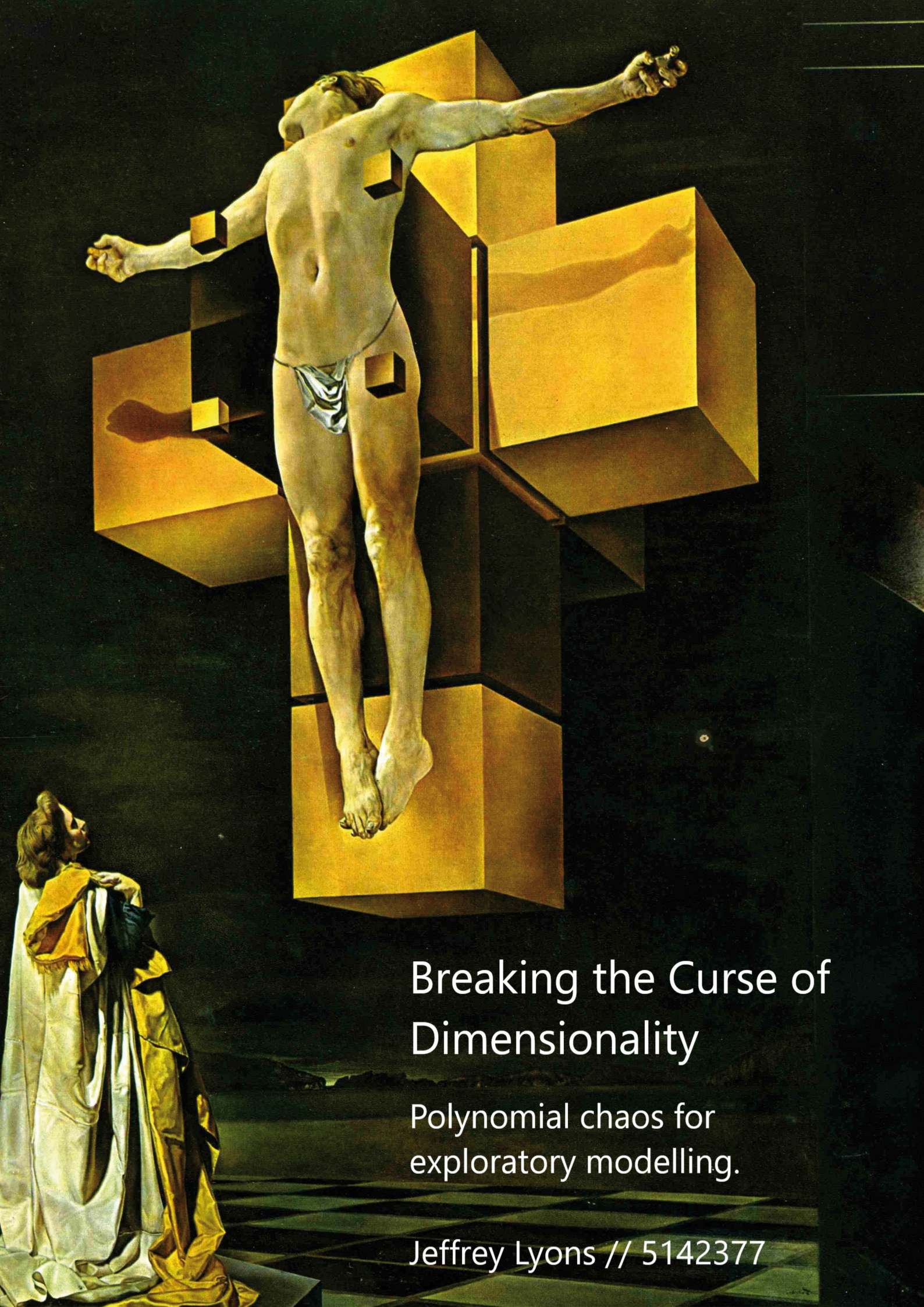


Breaking the Curse of Dimensionality

Polynomial chaos for exploratory modelling.

Jeffrey Lyons // 5142377

BREAKING THE CURSE: DIMENSION-ADAPTIVE POLYNOMIAL CHAOS FOR EXPLORATORY MODELLING.

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of

Master of Science in Complex Systems Engineering and Management

by

Jeffrey Lyons

June 2022

Jeffrey Lyons: *Breaking the Curse: Dimension-adaptive Polynomial Chaos for Exploratory Modelling.* (2022)
Coverpage: *Corpus Hypercubus* - Salvador Dali - oil-on-canvas, 1954

The work in this thesis was made in the:

Public policy group
Multi-Actor Systems
Faculty of Technology, Policy, & Management
Delft University of Technology

Supervisors: Prof. dr. Jan Kwakkel
Prof. dr. Igor Nikolic

GUIDING LIGHTS

Il faut cultiver notre jardin.

Voltaire

Or set upon a golden bough to sing
To lords and ladies of Byzantium
Of what is past, or passing, or to come.

W.B Yeats

The secret to modelling is not being perfect.

Karl Lagerfeld

ACKNOWLEDGEMENTS

I would like to extend my sincerest thanks to my Supervisors Prof. Dr. Ir. Jan Kwakkel and Prof. Dr. Ir. Igor Nikolic for both their academic and personal support. Without their time and patience this project would not have reached completion and I would likely still be stuck wandering the TPM faculty to this day. As a trio we shepherded this project through many extended delays, at least three waves of COVID19, and a host of other personal and professional challenges. Throughout the process they were more like friends and colleagues rather then overbearing supervisors, and I am incredibly grateful to have spent time under their respective wings. I would also like to thank my family for their unending love and support, without which I could, quite literally, not have completed this Master's degree. This document, and its corresponding examination, marks the end of a twenty two year long odyssey in education, and I will be forever thankful to them for their support, and for setting me back on my feet when I needed it most. My thanks to the Networks boys whose antics, voices, and friendship was a balm during long nights in TPM. My thanks also to the Galway Braintrust whose Wednesday power breakfasts provided motivation and succour throughout the winter period. In conclusion, I would like to thank my girlfriend Maureen for the countless meals, conversations, and hours spent dancing which helped shake off the effects of a day spent computing.

...

ABSTRACT

In light of worsening climate change and an increased interest in adapting infrastructure to cope with its effects, model-based decision support has become an essential tool for policy makers. In conditions of deep uncertainty, models may be used to explore a large space of possible system behaviours and so encourage a wider consideration of the possible futures. Such methods, where the focus is intentionally broad, fall under the remit of exploratory modelling and are a potential antidote to traditional predictive modelling methods where only a marginal treatment of uncertainty is attempted. One serious issue limiting the full exploitation of exploratory modelling is its computational intensity, the many computational experiments requiring large amounts on computing power which makes some analyses too expensive to attempt. In order to fully exploit the promise of exploratory modelling new methods of reducing computational intensity are needed. Polynomial chaos expansions (PCEs) are one class of methods which may fulfill this role. Our results conclusively demonstrate that PCEs are capable of accurately reproducing statistical moments and determining Sobol sensitivity indices significantly faster than direct-sampling methods, often requiring orders of magnitude fewer function evaluations. However, we found that the *curse of dimensionality* rendered conventional PCEs too costly for use with higher-dimensional models. We found that conventional sparse grids were effective at reducing the computational cost associated with fitting PCEs with high-dimensional models, as long as the model output was sufficiently smooth. For models where sparse grids were able to converge with reasonable accuracy, supplementing the PCCs with a modified Gersnter's dimension-adaptive algorithm further improved convergence times. The anisotropic refinement strategy employed by the algorithm allows for accurate determination of Sobol sensitivity indices with a minimum of computational effort.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Exploratory modelling (EM) is a broad methodology which serves as a counterpoint to traditional predictive modelling approaches and can be used to augment model-based decision support in the presence of irreducible uncertainties (Kwakkel and Pruyt, 2013). First proposed by Bankes in his seminal 1993 contribution, the aim of exploratory modelling is to understand and record the behaviour of a given model over a range of inputs by performing computational experiments (Bankes, 1993). Such computational experiments are akin to a 'what-if' analysis, and allow the modeller to investigate the impact of various assumptions on the model's behaviour and outputs.

