Short comparison on using spectral methods to analyze a random chaotic system

Dallas Foster

Department of Mathematics, Oregon State University

Abstract

We look at the the famous Lorenz 63 model with randomn initial conditions and deterministic parameters. The chaotic nature of this system magnifies small differences in the initial state at later times. We compare the ability of Monte Carlo Sampling, Stochastic Galerkin Projection and Collocation methods to capture the statistics of the random ordinary differential equation. We find that the accuracy of these methods, while being better than not assuming randomness, suffers from poor convergence and is dependent upon the time interval of the problem.

Keywords: Uncertainty Quantification, Lorenz, Collocation, Galerkin Projection

1. Problem Definition

In this paper we will consider the classic Lorenz system. First, we will consider the deterministic model for reference. The main thrust of the paper will be to recast the system as a random ordinary differential equation in which the initial conditions are unknown but come from a known distribution. In order to treat this uncertainty, we will consider three methods to determine the statistics of the solution: simple monte carlo sampling (2), stochastic Galerkin (2.3), and stochastic collocation (2.2). The latter two will rely on interpreting the solution in terms of a polynomial spectral expansion (2.1). The statistics of sufficiently large monte carlo samples will be used as a benchmark to test the error of the two spectral methods. We will conclude with a discussion of the convergence of the two methods.

1.1. Deterministic Model

Consider the Lorenz system of $u(t) = [x(t), y(t), z(t)]^T$ given by the following set of nonlinear differential equations

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= x(\rho - z) - y \\ \dot{z} &= xy - \beta z \end{aligned} \tag{1}$$

Email address: fostdall@oregonstate.edu (Dallas Foster)

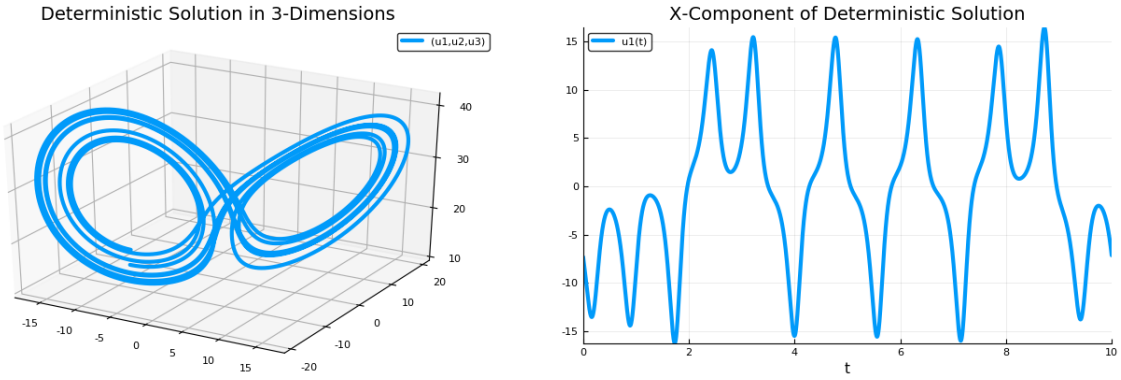
where σ, ρ, and β are parameters. We will consider these parameters as fixed and we will assign them the values

$$\sigma = 10.0, \qquad \rho = 28.0, \qquad \beta = 8/3$$

for the entirety of our tests. We consider the solution $u(t)$ on the set $t \in [0, 10]$ with initial conditions

$$\tilde{u}(0) = [\, -7.3,\ -11.5,\ 17.8\,]^T. \tag{2}$$

Solution are plotted below.



For the purposes of our analysis, we will mostly be concerned with the solution $x(t)$. This choice was arbitrary but representative of the dynamics of the tests that we have performed.

