig. 5 Exponential decay of the eigenvalues for the density KLE. Cutoff at $\alpha = 0.95$ results in $d_K = 14$; cutoff at $\alpha = 0.99$ results in $d_K = 64$.

$$Z(h) = \langle Z(h) \rangle + \sum_{i=1}^{\infty} \sqrt{\lambda_i} \phi_i(h) Y_i \quad (1)$$

$$\int_0^T C_{ZZ}(h_1, h_2) \phi_i(h_2) dh_2 = \lambda_i \phi_i(h_1) \quad (2)$$

$$Y_i = \frac{1}{\sqrt{\lambda_i}} \int_0^T Z(h) \phi_i(h) dh \quad (3)$$

In practice, the eigenvalues and eigenfunctions are sorted by descending magnitude of the eigenvalues and then the sum in Eq. 1 is truncated after some d_K number of sufficient terms. Determining the required d_K is problem-dependent, but in general d_K is chosen such that the mean-square norm of the approximation is within some relative error of the exact mean-square norm. In this study the heuristic method described in Eq. 4 is applied, where k is some sufficiently large number and α is close to 1 based on the desired level of permissible error (for a relative mean-square norms error of $(1 - \alpha) \times 100\%$).

$$d_K = \min \left\{ j : \frac{\sum_{i=1}^j \lambda_i}{\sum_{i=1}^{j+k} \lambda_i} \geq \alpha \right\} \quad (4)$$

Using this Mars-GRAM dataset, a value of $\alpha = 0.95$ results in $d = 14$ dimensions, while a value of $\alpha = 0.99$ results in $d = 64$ dimensions. Results are generated for both of these α values, referenced as the lower- and higher-dimension cases, respectively. The sorted eigenvalues and the truncation points for the lower- and higher-dimension cases are shown in Fig. 5.

Because the density profiles are reasonably well-modeled by a Gaussian random field, the expression for finding Y_i can be simplified. In fact, when $Z(h)$ is Gaussian the Y_i 's are all independent and identically distributed (i.i.d.) standard normal random variables:

$$Y_1, Y_2, \dots \sim \mathcal{N}(0, 1) \text{ i.i.d.} \quad (5)$$

In order to estimate the eigenvalues and eigenfunctions of the discrete variant of this random field, its covariance matrix must first be estimated. To that end, the sample covariance matrix is computed using the full dataset generated from Mars-GRAM2010. The unbiased estimate of the sample covariance matrix \mathbf{Q}_{ZZ} is shown in Eq. 6, where \mathbf{Z}_c is a matrix such that each column is an observation vector which has had the sample mean (a vector of the mean sample density at each altitude step) subtracted from it. The sample mean is used rather than treating the nominal output of GRAM as a population mean in order to keep the approach based entirely on the provided perturbed data, and thus more model-agnostic.

$$\mathbf{C}_{ZZ} \approx \mathbf{Q}_{ZZ} = \frac{1}{N-1} \mathbf{Z}_c \mathbf{Z}_c^T \quad (6)$$

Having computed a covariance matrix, it is straightforward to find the eigenvalues and eigenvectors of that matrix and sort them according to descending order of the eigenvalues, and the results are the λ_i and $\phi_i(h)$ in Eq. 1, respectively. Thus, all of the components to generate KLE realizations of the density profile are found from the GRAM data, repeated below for a single random trial substituting density as a function of altitude $\rho(h)$ as the random field:

$$\rho(h) \approx \langle \rho(h) \rangle + \sum_{i=1}^{d_K} \sqrt{\lambda_i} \phi_i(h) Y_i \quad Y_i, \dots, Y_d \sim \mathcal{N}(0, 1) \quad (7)$$

III. Uncertainty Quantification

This section presents the preliminary results of implementing the KLE representation of density variability in an uncertainty quantification framework for the aerocapture problem. In general, a standard Monte Carlo random sampling approach is very popular for this type of analysis [4, 6, 13, 14]. In addition to being simple to understand and implement, the greatest merit of Monte Carlo is that it avoids the so-called *curse of dimensionality* — that is, exponential growth in the number of samples required to achieve convergence as the (stochastic) dimensionality of the problem increases. The mean-square error of Monte Carlo is given in Eq. 8, where σ is the standard deviation of the QoI. The dimensionality of the problem d does not appear in Eq. 8, and thus the mean-square convergence of a Monte Carlo estimate is formally independent of the dimension of the problem [15]. Increasing the number of dispersed inputs does tend to increase the QoI variance and thus the number of samples required, but even so this strong performance for very high-dimensional problems sets Monte Carlo alone in terms of UQ techniques.

$$\langle (\bar{u}_N - \mu)^2 \rangle^{1/2} = \frac{1}{\sqrt{N}} \sigma \quad (8)$$

Though Eq. 8 captures the boon of Monte Carlo, it also reveals its bane: a convergence rate of $O(1/\sqrt{N})$ is relatively slow. This slow convergence limits Monte Carlo because, inherently, this technique oversamples the mean and undersamples the tails, making it an inefficient choice in the common scenario in which an analyst must understand behavior in the tails to pass or fail some requirement.

While popular, Monte Carlo is far from the only technique available to quantify the effect of uncertainty on some system model. Latin-hypercube sampling [16, 17], multi-level Monte Carlo [18, 19], sparse-grid stochastic collocation [8, 9], and polynomial chaos expansion [10, 11] all offer alternatives, to name just a very few. It is difficult to build a proper taxonomy of these methods, but a generalization is that these techniques offer faster convergence than Monte Carlo in exchange for some assumptions about the problem. However, in general these methods are also vulnerable to the curse of dimensionality, meaning their convergence rate suffers as the number of dimensions increases, to the point that Monte Carlo may out-perform them.

Though the KLE representation successfully provides a finite-dimensional representation of density variability that is still higher-fidelity than an exponential model, it results in $d_K = 14$ dimensions even with 5% mean-square error, or a much higher $d_K = 64$ dimensions for 1% mean-square error. Moreover, d_K only accounts for the dimensions from the KLE itself; an aerocapture scenario will require at least a few other dispersed inputs to be worthwhile, and may include tens or hundreds of additional dispersed inputs depending on the application and level of fidelity. Thus, even for a simple application with $d_K = 14$ and $d_i = 4$ other dispersed inputs, the 18-dimensional uncertainty effectively rules-out many UQ techniques [20]. This is the motivation for selecting polynomial chaos expansion (PCE) via compressed sensing for this problem; while it is not immune to the curse of dimensionality, compressed sensing-based PCE performs relatively well at high dimensions compared to other non-Monte Carlo techniques, while still promising faster convergence than Monte Carlo for a range of problems [21–23]. Fundamentally, the potential benefit of applying this method to the

aerocapture problem is to gain useful estimates of the statistics of QoIs with significantly fewer samples than would be required to form estimates of the same accuracy using standard Monte Carlo. The central question of this section could then be summarized as follows: how many samples does it take for the PCE method to stabilize/converge, and at that number of samples, how does the Monte Carlo estimate accuracy compare?

