

# HIGH DENSITY MODEL REDUCTION UNCERTAINTY QUANTIFICATION FOR MULTIGROUP DIFFUSION NEUTRONICS

**Paul W. Talbot\* and Anil K. Prinja**

University of New Mexico  
Department of Nuclear Engineering  
209 Farris Engineering Center  
Albuquerque, NM 87131  
talbotp@unm.edu; prinja@unm.edu

**Cristian Rabiti**

Idaho National Laboratory  
Nuclear System Design and Analysis Division  
P.O. Box 1625 MS 3870 cristian.rabiti@inl.gov

## ABSTRACT

While stochastic collocation for uncertainty quantification converges output domain uncertainty more quickly than Monte Carlo for low-dimensional input domains, such techniques suffer from the curse of dimensionality. We present here some techniques for mitigating the problems that high-dimension uncertainty spaces cause for collocation by using sparse grid sampling quadratures, high-density model reduction (HDMR), and anisotropic quadrature sampling.

*Key Words:* Uncertainty Quantification, Stochastic Collocation, HDMR, Sparse Grid

## 1 INTRODUCTION

The development of computational numerical models to simulate scientific processes often results in an algorithmic solver that takes inputs from the user and returns some quantity of interest. We label this algorithm as the *deterministic solver*. This may be misleading, as the deterministic solver might make use a stochastic algorithm; however, when we speak of deterministic or stochastic solvers, we refer to the dimensionality of the input space, not the algorithm itself.

While the deterministic solver is a powerful tool, it often fails to capture aleatory or epistemic uncertainty in the input parameters. Either because of randomness of the physics or uncertainty in measuring the input parameters, an accurate solver should quantify the uncertainty space of the input parameters and the resulting solution space for the quantity of interest. Solely making use of the deterministic solver potentially presents a naive picture of the solution space.

There are two major classes of uncertainty quantification methods, separated by their impact on the deterministic solver itself. Intrusive methods make use of parameter changes within the

---

\*talbotpne@gmail.com

algorithm, necessitating the deterministic code have uncertainty quantification coded in or added. Non-intrusive methods treat the deterministic solver as a black box. This allows the freedom to use methods on any deterministic solver that accepts inputs and delivers a quantity of interest. Non-intrusive methods are the focus of this work.

Among non-intrusive methods, we can further divide uncertainty quantification into stochastic and deterministic sampling methods. Stochastic methods are centered around randomly sampling the input parameter uncertainty space and deriving statistics from resulting quantities of interest, and include methods such as analog Monte Carlo, multilevel Monte Carlo, and Latin Hypercube. Deterministic methods sample from the uncertainty space in predefined locations, using characteristic or interpolating polynomials to create a Reduced-Order Model (ROM) of the uncertainty space, and include methods like stochastic Galerkin, stochastic least squares, and stochastic collocation.

There have been many recent advances in deterministic uncertainty quantification for computational physics [?]. We present ongoing work to demonstrate the efficiency of high-density model reduction (HDMR) techniques as well as sparse-grid stochastic collocation on sparse grids (SC) [?] in comparison with traditional analog Monte Carlo (MC) uncertainty quantification.

## 2 MOTIVATION

Traditionally, Monte Carlo methods are convergent but slow. Additionally, Monte Carlo for uncertainty quantification is dimension-agnostic. Stochastic collocation is based on using selected quadrature points within the uncertain input space to construct polynomials which represent the quantity of interest solution space. For small input space dimension  $N$  and reasonably smooth uncertain input space, stochastic collocation converges much more quickly than Monte Carlo. However, the quadrature set size grows exponentially with increasing number of dimensions. In this work, we explore methods to negate some of the curse of dimensionality and show improvement over analog Monte Carlo methods for larger numbers  $N$  of uncertain input parameters.

## 3 DETERMINISTIC MODEL

The uncertainty quantification methods we describe are agnostic of the deterministic problem whose uncertainty is quantified. To demonstrate our quantification methods, we apply them to three deterministic solvers, one each for an attenuation problem, a projectile with drag problem, and a nuclear reactor core  $k$ -eigenvalue problem.

### 3.1 Attenuation Problem Solver

Because of its analyticity, we make use of the attenuation problem to demonstrate accurate method convergence. This problem simulates the attenuation of a beam of particles through a fixed-size block subdivided into several sections with varying opacity. The quantity of interest is

the exit beam strength, and the problem can be solved analytically as

$$u(Y) = \prod_{n=1}^N \exp\left(-\frac{Y_n}{N}\right), \quad (1)$$

where  $u$  is the fractional emitted beam strength,  $N$  is the number of subdivided segments, and  $Y = Y_1, \dots, Y_N \in \Gamma$  are the product of uncertain cross sections and unit length, with  $\Gamma \in \mathbb{R}^N$  as the entirety of the combined uncertain input space. If  $Y_n$  is distributed uniformly as  $U \sim [a, b]$ , the mean can likewise be computed analytically as

$$\mathbb{E}[u(Y)] = \int_{\Gamma} u(Y) P(Y) d\Omega, \quad (2)$$

$$= \prod_n^N (b-a)^{-1} \int_a^b \exp\left(-\frac{Y_n}{N}\right) dY_n, \quad (3)$$

$$= \left[ \left( \frac{N}{b-a} \right) (e^{-a/N} - e^{-b/N}) \right]^N. \quad (4)$$

For this work,  $N = 5$ ,  $a = 1$ , and  $b = 6$ .

### 3.2 Projectile with Drag Problem Solver

This problem is included to demonstrate a problem without direct analytic solution and that uses a grid-based deterministic solver. In this problem we consider the range  $R$  of a spherical projectile with mass  $m$ , radius  $r$ , and drag coefficient  $C$  launched from a height of  $y_i$  at an angle of  $\theta$  with respect to parallel with the ground at an initial velocity of  $v$ , and density of the air  $\rho$  and gravitational acceleration constant  $g$ . We allow each of these to vary uncertainly with value as in Table ??.

Input	Mean	Range
$m$	0.145 kg	0.0725 kg
$r$	0.0336 m	0.00336 m
$C$	0.5	0.5
$\rho$	1.2 kg/m <sup>3</sup>	0.1 kg/m <sup>3</sup>
$v$	50 m/s	5 m/s
$\theta$	45°	10°
$g$	9.7988 m/s <sup>2</sup>	0.0349 m/s <sup>2</sup>
$y_i$	1 m	1 m

**Table I. Projectile with Drag Inputs**

The equations of motion are given in both the horizontal  $x$  and vertical  $y$  dimensions as

$$R = \frac{v_t}{g} (v \sin \theta + v_t) (1 - e^{-g t/v_t}) - v_t t, \quad (5)$$

$$x_f = \frac{v v_t \cos \theta}{g} (1 - e^{-g t/v_t}), \quad (6)$$

$$v_t = \frac{mg}{D}, \quad D = \frac{\rho C A}{2}, \quad A = \pi r^2. \quad (7)$$

The deterministic solver for the projectile problem uses a forward-Euler time-stepping scheme, and its accuracy depends strongly on the time step used.

### 3.3 Reactor Core $k$ -eigenvalue Solver

To demonstrate the uncertainty quantification performance on a difficult, nonlinear problem we introduce a solver for a nuclear reactor core neutronics problem. With this model we solve a two-dimensional nonlinear dual-energy group neutron diffusion criticality benchmark problem [?]. This problem simulates an orthogonal quarter-core reactor by imposing reflective conditions on the left and bottom boundaries, with vacuum no-traction conditions on the top and right. the problem makes use of five materials in 121 regions. Fig. ?? illustrates the core.

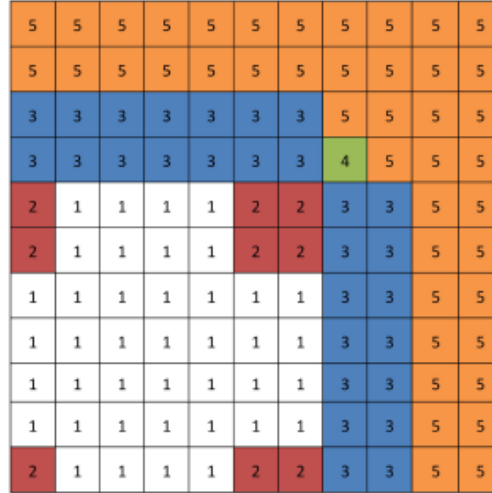


Figure 1. Core Geometry

### 3.4 Mathematical Model

The PDE describing steady-state neutron transport in the core is the two-group neutron diffusion criticality approximation,

$$-\nabla \cdot (D_1(\bar{x}) \nabla \phi_1(\bar{x})) + (\Sigma_a^{(1)}(\bar{x}) + \Sigma_s^{(1 \rightarrow 2)}(\bar{x})) \phi_1(\bar{x}) = \frac{1}{k(\phi)} \sum_{g'=1}^2 \nu_{g'} \Sigma_f^{(g')}(\bar{x}) \phi_{g'}(\bar{x}), \quad (8)$$

$$-\nabla \cdot (D_2(\bar{x}) \nabla \phi_2(\bar{x})) + \Sigma_a^{(2)}(\bar{x}) \phi_2(\bar{x}) = \Sigma_s^{(1 \rightarrow 2)}(\bar{x}) \phi_1(\bar{x}), \quad (9)$$

where we use the following parametric coefficients: the absorption cross section  $\Sigma_{g,a} = \Sigma_{g,c} + \Sigma_{g,f}$ ; the capture and fission cross sections  $\Sigma_{g,c}$  and  $\Sigma_{g,f}$ ; the diffusion coefficient  $D_g$  which depends

on the scattering cross section of the medium; and the fission multiplication factor  $\nu_g$ , the ratio of new neutrons per fission-producing neutron. The solution to this PDE is the neutron scalar flux  $\phi_g(\bar{x})$ . We apply no-traction conditions on the vacuum boundaries and zero-derivative current on the reflecting boundaries for both energy groups:

$$\left. \frac{\phi_g}{4} - \frac{D_g}{2} \frac{\partial \phi_g}{\partial x_1} \right|_{\partial\Omega_{\text{top}}} = 0, \quad g = 1, 2, \quad (10)$$

$$\left. \frac{\phi_g}{4} - \frac{D_g}{2} \frac{\partial \phi_g}{\partial x_2} \right|_{\partial\Omega_{\text{right}}} = 0, \quad g = 1, 2, \quad (11)$$

$$\left. -D_g \frac{\partial \phi_g}{\partial x_1} \right|_{\partial\Omega_{\text{bottom}}} = 0, \quad g = 1, 2, \quad (12)$$

$$\left. -D_g \frac{\partial \phi_g}{\partial x_2} \right|_{\partial\Omega_{\text{left}}} = 0, \quad g = 1, 2. \quad (13)$$