1.2. Random Model

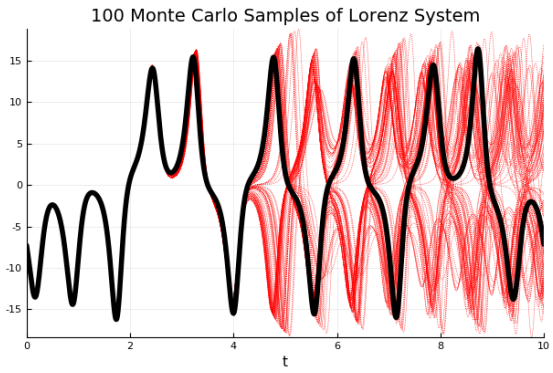
Consider the same system (1) with parameters given above. Now it is assumed that the initial condition is random,

$$u(0) \sim \mathcal{N}(\tilde{u}(0), \sigma^2).$$

That is, the initial conditions are normally distributed around the aforementioned initial conditions (2) with some small variance (taken in these experiments to be 0.1). The goal of this paper is to understand how the uncertainty in the initial condition translates into uncertainty in the solution. In particular, we will be interested in the mean and variance of the solution to our random differential equation.

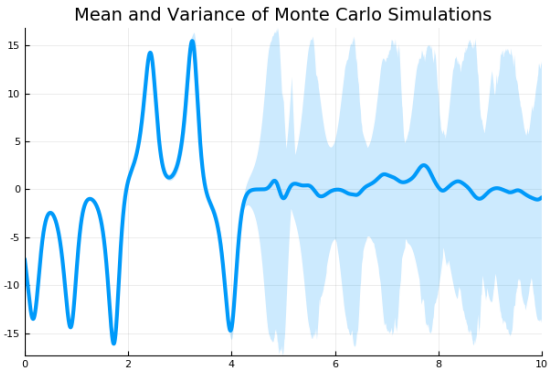
2. Modeling Randomness

The first and most naive way to understand the statistics of our solution is to sample the initial condition via random number generator, solve each differential equation using the sampled state and generate an ensemble of solutions. From this ensemble we can calculate the sample mean and variance at each point in time. This method, while being practical in this particular system, is not feasible in larger dimension model, or systems with systems with more than one random dimension. Even here, where we use $10,000$ monte carlo samples of the determinstic solution with step size $dt = 0.001$ as a benchmark for error analysis of the later methods, the computational cost on a personal computer has been prohibitive.



The picture to the left is the result of 100 samples.The black line is the solution with mean initial condition and each red line are members of the ensemble. Notice that at approximately $t = 4$, the ensemble solutions start diverging markedly from the deterministic solution. It is here that the chaotic nature of the Lorenz system starts impact the trajectories of the perturbed initial states. The slight change in initial condition actually changes the sheath that the periodic solution travels. It is difficult to consider the route of each trajectory and one cannot fit the trajectories neatly into a small number of bins using k-means clustering. Therefore we will just consider the simple sample mean and variance, as computed below for 10,000 Monte Carlo samples.

Notice that because of the nature of the sampling, the mean tends to small variations about zero as t increases but the variance, in order to capture all of the variations, is very large. These are the statistics that we are looking to recreate by using the stochastic collocation and Galerkin methods.



2.1. Polynomial Chaos

Here we follow the formulation as given in [1], but one can also view the development in [2]. Let Z be a random variable with a distribution function $F_Z(z)$ that has finite moments. The *generalized polynomial chaos* basis functions are orthogonal polynomial functions satisfying

$$\mathbb{E}\left[\Phi_m(Z)\Phi_n(Z)\right] = \gamma_n \delta_{mn}, \qquad m, n \in \mathbb{N}_0 \tag{3}$$

where γ_n are thought of as normalization constants and δ_{mn} is the Kronecker delta. In particular, if Z is a normally distribution random variable then the orthogonal polynomials are Hermite polynomials defined by the recursion relation,

$$H_0(Z) = 1, \qquad H_1(Z) = 2Z, \qquad H_{n+1}(Z) = 2ZH_n(Z) - 2nH_{n-1}(Z).$$

Because we are assuming uncertainty in our initial condition, $u(0) \sim \mathcal{N}(\tilde{u}(0), \sigma^2)$, our solution can be written in terms of a random variable $u(t, Z)$. Using these polynomials, we can write the generalized polynomial chaos expansion of our solution as