A. Entry Dynamics Simulation

To compare the performance of Monte Carlo and PCE for the aerocapture problem, all 130,000 simulations are completed using a simple entry dynamics simulation tool developed for this task, Petunia*. The simulation for this study is 3 degrees of freedom, and assumes constant aerodynamic coefficients, constant mass and no thrust, and point-mass gravity. In general gravity, drag, and lift forces all affect the trajectory of the vehicle, but for simplicity this study used a passive ballistic vehicle with zero lift. Gravity is computed as Eq. 9 where m is the mass of the vehicle, μ is the gravitational parameter of the Earth, \mathbf{r} is the position vector of the spacecraft and \mathbf{F}_g is the gravitational force vector.

$$\mathbf{F}_g = -\frac{m\mu}{|\mathbf{r}|^3}\mathbf{r} \quad (9)$$

$$|\mathbf{F}_D| = \frac{1}{2}\rho V_\infty^2 C_D A \quad (10)$$

The magnitude of the drag force is defined in Eq. 10 where C_D is the drag coefficient, A is the reference area, and V_∞ is the magnitude of the wind-relative velocity (not to be confused with the hyperbolic excess velocity). The drag force is oriented directly opposite the direction of this wind-relative velocity. For this study winds are neglected, and the atmosphere is assumed to rotate as if fixed to the planet. An ODE solver is then used to numerically integrate according to Newton's 2nd Law, $m\ddot{\mathbf{r}} = \sum \mathbf{F}$ inertially. The simulation integrates until one of two exit conditions is met. If the vehicle passes downward through some minimum altitude, set to 20 km for Mars, then the simulation terminates and that trajectory is categorized as an impact. Note that these trajectories also have a negative final flight path angle (are still traveling downwards), whereas the rest of the trajectories have a positive final flight path angle. If the vehicle passes upwards through a maximum altitude, defined as 135 km for Mars, then the simulation terminates. If the orbit mechanical energy at termination is less than zero, the trajectory was a successful aerocapture, and if the energy is positive then the vehicle is still on a hyperbolic trajectory and will escape.

Using this simulation tool, a nominal aerocapture at Mars is designed and distributions defined for several dispersed parameters. Each trajectory begins at the same location, with altitude equal to the atmospheric interface, defined to be 125 km for Mars. The drag coefficient C_D , mass m , entry flight path angle (EFPA), and initial velocity magnitude v are all randomly selected according to their distributions for each run. For the purpose of this study each of these parameters is modeled using a Gaussian distribution because this simplifies the PCE process. Some of these parameters, specifically C_D and m , would likely be better modeled as uniform distributions, and making this adjustment is planned for near-term future work.

For the current study, C_D has a mean of 1.59 and a variance equivalent to that of a uniform distribution with bounds of $\pm 10\%$. Likewise, m has a mean of 3000 kg and a variance equivalent to that of a uniform distribution with bounds of $\pm 5\%$. Again, note that neither of these values can physically be negative, yet they are modeled with Gaussian distributions. A uniform distribution or truncated Gaussian would remedy this problem, but for the purpose of these results note that negative values for drag coefficient or mass are never actually realized. EFPA and initial velocity magnitude are more naturally modeled using Gaussian distributions; EFPA has mean -10.6° and standard deviation $\sigma = 0.2^\circ/3$, while v has mean 6 km s^{-1} and standard deviation $\sigma = 10/3 \text{ m s}^{-1}$. These four distributions are all selected to be in the range of physically realistic values used in previous studies [24], but are then tuned such that most scenarios successfully completed aerocapture, at least some scenarios escaped, and some impacted the surface. This is done to reflect an early-stage study or parameter sweep where this high variance in results may occur, and ensures that the UQ approach tested in this study would still apply in those scenarios. Finally, the area A of the vehicle is not itself dispersed, but is instead computed by using a fixed ballistic coefficient of $\beta = 129 \text{ kg m}^{-2}$ then solving Eq. 11 to get the area given the current mass and drag coefficient values. This results in a nominal area of 14.6 m^2 . Table 1 summarizes these parameters. Two datasets are generated using the above simulation tool and input dispersions. A lower-dimensional dataset of 70,000 trajectories is generated using $\alpha = 0.95$, and a higher-dimensional dataset of 60,000 trajectories is generated using $\alpha = 0.99$.

*github.com/salbert21/petunia

$$\beta = \frac{m}{C_D A} \quad (11)$$

Table 1 Summary of simulation input parameters

Parameter	Mean	3- σ
C_D	1.59	0.275
m	3000 kg	259.8 kg
EFPA	-10.6°	0.2°
v	6 km s ⁻¹	10 m s ⁻¹
$\rho(h)$	Mars-GRAM2010 via KLE	Mars-GRAM2010 via KLE
β	129 kg m ⁻²	~

B. Bifurcation in the Aerocapture Problem

There is one other feature of the aerocapture problem that should be addressed before discussing the PCE implementation. When the variances of the dispersed parameters are sufficiently large, some trajectories may fail to aerocapture by impacting the surface, while others may fail by escaping (still having positive orbital energy at atmospheric exit). In general, the quantities of interest (QoIs) vary smoothly between aerocapture cases and escaped cases. However, trajectories that intersect the surface are terminated at a low altitude after they have already slowed down a great deal during a long atmospheric flight. The final states of these scenarios thus exist in a fundamentally different regime than the scenarios that either aerocapture or escape. This is reflected in QoIs that capture the final state of the vehicle, resulting in a bifurcation in the problem whenever the initial dispersions are large enough.

This bifurcation behavior is illustrated in Fig. 6. Figure 6a shows the discontinuity in final flight path angle and final orbital energy for the 60,000-case higher-dimensional Monte Carlo dataset. Figure 6b is generated for a nominal aerocapture scenario by incrementing the EFPA from -10.3° to -11.5° in steps of -0.025° while holding all other parameters constant (including the density profile). The shallowest trajectories, shown in red, are escape trajectories, which vary smoothly to become aerocapture trajectories, until in the green region the EFPA is steep enough that the trajectories impact the surface from there until the steepest magenta trajectory. Fig. 6b illustrates how this bifurcation unfolds over time; the impact cases follow a smooth transition at first, but continue decelerating deeper into the atmosphere until reaching a much lower final energy/velocity by the time the vehicle reaches the minimum altitude. The higher the inputs dispersions are, the more significant this bifurcating behavior will become, and it transitions from a negligible probability to some notable outliers then eventually to a bi-modal output distribution as the input variability is steadily increased.