The material properties are shown in Table ??, and the domain  $\Omega = [0, 200 \text{ cm}]^2$ . The reference flux solutions are plotted in Fig. ??, and for the reference problem  $k=1.00007605445$ .

Region	Group	$D_g$	$\Sigma_{a,g}$	$\nu\Sigma_{f,g}$	$\Sigma_s^{1,2}$
1	1	1.255	8.252e-3	4.602e-3	2.533e-2
	2	2.11e-1	1.003e-1	1.091e-1	
2	1	1.268	7.181e-3	4.609e-3	2.767e-2
	2	1.902e-1	7.047e-2	8.675e-2	
3	1	1.259	8.002e-3	4.663e-3	2.617e-2
	2	2.091e-1	8.344e-2	1.021e-1	
4	1	1.259	8.002e-3	4.663e-3	2.617e-2
	2	2.091e-1	7.3324e-2	1.021e-1	
5	1	1.257	6.034e-4	0	4.754e-2
	2	1.592e-1	1.911e-2	0	

**Table II. Reference Material Properties for Benchmark Core**

### 3.5 Deterministic Solver

The criticality eigenvalue and quantity of interest  $k(\phi)$  is converged on iteratively by

$$k^{n+1} = k^n \sum_{g=1}^2 \frac{\int_{\Omega} \nu \Sigma_f^{(g)} \phi_g^{n+1}(\bar{x}) d\bar{x}}{\int_{\Omega} \nu \Sigma_f^{(g)} \phi_g^n(\bar{x}) d\bar{x}}. \quad (14)$$

until a convergence tolerance is achieved. We solve  $\phi_1, \phi_2$ , and  $k$  nonlinearly and simultaneously using a globally- and locally-conservative finite volume approximation in space. We solve implicitly with a Jacobian-Free Newton Krylov method, using a GMRES package from nonlinear solver package `trilinos` [?].

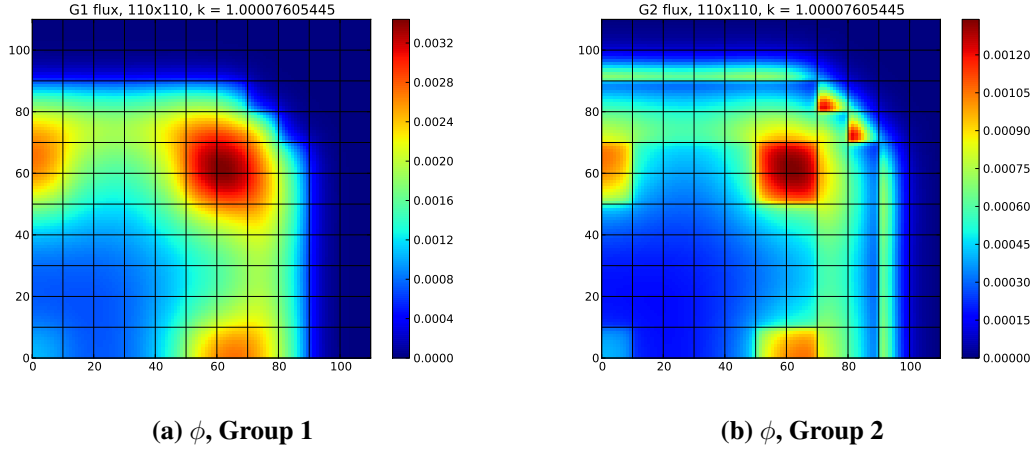


Figure 2. Reference Flux Profiles

To verify the deterministic solver, we compare flux profiles and  $k$  to the original benchmark [?] as well as demonstrate the convergence of  $k$  with increasing mesh grid refinement. The deterministic solver is in agreement with the benchmark, and convergence for  $k$  and both group fluxes are demonstrated in Fig. ???. Both fluxes as well as the  $k$ -eigenvalue converge between first and second order. The finite volume method is second order, but the overall convergence order is reduced by first-order boundary condition treatments. In all cases, reducing the size of  $h = \Delta x = \Delta y$  decreases the error in the solution. The reference solution is a level of refinement finer than the finest point shown. The flux error is calculated by considering the maximum error among several point-wise flux values, while  $k$  error is the global value for each refinement.

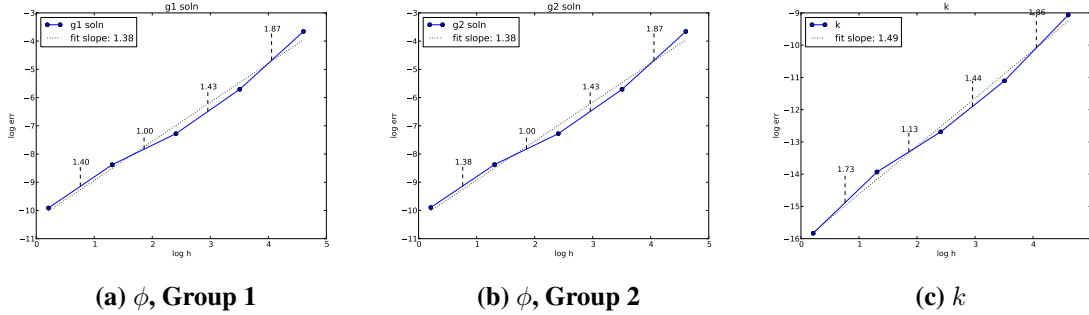


Figure 3. Deterministic Solver Convergence

## 4 UNCERTAINTY QUANTIFICATION

Throughout the remainder of this document, we will represent the solution to the  $k$ -eigenvalue problem generically as  $u(Y)$ , where  $u$  corresponds to the value of  $k$  and  $Y$  is the vector of uncertain input parameters.

There are 33 total input parameters in Table ??? that can be treated as uncertain. For simplicity,

any uncertainty introduced by these parameters is taken to be uniformly distributed within 10% of the reference value. We consider three cases for uncertainty quantification, distinguished chiefly by the number of input parameters with uncertainty ( $N$ ): material 1 uncertainty, where all the material properties for material 1 are treated as uncertain ( $N=7$ ); low-energy group uncertainty, where all material properties for group 2 are treated as uncertain ( $N=13$ ); and full uncertainty, where all material properties are treated as uncertain ( $N=33$ ).

The test for a method's effectiveness will be considering the level of error reached as a function of the number of calls to the deterministic solver required. While there is some difference in overhead cost between methods, this diminishes for sufficiently costly deterministic solutions.

In general, moments of the quantity of interest  $u(Y)$  are obtained by

$$\mathbb{E}[u(Y)^r] = \int_{\Gamma} u(Y)^r P(Y) dY, \quad (15)$$

where  $r$  is the desired moment,  $Y = [Y_1, Y_2, \dots, Y_N]$  are the uncertain inputs,  $P(Y)$  is the joint probability distribution of  $Y$ , and  $\Gamma$  is the uncertainty space spanned by  $Y$ .

#### 4.1 Monte Carlo

By way of benchmark, we consider analog Monte Carlo (MC) as a benchmark uncertainty quantification technique. Analog Monte Carlo determines the moments of a function by randomly sampling from the uncertainty space repeatedly until an accurate idea of the result is obtained. Monte Carlo benefits from a guaranteed consistent rate of convergence; however, this convergence rate is quite low; it typically converges as 1 over the square root of the number of deterministic solves. MC approximates moments of  $u(Y)$  as

$$\mathbb{E}[u(Y)^r] \approx \frac{1}{\eta} \sum_{m=1}^{\eta} u(Y^{(m)})^r, \quad (16)$$

where  $\eta$  is the total number of samples taken,  $m$  is the index of a sample, and  $Y^{(m)}$  is a point randomly sampled from  $\Gamma$ .