$$u(t, Z) = \sum_{k=0}^{\infty} \hat{u}_k(t)\Phi_k(Z), \qquad \hat{u}_k(t) = \frac{1}{\gamma_k}\mathbb{E}\left[u(t, Z)\Phi_k(Z)\right] \tag{4}$$

or, since we will only consider finite approximations, $u_N(t, Z)$, as the truncation of (4) to N terms. The coefficients $\hat{u}_k(t)$ are the quantities that hold the statistical information - the mean and variance - of our solution,

$$\mathbb{E}\left[u_N(t,Z)\right] = \hat{u}_0(t), \qquad \mathbb{E}\left[\left(u_N(t,Z) - \mathbb{E}\left[u_N(t,z)\right]\right)^2\right] = \sum_{0<k\leq N} \left[\gamma_k \hat{u}_k^2(t)\right]$$

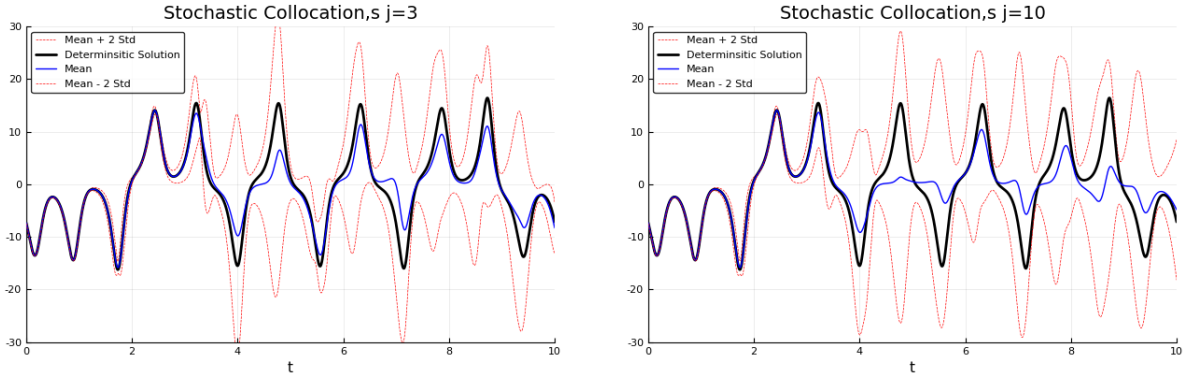
The two methods presented in this paper are different techniques for approximating these coefficients. First we will consider the collocation method, then the Galerkin projection method.

2.2. Stochastic Collocation

Consider the integrals in (4), we approximate them using a j order quadrature rule

$$\hat{u}_k(t) = \frac{1}{\gamma_k}\mathbb{E}\left[u(t,Z)\Phi_k(Z)\right] \approx \sum_{l=1}^{j} u(t,Z_l)\Phi_k(Z_l)w_l.$$

That is, we choose a set of nodes Z_l and weights w_l to achieve the highest possible order of algebraic exactness. Because the expectations are with respect to the weight function e^{-x^2} we implement Gauss-Hermite quadrature. For each node Z_l, we run the system (1) with initial condition $u(0) = \tilde{u}(0) + Z_l$, producing a sample $u(t, Z_l)$. With each sample in hand, the quadrature (2.2) is easy to calculate. Below we plot the results of computing $\hat{u}_0(t)$ and two standard deviation using $j = 3$ and $j = 10$. The solution with $u(0) = \tilde{u}(0)$ is also given for reference. To comment briefly on qualitative results, we notice that the



two standard deviation range encapsulates the deterministic solution and $\hat{u}_0(t)$ follows that solution somewhat closely. We expect that for large enough j we can recreate the statistics from the Monte Carlo sampling. Notice, however, that while the mean for $j = 10$ is flatter than $j = 3$, it does not quite have the same shape as the aforementioned sampling mean. We will quantify this in section (3). It is to be noted that the computational costs for collocation is significantly less than Monte Carlo sampling. Instead of sampling tens of thousands of time to produce the Monte Carlo sampling before, we essential solve the same system but with orders of magnitude fewer samples. Of particular importance is that there was no