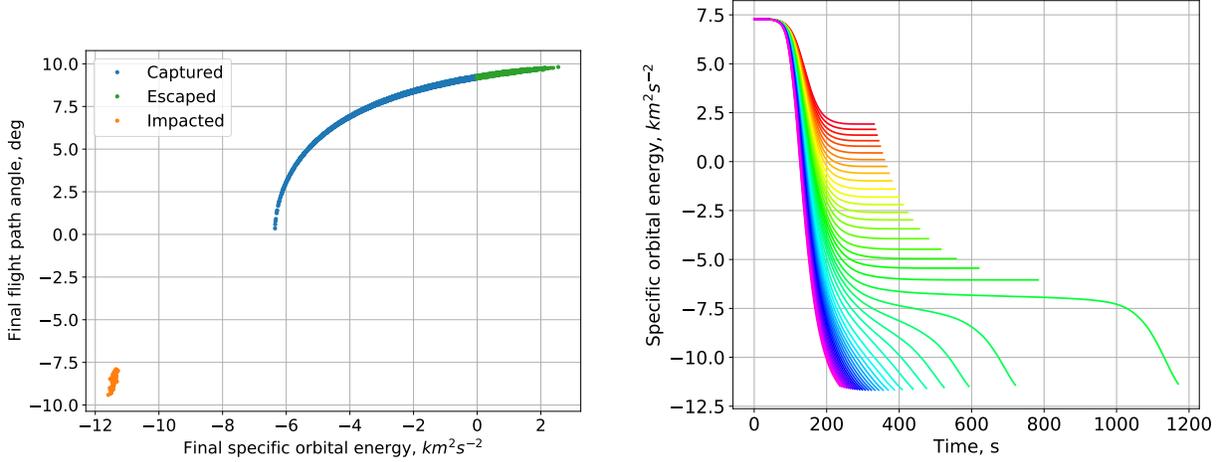
The bifurcation is relevant because the PCE method implemented in this study requires that the QoI depend smoothly on the dispersed inputs. Clearly, this discontinuity violates that assumption. Thus, if the input dispersions are large enough that the chance of impact is non-negligible (as is true for the dispersions summarized in Table 1), then the impact cases may disrupt the convergence of PCE by violating the smooth dependence assumption. The impact of this phenomenon is examined later by generating results for *filtered* and *unfiltered* datasets, where the filtered datasets have all impact trajectories removed before processing.

C. Polynomial Chaos Expansion

Polynomial chaos expansion works by representing the QoI $u = u(\mathbf{y})$ of a system as a generalized Fourier series expansion in a multi-dimensional polynomial basis orthonormal with respect to the joint probability distribution of the inputs \mathbf{y} , $\mathcal{P}_{\mathbf{y}}$:

$$\hat{u}(\mathbf{y}) = \sum_{j=0}^P c_j \psi_j(\mathbf{y}) \xrightarrow{\text{m.s.}} u(\mathbf{y}), \text{ as } P \rightarrow \infty \quad (12)$$

where in Eq. 12 u has finite variance and the approximation converges in the mean-square sense to the exact QoI value as the number of terms $P + 1$ approaches infinity. The Fourier coefficients c_j are given by Eq. 13 where $\langle \rangle$ is the



(a) Results from 60,000-case Monte Carlo with $\alpha = 0.99$ for $d = 68$ total dimensions (b) Results of gradually decreasing EFPA for an aerocapture simulation. Red is shallowest, magenta is steepest.

Fig. 6 Demonstrations of the bifurcation inherent in the aerocapture problem

expected value operator:

$$c_j = \langle u(\mathbf{y}) \psi_j(\mathbf{y}) \rangle \quad (13)$$

The orthonormal basis functions that form this approximation space are chosen based on the probability measure of the random inputs, \mathcal{P}_y . For this application, after applying a KLE representation to atmospheric density variability in Section II, the random inputs to this problem are d i.i.d. Gaussian random variables, where $d = d_K + d_i$ is the number of KLE terms for density plus the number of other scalar dispersed inputs. In this case, the basis functions are orthonormal Hermite polynomials [11]. Letting $\psi_{j_k}(\mathbf{y}_k)$ index these 1D polynomials in \mathbf{y}_k by their degree, $j_k = 0, 1, \dots$ for $k = 1, \dots, d$, the d -dimensional polynomials are defined by Eq. 14:

$$\psi_j(\mathbf{y}) = \prod_{k=1}^d \psi_{j_k}(\mathbf{y}_k) \quad (14)$$

Then, the expansion is truncated to a basis of *total order* p , by considering only those indices j_k for which $\sum_{k=1}^d j_k \leq p$. With this truncation method, the expansion will have $P + 1$ total terms as given by Eq. 15, noting that the number of expansion terms then grows exponentially with the number of dimensions d .

$$P + 1 = \frac{(p + d)!}{p!d!} \quad (15)$$

It is important to note that the quality of the approximation in Eq. 12 depends on the regularity of the QoI $u(\mathbf{y})$ with respect to the dispersed inputs \mathbf{y} . For an infinitely smooth $u(\mathbf{y})$ the convergence is exponential, but for non-smooth behavior the convergence can become slow or fail to converge, and is susceptible to Gibbs phenomenon.

The preceding steps provide a method for approximating the QoI, but have not covered how to compute the coefficients c_j beyond providing Eq. 13, which may be inefficient or infeasible to compute directly. Two fundamentally different approaches exist for solving for these coefficients: *intrusive* and *non-intrusive* methods. Intrusive methods such as Galerkin projection can be highly efficient [10], but by definition they require modifying the deterministic simulation code for the system of interest. This would also require modification whenever fidelity is added to the model or additional dispersed inputs are included. Thus, non-intrusive methods, which treat the deterministic solver as a black-box, are desirable.

There are a variety of non-intrusive methods to solve for the unknown PCE coefficients [25]. The basic approach is to generate some number of realizations of the QoI, then use these pilot trials to estimate the coefficients. For N samples

of the QoI, Eq. 12 can be arranged in matrix form as shown in Eq. 16. Note that the dimensions of the measurement matrix Ψ are $N \times (P + 1)$.

$$\begin{bmatrix} \psi_0(\mathbf{y}_1) & \cdots & \psi_P(\mathbf{y}_1) \\ \vdots & \ddots & \vdots \\ \psi_0(\mathbf{y}_N) & \cdots & \psi_P(\mathbf{y}_N) \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_P \end{bmatrix} \approx \begin{bmatrix} u(\mathbf{y}_1) \\ \vdots \\ u(\mathbf{y}_N) \end{bmatrix} \Rightarrow \Psi \mathbf{C} \approx \mathbf{U} \quad (16)$$

Solving the matrix equation in Eq. 16 via the standard least squares provides an estimate of the coefficients vector \mathbf{C} for an overdetermined problem, where the number of samples exceeds the number of coefficients ($N > (P+1)$). However, because the number of coefficients grows rapidly with the dimensionality as shown in Eq. 15, for high-dimensional problems this may require sufficiently many samples that the PCE convergence is no better than standard Monte Carlo, especially if a high truncation order p is required. This makes it desirable to apply methods to solve Eq. 16 for the underdetermined problem. In this case, however, there are infinitely many solutions, and the minimum ℓ_2 -norm solution becomes unstable under truncation error. This reveals a need for regularization of the problem.