The essence of the approach may be illustrated by considering a very simple model of disease spread in a population, where the number of infected individuals is assumed to grow exponentially over time. The model consists of a single equation which depends on a single growth parameter, α. The number of infected individuals at time t may be written as,

$$N_t = exp(\alpha t)$$

The behaviour of the model is determined by the value for α which is assumed by the modellers. A traditional modelling approach would mandate assuming a value for α based on the best available information and running the model to collect results. Any uncertainty in the value of α will lead to uncertainty in the model's outputs, reducing the accuracy of the results. In our example, in the early stages of an epidemic there may be little or no information as to the growth parameter of the disease so any assumption will come with large uncertainty. Instead of estimating a value based on possible incomplete data, the modellers could instead turn to exploratory modelling, and investigate how the model output changes for different values of the growth parameter α, sampled over a large range. In this respect, exploratory modelling allows the modellers to investigate a range of 'worlds' typified by different values for α.

Figure 1.1 shows the results of a computational experiment where the model was run for different values of the growth parameter. It is immediately clear that the number of new daily infections is highly sensitive to the value of the growth parameter, with small changes in α leading to large changes in model behaviour as reflected in the number of new daily infections. The sensitivity of the model output to changes in the growth parameter indicate that the accuracy of the results relies on minimising uncertainty in the value of α as much as possible. In the face of such parametric uncertainty, the modelling team may use their new-found knowledge on the complete behaviour of the model to guide their response to the epidemic. For instance, they could plan for the worst-case scenario, corresponding to $\alpha = 3$, or employ a wait-and-see approach, hoping that the growth parameter is lower. In the face of uncertainty, exploratory modelling allows for a more holistic understanding of the different world-states that can be produced by the model without an over-reliance on assumptions.

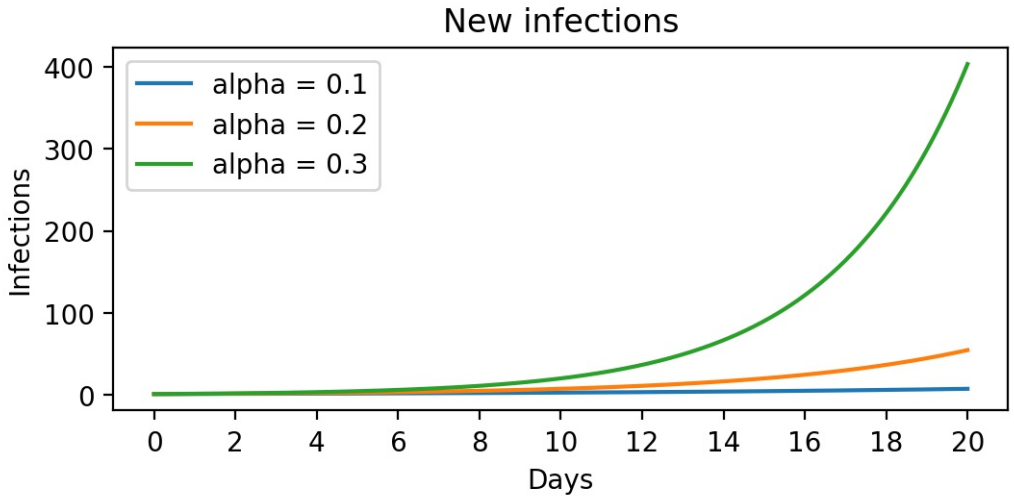


Figure 1.1: Figure showing the time evolution of new infections for our basic epidemiological model. The model is 'run' for each of the values for α pictured in the level, with significant changes in behaviour observed for variation in the growth parameters.

While in our example we only had to deal with a single uncertain parameter, it is not uncommon for decision support models to have very large numbers of uncertain parameters, each of which may have to be sampled over a large range in an exploratory modelling study. An example of a two-dimensional uncertainty space is illustrated in Figure 1.2. The model is run for different points in the uncertainty space which specify a value for each of the uncertain parameters. Sampling from such a large, multi-dimensional uncertainty spaces can quickly become computationally prohibitive, with computational time scaling exponentially with the number of dimensions (Gerstner and Griebel, 2003). This phenomenon is known as the *curse of dimensionality* and specifically refers to the exponential scaling of hyper-cube volume with increasing dimension (Verleysen and François, 2005). If the object of EM is to investigate the behaviour of a model for each point in an uncertainty space then the time required for such a study increases exponentially with the dimension of the space.

This computational bottleneck is a serious barrier to the wider implementation of exploratory modelling in service of policy design, particularly for organisations with limited computational resources. In order to further support the growth of robust policy design, new methods for managing the computational cost of high-dimensional exploratory modelling projects are needed.

Polynomial Chaos Expansions (PCE), a surrogate modelling approach to the sensitivity analysis of computational models, have the potential to reduce the computational intensity associated with EM projects. Surrogate, or meta-modelling, involves approximating the output of model with some appropriate mathematical structure (Sobester et al., 2008). This is equivalent to finding a function, which when fit to a certain number of model input-output pairs serves as a replacement, or surrogate, for the model itself. A sufficiently good surrogate model may then be used to calculate accurate output values for a given input without having to run the model itself, potentially reducing the computational intensity of the study. The concept is illustrated in Figure 1.3. PCEs can be used to approximate the response of a model with stochastic inputs. In the context of exploratory modelling, the uncertainty associated with a model's input parameters may transform a deterministic model into a stochastic one.

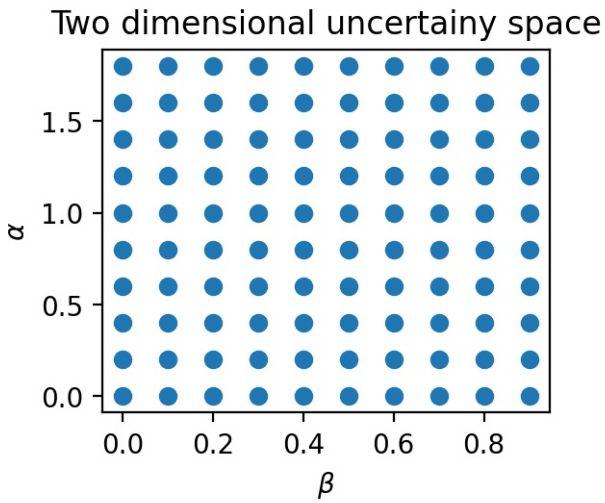


Figure 1.2: Figure showing a two dimensional uncertainty space and an exemplar sampling plan. In this example, the uncertainty ranges for each parameter are indicated on the corresponding axis. An example sampling plan is indicated by the blue points. The model which depends on the parameters in this space would be run for each point in the parameter space. The sampling plan can be made arbitrarily dense to better cover the uncertainty space.

Once the PCE has been fit with a certain amount of data from the model, the fitted PCE itself may be used as an accurate surrogate of the model. In addition to the aforementioned computational time advantages of surrogate modelling, PCEs have also been favoured owing to the ease with which they can be used to 'cheaply' calculate Sobol sensitivity indices (Sudret, 2008).

Sensitivity analysis involves assessing the influence that uncertainty in the input parameters have on uncertainty in the model output (Saltelli et al., 2008). A thorough sensitivity analysis is central to good modelling practice and is a central sub-method of EM . The example illustrated in Figure 1.1 achieves a basic sensitivity analysis in that it can readily be observed how sensitive the model is to uncertainty in the growth parameter α.

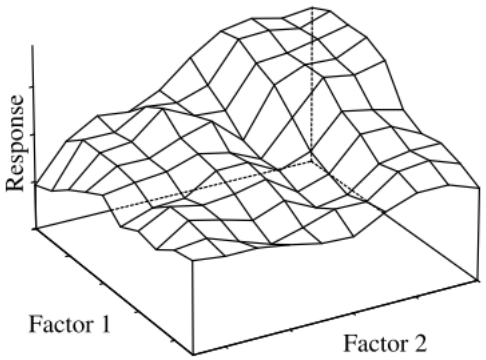


Figure 1.3: Figure showing a representation of a response surface for a two-factor model. The surface in the above model is approximated by a tessalation of planes, in a process not dissimilar to the interpolation of a smooth function by polynomials.

