Using Sparse Grid Interpolation to Facilitate Surrogate Modeling of Stochastic Dynamical Systems

Aditya Sai[a], Nan Kong[a]

[a] *Weldon School of Biomedical Engineering, Purdue University, West Lafayette, Indiana, USA*

Abstract

Certain dynamical models may be unwieldy to simulate repetitively in order to ascertain a global range of model behavior, especially if the model contains uncertainty in the form of stochastic terms. Surrogate modeling using sparse grid interpolation can alleviate the burden associated with increasing dimension of the parameter space. Previous stochastic differential equation (SDE)-based sparse grid approaches have not compared differences in sampling within the time domain and in the diffusion terms present in the equations. This study seeks to explore these features along with other factors considered important to the construction of the sparse grid, including the grid type and the numerical method used to solve the SDE. Results suggest that the grid type is the most important determinant for constructing a low-error interpolant, followed by the type of time points. Future studies will attempt to incorporate adaptive trajectory sampling from the uncertainty space into the sparse grid approach in order to minimize both the interpolation and convergence errors.

Keywords: sparse grid, interpolation, stochastic differential equation, surrogate modeling, systems biology

1. Introduction

1.1. Why use sparse grids and surrogate modeling?

Complex dynamical systems are often difficult to simulate when a large number of experimental factors are being considered [1, 2, 3]. Furthermore, local searches of the pertinent factors may be insufficient to characterize the wide range of possible behaviors exhibited by the system in question. Sparse grids allow for global, computationally efficient exploration of the uncertain space using tensor-product quadrature [4, 5, 6]. These approximations of the underlying model mitigate the curse of dimensionality associated with the increasing dimension of the uncertain space. The resulting surrogate model can be used in model-based control and optimization without sacrificing much of the modeling accuracy and incurring the cost of unnecessary model evaluations.

1.2. Why use stochastic differntial equations (SDEs)?

Most mechanistic and semi-mechanistic mathematical models are developed using ordinary differential equations (ODEs). Nevertheless, random external disturbances are known to influence such systems, necessitating the need to model them with stochastic processes [7].

The types of SDEs we focus on in this paper can be described by the following:

$$\mathbf{X}(t) = f(\mathbf{X}, \mathbf{u}, t, \boldsymbol{\theta})dt + g(\mathbf{X}, \mathbf{u}, t, \boldsymbol{\theta})d\mathbf{B}(t) \quad \mathbf{X}(0) = X_0 \qquad (1)$$

where $\mathbf{X} \in \mathbb{R}^n$ is a continuous time stochastic process, $\mathbf{B} \in \mathbb{R}^m$ is a Brownian motion process, t is time, $\mathbf{u}$ are inputs, $\boldsymbol{\theta}$ are model parameters, $f(\cdot) \in \mathbb{R}^{n\times n}$ is the drift term (deterministic component), and $g(\cdot) \in \mathbb{R}^{n\times m}$ is the diffusion term (stochastic component). We consider the uncertain space to be delineated by the ranges of the model parameters we are interested in.

The models explored in this paper were found in the systems biology literature. Systems biology is concerned with the systems level representation of biological phenomena [8]. Abstractions of these phenomena, represented as mathematical models, often have to be solved numerically using discretized approximations to the true solution. Randomness and heterogeneity plays a large role in influencing certain biological systems, like neurons [9, 10, 11]. Such random forces are prevalent in systems not isolated from the external environment [7].

Examples of SDE-based models can be found in systems biology: (1) the cerebellar granular cell [11], (2) a glucose regulatory system for type 1 diabetes patients [12], (3) the JAK-STAT pathway [13], and (4) the euglycemic hyperinsulinemic clamp [14]. There is a particular need to accurately model these biological phenomena with full consideration of the inherent variability. With surrogate models, researchers can fully study the spectrum of possible hypotheses coded by different model parameter values, without having to directly integrate the underlying model, which is often computationally prohibitive.

1.3. What has been done to adapt sparse grid interpolation to SDEs?

Sparse grids have been applied to stochastic partial differential equations (SPDEs) with random inputs [15, 16, 17, 18, 19, 20], backwards stochastic differential equations with random inputs [21], and differential algebraic equations with random parameters [22]. The objective was to understand how uncertainty propagates from inputs to outputs, an area known as uncertainty quantification. Such work employed the stochastic

collocation approach [23, 24], which approximates the stochastic solution with interpolating polynomials that solve the underlying deterministic problem at predetermined points in the parameter space.