Compressed sensing (also known as compressive sampling) provides one solution to this problem [21–23]. This approach enforces sparsity of the coefficient vector by applying appropriate norms. Initially, the solution with minimum ℓ_0 -norm is considered, where $\|\mathbf{C}\|_0 = \#\{j : c_j \neq 0\}$:

$$\hat{\mathbf{C}} = \arg \min_{\mathbf{C}} \|\mathbf{C}\|_0 \text{ s.t. } \Psi \mathbf{C} = \mathbf{U} \quad (17)$$

However, a number of issues arise here. The optimization is non-convex, the solution is not always unique, and this minimum is NP-hard to compute. A number of heuristic workarounds have been developed to address the complexity of finding this sparsest approximation. The approach that is taken in this study is convex relaxation via ℓ_1 -minimization. This solution seeks instead the minimum ℓ_1 -norm solution as defined in Eq. 18, where $\|\mathbf{C}\|_1 = \sum_{j=0}^P |c_j|$:

$$\hat{\mathbf{C}} = \arg \min_{\mathbf{C}} \|\mathbf{C}\|_1 \text{ s.t. } \Psi \mathbf{C} = \mathbf{U} \quad (18)$$

Eq. 18 is now a convex optimization problem, and may be solved using standard linear programs. Under some conditions, the minimum ℓ_1 -norm solution is unique and identical to the sparsest solution [26].

In practical implementation, there is truncation error, so quadratic programming can be employed to solve Eq. 19 for some tolerance $\delta > 0$. For this study, δ is selected by defining $\delta = \sigma \|\mathbf{U}\|_2$, where \mathbf{U} is a vector of N samples of the QoI \mathbf{u} . σ is tuned manually, as described further in Section III.D.

$$\hat{\mathbf{C}} = \arg \min_{\mathbf{C}} \|\mathbf{C}\|_1 \text{ s.t. } \|\Psi \mathbf{C} - \mathbf{U}\|_2 \leq \delta \quad (19)$$

A number of tools exist for ℓ_1 -minimization. The tool used for this study is the basis pursuit denoise (BPDN) problem solver provided by SPGL1: Spectral Projected Gradient for L1 minimization [27], a Python port of the original MATLAB solver [28] and based on the theory outlined by van den Berg and Friedlander [29].

Once the PCE coefficients are known or estimated, the mean and variance of the QoI can be trivially computed as shown in Eqs. 20 and 21. Though not part of this study, further stochastic properties of the QoI such as its cumulative distribution function or probability density function can be estimated by inexpensively sampling Eq. 12 now that the coefficients are known. Furthermore, sensitivity analysis is a natural byproduct of a PCE solution, and Sobol indices can be easily determined from the coefficients [30, 31].

$$\langle \mathbf{u} \rangle \approx \langle \hat{\mathbf{u}} \rangle = c_0 \quad (20)$$

$$\sigma^2 \approx \sum_{i=1}^P c_i^2 \quad (21)$$

D. Results and Discussion

There are many parameters in the aerocapture problem that could be relevant quantities of interest, including peak heat-rate, peak deceleration load, total integrated heat load, final apoapsis radius, final energy, etc. Furthermore, a variety of statistics could be of interest for each of these QoIs, such as the 99% confidence interval for peak heat-rate

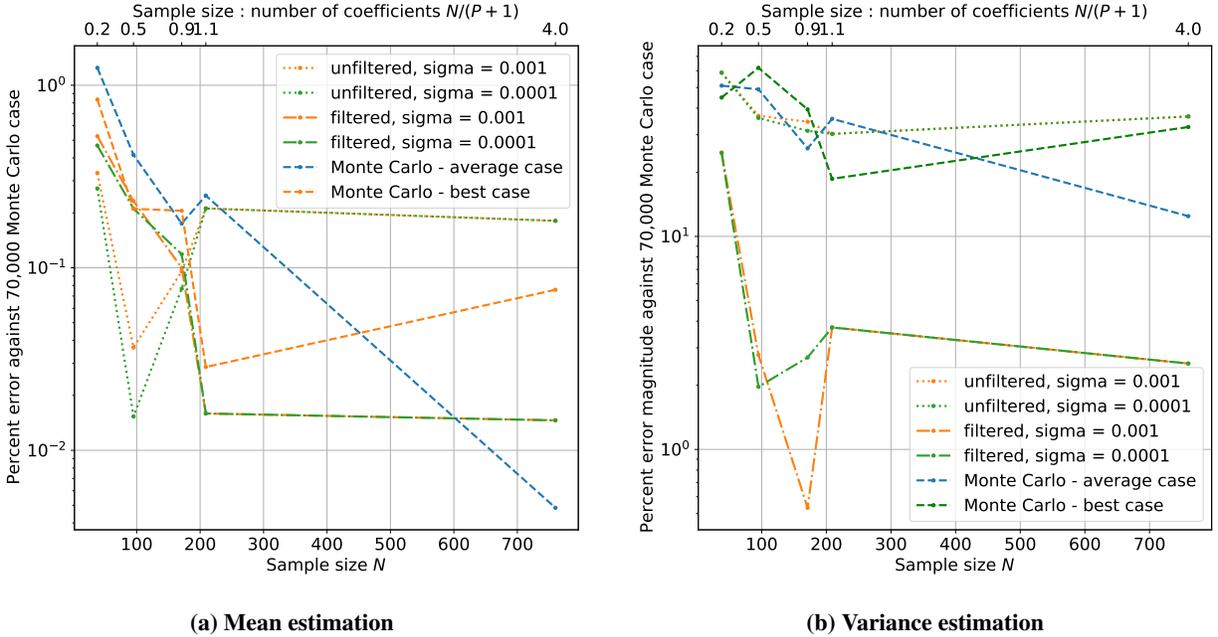


Fig. 7 Final flight path angle results for $\alpha = 0.95$, $p = 2$

or the sensitivity of the final apoapsis radius to density variability. To simplify the scope for a more straightforward comparison, in this study two QoIs are considered, the final flight path angle and final specific orbital energy. For each of these two QoIs, the statistics of interest are simply the mean and variance.

The results of estimating these QoIs by applying compressed sensing PCE to the aerocapture problem are summarized in Figs. 7 - 12. In each case, the sample mean or variance computed from the full 60,000- or 70,000-sample Monte Carlo dataset is used as the truth value for comparison, because this large number of samples was shown to be more than enough to achieve convergence. The y-axis of each plot then displays the absolute value of the percent error of each estimate compared to this full Monte Carlo value. Note the log scale for the y-axis. While the actual number of samples included at each point are labeled on the lower x-axis, the upper x-axis provides the sample size in terms of the factor of the number of coefficients $P + 1$; i.e., when this value is 0.9 the problem is underdetermined, and at 1.1 it is overdetermined and SPGL1 returns a least-squares estimate.

The unfiltered cases consider all samples and are compared against the full Monte Carlo dataset. The filtered cases do not include any impact trajectories, and are compared against a filtered version of the full Monte Carlo dataset that has also had the impact cases removed. When computing all PCE results, the first N samples (or the first N non-impact samples, for filtered results) in the full dataset are used to generate the PCE estimate. Note that this leaves the PCE results somewhat vulnerable to the particular makeup of this set of samples compared to the full dataset; this is addressed further in Section IV. To capture the results for various tunings of the tolerance parameter σ , PCE results are computed for $\sigma = \{0.01, 0.001, 0.0001\}$ in each case. The two best-performing results are then selected for display. The PCE results should be able to perform at least as well as the best-performing value of σ when properly optimized, as discussed further in Section IV.