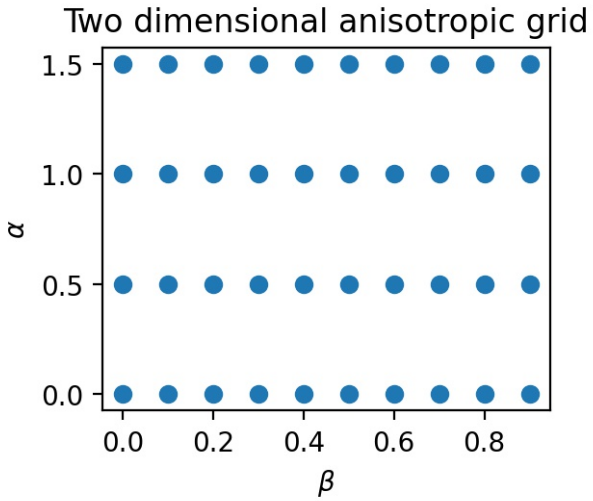


Figure 1.4: Figure showing a two-dimensional anisotropic sampling plan for two uncertain variables, α and β. The anisotropy is manifested in the direction dependent sampling density.

PCEs have been supplemented in the literature through the use of a *dimension-adaptive* sampling schemes, first developed by (Gerstner and Griebel, 2003) for the integration of high-dimensional integrals (Edeling et al., 2021). Dimension-adaptive sampling aims to reduce the computational cost associated with computing high dimensional integrals by concentrating function evaluations on dimensions where the greatest resolution is required. With respect to fitting PCEs, a process equivalent to integration, such an algorithm may be used to construct an *anisotropic-grid* where certain dimensions are sampled at higher densities than others. An example of such a grid is shown in Figure 1.4. The combination of both methods shows promise for executing sensitivity analysis for large numbers of uncertain parameters where traditional *pseudo-random* methods are computationally prohibitive (Edeling et al., 2021).

1.0.1 Research gap

The potential of exploratory modelling as a method class for supporting model-based decision making is hampered by the *curse of dimensionality*. Decision makers and modellers without access to high-performance computing resources are unable to fully leverage the power of exploratory modelling to better understand the behaviour of their models under situations of deep uncertainty. In order to realise the potential of EM, new methods of mitigating the exponential cost scaling of high-dimensional models are needed. Polynomial chaos expansions have been highlighted in the literature for their ability to model the output of stochastic models. Particularly with respect to sensitivity analysis, they have been shown to converge significantly faster than pseudo-random and brute-force methods for problems with moderately low-dimensions. Studies which have aimed to assess the performance of PCEs under a range of conditions and test functions, have so far concentrated mainly on low-dimensional, 'well-behaved' analytical test functions. In addition, although there is precedent in the literature for the supplementation of PCEs with dimension adaptive sampling, there has been little focus on applying the combined methods to complex policy models. Before these novel methodologies are adopted by the EM field, a more rigorous examination of their fitness is needed on problems which are more representative of those commonly approached in the

field where many-dimensional, highly non-linear and non-smooth models are the norm.

1.0.2 Reseach questions

In order to address the research gap illustrated in the previous section, the following research questions have been formulated:

MRQ: How can polynomial chaos expansions with dimension adaptive sampling be applied to decrease computational cost in exploratory modelling projects?

1. What are the relevant features of polynomial chaos expansions and how may they be employed in exploratory modelling projects?
2. How can dimension-adaptive sampling be used to facilitate faster convergence of polynomial chaos expansions?
3. Which measures of performance may be used to evaluate the efficacy of these methods in exploratory modelling projects?
4. What is the relative efficacy of these methods versus other *best-in-class* algorithms?

1.0.3 A note on the research approach

The research questions defined above constitute a logical and summative decomposition of the main research question. Answering each in turn should contribute to satisfactorily answering the main research question. Although the sub-questions are related, their individual nature necessitates a tailored research approach.

As a cohesive research plan, the phases outlined below form a *mixed-methods* research project, with a focus on the quantitative evaluation of a novel class of algorithms. Phases I - II will consist of desk research, with the aim of building an epistemic foundation for the quantitative portion of the study. In Phase I, the theoretical basis for both polynomial chaos expansions and dimension-adaptive sampling will be explored and the relevant features elaborated. This includes their mathematical construction, as well as their implementation as computer algorithms. In addition, the existing use-cases of these methods in adjacent fields of study, as well as their likely role in EM, will be explored.

Phase III will elucidate the relevant performance measures according to which the novel methods will be judged, as well as ascertaining which existing algorithms they should be judged against. In this respect, Phase III aims to delineate the relevant experimental parameters, test problems, and contexts under which the performance of the novel algorithms will be evaluated.

Phase IV forms the quantitative centre of this study since it will provide data as to the performance of the algorithms against others in the field. The results gained in this phase will allow for a conclusion to be made as to value of polynomial chaos expansions with dimension-adaptive sampling for exploratory modelling applications.

1.0.4 Research flow

The character of the research sub-questions and the research methods required to answer them split the research project into three distinct phases, illustrated in 5.3.

	Research question	Research method
Phase I	1,2	Literature review
Phase II	3	Exploration & Experimental design
Phase III	4	Experiments and analysis

Table 1.1: Outline of the given project phases and research methods.

Phase I - Exploration: The algorithms, their structure, features, and applications

Phase I aims to assess the literature surrounding both polynomial chaos expansions and dimension-adaptive sampling methods. In answering sub-questions one and two it will organise the existing literature and terminology of the field, will form the epistemic foundation of the research project, and will contribute the knowledge necessary to implement them in the quantitative portion of the study. The main deliverable of this research phase will be a literature review and a detailing of the relevant mathematical foundation . As such, this phase constitutes a desk research phase.

Aspects to be examined in this phase are: the mathematical bases of both methods, as well as where they are positioned in their original fields of study, how they are usually implemented as algorithms, and the problems they are usually used to address. Based on the existing use-cases identified in the literature review a proposition will be made as to how the algorithms may contribute to solving issues which arise in EM projects.

Phase II - Exploration: The experimental arena

Phase II involves the task of constructing the set of computational experiments which will be used to evaluate the efficacy of both algorithms. Relevant measures of performance will be selected with justification from the literature regarding both the orthodox use of PCE+DAS, as well as its proposed use in EM projects.

A set of test-problems will be found which are representative of the 'real-world', high-dimensional, and non-smooth policy models which are commonly approached through the use of EM. In order to provide for the widest possible applicability of our results, these models should be chosen such that the measures of performance can be determined under a number of different contexts, for example, with varying model dimension or complexity. To achieve this condition, it may be required to use more than one test-problem

The main data source for this phase will be the literature which implement PCE and DAS for use with various models, as well as the exploratory modelling literature which deals with similar problems.

Where we seek to implement PCE+DAS for the sensitivity analysis of high-dimensional models, the computational experiments will involve the computation of first, second, and total-order Sobol sensitivity indices for each of the model parameters. Relevant performance mea-

sures in this case include the speed of convergence as measured by the number of function evaluations until a pre-defined halting condition is reached. In the case a model is used for which the Sobol indices are known *a priori* an error metric can be calculated. Where an error measure can be computed it will be of interest to examine the trade-off between computational speed and accuracy as illustrated in (Crestaux et al., 2009).

On a practical level the algorithms will be implemented in this phase in accordance with the experimental design and factors determined through the literature review.

Phase III: Experimentation - 'A fight to the ϵ'

Phase III forms the quantitative heart of the study and aims to assess the performance of the novel algorithms implemented in Phase II.

The algorithms will be assessed according to the performance measures decided on in Phase II under the different experimental contexts.

2 RELATED WORK

2.1 SENSITIVITY ANALYSIS

The field of sensitivity analysis (SA) encompasses a range of methods for evaluating how the output of computational models are affected by changes in their inputs (Razavi et al., 2021). Such models are subject to different types of uncertainty from epistemic, to ontic, and Knightian or deep uncertainty, each of which may propagate through the model and effect the model output (Pruyt and Kwakkel, 2014). Specifically, sensitivity analysis is concerned with assessing how uncertainty in the output of a model can be apportioned to different sources of uncertainty in the model input (Saltelli et al., 2004). In decision support, SA has been identified as an essential step in the modelling process, since it affords a more complete understanding of how ever-present uncertainty affects the model output, and by extension, the fidelity of the model on which policy designs and decisions may be based (Razavi et al., 2021), (Saltelli et al., 2000).