#### 4.2 Stochastic Collocation

In stochastic collocation, we approximate  $u(Y)$  as the sum of  $u$  evaluated at  $\eta$  collocated points multiplied by multidimensional Lagrangian polynomials. We make use of the quadrature index  $k$ , not to be confused with the criticality factor  $k(Y)$  (represented by  $u(Y)$ ).

$$u(Y) \approx u_{h,\eta,\Lambda(L)}(Y) = \sum_{k=0}^{\eta} u(Y^{(k)}) \mathcal{L}_k(Y), \quad (17)$$

$$\mathcal{L}_k(Y) = \prod_{n=1}^N \mathcal{L}_{k_n}(Y_n), \quad (18)$$

$$\mathcal{L}_{k_n}(Y_n) = \prod_{j=1}^i \frac{Y_n - Y_n^{(i)}}{Y_n^{(k_n)} - Y_n^{(i)}}, \quad (19)$$

$$\mathbb{E}[u(Y)] \approx \mathbb{E}[u_h(Y)] = \sum_{k=1}^{\eta} w_k u_h(Y^{(k)}), \quad (20)$$

where  $u_h(Y)$  is the spatially-discretized PDE solution, and  $Y^{(k)} = [Y^{(k_1)}, \dots, Y^{(k_N)}]$  are realizations of  $Y$  similar to  $Y_m$  in Monte Carlo but chosen at quadrature points  $Y^{(k)}$  with corresponding weights  $w_k$ . For this study, uniformly-distributed uncertain parameters suggest Gauss-Legendre quadrature to obtain collocation points and weights. The necessary quadrature points are obtained based on polynomial expansion orders from an index set  $\Lambda(L)$ .

#### 4.2.1 Index Set $\Lambda(L)$

The index set  $\Lambda(L)$  provides the basis for most of the polynomial representation, including determining the quadrature set to evaluate the sum in Eq. ???.  $L$  is the polynomial degree of the stochastic collocation expansion. For example, for a single uncertain parameter ( $N = 1$ ) and a fourth-order polynomial approximation ( $L = 4$ ),  $\Lambda$  includes all polynomial orders from 0 to 4, ( $\Lambda = [0, 1, 2, 3, 4]$ ). Each index point  $p \in \Lambda$  corresponds to a polynomial expansion moment of order  $i$ .

In the multivariate case, there are several methods to determine what index set to use. In the most naive case,  $\Lambda(L)$  is a tensor product of polynomial expansion orders,

$$\Lambda_{\text{TP}}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \max_{1 \leq n \leq N} p_n \leq L \right\}, \quad |\Lambda_{\text{TP}}(L)| = (L + 1)^2. \quad (21)$$

For example, for  $N = 2$  and  $L = 4$ , the index set  $\Lambda_{\text{TP}}(L)$  includes all combinations of the expansion orders  $[0, 1, 2, 3, 4] \otimes [0, 1, 2, 3, 4]$  for a total of 25 polynomial expansion indices. The collection of expansion points in this example index set is shown in Fig. ??.

Other index sets with less cardinality can be employed to reduce the number of collocation points. Two in particular are the “total degree” set (see Fig. ??), which is ideal for quantities that are analytic in stochastic space,

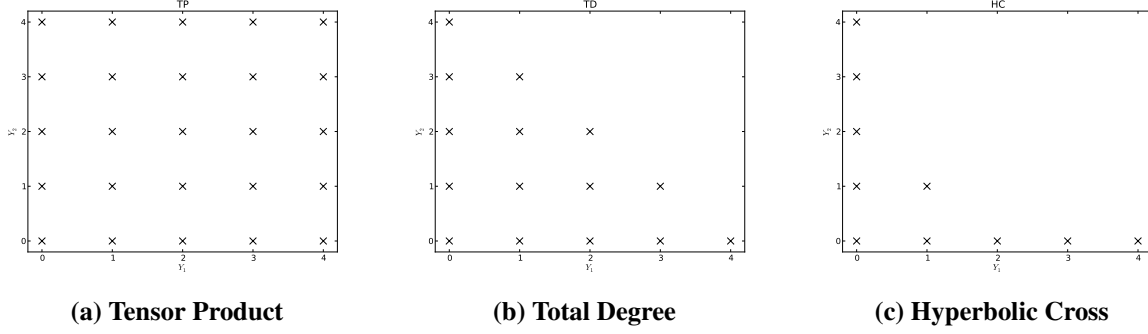
$$\Lambda_{\text{TD}}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \sum_{n=1}^N p_n \leq L \right\}, \quad |\Lambda_{\text{TD}}(L)| = \binom{L + N}{N}, \quad (22)$$

and the “hyperbolic cross” index set (see Fig. ??), for quantities that have limited smoothness in stochastic space,

$$\Lambda_{\text{HC}}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \prod_{n=1}^N p_n + 1 \leq L + 1 \right\}, \quad |\Lambda_{\text{HC}}(L)| \leq (L + 1)(1 + \log(L + 1))^{N-1}. \quad (23)$$

Figure ?? shows each of these index sets for  $N = 2, L = 4$ . The savings of total degree and hyperbolic cross help fight the curse of dimensionality present in tensor product.





**Figure 4. Index Set Examples:**  $N = 2, L = 4$

#### 4.2.2 Sparse Grid Quadrature

The collocation points used in the Lagrange polynomial expansion are obtained based on the index set chosen. To uniquely specify an isotropic Smolyak-like sparse grid, it is necessary to determine the number of uncertain variables  $N$ , the desired expansion order  $L$ , the index set  $\Lambda(L)$ , the quadrature rule  $p(i)$ , and the quadrature points-and-weights generating function for one-dimensional quadratures. This provides the isotropic sparse grid approximation

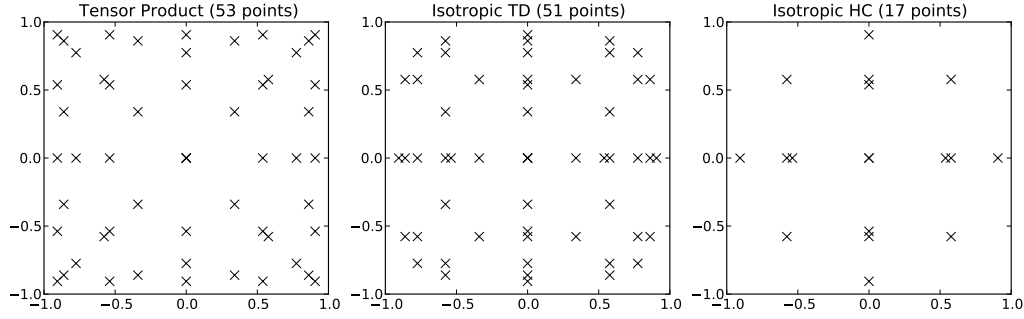
$$u(Y) \approx \mathcal{S}_{N,\Lambda(L)}[k](Y) = \sum_{\mathbf{i} \in \Lambda(L)} c(\mathbf{i}) \bigotimes_{n=1}^N \mathcal{U}_{n,p(i_n)}[u](Y), \quad (24)$$

$$c(\mathbf{i}) = \sum_{\substack{\mathbf{j}=\{0,1\}^N, \\ \mathbf{i}+\mathbf{j} \in \Lambda(L)}} (-1)^{|\mathbf{j}|_1}, \quad (25)$$

$$\bigotimes_{n=1}^N \mathcal{U}_{n,p(i_n)}[u](Y) \equiv \sum_{k_1=0}^{p(i_1)} \cdots \sum_{k_N=0}^{p(i_N)} u_h(Y^{(k_1)}, \dots, Y^{(k_N)}) \prod_{n=1}^N \mathcal{L}_{k_n}(Y_n), \quad (26)$$

$$= \sum_k^{p(\vec{i})} u_h(Y^{(k)}) \mathcal{L}_k(Y), \quad (27)$$

where  $p(i)$  is the “quadrature rule” used to obtain the number of quadrature points for a given expansion order. While this can be chosen arbitrarily, it is common to select  $p(i) = i$  for Gauss quadrature and  $p(i) = 2^i$  for Clenshaw-Curtis quadrature. The sparse grid quadrature is obtained through the tensor product of smaller quadratures, which has less cardinality than the full tensor quadrature. The savings in collocation points are demonstrated in Figure ?? for two identically distributed uniform variables, each on  $[-1,1]$ . The reduction in collocation points grows with the number of input parameters  $N$  and the expansion order  $L$ . A visual example of sparse grids for two variables uniformly distributed between -1 and 1 using Gauss-Legendre quadrature is shown in Fig. ?. For comparison, we show the number of index points for two input spaces of dimensionality  $N$  and several expansion levels  $L$  for all three index sets, as well as the number of collocation points for total degree and hyperbolic cross rules in Table ?. We note that accuracy cannot be



**Figure 5. Sparse Grids,  $N = 2$ ,  $L = 4$ ,  $p(i) = i$ , Legendre points**

directly drawn from polynomial expansions; for this problem the same number of collocation points results in a similar magnitude error in both total degree and hyperbolic cross. This implies, for this problem, that a much lower-order polynomial expansion constructed using the total degree rule is comparable in error to a larger-order polynomial expansion constructed using the hyperbolic cross rule. While the hyperbolic cross collocation points are significantly more sparse than the total