SDEs with white noise were also explored with the sparse grid method [25], with the intent of determining the optimal time step and noise level for first-order convergence of the method. When the integration and discretization step sizes were varied, the increase in the number of random variables along the stochastic process led to an increase in the number of points needed to construct the sparse grid, which we refer to as support nodes.

1.4. Gaps Identified and Contributions of the Paper

One of the poignant gaps revealed by existing SDE-based sparse grid methods is the use of uniformly spaced time points, as in [25]. An alternative approach was proposed in [2], which uses the extrema of Chebyshev polynomials for efficient time domain interpolation of ODE models. Extending this selection scheme to SDEs may provide the reduced memory and improved performance of the sparse grid interpolation algorithm, as opposed to uniform selection of time points.

Another gap concerns the diffusion function, which represents the stochastic component of the SDE and contributes to the system uncertainty. The complexity of the diffusion function is also of particular interest when constructing surrogate models for SDEs. Previous work has not delved into the effect different diffusion functions can have on the accuracy of the surrogate model. Therefore, we also plan to adapt sparse grid methods to accommodate different diffusion functions.

2. Background

2.1. Sparse Grid Interpolation

In sparse grid interpolation, the support nodes are selected in a predefined manner, and there is a nested, hierarchical sampling scheme [5, 6, 26] such that nodes from a previous interpolation depth are used in subsequent interpolation depths.

A mathematical formulation of sparse grids now follows from [4, 5, 27, 28, 29]. Consider a function $f : [0,1]^d \to \mathbb{R}$ that is to be interpolated on a finite number of support nodes. Rescaling can be performed on dimensions that are not of unit length. Here, f is a sample average of multiple SDE trajectories:

$$f(x) = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(x, \omega_i) \tag{2}$$

where $\hat{f}(\cdot)$ is a SDE trajectory, x is the support node, ω_i is the i^{th} SDE trajectory at x, and N is the number of SDE trajectories. A trajectory consists of observations of the underlying process at specific sampling times. f represents the average of all such observed trajectories for each sampling time.

A univariate interpolation function can be constructed:

$$\mathcal{U}^i(f) = \sum_{j=1}^{m_i} a_j^i \cdot f(x_j^i) \tag{3}$$

where $i \in \mathbb{N}$, $a_j^i \in C([0,1])$, $a_j^i(x_l^i) = \delta_{jl}$, $l \in \mathbb{N}$ are the univariate basis functions, $x_j^i \in X_i = \{x_1^i, \ldots, x_{m_i}^i\}$, $x_k^i \in [0,1]$, $1 \le k \le m_i$, are the support nodes.

Extending this interpolation function to multiple dimensions, the multivariate formula, using the full tensor product formulation, is:

$$(\mathcal{U}^{\rangle_\infty} \otimes \cdots \otimes \mathcal{U}^{\rangle_\Gamma})(f) = \sum_{j_i=1}^{m_{i_1}} \cdots \sum_{j_d=1}^{m_{i_d}} (a_{j_1}^{i_1} \otimes \cdots \otimes a_{j_d}^{i_d}) f(x_{j_1}^{i_1}, \ldots, x_{j_d}^{i_d}). \tag{4}$$

The number of support nodes required for the full tensor product representation is $\prod_{j=1}^{d} m_{i_j}$, which is computationally demanding for high dimensions d. The Smolyak construction aims to substantially decrease the number of support nodes used while preserving the interpolation properties in 1-dimensional cases.

Define the difference function $\Delta^i = \mathcal{U}^{\rangle} - \mathcal{U}^{\rangle_{-\infty}}$, $\mathcal{U}' = 0$ and multi-index $\mathbf{i} \in \mathbb{N}^d$, $|\mathbf{i}| = i_{i_1} + \cdots + i_{i_d}$. Now, define the Smolyak interpolant as:

$$A_{n+d,d}(f) = \sum_{k=0}^{n} \sum_{|\mathbf{i}|=k+d} (\Delta^{i_1} \otimes \cdots \otimes \Delta^{i_d})(f). \tag{5}$$