In general, sensitivity analysis methods use parameter value sets which are defined *a priori* and for which the model is evaluated. A variety of sampling methods may be used to select the parameter values (Santner et al., 2003). Pseudo-random sampling methods such as MC and qMC are immune to the the curse of dimensionality and are therefore favoured for problems with high-dimensional input parameter spaces.

One of the most widespread SA methods is that of variance-based or Sobol sensitivity analysis, which apportions variance in the model output to the uncertainty in the model inputs (Sobol, 1993). Although popular, owing to its interpretability, and its capacity to deal with interaction effects between parameters, Sobol sensitivity analysis is computationally expensive, which serves as a barrier to its widespread adoption in non-computational fields. While pseudo-random methods do not suffer from the curse of dimensionality directly, they nevertheless converge slowly for models with high-dimensional parameter spaces (Gerstner and Griebel, 2003). For teams with no access for high-performance computing resources such a bottleneck serves to limit the size of the SA campaigns which can be undertaken.

This bottleneck may be side-stepped by using a meta-modelling or surrogate modelling approach, where the Sobol indices are computed from a meta-model with fewer model evaluations than with direct sampling methods (Sudret, 2008). Polynomial Chaos Expansions (PCE) are one such meta-modelling approach which have been used in a wide-range of domains for executing SA and UQ campaigns (Edeling et al., 2021), (Loukrezis et al., 2019).

2.2 POLYNOMIAL CHAOS EXPANSION

PCE expansions, first introduced by Wiener, can approximate any second-order random variable with a series of polynomials in centered, normalised random variables (Wiener, 1938) (Xiu and Karniadakis, 2002). With respect to simulation models, this is equivalent to projecting the model output joint probability density function onto a basis of orthogonal stochastic polynomials over the random inputs (Xiu, 2010). The main advantage of the method is its convergence speed when compared against the direct sampling approaches outlined above (Crestaux et al., 2009). The coefficients of the PCE are determined by requiring that the meta-model equal the actual model output at a certain number of points. As such, the model output can be well approximated using an order of magnitude fewer model evaluations than standard MC methods in some experimental contexts (Crestaux et al., 2009). A further advantage of PCE is that the Sobol indices may be deduced directly from the coefficients of the PCE, making it ideal for the sensitivity analysis of complex simulation models which would be costly to adapt for a secondary analysis (Edeling et al., 2021).

The construction of the meta-model from orthogonal polynomials allows for the determination of the expansion coefficients by taking the inner-product of the joint probability distribution with the polynomial corresponding to a given dimension. The inner-product may then be computed by integration to yield the expansion coefficients. Since the analytical integration of such expressions is non-trivial, the multivariate integrals are usually computed numerically using a quadrature rule such as nested 1D Gaussian quadrature. The use of PCEs for SA then depends on the evaluation of the following integral for each of the input parameters, k,

$$I_k \approx \int \Upsilon(\xi)\Psi_k(\xi)p(\xi)d\xi \tag{2.1}$$

where $\Upsilon(\xi)$ is the output polynomial, $\Psi_k(\xi)$ is the polynomial basis, and $p(\xi)$ is the joint distribution of the input uncertainties.

For a single parameter, an approximation of level l requires the function to be evaluated l times and for d parameters, each approximated to level l, requires l^d evaluations. In this respect PCEs suffer from the curse of dimensionality, since the number of function evaluations scales exponentially with the dimension of the input space, potentially limiting their use on models with more than around 10 input parameters (Crestaux et al., 2009). In this context, the curse of dimensionality is defined by the exponential scaling of the integration error that is made by a quadrature rule of level l with N_l points,

$$E_l = O(N^{\frac{-r}{d}})$$

where d is dimension and r is the regularity of the function (Gerstner and Griebel, 2003).

Whereas PCEs have been found to converge significantly faster than direct-sampling methods for $d < 20$, they begin to struggle with higher-dimensional problems (Crestaux et al., 2009). Therefore, where PCEs are considered as a potentially useful sensitivity analysis method for exploratory modelling purposes, especially where factor prioritisation and fixing are concerned, this represents a considerable limitation to their application.

The computational intensity of PCEs arises specifically from the form of the quadrature scheme used to approximate the integrals represented by equation 2.1. A 1D Gaussian quadrature scheme of level l requires l function evaluations. In approximating multi-dimensional

FULL CARTESIAN GRID VERSUS CLASSICAL SPARSE GRID[a]

Dimension d	Full Grid $\lvert V_4 \rvert$	Sparse Grid $\lvert V_4^S \rvert$
1	15	15
2	225	49
3	3,375	111
4	50,625	209
5	759,375	351
10	$5.77 \cdot 10^{11}$	2,001
20	$3.33 \cdot 10^{23}$	13,201
50	$6.38 \cdot 10^{58}$	182,001
100	>Googol	1,394,001

Figure 2.1: The above table well illustrates how the curse of dimensionality arises when applying Clenshaw Curtis quadrature to high-dimensional spaces. With a constant quadrature level of 4 for each dimension, the number of function evaluations explodes and quickly becomes prohibitive for even modestly dimensional spaces. The table shows values for hierarchical finite-element bases. The sparse grid approach samples at significantly fewer points and so dodges the curse.

integrals, where each dimensions is approximated by 1D quadrature schemes at the same order the number of function evaluations required scales with $\mathcal{O}(N^d)$, quickly making the cost of such computations unfeasible in situations where computing power or time is limited. The existence of this curse is the reason for ubiquity of Monte-Carlo (MC) methods in numerical integration applications. With MC methods, the function is evaluated at pseudo-randomly chosen points and the integrand is approximated as the average of the function values. Although MC methods do not suffer from the curse of dimensionality, their computation time scaling with the square-root of dimension, they are still too slow for problems with even modest numbers of dimensions (Gerstner and Griebel, 2003). Quasi-Monte Carlo methods (QMC) use low-discrepancy sequences to generate the evaluation points and as such converge significantly faster than pure MC methods. Nevertheless, QMC methods suffer a logarithmic dependence on dimension, limiting their use for high-dimensional problems.

The sparse-grid approach, pioneered by Smolyak, attempts to bypass the curse of dimensionality by taking advantage of the smoothness of the integrands (Smolyak, 1963). With this approach, the multi-variate integrand is evaluated using quadrature formulas which are constructed from tensor-products of 1D, or univariate, quadrature formulas. The quadrature level l for each dimensions is tracked through the use of a multi-index (Bungartz and Griebel, 2004). The sparse-grid approach results in a less fine-grained sampling of the input parameter space and is therefore less computationally intensive. An example of a sparse grid is shown in figure 2.2 alongside a full or *Cartesian* grid.

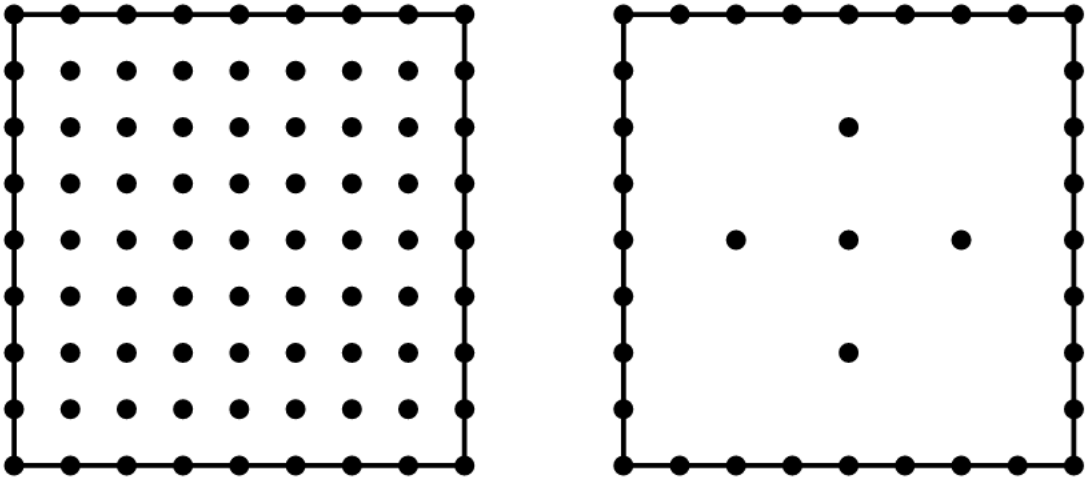


Figure 2.2: The above diagram well illustrates the difference between a full (*left*) and sparse grid (*right*) for $\mathbb{R}^2$. The full Cartesian grid is a product rule space with each dimension sampled at the same level of refinement. The sparse grid is a tensor-product space where only rules whose indices are on the unit simplex are considered.