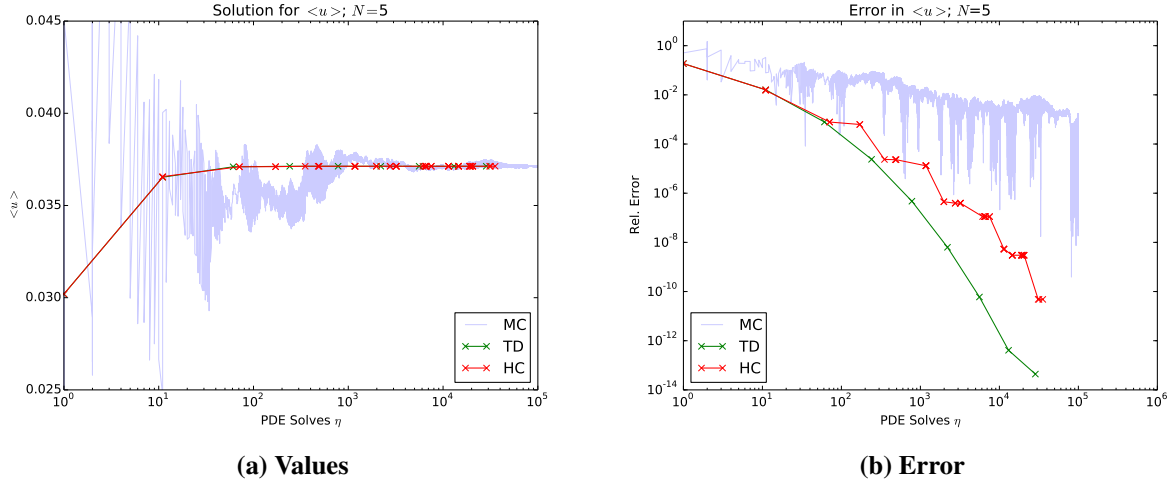
$N$	$L$	TP	TD		HC	
		$ \Lambda(L) $	$ \Lambda(L) $	$\eta$	$ \Lambda(L) $	$\eta$
3	4	125	35	165	16	31
	8	729	165	2,097	44	153
	16	4,913	969	41,857	113	513
5	2	293	21	61	11	11
	4	3,125	126	781	31	71
	8	59,049	1,287	28,553	111	481

**Table III. Index Set and Collocation Size Comparison**

degree collocation points, we note that the increased number of polynomial expansion moments in total degree make it much more accurate for the same total polynomial expansion level  $L$ . The results of this work demonstrate that for this problem, the same number of collocation points using either total degree or hyperbolic cross results in a similar magnitude of error.

### 4.3 Attenuation Problem Results

Figure ?? shows the improvement of the various UQ methods discussed as a function of the number of deterministic solves required. Because each variable contributes identically, anisotropic methods did not show any improvement over collocation. Given the unlimited smoothness of the uncertainty space, we are not surprised to see the total degree sparse grid for stochastic collocation outperform both Monte Carlo and hyperbolic cross sparse grid.



**Figure 6. Attenuation,  $\mathbb{E}[u]$**

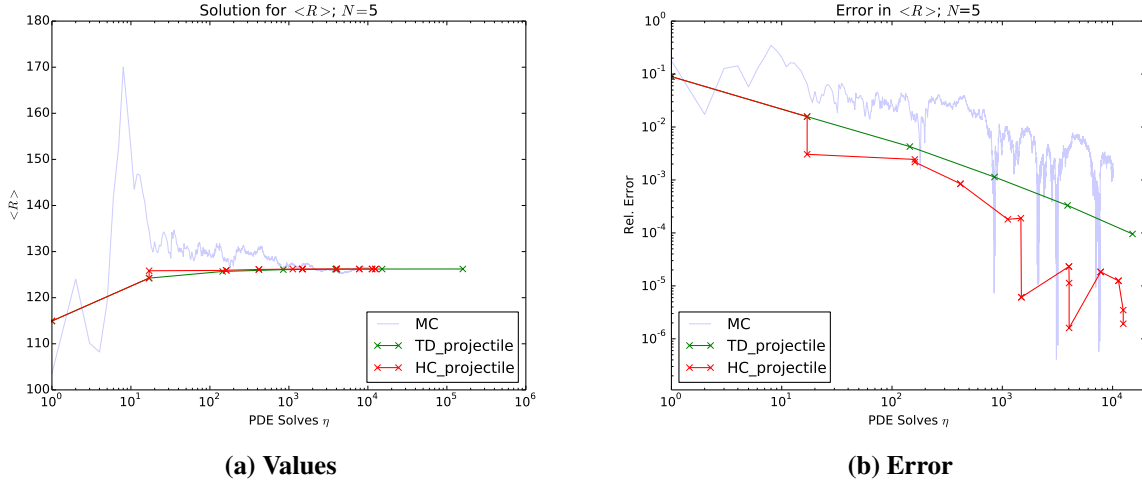
#### 4.4 Projectile Problem Results

Figure ?? shows the improvement of the various UQ methods discussed as a function of the number of deterministic solves required. Because each variable contributes identically, anisotropic methods did not show any improvement over collocation. The uncertainty space for this problem is significantly less smooth than for the analytic attenuation problem, so we see hyperbolic cross converging more efficiently than total degree.

There are two particular points of interest in Figure ?. First, there is a vertical drop in error at  $\eta = 11$  solves. For hyperbolic cross, it is not uncommon for a higher-order polynomial set to require the same number of computational solves as the polynomial order just smaller in order, and yet increase accuracy. This comes from spending less effort on resolving interactions between uncertainties and more effort on resolving contributions from single inputs. Second, there is oscillation in the error convergence below an error of about  $10^{-5}$ . This is an artifact that arises from the deterministic solver. Because of the forward-Euler method of the solver, it converges only as well as the time step selected. In this case, the relative error converged to in the deterministic solver time step was on the order of  $10^{-5}$ , so the stochastic solver cannot converge the uncertain space any further, and any apparent convergence is incidental. Tightening the tolerance in the deterministic solver extends the convergence range of the stochastic solver.

#### 4.5 Diffusion Solver Results

In order to compare magnitude of error, we use a “benchmark” solution generated with a high number of collocation points using isotropic sparse grid stochastic collocation as a reference solution. Monte Carlo and stochastic collocation solutions for varying numbers of PDE solves are computed, and relative error is plotted as a function of number of solves. The error is in the

**Figure 7. Projectile,  $\mathbb{E}[R]$** 

moments  $r$  of the quantity of interest  $k(Y) = u(Y)$ , and is given by

$$\epsilon_h^{(r)} = \frac{|\mathbb{E}[u^{(r)}] - \mathbb{E}[u_{\text{ref}}^{(r)}]|}{\mathbb{E}[u_{\text{ref}}^{(r)}]}, \quad (28)$$

$$\mathbb{E}[u^{(r)}] \approx \mathbb{E}[S_{N, \Lambda_{\text{TD}}(L)}[u](Y)^{(r)}] = \sum_{k=1}^{\eta} w_k u^{(r)}(Y^{(k)}), \quad (29)$$

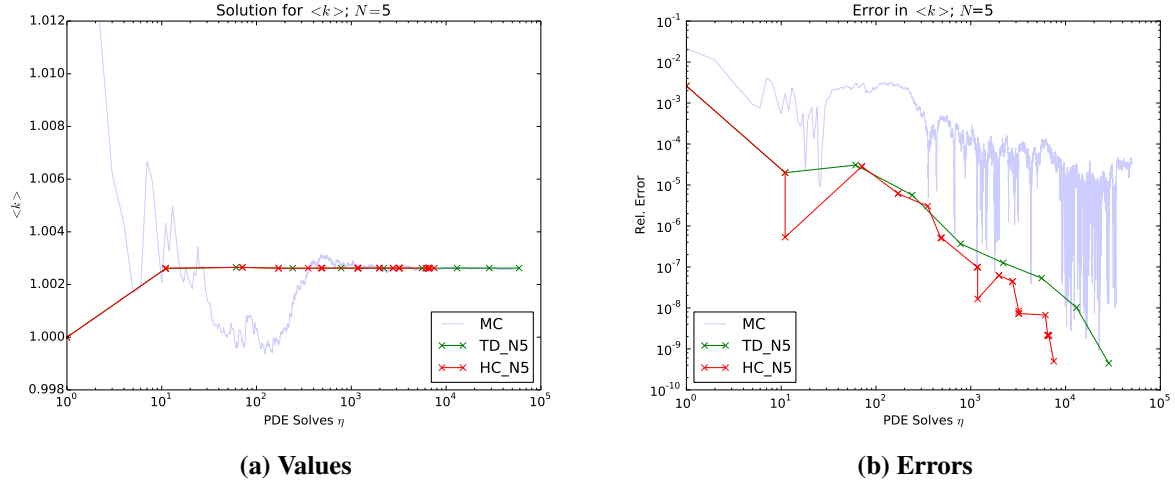
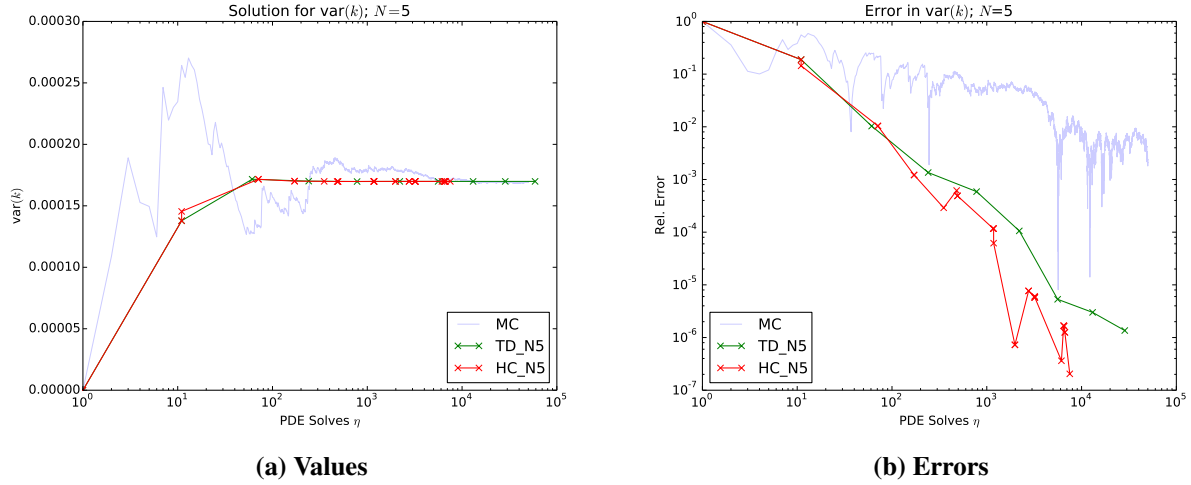
where the weights  $w_k$  and points  $Y^{(k)}$  come from the multivariate quadrature used in stochastic collocation.