The inner sum can be expressed as

$$\sum_{|\mathbf{i}|=k+d} \sum_{\mathbf{j}} (a_{j_1}^{i_1} \otimes \cdots \otimes a_{j_d}^{i_d})(f(\mathbf{x}_\mathbf{j}^\mathbf{i}) - A_{k+d-1,d}(f)(\mathbf{x}_\mathbf{j}^\mathbf{i})), \tag{6}$$

where $\mathbf{j}$ is the multi-index $(j_1, \ldots, j_d)$, $j_l = 1, \ldots, m_{i_l}^\Delta$, $l = 1, \ldots, d$, and the points $\mathbf{x}_\mathbf{j}^\mathbf{i} = (x_{j_1}^{i_1}, \ldots, x_{j_d}^{i_d})$, $x_{j_l}^{i_l}$ is the j_lth element of $X_\Delta^{i_1} = X^{i_l} \backslash X^{i_l - 1}$, $X^0 = \emptyset$, and $m_{i_l}^\Delta = |X_\Delta^{i_l}|$.

It is also useful to compute the relative and absolute errors of the Smolyak interpolant using correction terms known as hierarchical surpluses:

$$w_\mathbf{j}^{k,\mathbf{i}} = f(x_\mathbf{j}^\mathbf{i}) - A_{k+d-1,d}(f)(x_\mathbf{j}^\mathbf{i}), \tag{7}$$

$$E_{abs}^n = \max w_\mathbf{j}^{n,\mathbf{i}}, \tag{8}$$

$$E_{rel}^n = \frac{\max w_\mathbf{j}^{n,\mathbf{i}}}{\max f(x_\mathbf{j}^\mathbf{i}) - \min f(x_\mathbf{j}^\mathbf{i})}. \tag{9}$$

From this formulation, it is evident that one need only the function evaluations at certain support nodes. Furthermore, the nodes can be chosen in an hierarchical fashion such that $X^i \subset X^{i+1}$.

2.2. Support Node Selection/Grid Type

Sparse grid interpolation commonly employs two types of basis functions that compose the underlying grids:

- Piecewise (multi)linear
- Polynomial

Piecewise linear grids present a reasonable tradeoff between accuracy and cost [26]. Clenshaw-Curtis is the best piecewise linear grid to work with, allowing for dimension adaptivity, where more nodes are sampled along the dimensions of the uncertain parameter space with the highest error. The node selection scheme is:

$$m_i = \begin{cases} 1, & i = 1 \\ 2^{i-1} + 1, & i > 1 \end{cases} \tag{10}$$

$$x_j^i = \begin{cases} \frac{(j-1)}{m_i - 1}, & m_i > 1 \\ \frac{1}{2}, & m_i = 1, \end{cases} \tag{11}$$

where m_i is the number of support nodes for level i, and x_j^i is the position of the j^{th} node at level i, $j = 1, \ldots, m_i$.

Higher-order basis functions, like polynomial grids, can be applied to systems where the function to be interpolated is smooth and higher accuracy is required. These grids include the Chebyshev-Gauss-Lobatto [28] and the Gauss-Patterson grids [27]. For the Chebyshev-Gauss-Lobatto grids,

$$m_i = \begin{cases} 1, & i = 1 \\ 2^{i-1} + 1, & i > 1 \end{cases} \tag{12}$$

$$x_j^i = \begin{cases} -\cos \frac{\pi \cdot (j-1)}{m_i - 1}, & m_i > 1 \\ 0, & m_i = 1. \end{cases} \tag{13}$$

Refer to [27] for a discussion of node selection for the Gauss-Patterson grids.

2.3. Time Interval Selection

Time intervals can be either uniform or non-uniform. With non-uniform time points, a possibility is to utilize the extrema of the Chebyshev polynomials as was done in [2] for ODE models:

$$T_s^i = T_{min}^i + (1 - \cos(\frac{\pi s_i}{d})) \frac{T_{max}^i - T_{min}^i}{2}, \tag{14}$$

where $i \in \{1, \ldots, p\}$ is a vector of indices corresponding to model outputs, d is the degree of the interpolating Lagrange polynomial, T_s^i is a vector of sampling times, T_{min}^i is the minimum time, T_{max}^i is the maximum time, and $s_i = [0, \ldots, d]$.