2.3 ADAPTIVE SPARSE GRIDS

The 'full-grid' sampling plan described above, where each dimension is numerically integrated at the same level, implicitly assumes that each of the dimensions are equally important. With respect to sensitivity analysis, this is equivalent to assuming that each uncertain input has a significant influence on the output variance. In reality, such models usually exhibit a lower effective dimension, a result of the sparsity of effects principle (Box and Meyer, 1986). The use of sensitivity analysis for factor fixing is predicated on the principle that a relatively small proportion of the input variables will have a disproportionately large influence on the output variance (Saltelli et al., 2008).

If the relative importance of the various dimensions was known *a priori*, then function evaluations could be concentrated on those dimensions which confer the largest gain in approximation accuracy. This is achieved in practice by increasing the level of the cubature scheme in those dimensions which are determined to be of greater importance. *Griebel et al.* proposes one such scheme, a generalisation of Smolyak's sparse grid approach, where the quadrature order of important dimensions is adaptively increased, while less important dimensions are evaluated using lower level quadrature rules (Gerstner and Griebel, 2003).

In the scheme proposed by *Griebel et al.* the multi-index is first populated with zeros, indicating that each dimension is integrated with zeroth-order, or level-one, quadrature rules. A 'look-ahead' step is then performed during which a local error measure is calculated. The dimension which confers the largest decrease in approximation error has its index value increased by one, indicating a higher-order level of quadrature. In this way the dimensions to which the model output variance is most sensitive will be sampled in detail while unimportant dimensions will be sampled to a lower order degree. This dimension-adaptivity serves to significantly decrease the computational intensivity of such numerical integration problems.

Authors in the literature from a variety of fields have applied the adaptive sparse-grid approach of (Gerstner and Griebel, 2003) to solve a range of high-dimensional modelling prob-

lems. (Brumm and Scheidegger, 2017) utilised an adaptive-sparse grid approach to solve a dynamic stochastic model with over 100 dimensions. In a departure for the now standard 'look-ahead' algorithm of (Gerstner and Griebel, 2003), (Dwight et al., 2016) suggest using sparse grids to compute Sobol indices directly and subsequently refining the sampling for those input parameters with the largest Sobol indices. Their algorithm mitigates the computational cost of the look-ahead step by allowing the refinement of more than one dimension during each step. This 'grouping' approach was also implemented by (Bungartz and Dirnstorfer, 2003) albeit using a local error measure rather than Sobol indices. Their results suggest that around 10% of the candidate index values should be chosen for refinement for problems with less than 10 dimensions and 2-5% for significantly larger D. Dimension-adaptivity also finds use in the related method class of interpolation where it has been applied to power flow calculations (Tang et al., 2015).

2.4 SUPPLEMENTATION OF PCES WITH DIMENSION-ADAPTIVE SPARSE GRIDS

Given the aforementioned issues approximating high-dimensional probability density functions with PCEs, the use of sparse and adaptive grid schemes has been examined in the literature. One such project, the study of (Edeling et al., 2021) utilises PCEs with DAS for uncertainty quantification of the CovidSim epidemiological code. Their UQ campaign analysed the effect of 60 input parameters, and found a subset of 19 parameters to which the code was most sensitive. Uncertainty in these parameters was found to be magnified in the outputs by 300%. The study of (Edeling et al., 2021) is in the vein of traditional sensitivity analysis, as the authors sought to establish the extent to which the output of a large, complicated model was dependent on uncertainty in a small subset of its parameters($< 10\%$). Other groups have implemented the combined methodology for UQ in solid-rocket motor models as well as for the calculation of robustness metrics in airfoil optimisation problems (Fang et al., 2019) (Li et al., 2019).

3 BACKGROUND THEORY

3.1 SENSITIVITY ANALYSIS

The field of sensitivity analysis (SA) encompasses a range of methods for dealing with the presence of uncertainty in simulation models (Razavi et al., 2021). Such models are subject to different types of uncertainty which may propagate through the model and affect the model output (Pruyt and Kwakkel, 2014). Specifically, sensitivity analysis is concerned with assessing how uncertainty in the output of a model can be apportioned to different sources of uncertainty in the model input (Saltelli et al., 2004). Since complicated mathematical models have become an essential part of the modern scientific toolkit, it is necessary to understand how the presence of uncertainties in the model inputs affect the model outputs. Inputs of interest are referred to in the literature as 'factors', a term inherited from closely related field of *Design of Experiments* (Santner et al., 2003).

In the practice of systems modelling, SA aids the researcher in understanding a number of fundamental system properties as they are formalised in the model, such as the exploration of how different parameters, boundary conditions, and processes affect the output of a model (Razavi et al., 2021). For models with large numbers of input dimensions SA can also be used to identify less influential dimensions, which may then be removed or fixed to a certain value during subsequent analysis , a process known as model order reduction (Tarantola et al., 2007). Finally, in decision support, SA has been identified as an essential step in the modelling process since it affords a more complete understanding of how ever-present uncertainty affects the model output, and by extension, the fidelity of the model on which policy designs and decisions may be based (Razavi et al., 2021) (Saltelli et al., 2000). The importance of SA in such contexts is well exemplified by the work of (Edeling et al., 2021), who executed a SA campaign on Imperial University's CovidSIM, the UK's primary epidemiological model used to guide their response to the COVID19 pandemic of 2020 and 2021. *Edeling et al.* found that of the 940 free parameters which specified the structure of the model, uncertainty in a subset of 19 parameters were most influential, with uncertainty in these parameters magnified by 300% in the model output. SA methods exploit the so called 'sparsity of factors' principle, a heuristic which states that often it is only a small subset of factors which have a significant effect on a given model output (Box and Meyer, 1986).

There are two broad classes of sensitivity analysis methods, *local* and *global*. Whereas local sensitivity analysis is concerned with how a model's output is affected by small perturbations in a parameter's value around a certain fixed point, global methods vary the input factors simultaneously and sample each over its complete range. As such, global methods afford a more complete picture of the sensitivity landscape of a model. Within global sensitivity analysis there are a range of distinct methods, each of which fulfill the goals of SA in a different way. In the sequel we focus on the *variance-based* approach to SA, which aims to decompose the variance of a model's output distribution and attribute it to variance in the model's inputs.

3.1.1 Variance-based sensitivity analysis

Variance based sensitivity analysis, or Sobol sensitivity analysis, after its progenitor Ilya Sobol, decomposes a model's output variance and attributes it to variance, or uncertainty, in the model's inputs (Sobol, 1993). As an example, consider a model with two uncertain inputs and a single output. It might be the case that, following a SA campaign, we find that 60% of the model's output variance is caused by the variance of the first input, 20% by the variance of the second, while the remaining 20% is due to interactions between the two. According to Sobol's method, these fractional variances can be directly interpreted as measures of sensitivity, describing both the *main*, and *higher-order* effects a given factor has on the output of a model. In order to illustrate the method more completely, and to lay the theoretical foundation for subsequent sections we will now derive Sobol's decomposition as laid out by (Saltelli et al., 2008). In the following, the terms *input variable* and *factor* are used interchangeably to refer to the same thing.

Consider a generic model $\mathbf{f}$ of the form,

$$\mathbb{Y} = \mathbf{f}(\mathbf{x}) \tag{3.1}$$

where $\mathbf{x} = \{x_1, ..., x_d\}^T \in \mathbb{R}^d$ is the finite vector of deterministic input variables and $\mathbb{Y} = \{y_1, ...y_q\}^T \in R^q$ is the vector of quantities of interest which the model provides, referred to in the sequel as the *model-response*. For the purposes of this derivation we will assume that the model output is uni-variate, corresponding to a single output quantity of interest. The model then maps $\mathbb{R}^d \longrightarrow \mathbb{R}$.