The Monte Carlo lines represent the Monte Carlo estimate for just N samples, compared to the Monte Carlo estimate using all 60,000 or 70,000 samples. These estimates are always unfiltered. Because Monte Carlo is in general more sensitive to the makeup of the particular set of samples, for each plot 10 Monte Carlo lines are generated using N randomly selected (uniform without replacement) samples from the total dataset. Two of these ten results are then selected for display, one with the best convergence and another with average convergence. This approach is meant to somewhat mitigate the effect of the random makeup of a single set of N samples where N is small.

Figs. 7 and 8 gives the results for a KLE with $\alpha = 0.95$, which results in a total dimensionality of $d = 18$, for a 2nd-total-order truncation of the PCE expansion. Figs. 9 and 10 provide the same but for a 3rd-total-order truncation of the PCE expansion. Finally, Figs. 11 and 12 show the results for a KLE with $\alpha = 0.99$, which results in a total dimensionality of $d = 68$, for a 2nd-total-order truncation of the PCE expansion. Note that since the KLE implementation

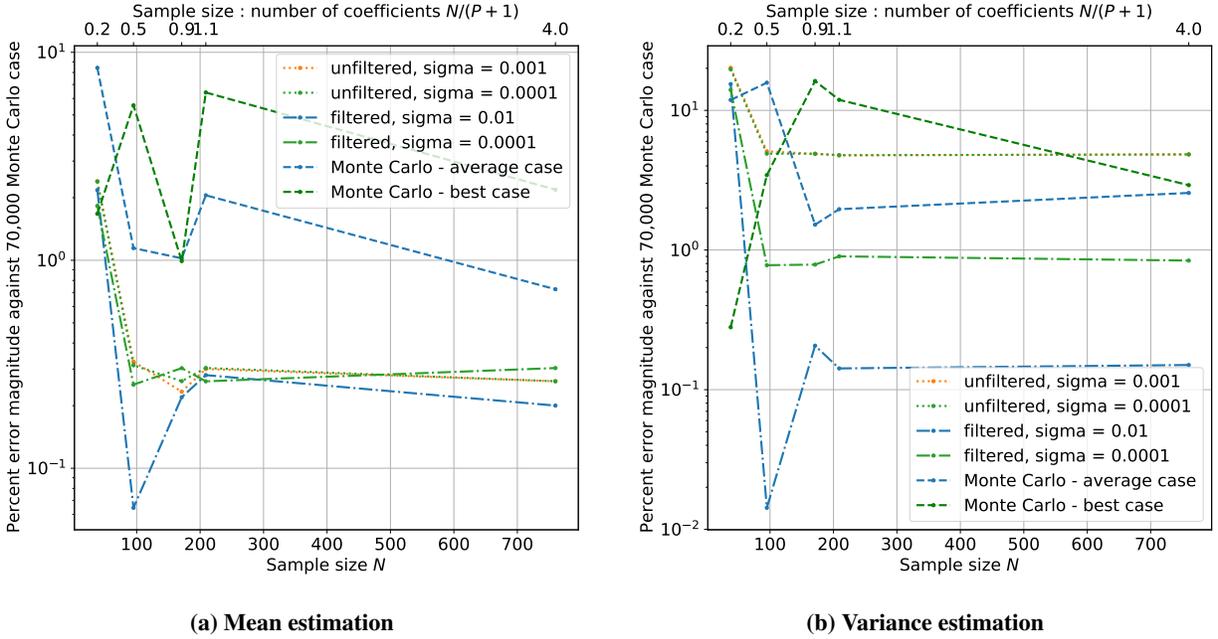


Fig. 8 Final orbital energy results for $\alpha = 0.95$, $p = 2$

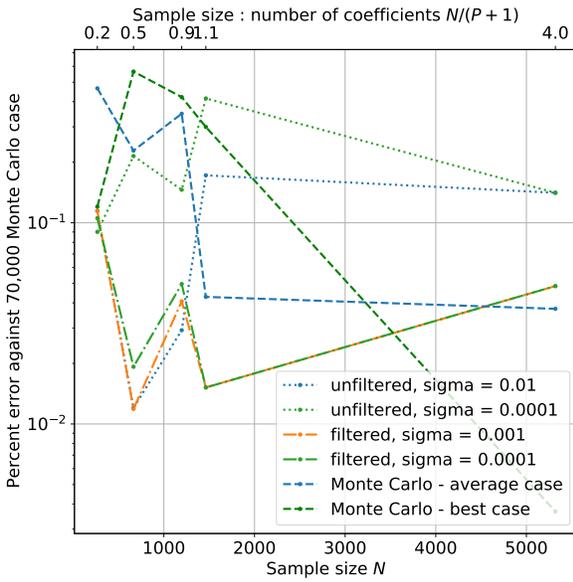
is different for this last case, the Monte Carlo dataset is also different, though the input dispersion settings are identical.

These results are preliminary, and further work is needed to make definitive statements about the performance of Monte Carlo vs. PCE for this problem. However, some early observations can be made. In general, Monte Carlo estimates will steadily decrease in error as the number of samples is increased. For PCE, the p -total-order truncation limits the minimum error of the estimate, so although the convergence may be faster than Monte Carlo the estimates will then plateau. This behavior is observed in some cases, most clearly in Fig. 10a. However, the expected trend is often obscured by the randomness inherently present in these results; while the convergence predicts a monotonic decrease in error for the Monte Carlo results, this is often not the case due to outliers present in small sample sizes.

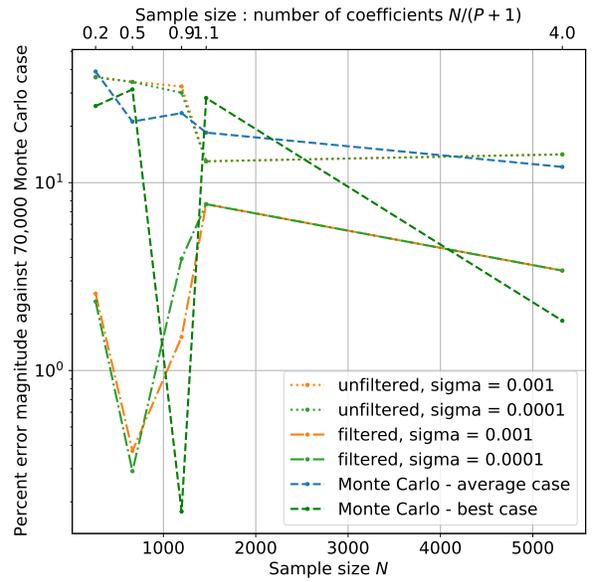
In general, the filtered PCE results do perform the best of the three categories. The unfiltered results are typically worse than the filtered results as expected, with the exceptions of Figs. 8a and 12a where they perform roughly equally. In some ways, the more surprising result is that the unfiltered cases do not perform significantly worse than they did — after all, they clearly violate the PCE assumption of a smooth input-QoI relationship. This seems to be because the impact cases represent such a small number of the total trajectories, only about 0.1%. If the input dispersions had significantly higher variance, or if the problem setup were otherwise changed such that the impact cases represent a larger portion of the results, this discontinuity may become a larger obstacle. Thus, the importance of the bifurcation issue is strongly dependent on the specific problem and input dispersions.