Once the model outputs are sampled at these sampling times, they can be evaluated at other times t, $T_{min}^i \leq t \leq T_{max}^i$:

$$\tilde{y}_i(\theta, t) = L_d^i(t) \cdot \hat{y}_i(\theta, T_s^i), \tag{15}$$

where $\tilde{y}_i(\theta, t)$ is the interpolated model output with parameters θ at time t, $\hat{y}_i(\theta, T_s^i)$ is the sparse grid model output sampled at the times T_s^i, L_d^i is the Lagrange interpolating polynomial for the i^{th} model output, defined in [30].

2.4. SDE Numerical Methods

Two types of numerical methods are available for use in integrating SDEs: the Euler-Mayurama (Equation 16) and Milstein (Equation 17) Methods, which represent first and second-order stochastic Taylor expansions, respectively [31, 32, 33]:

$$X(i+1) = X(i) + f(X(i), u, i\delta t, \theta)\delta t + g(X(i), u, i\delta t, \theta)(B(i) - B(i-1)), \tag{16}$$

$$X(i+1) = X(i) + f(X(i), u, i\delta t, \theta)\delta t + g(X(i), u, i\delta t, \theta)(B(i) - B(i-1)) + \frac{g(X(i), u, i\delta t, \theta) \cdot g'(X(i), u, i\delta t, \theta)((B(i) - B(i-1))^2 - \delta t)}{2}, \tag{17}$$

where δt is the integration time step and $g'(\cdot)$ is the derivative of $g(\cdot)$ with respect to t.

The approximation accuracy for higher-order diffusion functions will be affected by the method used due to the presence of higher order terms in the Taylor expansion of the more accurate Milstein Method. For constant diffusion functions, there is no difference between the two methods.

3. Methods

3.1. Implementation

The underlying code for the approach discussed here was written in Matlab. This involved two steps. First, the relevant specifications for each model were determined. These specifications include the timespan of the simulation, relevant model outputs, and parameters of interest. Parameters of interest are those parameters of biological importance for the system under study. Second, after the simulation conditions were specified, the model outputs were simulated using the numerical methods discussed and subsequently interpolated using the sparse grid method. Sparse grid interpolation was done using the Sparse Grid Interpolation Toolbox [34]. The factors that were varied for each run are described in the next section.

3.2. Computational Experiments

We list the factors for each simulation run here, followed by simulation conditions for each of the four models (see Appendix A.1- Appendix A.4 for their mathematical representations).

Grid Type	Clenshaw-Curtis (CC) Chebyshev-Gauss-Lobatto (CGL) Gauss-Patterson (GP)
Type of Time Interval	Uniform (U) Non-uniform (NU)
# of Time Points (TP)	16, 32, 64
SDE Numerical Method	Euler-Mayurama (EM) Milstein (M)

Table 1: Factors for Computational Experiments

The metrics with which we will measure accuracy of the resulting interpolant is:

1. relative error of the interpolant computed at the support nodes,
2. # of support nodes used to construct the interpolant,

Timespan of Simulation	$[0, 2]$
Model Outputs	X
Parameters of Interest [Range]	μ [0, 1], σ [0, 1]

Table 2: Simulation Conditions for Arithmetic Brownian Motion (ABM) and Geometric Brownian Motion (GBM) Models.

Timespan of Simulation	[0, 5]
Model Outputs	Q_P,Q_I
Parameters of Interest [Range]	F_T [45, 50], PS [10, 15], V_e [25, 30], σ_1 [0, 0.1], σ_2 [0, 0.1]

Table 3: Simulation Conditions for Dynamic Contrast Enhanced Imaging (DCEI) Model. Parameter ranges were inferred from [7].

Timespan of Simulation	[0, 1]
Model Outputs	x,y
Parameters of Interest [Range]	a [1.9, 2.1], b [0.7, 0.9], c [0.6, 0.8], α [0, 0.5], β [0, 0.5]

Table 4: Simulation Conditions for Predator-Prey (PP) Model. Parameter ranges were inferred from [35].