We now account for parametric uncertainty in f by assuming that the input parameters are stochastic variables. The model input then becomes a random vector, $\boldsymbol{\xi} = \{\xi_1, ...\xi_d\}^T \in \mathbb{R}^d$. Since the model output depends on these stochastic input variables, the model output $\mathbb{Y}(\boldsymbol{\xi})$ may itself may be treated as a stochastic variable of unknown form and dimension q. For the purposes of this derivation we will assume that the model output is uni-variate, corresponding to a single output quantity of interest. The extent of the variance of $\mathbb{Y}(\boldsymbol{\xi})$ is a measure of the uncertainty in the model output caused by uncertainty in the model inputs.

For the sake of readability, we denote the now random input variables by x in place of the usual ξ and request that the reader keep in mind that the variables contained in the vector $\mathbf{x}$ are functionals, rather than real-valued numbers. Each input variable has a non-zero range of variation, corresponding to the assumed parametric uncertainty surrounding that input. We now ask what would happen to the uncertainty in $\mathbb{Y}(\mathbf{x})$ if we were to fix a factor, x_i at a particular value, given by x_i^*. Let $V_{x\sim i}(\mathbb{Y}|x_i = x_i^*)$ denote the resulting variance of $\mathbb{Y}$ taken over $x_{\sim i}$, all factors except x_i. $V_{x\sim i}$ is termed the conditional variance, as it is conditional on the fixing of uncertain parameter x_i to a specific value x_i^*. It is tempting to think that this quantity might serve as an appropriate measure to quantify the sensitivity of our model output to the variable x_i, since the smaller $V_{x\sim i}$ the greater the presumed contribution of varying x_i alone to the total variance of the output; however, such a measure would depend on x_i^* only, the specific value to which the parameter was fixed. In order to determine a metric which accounts for all possible values over the range of x_i we take the average of the conditional variance over the same range. This may be expressed as,

$$E_{x_i}\left[V_{x\sim i}\left(\mathbb{Y}|x_i\right)\right] \tag{3.2}$$

where E_{x_i} is the empirical mean, or expectation value, of $V_{x\sim i}$ over the range of x_i.

We may apply the law of total variance to yield,

$$E_{x_i}\left[V_{x\sim i}\left(\mathbb{Y}|x_i\right)\right] + V_{x_i}\left(E_{x\sim i}\left[\mathbb{Y}|x_i\right]\right) = V\left(\mathbb{Y}\right) \tag{3.3}$$

The above expression shows that a small $E_{x_i}\left[V_{x\sim i}\left(\mathbb{Y}|x_i\right)\right]$ or a large $V_{x_i}\left(E_{x\sim i}\left[\mathbb{Y}|x_i\right]\right)$ will imply that x_i is an influential factor. The conditional variance $V_{x_i}\left(E_{x\sim i}\left[\mathbb{Y}|x_i\right]\right)$ is then the first-order effect of variation in x_i on the variance of the model output, $\mathbb{Y}$. We may then define a sensitivity measure,

$$S_{x_i} = \frac{V_{x_i}\left(E_{x\sim i}\left[\mathbb{Y}|x_i\right]\right)}{V(\mathbb{Y})} \tag{3.4}$$

which is known as the first-order Sobol sensitivity index of variable x_i on $\mathbb{Y}$, with higher values indicating a more influential variable. Since it follows from the equation above that $V_{x_i}\left(E_{x\sim i}\left(\mathbb{Y}|x_i\right)\right) \leq V(\mathbb{Y})$, S_{x_i} takes a value between 0 and 1. In the rest of the document we use S_1 to denote the first-order Sobol sensitivity index of a given variable.

Computing S_1 for each model variable will allow us to rank each variable by its main, or first-order, effect on the variance of the output. That is, the expected change in output variance by fixing each valuable alone, averaged over their ranges. In order to derive an expression for second-order effects, we may ask, what portion of the output variance is conditional on the values of two variables taken together? Replicating out expression for $S_1(x_i)$ we have,

$$\frac{V(E[Y\,|x_i, x_j])}{V(\mathbb{Y})} \tag{3.5}$$

for $i \neq j$. Here we have dropped the subscripts from both the variance and expectation operators since it should be clear that the range over which we apply the inner operator is the same as that over which we apply the outer two operators, ie we write $V(\cdots)$ instead of $V_{x_i x_j}$. If we calculated the expression in equation 3.5 for two factors in a given non-additive model we would find that the following holds,

$$\frac{V(E[Y\,|x_i, x_j])}{V(\mathbb{Y})} > S_{x_i} + S_{x_j} \tag{3.6}$$

The above inequality implies that the second-order effect cannot be written as a superposition of the separate main order effects of x_i and x_j. To account for discrepancy we introduce a term representing this higher-order effect.

$$V(E[\mathbb{Y}|x_i, x_j]) = V_i + V_j + V_{ij} \tag{3.7}$$

where

$$V_i = V(\mathbb{E}[\mathbb{Y}|x_i]) \tag{3.8}$$

and,

$$V_j = V(\mathbb{E}[\mathbb{Y}|x_j]) \tag{3.9}$$

Dividing both sides of equation 3.7 by the variance of the model output $V(\mathbb{Y})$ and rearranging yields an expression for the second-order Sobol sensitivity index of x_i and x_j on $\mathbb{Y}$. Just

as it is possible to calculate the first-order Sobol indices for each variable in a model, so too is it possible to determine the second-order index of any two variables. Naturally, the only proviso for the presence of a second-order effect is that the model output is influenced by at least two factors. In general, for a model with k-factors there are,

$$\binom{k}{2} \tag{3.10}$$

second-order interactions for which we can calculate a second-order Sobol sensitivity index. The same process of conditional variance can be followed to derive expressions for successively higher-order interaction effects. As the order of the interaction is increased, we expect the calculated terms to account for a smaller amount of the output variance, as suggested by the sparsity of effects principle (Box and Meyer, 1986). This fact, of generally diminishing 'returns' with the addition of higher-order terms, as well as each index being normalised to the total variance, is naturally suggestive of approximation by expansion. In calculating each Sobol sensitivity index for each factor, or group of factors, we are attempting to fully account for the variance of the output by apportioning it to variance in the model's input factors. If we followed the logic of the above expressions and attempt to sum each the Sobol indices of each order for each model factor we would have the following,

$$\sum_i S_i + \sum_i \sum_{j>i} S_{ij} + \sum_i \sum_{j>i} \sum l > i S_{ijl} + \cdots + S_{123\ldots k} = 1 \tag{3.11}$$

for a model with k-factors.

We address this similarity more completely in the next section where we examine how a given output distribution can be fully decomposed into partial variances attributable to different factors.

3.1.2 Sobol's decomposition

Taking our original stochastic model as a starting point,

$$\mathbb{Y} = \mathbf{f}(\mathbf{x}) \tag{3.12}$$

where the model output $\mathbb{Y}(x)$ is a scalar, and each of the input factors are independent random variables with known probability distributions which characterise our uncertainty about their value.

Sobol's innovation was to formulate a square integrable function f defined over the domain of the k-dimensional unit hypercube, Ω^k,

$$\Omega^k = (X|0 \leq x_i \leq 1; i = 1, \cdots, k) \tag{3.13}$$

Sobol proposes decomposing the function f into a finite series of successively higher-dimension terms where each of the constitutive function are also square-integrable over the domain of Ω^k, and each term is only a function of the factors which appear in its index, for example,

$f_i = f(x_i)$ and $f_{ij} = f(x_i, x_j)$. The expansion may be written as,

$$f = f_0 + \sum_i f_i + \sum_i \sum_{j>i} f_{ij} + \cdots + f_{12\cdots k} \tag{3.14}$$

The decomposition has 2^k terms, the first of which, f_0, is a constant, followed by k first-order terms, $\binom{k}{2}$ second-order terms and $\binom{k}{n}$ terms of order n. Such an expansion is termed a high-dimensional model representation and is non-unique. If we attempt to represent the output of a model $\mathbb{Y}(\mathbf{x})$ with the above expansion, the link with our previously derived expressions becomes clear.