Comparing the results for 2nd- and 3rd-total-order truncation between Figs. 7 & 8 and Figs. 9 & 10, respectively, the $p = 3$ results do not appear to outperform Monte Carlo by as much overall. This comparison represents an interesting tradeoff in that $p = 3$ does provide more accurate PCE results, but it also significantly increases the number of coefficients and thus the number of samples required for a given sample size factor. Because the number of samples required is nearly an order of magnitude larger, Monte Carlo also in general performs better. The choice of total order truncation is strongly application dependent; e.g., is a 1% error acceptable or not for the mean estimate? The order of truncation required to achieve the desired accuracy then affects the Monte Carlo vs. PCE tradeoff.

Comparing the $\alpha = 0.95$ results with the $\alpha = 0.99$ results, overall the higher-dimensional cases perform better than expected. The larger sample sizes required again drive better Monte Carlo performance, but unlike increasing p , increasing α only improves the accuracy of the KLE approximation, not the PCE convergence. While the results in Figs. 11 and 12 are perhaps incrementally worse than the corresponding $\alpha = 0.95$ results, the difference is small and again obfuscated by the randomness present in all of the results.

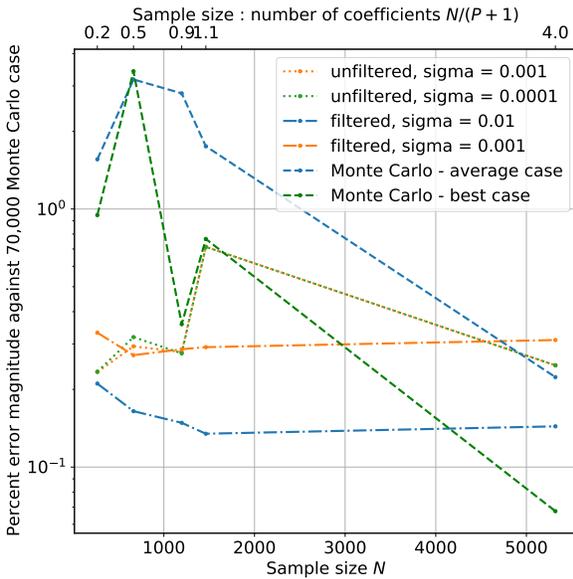


(a) Mean estimation

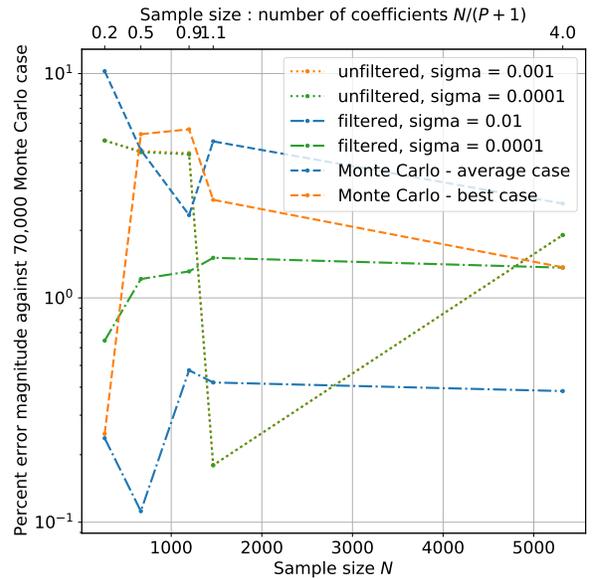


(b) Variance estimation

Fig. 9 Final flight path angle results for $\alpha = 0.95, p = 3$

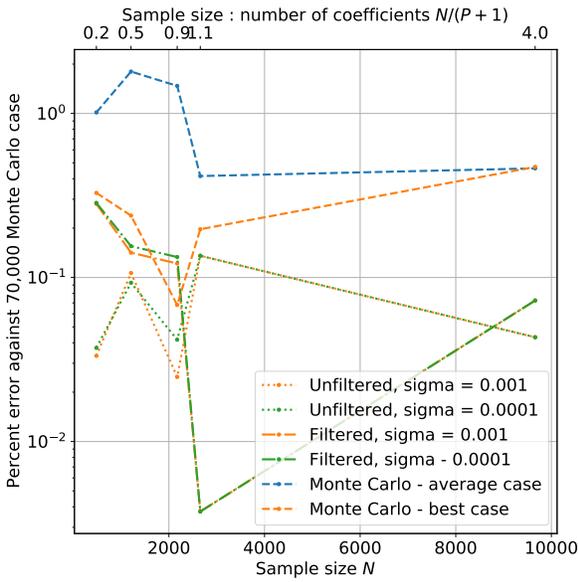


(a) Mean estimation

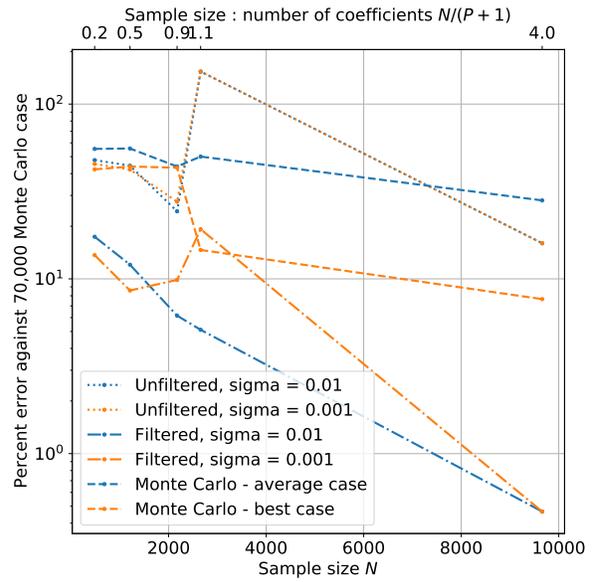


(b) Variance estimation

Fig. 10 Final orbital energy results for $\alpha = 0.95, p = 3$

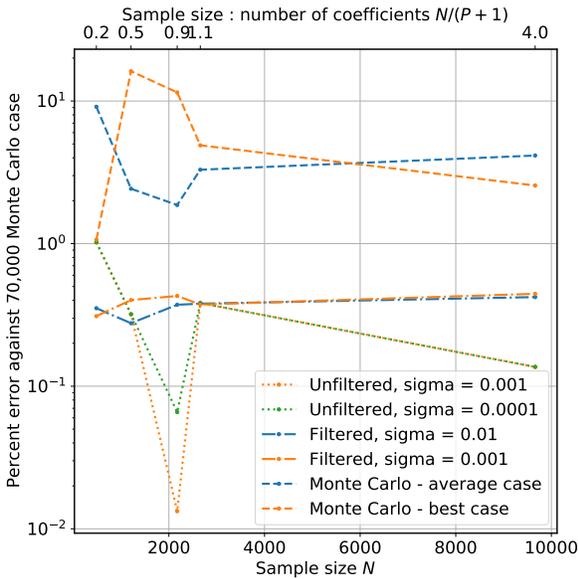


(a) Mean estimation

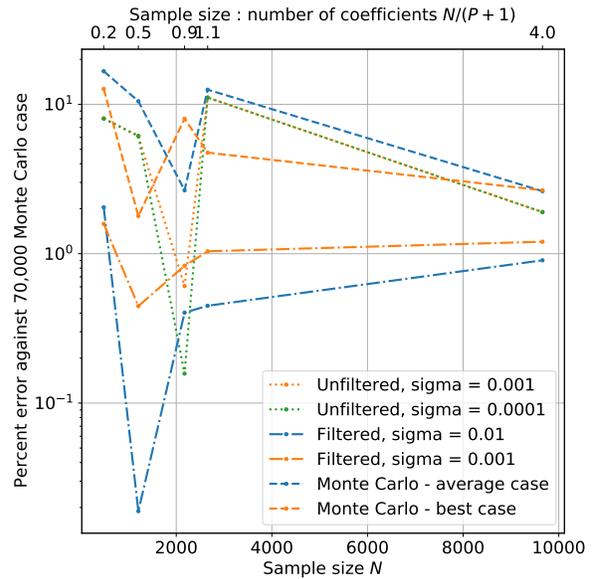


(b) Variance estimation

Fig. 11 Final flight path angle results for $\alpha = 0.99, p = 2$



(a) Mean estimation



(b) Variance estimation

Fig. 12 Final orbital energy results for $\alpha = 0.99, p = 2$

IV. Challenges and Future Work

There is significant further work to pursue in analyzing the performance of Monte Carlo vs. PCE for this problem, as well as a number of open questions. To begin, a near-term step is to model some of the dispersed inputs using a uniform distribution, and to truncate Gaussian distribution where necessary (such as in the KLE expansion for density, which cannot be negative). While the measurement matrix Ψ is built using only orthonormal Hermite polynomials for this study, it will be relatively straightforward to modify this matrix to use orthonormal Legendre polynomials for the input variables modeled using uniform distributions [11]. Another future addition is correlations between the random inputs, which will require modification of the PCE approach but is feasible [32]. Considering alternative modeling steps for the density KLE, such as representing the log of the density using a KLE in order to fit a log-normal distribution, is also a potential future addition.

One of the significant limitation of these results so far is the randomly-varying nature of the PCE and Monte Carlo performance. A near-term step is to implement k-fold cross-validation for the PCE results to obtain a more robust measure of the PCE performance compared to Monte Carlo [33]; this same approach could also be applied to the Monte Carlo results that used a subset of the full data. In addition, the value of the tuning parameter σ should be optimized for each scenario, including varying σ with the number of samples N [34]. The current implementation heuristically compares three values of σ held constant with sample size, and saw some significant variation, so an optimized and varying σ should result in notable improvement for the PCE results.