4. Results & Discussion

4.1. Identifying the Sufficient Number of Realizations

Before determining the superior configuration of factors for the given models, we first attempt to find the necessary number of realizations to ensure an upper bound on the underlying convergence error. For each of the four models, we increased the number of realizations, calculating the relative convergence error each time, stopping once the error was below some predefined threshold, 10^{-4} for the constant diffusion term models and 10^{-2} for the linear diffusion term models. The equations used to compute these relative convergence errors are shown in Appendix A.5. Specifically, Equation A.11 was used for the first two models, and Equation A.12 was used for the latter two. For the ABM and GBM models, 100 uniformly spaced points were selected in the 2-dimensional parameter space in a grid format, and their relative convergence errors were computed. For the DCEI and PP models, 100 points were selected using Latin Hypercube Sampling (using the Matlab command *lhsdesign*) and their relative convergence errors were computed.

Once this procedure was completed for all the models, 1500, 1200, 400, and 400 realizations were deemed sufficient for the ABM, GBM, DCEI and PP models, respectively. The corresponding plots of the error values at these realizations are shown in Figs. 1-4.

4.2. Determining the Superior Configuration

Once the sufficient number of realizations was identified for each model, we proceeded to run different configurations of the factors listed in Table 1 to determine the superior configuration and interpolant. For brevity, we chose to run the full factorial design of configurations for the ABM model only, while running a limited number of configurations for the remaining models. The baseline configuration to run for all models was {EM,CC,U,16 TP}. For the ABM model, all 18 possible configurations were run, whereas for the other models, only 1 factor was varied for each run, resulting in six configurations for the DCEI model and seven for the remaining two.

The results of these runs are shown in Figs. 5- 8. For all models, the polynomial interpolant grids (e.g., CGL, GP) performed better than the piecewise linear grid (e.g., CC). This is to be expected as polynomial interpolants have an extra degree of freedom with which to interpolate these nonlinear models. The use of the Milstein Method provided no additional benefit, incurring a higher relative error and a higher number of support nodes in the GBM model. Uniform time points performed as well as or better than the non-uniform counterparts. Increasing the number of time points did not bring a clear improvement for any model.

Based on these findings, we would suggest that at a minimum, using the Chebhsev-Gauss-Lobatto or the Gauss-Patterson grids is good, as they clearly demonstrated a lower relative error, along with uniform time points. It is possible that using more time points may make a difference in larger models, as only models with at most 2 states were considered here.

5. Conclusions & Future Work

The approach discussed here interpolates the solution provided by an average SDE trajectory at each support node. It does so by analyzing a variety of computational factors that influence the accuracy of the solution, in order to suggest possible configurations that produce superior interpolants. Future work will more compactly and adaptively represent multiple realizations of the SDE solver at each point to account for the stochastic properties of the underlying system by taking advantage of the sparse grid structure. This paper serves as an exploration of the true stochastic dynamics of SDE models using computationally expedient surrogate modeling.

Appendix A. Models Used

Appendix A.1. Arithmetic Brownian Motion (ABM)

$$dX = \mu dt + \sigma dB, \quad X(0) = 1 \tag{A.1}$$

$$X = X_0 + \mu t + \sigma B \tag{A.2}$$

Appendix A.2. Geometric Brownian Motion (GBM)

$$dX = \mu X dt + \sigma X dB, \quad X(0) = 1 \tag{A.3}$$

$$X = X_0 \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma B\right) \tag{A.4}$$

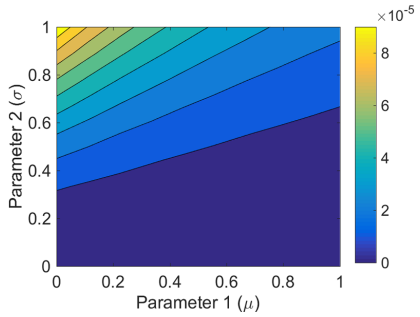

Figure 1: Contour plot of relative convergence errors for parameter space of ABM model with 1500 realizations.

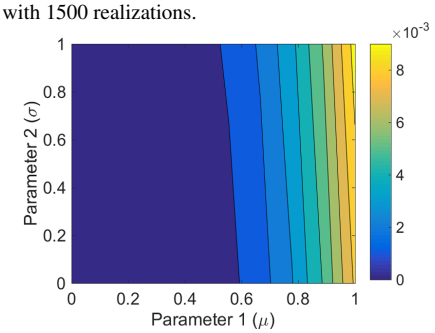

Figure 2: Contour plot of relative convergence errors for parameter space of GBM model with 1200 realizations.