Figs. ??-?? show the comparison of Monte Carlo convergence to stochastic collocation for  $N = 1, 3, 5$  where  $N$  is the number of uncertain parameters. Both total degree (TD) and hyperbolic cross (HC) index sets have been included for  $N > 1$  (they are indistinguishable for  $N = 1$ ). Additionally, we include two low-anisotropy anisotropic grids.

Because of the regularity of  $k(Y) = u(Y)$ , the total degree index set is as cost effective as the hyperbolic cross index set. For a less regular stochastic solution, we expect hyperbolic cross would be more efficient. We also expect the convergence rate to diminish with increasing  $N$ , and that trend can be seen in the figures for the mean and variance of  $k(Y)$ . Both the magnitude of the error as well as the convergence rate of stochastic collocation outperforms Monte Carlo for any number of runs. In addition, heuristic selection of importance weighting improved the accuracy of sparse grid methods by approximately half an order of magnitude.

## 5 ADDITIONAL IMPROVEMENTS

Besides using sparse grid quadrature for stochastic collocation, there are a number of other approaches to stochastic collocation that can further reduce the sample space and improve convergence.

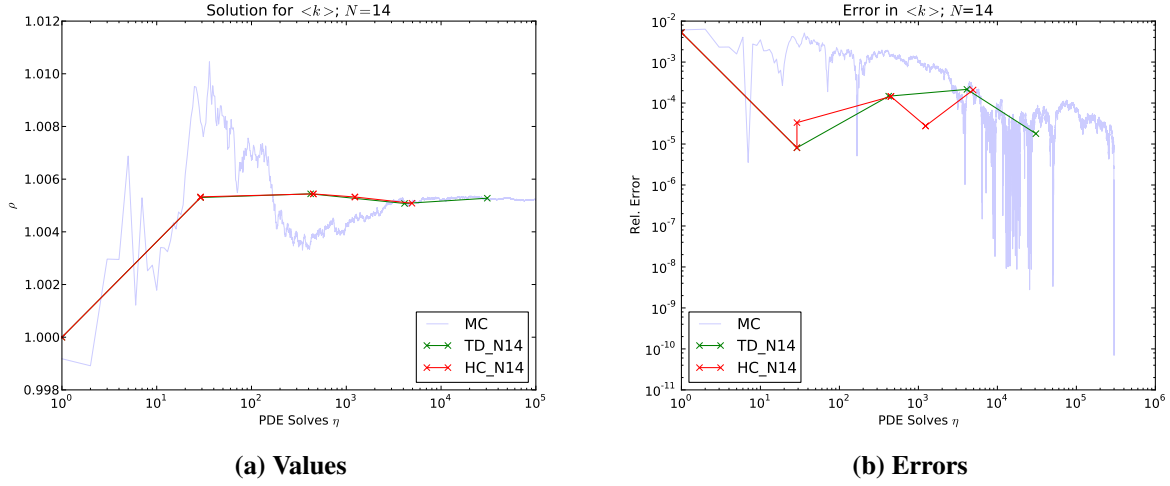
Figure 8. Diffusion,  $N = 5$ , MeanFigure 9. Diffusion,  $N = 5$ , Variance

### 5.1 Anisotropic Sparse Grid

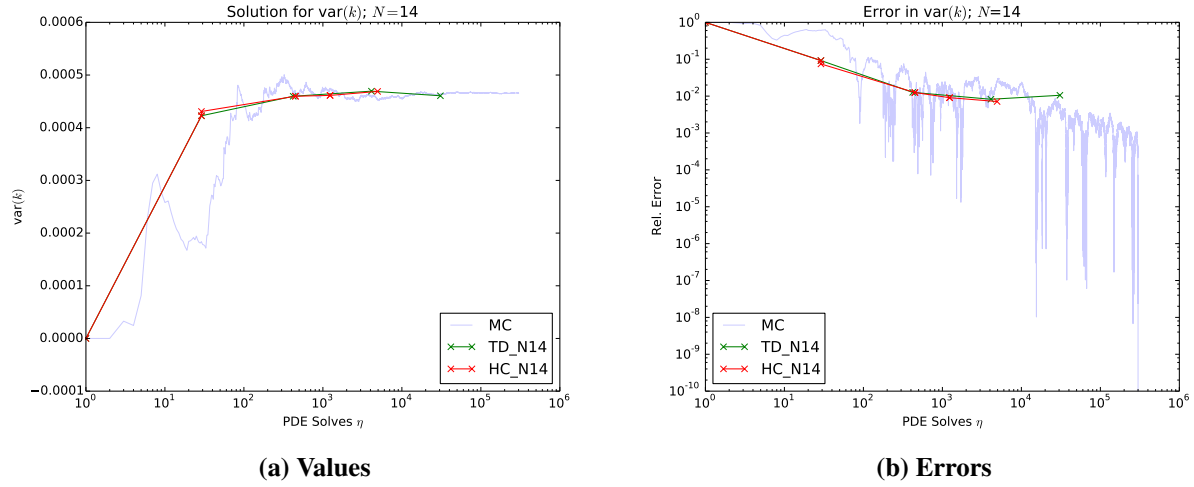
One disadvantage of sparse grids as defined above is that it treats all dimensions of uncertainty space equally. In many cases, however, the sensitivity of the stochastic solution to one dimension is less than another. For example, we heuristically expect the sensitivity of the  $k$ -eigenvalue to the reflecting material (material 5) diffusion coefficient to be much less than the sensitivity to the fission cross section in the main fissile material (material 1).

We can leverage the varying sensitivity by introducing importance parameters  $\vec{\alpha} = [\alpha_1, \dots, \alpha_N]$  that parametrize the sensitivity of the stochastic solution to each dimension. Using the one-norm

$$|\vec{\alpha}|_1 \equiv \frac{1}{N} \sum_{n=1}^N \alpha_n, \quad (30)$$



**Figure 10. Diffusion,  $N = 14$ , Mean**



**Figure 11. Diffusion,  $N = 14$  Variance**

these weight parameters adjust the index set rules  $\Lambda(L)$  for total degree and hyperbolic cross as

$$\tilde{\Lambda}_{\text{TD}}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \sum_{n=1}^N \alpha_n p_n \leq |\bar{\alpha}|_1 L \right\}, \quad (31)$$

$$\tilde{\Lambda}_{\text{HC}}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \prod_{n=1}^N (p_n + 1)^{\alpha_n} \leq (L + 1)^{|\bar{\alpha}|_1} \right\}. \quad (32)$$

In this formulation, greater values of  $\alpha_n$  result in less quadrature points for uncertain input  $Y_n$ . Smaller values of  $\alpha_n$  are assigned to more sensitive dimensions to prioritize collocation points.

## 5.2 HDMR

Despite the gains in efficiency from using anisotropic sparse grids, stochastic collocation still suffers from the curse of dimensionality, becoming computationally unwieldy for more than a dozen uncertain inputs. We can further improve the efficiency of deterministic uncertainty quantification by making use of high-density model reduction (HDMR) techniques [?]. These methods have been applied recently to neutronics problems for significant gain in efficiency [?].

Cut-HDMR makes use of representative “slices” or cuts of the uncertainty space. This slice in uncertainty space makes use of “reference values” for each of the uncertain inputs, often the mean or most likely value for the parameter. We denote the reference value for input parameter  $Y_n$  as  $Y_n^{(0)}$ . Cut-HDMR uses reference values for the majority of inputs at a time and varies small numbers of parameters, then constructs the results as a linear sum of the contributions. It approximates a function as a sum of increasing interactions,

$$u(Y) = H[u](Y) = u_0 + \sum_{n_1=1}^N u_{n_1} + \sum_{n_1=1}^N \sum_{n_2=1}^{n_1} u_{n_1, n_2} + \sum_{n_1=1}^N \sum_{n_2=1}^{n_1} \sum_{n_3=1}^{n_2} u_{n_1, n_2, n_3} + \dots, \quad (33)$$

$$u_0 = u(Y^{(0)}), \quad (34)$$

$$u_n \equiv u(Y_n) - u_0, \quad (35)$$

$$u_{i,j} \equiv u(Y_i, Y_j, \bar{Y}) - u_i - u_j - u_0, \quad (36)$$

and so on. Here  $N$  is the total number of uncertain inputs, and

$$u(Y_n) = u|_{Y_i=Y_i^{(0)}, i \neq n} \equiv u(Y_1^{(0)}, Y_2^{(0)}, \dots, Y_n, \dots, Y_N^{(0)}), \quad (37)$$

indicates holding any variables not explicitly listed at the reference value. For multiple listed uncertain inputs,

$$u(Y_m, Y_n) \equiv u(Y_1^{(0)}, Y_2^{(0)}, \dots, Y_m, \dots, Y_n, \dots, Y_N^{(0)}), \quad (38)$$