The following orthogonality relation holds for any two summands in the decomposition,

$$\int_{\Omega^k} f_{i_1\cdots i_x}(x_i \cdots) f_{j_1\cdots j_x}(x_j \cdots) = 0 \tag{3.15}$$

The mutual orthogonality of the expansion terms allows us to calculate their values using the conditional variances of the model output. For example, applying the standard expression for the expectation value of a continuous variable we have,

$$E[\mathbb{Y}(x)] = f_o \int_{-1}^{1} \mathbb{Y}(x) dx = f_0 \tag{3.16}$$

Since, by construction, each term in the expansion is orthogonal to the model output, the rest of the terms in the above computation reduce to zero, leaving us with an expression for f_0,

$$f_0 = E[\mathbb{Y}(\mathbf{x})] \tag{3.17}$$

which may be empirically determined.

Similarly, for the higher order terms we have,

$$f_i = E[\mathbb{Y}|x_i] - E[\mathbb{Y}(\mathbf{x})] \tag{3.18}$$

and

$$f_{ij} = E[\mathbb{Y}|x_i, x_j] - f_i - f_j - E[\mathbb{Y}(\mathbf{x})] \tag{3.19}$$

Representing each term in the decomposition as the conditional expectation of the model response yields the expression from equation 3.14 and we find that the variances of each term in the decomposition are the sensitivity measures which we desire.

$$\sum_i S_i + \sum_i \sum_{j>i} S_{ij} + \sum_i \sum_{j>i} \sum_{l>i} S_{ijl} + \cdots + S_{123,\cdots,k} = 1 \tag{3.20}$$

3.2 POLYNOMIAL CHAOS EXPANSIONS

3.2.1 Mathematical basis

In essence, polynomial chaos expansions approximate any second-order random variable as an infinite series of polynomials in normalised Gaussian variables. Given a second-order random variable $\mathbb{Y}$, and an infinite set of orthogonal random variables $\{\xi\}_1^\infty$ we may write,

$$\mathcal{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \alpha_k \Psi_k(\boldsymbol{\xi}), \qquad \boldsymbol{\xi} = \{\xi_1, \xi_2, ...\} \tag{3.21}$$

where α_k are the expansion coefficients and $\Psi_k(\cdot)$ are the orthogonal basis polynomials in each of the random variables, which are held in the random vector, $\boldsymbol{\xi}$.

More rigorously, we define a space Ω of random events and Θ, the space of functionals which map the elements $\omega \in \Omega$ to a value in $\mathbb{R}$. A random variable is therefore the function $\theta : \omega \in \Omega \mapsto \mathbb{R}$. Let $\{\xi_i\}_1^\infty$ be an infinite but countable set of independent normalised Gaussian variables. The PC expansion of a second-order random variable is then,

$$\begin{aligned} \mathcal{Y}(\boldsymbol{\xi}(\omega)) = a_0\Gamma_0 + \sum_{i_1=1}^{\infty} a_{i_1}\Gamma_1(\xi_{i_1}(\omega) \\ + \sum_{i_1}^{\infty}\sum_{i_2=1}^{i_1} a_{i_1 i_2}\Gamma_2(\xi_{i_1}(\omega), \xi_{i_2}(\omega)) \\ + \sum_{i_1}^{\infty}\sum_{i_2=1}^{i_1}\sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3}\Gamma_3(\xi_{i_1}(\omega), \xi_{i_2}(\omega), \xi_{i_3}(\omega)) + \cdots \end{aligned} \tag{3.22}$$

where Γ_p is the set of all polynomials of order p in the random variables $\{\xi_i\}_1^\infty$ and is called the polynomial chaos of order p. The above expression demonstrates that the PC of order p is the set of all polynomials of degree p in all possible combinations of the random variables in $\{\xi_i\}_1^\infty$. Grouping the terms in the expansion according to their polynomial families Ψ_k where there is a one-to-one correspondence between $\Gamma(\cdots)$ and $\Psi(\cdots)$, returns us to the original expression,

$$\mathcal{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \alpha_k \Psi_k(\boldsymbol{\xi}), \qquad \boldsymbol{\xi} = \{\xi_1, \xi_2, ...\} \tag{3.23}$$

In equation 3.23 we have dropped the symbol ω from out notation and will continue with this notation for the rest of the document. The inclusion of the symbol ω is intended to rigorously account for the character of the random variable ξ as a *function* defined over a space of random events. With this point made we continue with the derivation.

Since the random variables in $\boldsymbol{\xi}$ were defined to have Gaussian distribution functions, the expansion given above is a homogeneous chaos expansion as originally defined by (Wiener, 1938). As such, it employs Hermite polynomials in terms of the Gaussian random variables as basis functions for the expansion (Xiu and Karniadakis, 2002), where the general expression for Hermite polynomials of order n in all random variables is,

$$H_n(\xi_{i_1}, \cdots, \xi_{i_n}) = e^{\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}}(-1)^n \frac{\partial^n}{\xi_{i_1}\cdots\xi_{i_n}} e^{-\frac{1}{2}\boldsymbol{\xi}^T\boldsymbol{\xi}} \tag{3.24}$$

While the Hermite-chaos expansion as defined above is effective in approximating stochastic variables using Gaussian inputs, its convergence suffers with non-Gaussian inputs. Where the order of the expansion is to be truncated for application in computational studies this will adversely affect the ability of finite PCEs to approximate variables of interest.

In order to deal with more general random inputs (Xiu and Karniadakis, 2002) introduced the Wiener-Askey polynomial chaos, where the approximating polynomials are not restricted

	Random variables ξ	Wiener-Askey Chaos $\{\Psi(\xi)\}$	Support
Continuous	Gaussian	Hermite-chaos	$(-\inf, \inf)$
	gamma	Laguerre-chaos	$[0, \inf]$
	beta	Jacobi-chaos	$[a, b]$
	Uniform	Legendre-chaos	$[a, b]$
Discrete	Poisson	Charlier-chaos	$\{0, 1, 2, \cdots\}$
	binomial	Krawtchouk-chaos	$\{0, 1, 2, \cdots, N\}$
	negative binomial	Meixner-chaos	$\{0, 1, 2, \cdots\}$
	hypergeometric	Hahn-chaos	$\{0, 1, 2, \cdots, N\}$

Table 3.1: The correspondence between the input variable type and the corresponding polynomial chaos. The supports for each polynomial family are also given. Since each of the polynomial families form a complete Hilbertian basis they can be expected to converge to any L_2 functional in the L_2 sense for their respective variable types as a result of the generalised Cameron-Martin theorem. Of interest to sensitivity analysis applications is the uniform distribution which assumes no prior knowledge of the form of the parametric uncertainty and is well approximated by *Legendre* polynomials. Proper table here

to Hermite polynomials. Instead, they can be any type of orthogonal polynomial from the Askey scheme where input random variable types are matched with the polynomial family that best provides for their convergence in a chaos expansion (Koekoek and Swarttouw, 1996) .

3.2.2 Computation

For practical calculations a finite number d, of random variables are used, which leads to a *finite dimensional* PCE,

$$\mathbb{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \alpha_k \Psi_k(\boldsymbol{\xi}), \qquad \boldsymbol{\xi} = \{\xi_1, ..., \xi_d\} \in \mathbb{R}^d \tag{3.25}$$

This is not a limitation since in practical contexts we are usually only concerned with a finite number of random input variables corresponding to parametric uncertainty in a model's inputs. Note that at this stage, although the expansion is *finite-dimensional*, it still contains an infinite number of terms owing to the infinite series of basis polynomials in each of the d variables.

In order to implement PCEs practically the expansion may be truncated to a certain order, p. A PCE truncated to order p contains only those multi-variate polynomials in the complete multi-variate basis with combined order $\leq p$, where all polynomials of order p are termed the p^{th} polynomial chaos and are contained in Ψ_p. We therefore have the following expression for a truncated PCE,

$$\mathbb{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{P} \alpha_k \Psi_k(\boldsymbol{\xi}) \tag{3.26}$$

where the number of terms in the PCE, and consequently the number of basis vectors P is given by,

$$P + 1 = \binom{d + p}{p} \tag{3.27}$$

where d is the dimension of the input space and p is the truncation order of the PCE.

3.2.3 Surrogate modelling

We now consider the case of the above technique when applied to the kind of finite model which might be encountered in the field of sensitivity analysis.