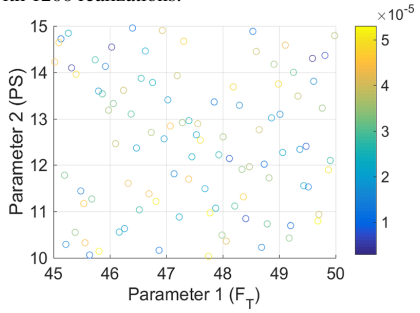

Figure 3: Relative convergence errors for parameter space of DCEI model with 400 realizations. Only the first two dimensions are shown.

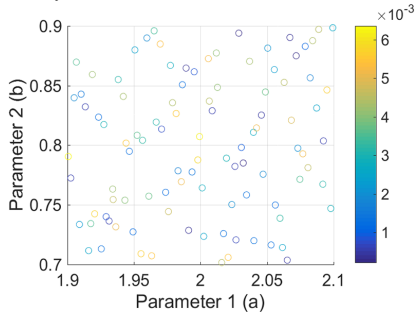

Figure 4: Relative convergence errors for parameter space of PPM model with 400 realizations. Only the first two dimensions are shown.

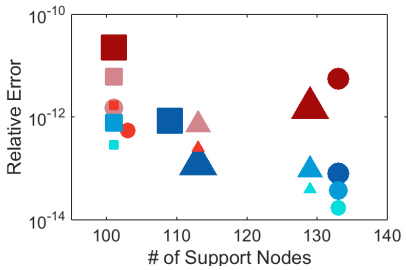

Figure 5: Scatter plot of relative errors and number of support nodes for each configuration of the ABM Model. Circles = CC, Squares = CGL, Triangles = GP, Blue = U, Red = NU, and Size = # of Time Points.

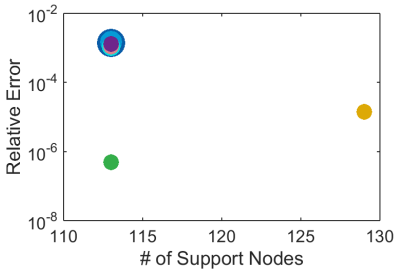

Figure 6: Scatter plot of relative errors and number of support nodes for each configuration of the GBM Model. Light Blue = 16 TP, Medium Blue = 32 TP, Dark Blue = 64 TP, Purple = NU, Green = CGL, Yellow = GP.

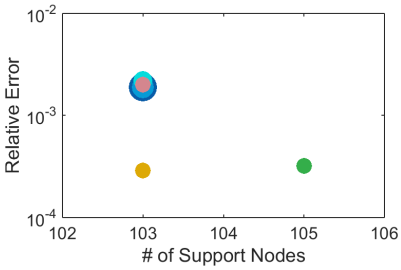

Figure 7: Scatter plot of relative errors and number of support nodes for each configuration of the DCEI Model. Light Blue = 16 TP, Medium Blue = 32 TP, Dark Blue = 64 TP, Purple = NU, Green = CGL, Yellow = GP.

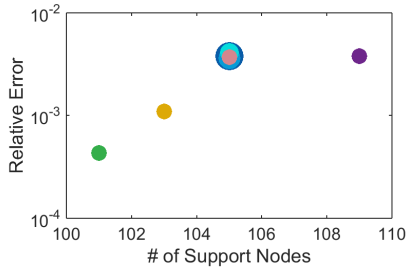

Figure 8: Scatter plot of relative errors and number of support nodes for each configuration of the PP Model. Light Blue = 16 TP, Medium Blue = 32 TP, Dark Blue = 64 TP, Purple = NU, Green = CGL, Yellow = GP.

Appendix A.3. Dynamic Contrast Enhanced Imaging (DCEI)

$$[Q_P, Q_I] = [0, 0] \tag{A.5}$$

$$dQ_P = \left(\frac{F_T}{1-h} - \frac{PS}{V_B(1-h)}Q_P + \frac{PS}{V_e}Q_I - \frac{F_T}{V_B(1-h)}Q_P\right)dt + \sigma_1 dB_1 \tag{A.6}$$

$$dQ_I = \left(\frac{PS}{V_B(1-h)}Q_P - \frac{PS}{V_e}Q_I\right)dt + \sigma_2 dB_2 \tag{A.7}$$

where Q_P (Q_I) is the quantity of contrast agent in the blood plasma (interstitial space), F_T is the blood perfusion flow, h is the hematocrit fraction, PS is the permeability surface area product of tissue, and V_b (V_e) is the blood (extracellular) volume.