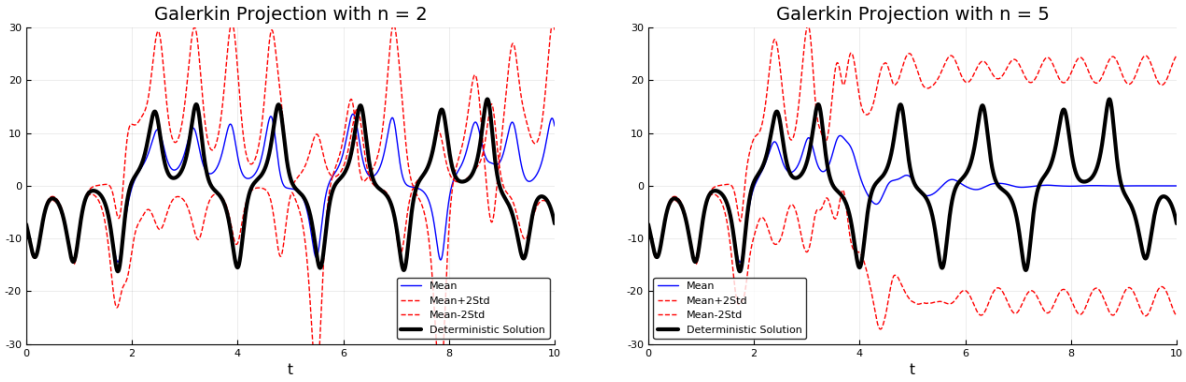
need to alter the model code with this method, which is why collocation is referred to as a non-intrusive method. In contrast, the Galerkin Projection method is intrusive and can require considerable change to the underlying model.

2.3. Stochastic Galerkin Projection

Galerkin Projection still seeks the solution as a polynomial chaos expansion (4) but is an extension of the classic Galerkin approach for deterministic systems. In particular, we seek a solution to the system of equations

$$\mathbb{E}\left[\frac{d}{dt}\begin{pmatrix} x_N(t,Z) \\ y_N(t,Z) \\ z_N(t,Z) \end{pmatrix}\Phi_k\right] = \mathbb{E}\left[\begin{pmatrix} \sigma(x_N(t,Z)-y_N(t,z)) \\ x_N(t,Z)(\rho-y_N(t,z))-z_N(t,Z) \\ x_N(t,Z)y_N(t,Z)-z_N(t,Z) \end{pmatrix}\Phi_k(Z)\right] \tag{5}$$

with initial condition $\mathbb{E}[u_N(0)\Phi_k(Z)]$, for $k = 1, ..., N$. This creates a $3N$ dimensional system. This integration can be done easily for linear systems and even here where the system only has polynomial non-linearity. For linear systems this requires minimal change of the original system. For nonlinear systems this process requires completely rewriting the original model, which even in our simple model is prohibitive for $N > 3$. For this reason, we calculate the expectations in (5) using the same quadrature rule as in (2.2). Since the integrals in (5) are of polynomials, we can easily calculate the order of quadrature necessary to give zero quadrature error. For instance, the largest polynomial in this system is of order $2N$, so a $N + 1$ order quadrature rule will be exact. The graphs above show the results