This study compares a higher-dimensional case with two lower-dimensional sets of results, but further comparison of how PCE performance is affected by increasing dimensionality driven by increasing KLE accuracy is an item of future interest. The expected result is that as the dimensionality of the problem increases, at some point PCE will perform categorically worse than Monte Carlo due to the curse of dimensionality, but identifying roughly where this transition occurs for the aerocapture problem is of interest.

Another avenue for future work is to consider other aerocapture scenarios, such as lift-modulated control or scenarios at other planets, and to consider other QoIs. In this study PCE results for final orbital energy seemed to perform somewhat better than the results for final flight path angle, so it is possible that other QoIs such as peak heat-rate are significantly more- or less-suitable for PCE.

Finally, an open question of general interest as well as to this study in particular is: when is an exponential atmosphere model good enough? Certainly the answer is dependent not only on the aerocapture scenario under study, but more importantly on the required fidelity/accuracy for that particular application. It is relevant to this work, however, because these results show that PCE is most useful when the KLE representation of density can tolerate around 5% error, and when the number of other dispersed inputs is not too high. Because the $\alpha = 0.95$ KLE approximation already sacrifices some accuracy, it is not immediately clear to what extent this model of density is more accurate than an exponential model. Using an exponential model would simplify the problem significantly, and the much lower dimensionality would allow for a range of other UQ techniques that in this study are precluded by the 18-dimensional problem [1]. In other words, in order for it to be worth it to use this UQ framework with a KLE density model, that density model should still gain sufficient accuracy over the exponential model.

V. Conclusions

This study demonstrates that a KLE expansion is a practical method to represent high-fidelity atmospheric density data in a finite-dimensional way. This representation then enables the use of UQ techniques beyond Monte Carlo, and PCE via compressed sensing is particularly well-suited to the problem since the dimensionality is still relatively high. Preliminary results show that PCE does outperform Monte Carlo when capturing the mean and variance of the final flight path angle and specific orbital energy from an aerocapture scenario at Mars. Filtering out trajectories that impact the surface noticeably improves performance, at the expense of being unable to capture statistics that include those cases. Because the benefit of PCE is faster convergence, and because it does still suffer from high dimensionality, the methods outlined in this study are most applicable to early-phase studies or broad parametric sweeps. In those applications, the benefits of the smaller required sample size can be realized while the approximation made by the KLE can be tolerated, and the number of other dispersed inputs is still relatively small.

Acknowledgments

This work is supported by a NASA Space Technology Research Fellowship. A. Doostan acknowledges the support of the NSF grant CMMI 1454601.