Appendix A.4. Predator-Prey System (PP)

$$[x_0, y_0] = [0.6, 0.8] \tag{A.8}$$

$$dx = x\left(a - bx - \frac{cy}{m+x}\right)dt + \alpha x dB_1 \tag{A.9}$$

$$dy = y\left(r - \frac{fy}{m+x}\right)dt + \beta y dB_2 \tag{A.10}$$

where x (y) is the prey (predator) population, a is the growth rate of x, b measures the strength of intra-species competition among x, f and c are the maximum values of the per-capita reduction rate of x due to y, m is the environment protection rate of x and y, r is the growth rate of y, α and β are the intensities of the noise processes.

Appendix A.5. Relative Convergence Error

$$K^* = \arg\min_K \max_p \frac{\frac{1}{N}\sum_{i=1}^{N}[x_{p,K}(i) - \hat{x}_{p,K}(i)]^2}{\frac{1}{N}\sum_{i=1}^{N}\hat{x}_{p,K}^2(i)} \leq \epsilon \tag{A.11}$$

$$K^* = \arg\min_K \max_{p,t} \left|\frac{x_{p,K}(t) - x_{p,K-\delta}(t)}{x_{p,K}(t)}\right| \leq \epsilon \tag{A.12}$$

where K^* (K) is the sufficient (actual) number of realizations, $x_{p,K}(i)$ ($\hat{x}_{p,K}(i)$) is the interpolated (expected) model state with K realizations, for point p in the parameter space, and at discretized time point i, N is the number of time points, ϵ is the error threshold, and δ is the increment in the number of realizations.

[1] M. M. Donahue, G. T. Buzzard, A. E. Rundell, Experiment design through dynamical characterisation of non-linear systems biology models utilising sparse grids, IET systems biology 4 (2010) 249–62.

[2] J. N. Bazil, G. T. Buzzard, A. E. Rundell, A global parallel model based design of experiments method to minimize model output uncertainty, Bulletin of mathematical biology 74 (2012) 688–716.

[3] T. Mdluli, G. T. Buzzard, A. E. Rundell, Efficient optimization of stimuli for model-based design of experiments to resolve dynamical uncertainty, PLOS Computational Biology 11 (2015) e1004488.

[4] T. Gerstner, M. Griebel, Dimension-Adaptive Tensor-Product Quadrature, Computing 71 (2003) 65–87.

[5] H.-J. Bungartz, M. Griebel, Sparse grids, Acta Numerica 13 (2004) 147.

[6] T. Gerstner, M. Griebel, Sparse grids (Quantitative Finance), Encyclopedia of Quantitative Finance 13 (2008) 5.

[7] S. Ditlevsen, A. Samson, Introduction to Stochastic Models in Biology, in: M. Bachar, J. Batzel, S. Ditlevsen (Eds.), Springer-Verlag Berlin Heidelberg, volume 2058 of *Lecture Notes in Mathematics*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 3–35.

[8] T. Székely, K. Burrage, Stochastic simulation in systems biology, Computational and Structural Biotechnology Journal 12 (2014) 14–25.

[9] T. Manninen, M.-L. Linne, K. Ruohonen, Developing Itô stochastic differential equation models for neuronal signal transduction pathways., Computational biology and chemistry 30 (2006) 280–291.

[10] A. Saarinen, M.-L. Linne, O. Yli-Harja, Modeling single neuron behavior using stochastic differential equations, Neurocomputing 69 (2006) 1091–1096.

[11] A. Saarinen, M.-L. Linne, O. Yli-Harja, Stochastic differential equation model for cerebellar granule cell excitability., PLoS computational biology 4 (2008) e1000004.

[12] A. K. Duun-Henriksen, S. Schmidt, R. M. Røge, J. B. Møller, K. Nørgaard, J. B. Jørgensen, H. Madsen, Model identification using stochastic differential equation grey-box models in diabetes., Journal of diabetes science and technology 7 (2013) 431–40.