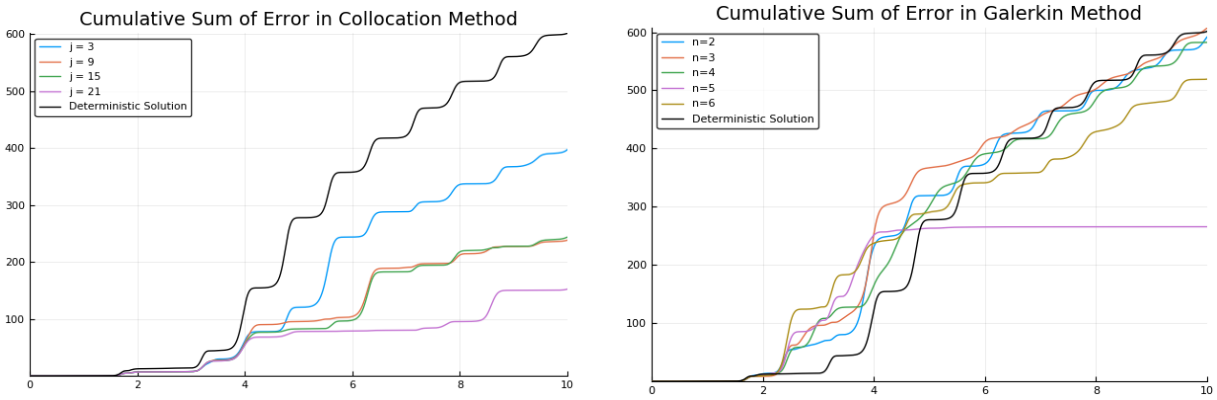
galerkin projection with $N = 2$ and $N = 5$ degree expansions. Notice two things, the results for $N = 2$ are not particularly impressive. Not only do the confidence bounds not contain the deterministic solution, but the mean does not follow that solution closely like it did in the collocation case. With $N = 5$ we do notice that the long term behavior of the mean does appear to be similar to the behavior that we notice in the monte carlo sample mean. Again, we will quantify this in the results section.

3. Results and Analysis

Here we perform error analysis for the two spectral methods described above. Consider the following metric for the error,

$$\text{error} \ := \| \ \mathbb{E}\left[u(t,Z)\right] - \mathbb{E}\left[u_N(t,Z)\right] \ \|_2$$

where $\mathbb{E}\left[u(t,Z)\right]$ is approximated by taking the sample mean of $10,000$ Monte Carlo samples. Below are plots of the cumulative sum of errors over $[0,10]$ for both the collocation and galerkin projection method. In each graph, the black line is given by the solution to the



original deterministic solution. We consider this in both diagrams because we are not only concerned with the accuracy of each method with respect to the sample mean, but also with respect to not incorporating randomness at all. In the graph on the left, notice that each of the approximations have smaller error than the nonrandom system for all times and that there is clear declining error with increasing quadrature order. We can compute this order of convergence,

$$\| \ \mathbb{E}\left[u(t,Z)\right] - \mathbb{E}\left[u_N(t,Z)\right] \ \|_2 = O(n^{\gamma})$$

where $\gamma \approx 0.49$. This is in stark contrast to the error for the Galerkin method. In the image on the right, there is no clear error convergence and for $t < 4$ none of the approximations have smaller error than the nonrandom solution. The case where $n = 5$ seems like an outlier in comparison to the other approximations. Note that the jth order quadrature method and jth order Galerkin system are not equivalent. In this particular system, 7th order Galerkin projection is equivalent to an 11 quadrature points.

4. Discussion and Conclusion

The Lorenz 63 system suffers from polynomial non-linearity and chaotic behavior. The chaotic structure of the solution means that small perturbations in the initial condition can produce large deviations at later times. This leads to a sample mean that is nearly zero for large t. The small number of dimensions in this model lend itself to Monte Carlo sampling more real-world scenarios. In practice, Monte Carlo sampling might be too computationally

expensive to perform. There are other advances in sampling methods like MCMC and Multigrid Monte Carlo that also try to address the slow convergence of Monte Carlo. In this paper we compared to spectral methods to capture the statistics of teh system. Stochastic Galerkin Projection, beyond being difficult to implement because of its intrusive nature, did not produce convergence. Stochastic Collocation is non-instrusive, fairly easy to implement, and fast compared to Monte Carlo when using sparse grid collocation. In particular, the error was better in our model than simply not considering randomness at all. One may want to consider higher order Galerkin methods in the future. Our results were limited by the number of sparse grid levels that were available. In addition, one may want to test the error in higher statistical moments than the mean. We computed the variance but did not compare errors. Higher statistical moments are easier to calculate in the collocation framework than the Galerkin projection framework.

References

[1] D. Xiu, Numerical Methods for stochastic computations : a spectral method approach, Princeton University Press, 2010.

[2] R. C. Smith, Uncertainty Quantification Theory, Implementation, and Application, Society for Industrial and Applied Mathematics, 2014.