This Cut-HDMR representation allows us to approximate by truncating at a particular interaction level  $H$ . For instance, an  $H2$  approximation would only include the terms

$$u(Y) \approx H_2[u](Y) = u_0 + \sum_i^N u_i + \sum_i^N \sum_j^{i-1} u_{i,j}. \quad (39)$$

A variety of methods can be used to represent  $u(Y)$ . In our case, we make use of stochastic collocation on sparse grids, which is most efficient for a moderate number of uncertain inputs. For example,

$$u(Y) \approx H_2[u](Y) \approx S_0 + \sum_i^N S_i + \sum_i^N \sum_j^{i-1} S_{i,j}, \quad (40)$$

$$S_0 \equiv S[u](\bar{Y}), \quad (41)$$

$$S_i \equiv S[u](Y_i, \bar{Y}) - S_0, \quad (42)$$

$$S_{i,j} \equiv S[u](Y_i, Y_j, \bar{Y}) - S_i - S_j - S_0. \quad (43)$$

Moments  $r$  of this HDMR approximation can be obtained simply as - TODO is this true? -

$$\mathbb{E}[u(Y)^r] = \mathbb{E}[H[u](Y)^r] = u_0^r + \sum_i^N u_i^r + \sum_i^N \sum_j^i u_{i,j}^r + \dots \quad (44)$$

### 5.2.1 Attenuation Problem Results

Figures ?? and ?? compare the convergence of the additional methods described here as a function of the number of deterministic solves required. Because each variable contributes identically, anisotropic methods did not show any improvement over collocation and are omitted for this problem. We note that, for five input parameters, it appears HDMR at best can tie the uncertainty convergence obtained by full collocation. This is sensible, as using sparse grid quadrature already removes most of the higher-order interactions between variables from consideration. We expect that as the number of input parameters increases, the benefit of HDMR will as well.

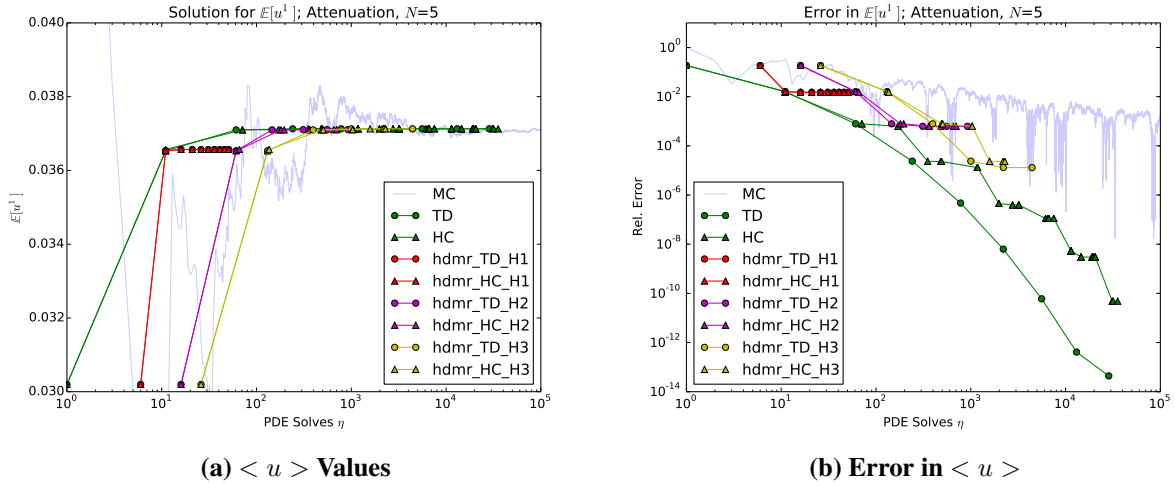


Figure 12. Attenuation UQ Results, Mean

### 5.2.2 Projectile Problem Results

### 5.2.3 Diffusion Problem Results

## 5.3 Anisotropic Sparse Grid

One disadvantage of sparse grids as defined above is that it treats all dimensions of uncertainty space equally. In many cases, however, the sensitivity of the stochastic solution to one dimension is less than another. For example, we heuristically expect the sensitivity of the  $k$ -eigenvalue to the



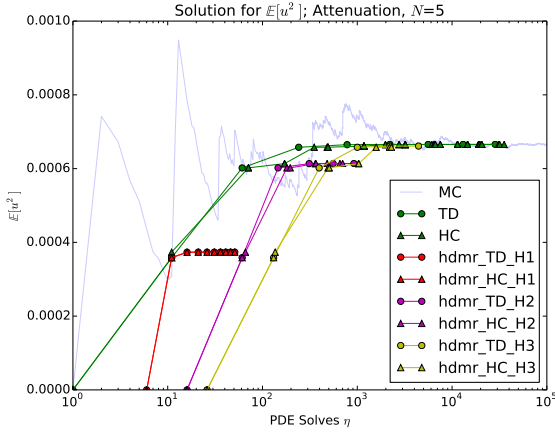
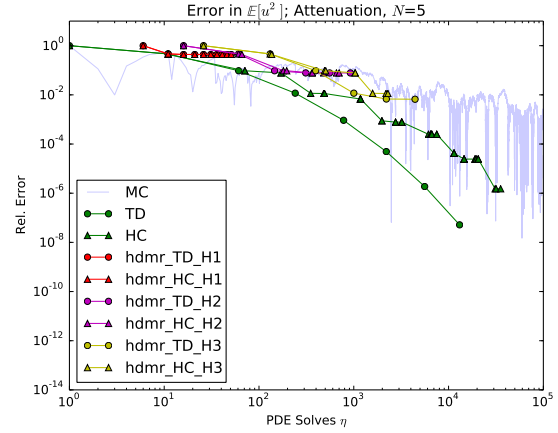

 (a)  $\langle u \rangle$  Values

 (b) Error in  $\langle u \rangle$ 

Figure 13. Attenuation UQ Results, Variance

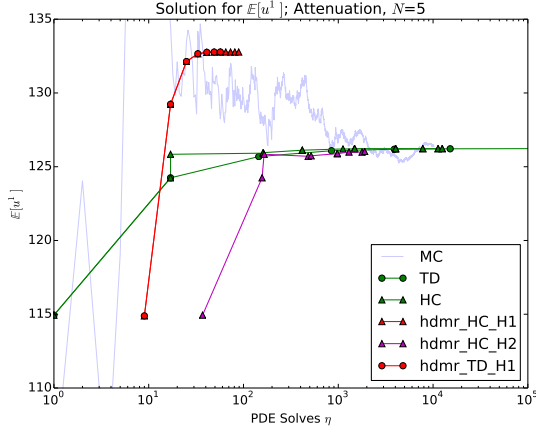
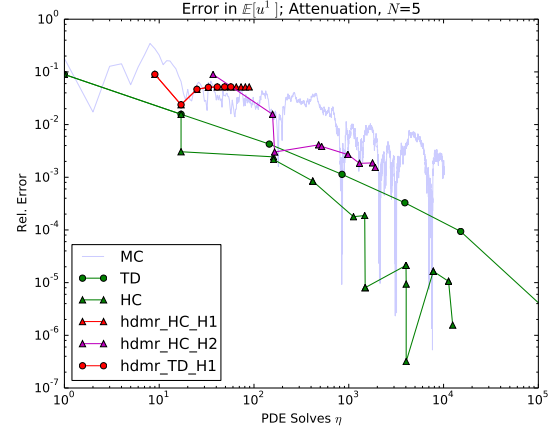

 (a)  $\langle u \rangle$  Values

 (b) Error in  $\langle u \rangle$ 

Figure 14. Projectile UQ Results, Mean

reflecting material (material 5) diffusion coefficient to be much less than the sensitivity to the fission cross section in the main fissile material (material 1).

We can leverage the varying sensitivity by introducing importance parameters  $\vec{\alpha} = [\alpha_1, \dots, \alpha_N]$  that parametrize the sensitivity of the stochastic solution to each dimension. Using the one-norm

$$|\vec{\alpha}|_1 \equiv \frac{1}{N} \sum_{n=1}^N \alpha_n, \quad (45)$$

these weight parameters adjust the index set rules  $\Lambda(L)$  for total degree and hyperbolic cross as

$$\tilde{\Lambda}_{TD}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \sum_{n=1}^N \alpha_n p_n \leq |\vec{\alpha}|_1 L \right\}, \quad (46)$$

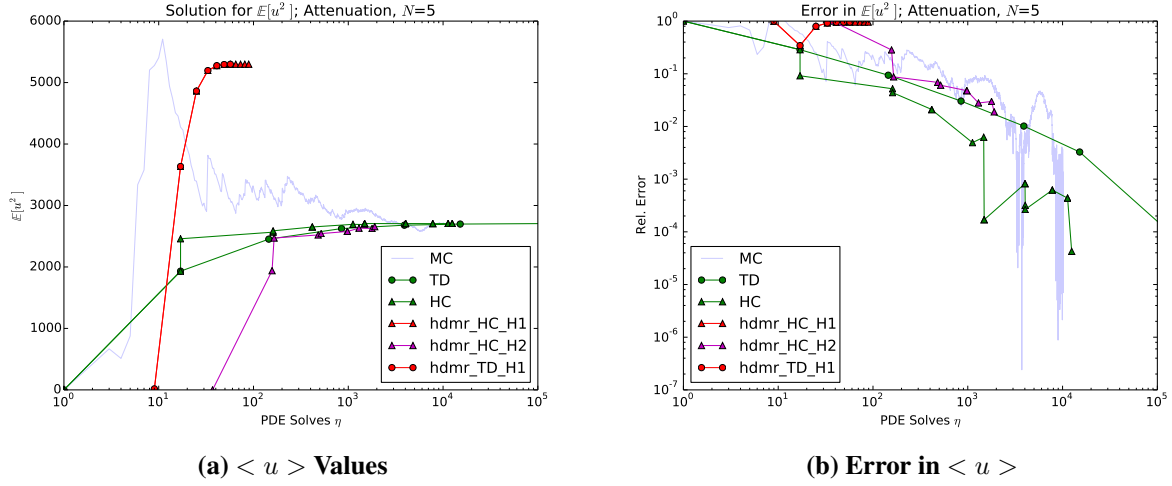


Figure 15. Projectile UQ Results, Variance

$$\tilde{\Lambda}_{\text{HC}}(L) = \left\{ \bar{p} = [p_1, \dots, p_N] : \prod_{n=1}^N (p_n + 1)^{\alpha_n} \leq (L + 1)^{|\bar{\alpha}|_1} \right\}. \quad (47)$$

In this formulation, greater values of  $\alpha_n$  result in less quadrature points for uncertain input  $Y_n$ . Smaller values of  $\alpha_n$  are assigned to more sensitive dimensions to prioritize collocation points.