[13] C. Surulescu, N. Surulescu, Some Classes of Stochastic Differential Equations as an Alternative Modeling Approach to Biomedical Problems, 2013, pp. 269–307.

[14] U. Picchini, S. Ditlevsen, A. De Gaetano, Modeling the euglycemic hyperinsulinemic clamp by stochastic differential equations, Journal of Mathematical Biology 53 (2006) 771–796.

[15] F. Nobile, R. Tempone, C. G. Webster, A Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data, SIAM Journal on Numerical Analysis 46 (2008) 2309–2345.

[16] X. Ma, N. Zabaras, An adaptive hierarchical sparse grid collocation algorithm for the solution of stochastic differential equations, Journal of Computational Physics 228 (2009) 3084–3113.

[17] N. Agarwal, N. R. Aluru, A domain adaptive stochastic collocation approach for analysis of MEMS under uncertainties, Journal of Computational Physics 228 (2009) 7662–7688.

[18] N. Agarwal, N. R. Aluru, Weighted Smolyak algorithm for solution of stochastic differential equations on non-uniform probability measures, International Journal for Numerical Methods in Engineering 85 (2011) 1365–1389.

[19] M. Liu, Z. Gao, J. S. Hesthaven, Adaptive sparse grid algorithms with applications to electromagnetic scattering under uncertainty, Applied Numerical Mathematics 61 (2011) 24–37.

[20] S. Sankaran, A. L. Marsden, A stochastic collocation method for uncertainty quantification and propagation in cardiovascular simulations., Journal of biomechanical engineering 133 (2011) 31001.

[21] G. Zhang, M. Gunzburger, W. Zhao, A Sparse-Grid Method for Multi-Dimensional Backward Stochastic Differential Equations, Journal of Computational Mathematics 31 (2013) 221–248.

[22] R. Pulch, Stochastic collocation and stochastic Galerkin methods for linear differential algebraic equations, Journal of Computational and Applied Mathematics 262 (2014) 281–291.

[23] L. Mathelin, M. Y. Hussaini, A Stochastic Collocation Algorithm for Uncertainty Analysis, NASA Scientifi and Technical Information (STI) (2003).

[24] I. Babuška, F. Nobile, R. Tempone, A Stochastic Collocation Method for Elliptic Partial Differential Equations with Random Input Data, SIAM Journal on Numerical Analysis 45 (2007) 1005–1034.

[25] Z. Zhang, M. V. Tretyakov, B. Rozovskii, G. E. Karniadakis, A Recursive Sparse Grid Collocation Method for Differential Equations with White Noise, SIAM Journal on Scientific Computing 36 (2014) A1652–A1677.

[26] A. Klimke, B. Wohlmuth, Algorithm 847: spinterp: Piecewise multilinear hierarchical sparse grid interpolation in MATLAB, ACM Transactions on Mathematical Software 31 (2005) 561–579.

[27] T. Gerstner, M. Griebel, Numerical integration using sparse grids, Numerical Algorithms 18 (1998) 209–232.

[28] V. Barthelmann, E. Novak, K. Ritter, High dimensional polynomial interpolation on sparse grids, Advances in Computational Mathematics 12 (2000) 273–288.

[29] A. Klimke, K. Willner, B. Wohlmuth, Uncertainty modeling using fuzzy

arithmetic based on sparse grids: applications to dynamic systems, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 12 (2004) 745–759.

[30] G. T. Buzzard, D. Xiu, Variance-based global sensitivity analysis via sparse-grid interpolation and cubature, Communications in Computational Physics 9 (2011) 542–567.

[31] D. J. Higham., An Algorithmic Introduction to Numerical Simulation of Stochastic Differential Equations, SIAM Review 43 (2001) 525–546.

[32] T. Sauer, Numerical Solution of Stochastic Differential Equations in Finance, in: Handbook of Computational Finance, volume 47, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 529–550.

[33] T. Sauer, Computational solution of stochastic differential equations, Wiley Interdisciplinary Reviews: Computational Statistics 5 (2013) 362–371.

[34] A. Klimke, Sparse Grid Interpolation Toolboxuser's guide, IANS report (2006).

[35] C. Ji, D. Jiang, N. Shi, A note on a predator-prey model with modified Leslie-Gower and Holling-type II schemes with stochastic perturbation, Journal of Mathematical Analysis and Applications 377 (2011) 435–440.