References

- [1] Jiang, X., “Uncertainty Quantification for Mars Atmospheric Entry using Polynomial Chaos and Spectral Decomposition,” *2018 AIAA Guidance, Navigation, and Control Conference*, 2018. <https://doi.org/10.2514/6.2018-1317>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2018-1317>.
- [2] Halder, A., and Bhattacharya, R., “Dispersion Analysis in Hypersonic Flight During Planetary Entry Using Stochastic Liouville Equation,” *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 2, 2011, pp. 459–474. <https://doi.org/10.2514/1.51196>, URL <https://doi.org/10.2514/1.51196>.
- [3] Heidrich, C. R., and Braun, R. D., “Aerocapture Trajectory Design in Uncertain Entry Environments,” *AIAA Scitech 2020 Forum*, 2020. <https://doi.org/10.2514/6.2020-1741>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2020-1741>.
- [4] Spencer, D. A., and Braun, R. D., “Mars Pathfinder atmospheric entry - Trajectory design and dispersion analysis,” *Journal of Spacecraft and Rockets*, Vol. 33, No. 5, 1996, pp. 670–676. <https://doi.org/10.2514/3.26819>, URL <https://doi.org/10.2514/3.26819>.
- [5] Kazmierczak, E. J., and Nakhjiri, N., “Improvements to Modeling and Trajectory Simulation of Mars Aerocapture and Aerobraking,” *2018 Space Flight Mechanics Meeting*, 2018. <https://doi.org/10.2514/6.2018-2228>, URL <https://arc.aiaa.org/doi/abs/10.2514/6.2018-2228>.
- [6] Desai, P. N., Mitcheltree, R. A., and Cheatwood, F. M., “Entry Dispersion Analysis for the Stardust Comet Sample Return Capsule,” *Journal of Spacecraft and Rockets*, Vol. 36, No. 3, 1999, pp. 463–469. <https://doi.org/10.2514/2.3467>, URL <https://doi.org/10.2514/2.3467>.
- [7] Regan, F. J., and Anandkrishnan, S. M., *Dynamics of Atmospheric Re-Entry*, AIAA, 1993.
- [8] Xiu, D., and Hesthaven, J., “High-Order Collocation Methods for Differential Equations with Random Inputs,” *SIAM J. Sci. Comput.*, Vol. 27, No. 3, 2005, pp. 1118–1139.
- [9] Nobile, F., Tempone, R., and Webster, C., “A Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data,” *SIAM J. Numer. Anal.*, Vol. 46, No. 5, 2008, pp. 2309–2345.
- [10] Ghanem, R., and Spanos, P., *Stochastic Finite Elements: A Spectral Approach*, Dover, 2002.
- [11] Xiu, D., and Karniadakis, G., “The Wiener-Askey polynomial chaos for stochastic differential equations,” *SIAM Journal on Scientific Computing*, Vol. 24, No. 2, 2002, pp. 619–644.
- [12] Loeve, M., “Elementary probability theory,” *Probability theory i*, Springer, 1977, pp. 1–52.
- [13] Lockwood, M. K., Queen, E. M., Way, D. W., Powell, R. W., Edquist, K., Starr, B. W., Hollis, B. R., Zoby, C. E., Hrinda, G. A., and Bailey, R. W., “Aerocapture Systems Analysis for a Titan Mission,” Tech. Rep. TM-2006-214273, NASA, 2006.
- [14] Way, D. W., Powell, R. W., Chen, A., Steltzner, A. D., Martin, A. M. S., Burkhart, P. D., and Mendek, G. F., “Mars Science Laboratory: Entry, Descent, and Landing System Performance,” *2007 IEEE Aerospace Conference*, 2007, pp. 1–19. <https://doi.org/10.1109/AERO.2007.352821>.
- [15] Caffisch, R. E., et al., “Monte carlo and quasi-monte carlo methods,” *Acta numerica*, Vol. 1998, 1998, pp. 1–49.
- [16] Stein, M., “Large sample properties of simulations using Latin hypercube sampling,” *Technometrics*, Vol. 29, No. 2, 1987, pp. 143–151.
- [17] Helton, J. C., and Davis, F. J., “Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems,” *Reliability Engineering & System Safety*, Vol. 81, No. 1, 2003, pp. 23–69.
- [18] Heinrich, S., “Multilevel monte carlo methods,” *International Conference on Large-Scale Scientific Computing*, Springer, 2001, pp. 58–67.
- [19] Giles, M. B., “Multilevel monte carlo path simulation,” *Operations research*, Vol. 56, No. 3, 2008, pp. 607–617.
- [20] Smith, R. C., *Uncertainty quantification: theory, implementation, and applications*, Vol. 12, Siam, 2013.
- [21] Doostan, A., and Owhadi, H., “A non-adapted sparse approximation of PDEs with stochastic inputs,” *Journal of Computational Physics*, Vol. 230, 2011, pp. 3015–3034.

- [22] Blatman, G., and Sudret, B., “Adaptive sparse polynomial chaos expansion based on least angle regression,” *Journal of Computational Physics*, Vol. 230, 2011, pp. 2345–2367.
- [23] Hampton, J., and Doostan, A., “Compressive Sampling of Polynomial Chaos Expansions: Convergence Analysis and Sampling Strategies,” *Journal of Computational Physics*, Vol. 280, 2015, pp. 363–386.
- [24] Austin, A., Nelessen, A., Strauss, B., Ravich, J., Jesick, M., Venkatapathy, E., Beck, R., Wercinski, P., Aftosmis, M., Wilder, M., Allen, G., Braun, R., Werner, M., and Roelke, E., “SmallSat Aerocapture to Enable a New Paradigm of Planetary Missions,” *2019 IEEE Aerospace Conference*, 2019, pp. 1–20. <https://doi.org/10.1109/AERO.2019.8742220>.
- [25] Hadigol, M., and Doostan, A., “Least squares polynomial chaos expansion: A review of sampling strategies,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 332, 2018, pp. 382 – 407. <https://doi.org/https://doi.org/10.1016/j.cma.2017.12.019>, URL <http://www.sciencedirect.com/science/article/pii/S0045782517307697>.
- [26] Peng, J., Hampton, J., and Doostan, A., “A weighted ℓ_1 -minimization approach for sparse polynomial chaos expansions,” *Journal of Computational Physics*, Vol. 267, 2014, pp. 92–111.
- [27] Doll, A., Ravasi, M., and Relyea, D., “SPGL1: A solver for large-scale one-norm regularized least squares,” , December 2020. <https://github.com/drrelyea/spgl1>.
- [28] van den Berg, E., and Friedlander, M. P., “SPGL1: A solver for large-scale sparse reconstruction,” , December 2019. <https://friedlander.io/spgl1>.
- [29] van den Berg, E., and Friedlander, M. P., “Probing the Pareto frontier for basis pursuit solutions,” *SIAM Journal on Scientific Computing*, Vol. 31, No. 2, 2008, pp. 890–912. <https://doi.org/10.1137/080714488>, URL <http://link.aps.org/link/?SCE/31/890>.
- [30] Sudret, B., “Global sensitivity analysis using polynomial chaos expansions,” *Reliability Engineering & System Safety*, Vol. 93, No. 7, 2008, pp. 964 – 979. <https://doi.org/https://doi.org/10.1016/j.res.2007.04.002>, URL <http://www.sciencedirect.com/science/article/pii/S0951832007001329>, bayesian Networks in Dependability.
- [31] Sudret, B., and Mai, C., “Computing derivative-based global sensitivity measures using polynomial chaos expansions,” *Reliability Engineering & System Safety*, Vol. 134, 2015, pp. 241 – 250. <https://doi.org/https://doi.org/10.1016/j.res.2014.07.009>, URL <http://www.sciencedirect.com/science/article/pii/S0951832014001641>.
- [32] Navarro, M., Witteveen, J., and Blom, J., “Polynomial chaos expansion for general multivariate distributions with correlated variables,” *arXiv preprint arXiv:1406.5483*, 2014.
- [33] Huan, X., Safta, C., Sargsyan, K., Vane, Z. P., Lacaze, G., Oefelein, J. C., and Najm, H. N., “Compressive sensing with cross-validation and stop-sampling for sparse polynomial chaos expansions,” *SIAM/ASA Journal on Uncertainty Quantification*, Vol. 6, No. 2, 2018, pp. 907–936.
- [34] Diaz, P., Doostan, A., and Hampton, J., “Sparse polynomial chaos expansions via compressed sensing and D-optimal design,” *Computer Methods in Applied Mechanics and Engineering*, Vol. 336, 2018, pp. 640 – 666. <https://doi.org/https://doi.org/10.1016/j.cma.2018.03.020>, URL <http://www.sciencedirect.com/science/article/pii/S0045782518301452>.