One method to quantify the sensitivity of the quantity of interest to changes in each input parameter is by using an ANOVA-like approach with a well-converged HDMR calculation. We derive an importance factor based on the contribution of each single variable to the overall second moment.

$$\mathbb{E}[u(Y)^2] \approx \mathbb{E}[H_3[u](Y)^2] = u_0^2 + \sum_{n_1=1}^N u_{n_1}^2 + \sum_{n_1=1}^N \sum_{n_2=1}^{n_1} u_{n_1, n_2}^2 + \sum_{n_1=1}^N \sum_{n_2=1}^{n_1} \sum_{n_3=1}^{n_2} u_{n_1, n_2, n_3}^2, \quad (48)$$

$$\alpha_n = \text{ceil} \left[ -\log \left( \frac{u_n^2}{\mathbb{E}[H[u](Y)^2]} \right) \right]. \quad (49)$$

While not the only method for calculating weight factors, this formulation ensures positive integers with a minimum of 1 for each weight factor.

We omit results for the attenuation case, as each variable contributes identically and no anisotropy performs better than isotropic sparse grid for this case.

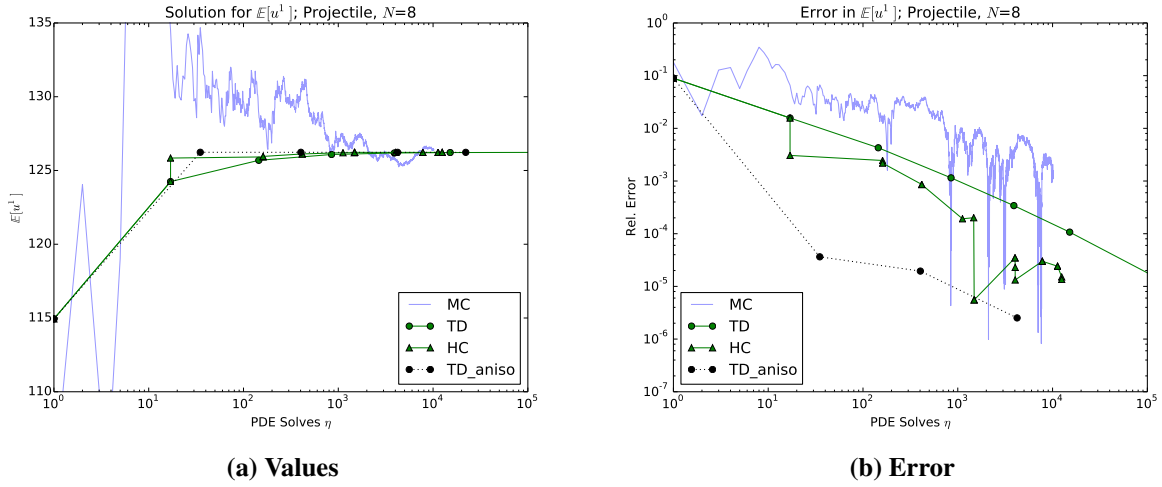
### 5.3.1 Projectile Problem

Table ?? shows values found using an order  $H_1[u](Y)$  approximation for the projectile problem. The drag coefficient  $C$  is clearly the input with the most impact, while the gravitational acceleration constant  $g$  has the least effect. We note that these sensitivities are based on the uncertain ranges given above; changing the uncertainty ranges or distribution could impact the weighting factors  $\alpha$ . As

shown in Figures ?? and ??, employing a variance-based anisotropic sparse grid collocation shows orders of magnitude improvement in convergence and overall error when compared to isotropic sparse grid collocation.

$Y_n$	$Y_n^2$	$\frac{Y_n^2}{H_1^2[u](Y)}$	$\alpha_n$
$m$	2.57E4	6.2E-5	5
$r$	2.61E4	6.3E-5	5
$C$	9.45E7	2.3E-1	1
$\rho$	1.52E3	3.6E-6	6
$v$	4.18E1	1.0E-7	7
$\theta$	1.93E7	4.6E-2	2
$g$	2.77E-3	6.6E-12	12
$y_i$	1.56E4	5.7E-5	5

**Table IV. Anisotropies for Projectile Problem**



**Figure 16. Projectile UQ Results, Mean**

### 5.3.2 Diffusion Problem, $N = 5$

Table ?? shows values found using an hdmr-based  $H_1[u](Y)$  approximation for the diffusion problem. As expected, the capture and fission cross sections in the most prevalent materials have the greatest effect on the  $k$ -eigenvalue. As shown in Figures ?? and ??, employing a variance-based anisotropic sparse grid collocation shows some improvement over an isotropic sparse grid. There also is some indication of the resolution of the benchmark solution; the convergence of the mean becomes unstable around  $10^{-7}$  ( $10^{-5}$  for the variance). We expect a further-resolved benchmark would demonstrate continued convergence in all methods.

TODO Update table values with new ANOVA run

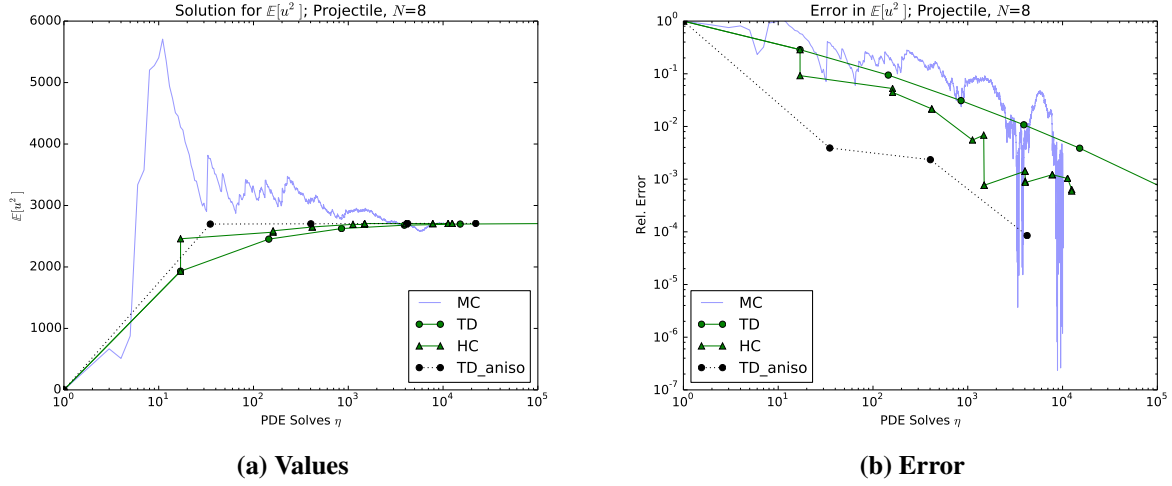


Figure 17. Projectile UQ Results, Variance

$Y_n$	$Y_n^2$	$\frac{Y_n^2}{H_1^2[u](Y)}$	$\alpha_n$
$D_2^{(1)}$	1.75E-13	1.8E-13	13
$\Sigma_{c,2}^{(1)}$	9.86E-6	9.9E-6	6
$\Sigma_{f,2}^{(1)}$	4.80E-6	4.8E-6	6
$D_2^{(4)}$	2.12E-14	2.1E-14	14
$\Sigma_{c,2}^{(4)}$	2.05E-9	2.1E-9	9
$\Sigma_{f,2}^{(4)}$	4.59E-10	4.6E-10	10
$D_2^{(5)}$	6.18E-13	6.2E-13	13

Table V. Anisotropies for Projectile Problem

### 5.3.3 Diffusion Problem, $N = 14$

Table ?? shows values found using an  $H_1[u](Y)$  approximation for the projectile problem. As expected, the capture and fission cross sections in the most prevalent materials have the greatest effect on the  $k$ -eigenvalue.

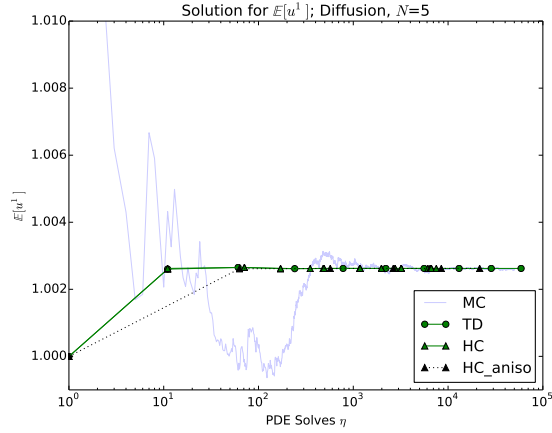
As shown in Figures ?? and ??, employing a variance-based anisotropic sparse grid collocation shows some improvement over an isotropic sparse grid.

## 6 CONCLUSION

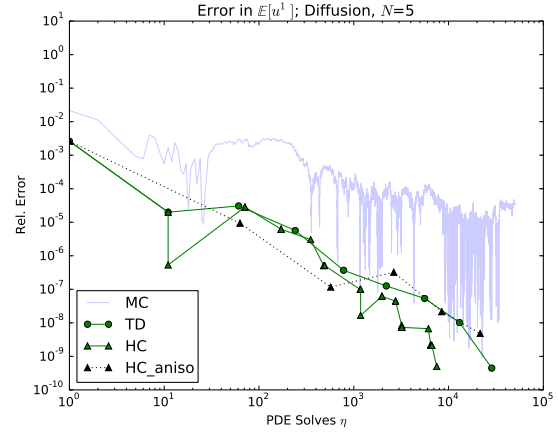
TODO

Ongoing:

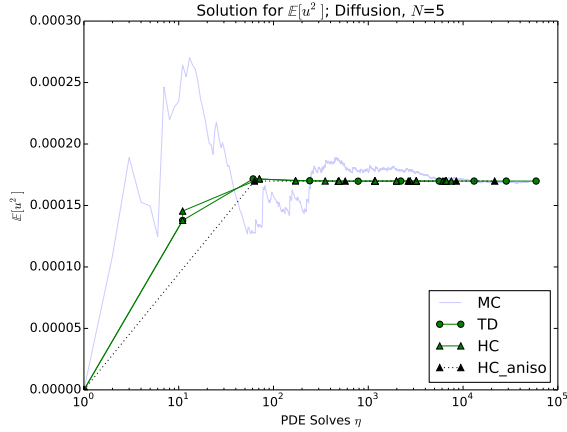
- Adaptive Anisotropic Sparse Grid
- Adaptive HDMR



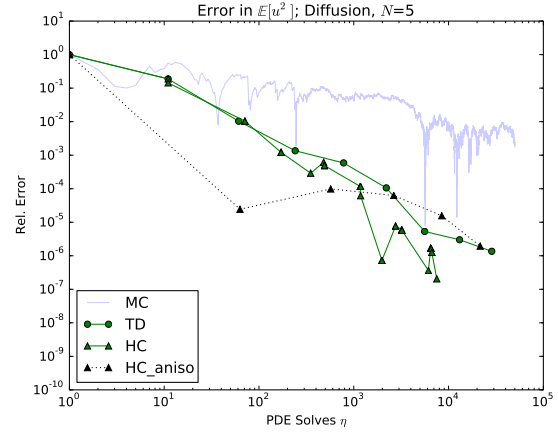
(a) Values



(b) Error

**Figure 18. Diffusion ( $N = 5$ ) UQ Results, Mean**

(a) Values



(b) Error

**Figure 19. Diffusion ( $N = 5$ ) UQ Results, Variance**

## 7 ACKNOWLEDGMENTS

This research was supported by research grants from Idaho National Laboratory.

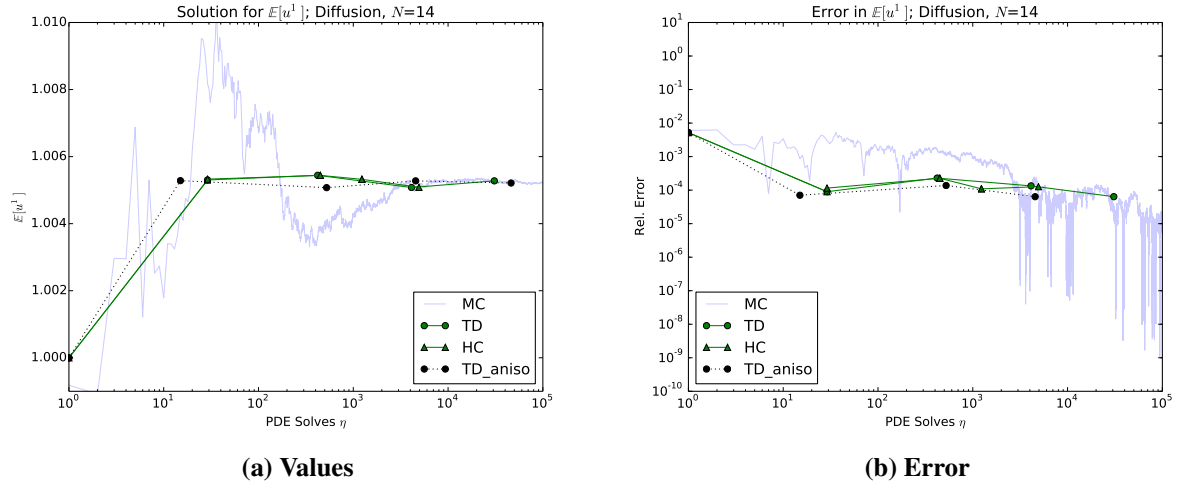
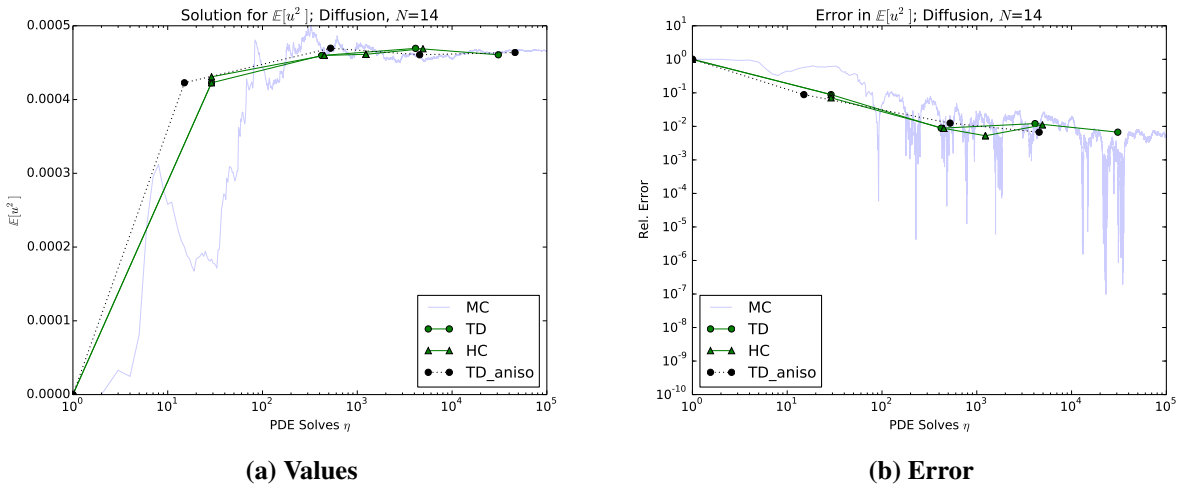
## 8 REFERENCES

- [1] Ivo Babuska, Fabio Nobile, and Raul Tempone. A Stochastic Collocation Method for Elliptic Partial Differential Equations with Random Input Data. *SIAM Journal on Numerical Analysis*, 45, 2007.
- [2] Michael Heroux et. al. An Overview of Trilinos. Technical Report SAND2003-2927, Sandia National Laboratories, 2003.

$Y_n$	$Y_n^2$	$\frac{Y_n^2}{H_1^2[u](Y)}$	$\alpha_n$
$D_2^{(1)}$	1.75E-13	1.8E-13	13
$\Sigma_{c,2}^{(1)}$	9.86E-6	9.9E-6	6
$\Sigma_{f,2}^{(1)}$	4.80E-6	4.8E-6	6
$D_2^{(2)}$	1.76E-13	1.8E-13	13
$\Sigma_{c,2}^{(2)}$	1.44E-6	1.4E-6	6
$\Sigma_{f,2}^{(2)}$	2.67E-7	2.7E-7	7
$D_2^{(3)}$	2.78E-14	2.8E-14	14
$\Sigma_{c,2}^{(3)}$	6.24E-6	6.2E-6	6
$\Sigma_{f,2}^{(3)}$	2.20E-6	2.2E-6	6
$D_2^{(4)}$	2.12E-14	2.1E-14	14
$\Sigma_{c,2}^{(4)}$	2.05E-9	2.1E-9	9
$\Sigma_{f,2}^{(4)}$	4.59E-10	4.6E-10	10
$D_2^{(5)}$	6.18E-13	6.2E-13	13
$\Sigma_{c,2}^{(5)}$	3.24E-10	3.2E-10	10

**Table VI. Anisotropies for Projectile Problem**

- [3] Zhengzheng Hu, Ralph C. Smith, Jeffrey Willert, and C. T. Kelley. High Dimensional Model Representations for the Neutron Transport Equation. *NS&E*, 177, 2014.
- [4] Argonne National Laboratory. Argonne code center: Benchmark problem book. *ANL-7416 M&C Division of ANS*, 1968.
- [5] G. Li, C. Rosenthal, and H. Rabitz. High Dimensional Model Representations. *J. Phys. Chem. A*, 105, 2001.
- [6] Nobile, Tempone, and Webster. A Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data. *SIAM Journal on Numerical Analysis*, 46, 2008.

Figure 20. Diffusion ( $N = 14$ ) UQ Results, MeanFigure 21. Diffusion ( $N = 14$ ) UQ Results, Variance