Consider first a physical model $\mathbb{M}$ with the deterministic mapping given by,

$$\mathbb{Y} = \mathbb{M}(\mathbf{x}) \tag{3.28}$$

where $\mathbf{x} = \{x_1, ..., x_d\}^T \in \mathbb{R}^d$ is the finite vector of deterministic input variables and $\mathbb{Y} = \{y_1, ...y_q\}^T \in R^q$ is the vector of quantities of interest which the model provides, referred to in the sequel as the *model-response*.

We now account for parametric uncertainty in $\mathbb{M}$ by assuming that the input parameters are stochastic variables. The model input then becomes a random vector, $\boldsymbol{\xi} = \{\xi_1, ...\xi_d\}^T \in \mathbb{R}^d$ corresponding to an event $\omega \in \Omega$. At this stage, given the redefinition of the input vector as a set of random variables, our once deterministic model may now be considered equivalent to a stochastic model, where internal parameters and inputs take values drawn from a distribution. Although at its core the model defined above is completely deterministic, we have introduced random variables in order to account for our uncertainty about the true values of the deterministic input parameters.

The random variables in ¸ are assumed to be independent, which allows for the following definition of the input joint probability distribution,

$$P_{\tilde{\xi}}(\boldsymbol{\xi}) = \prod_{j=1}^{d} P_{\tilde{\xi}_j}(\xi_j) \tag{3.29}$$

The model output is also assumed to have finite variance such that,

$$E[\mathbb{Y}^2] = \int_{\mathcal{D}_{\tilde{\xi}}} \mathbb{M}^2(\xi) P_{\tilde{\xi}}(\xi) d\xi < \infty \tag{3.30}$$

Since the model takes random variables as input, the model output $\mathbb{Y}(\boldsymbol{\xi})$ may itself be treated as a second-order random variable of unknown form and of dimension q.

The techniques outlined in the previous section allow us to recast the unknown model output as an infinite sum of polynomials in the uncertain inputs. This is equivalent to projecting the output probability density function onto a basis of orthogonal polynomials in the random inputs (Xiu, 2010). For a finite set of d random inputs we may write the model output as,

$$\mathbb{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{\infty} \alpha_k \Psi_k(\hat{\xi}), \qquad \boldsymbol{\xi} = \{\xi_1, ..., \xi_d\} \tag{3.31}$$

Truncating to a fixed order p the infinite series reduces to a finite sum over $P + 1 = \binom{d+p}{p}$ terms, namely,

$$\mathbb{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{P} \alpha_k \Psi_k(\hat{\xi}), \qquad \boldsymbol{\xi} = \{\xi_1, ..., \xi_d\} \tag{3.32}$$

It is important to note at this point that the polynomials, Ψ_k which form the basis of the expansion are multi-variate, and are constructed by tensorisation of uni-variate polynomials, each only depending on a single random variable ξ_j. We will now to proceed with an example to illustrate how such a polynomial basis may be formed.

Consider a model which takes as input two random Gaussian variables with *zero mean* and *unit variance*, contained in a random vector $\boldsymbol{\xi} \in \mathbb{R}^2$. The model produces an output $\mathcal{Y}(\_)$ which is dependent on the two inputs. We assume for the case of illustration that the model output is uni-dimensional. In line with the Wiener-Askey scheme we will use Hermite polynomials in ξ_1 and ξ_2 to construct the basis vectors over which to expand $\mathcal{Y}(\boldsymbol{\xi})$. For the sake of brevity, we compute only the first three Hermite polynomials in each variable. These are shown in Table 3.2.

	$\Xi_1(\xi_1)$	$\Xi_2(\xi_2)$
$He_0(\xi_j)$	1	1
$He_1(\xi_j)$	ξ_1	ξ_2
$He_2(\xi_j)$	$\xi_1^2 - 1$	$\xi_1^2 - 1$
$\vdots$	$\vdots$	$\vdots$
$He_n(\xi_j)$	$He_n(\xi_1)$	$He_n(\xi_2)$

Table 3.2: Table showing the first three Hermite polynomials for the two input variables ξ_1 and ξ_2.

The polynomials in each of the columns form a complete uni-variate, ortho-normal basis in each of the uncertain model inputs. To construct the full set of multi-variate basis vectors we take the tensor product of both sets, where Ξ_1 and Ξ_2 are the infinite sets of Hermite polynomials in ξ_1 and ξ_2 respectively. The basis vectors arising from tensorising the first three terms in each set are,

$$\Xi_1 \bigotimes \Xi_2 = \left[\xi_1, \quad \xi_2, \quad \xi_1\xi_2, \quad \xi_1^2 - 1, \quad \xi_2^2 - 1, \quad \xi_1\xi_2^2 - \xi_1, \quad \xi_2\xi_1^2, -\xi_2 \quad \xi_2^2\xi_1^2 - \xi_1^2 - \xi_2^2 + 1, \quad \cdots \right] \tag{3.33}$$

If we truncate the set of basis vectors to order three, constructed by tensorising the infinite series of Hermite polynomials in each variable, we are left with a set of 10 basis vectors. Grouping these terms according to their combined order yields the polynomials, Ψ_p, which feature in equation 3.23. To arrive at the simplified expression given in eq 3.23, we no longer sum over the polynomials in each of the polynomial chaoses of different orders but rather sum over the group as a whole in a single sum with a single index variable, k.

We may now write the complete polynomial chaos expansion of $\mathcal{Y}$ in terms of the basis polynomials in ξ_1 and ξ_2,

$$\begin{aligned} \mathcal{Y}(\boldsymbol{\xi}) = \sum^{P} \beta_k \Psi_k(\hat{\xi}) = \alpha_0 + \alpha_1\xi_1 + \alpha_2\xi_2 + \alpha_3\xi_1\xi_2 + \alpha_4(\xi_1^2 - 1) + \\ \alpha_5(\xi_2^2 - 1) + \alpha_6(\xi_1\xi_2^2 - \xi_1) + \alpha_7(\xi_2\xi_1^2 - \xi_2) + \\ \alpha_8(\xi_2^2\xi_1^2 - \xi_1^2 - \xi_2^2 + 1) + \alpha_9(\xi_1^3 - 3\xi_1) + \alpha_{10}((\xi_2^3 - 3\xi_2)) \end{aligned} \tag{3.34}$$

Since constructing the multi-variate basis for a finite set of inputs is trivial, the task of approximating a random variable in a PCE reduces to computing each of the P expansion coefficients. Once the coefficients have been determined the polynomial expansion constitutes a surrogate model of $\mathbb{M}(\boldsymbol{\xi})$ and can be used to approximate the output of M without requiring potentially

j	p, Order of the Homogeneous Chaos	j^{th} Polynomial Chaos Ψ_j
0	$p = 0$	1
1	$p = 1$	ξ_1
2		ξ_1
3	$p = 2$	$\xi_1^2 - 1$
4		$\xi_1\xi_2$
5		$\xi_2^2 - 1$
6	$p = 3$	$\xi_1^3 - 3\xi_1$
7		$\xi_1^2\xi_2 - \xi_2$
8		$\xi_1\xi_2^2 - \xi_1$
9		$\xi_2^3 - 3\xi_2$

Table 3.3: Table showing the relationship between the various polynomial chaoses, order p, and index j for two Gaussian variables, ξ_1 and ξ_1

expensive model evaluations.

There are a variety of methods by which the coefficients of a polynomial chaos expansion may be determined. They may be divided into two main classes based on whether they require alteration to the model code or whether they require model realisations only. Intrusive methods usually require that some alteration be made to the numerical code in order to determine the expansion coefficients. The Galerkin projection is one such method which formulates the problem as a set of P+1 coupled equations which may then be solved to yield the coefficients. While such an intrusive approach may be readily achieved with a simple model, for applications with relatively complex models where ex-post alteration may be time-consuming and difficult to achieve, so called non-intrusive methods are preferred. Non-intrusive methods are those which require model evaluations only and do not require *ex-post* alteration of the model code. Examples of non-intrusive methods include Least Square Regression and Non-Intrusive Spectral Projection (NISP).

3.2.4 PCEs and sensitivity analysis

The astute reader will already have noticed the similarity in form between Sobol's decomposition for general square-integrable functions, $f(\boldsymbol{\xi})$, and our surrogate model approximation $\mathcal{Y}(\boldsymbol{\xi})$. It is this isomorphism, first exploited by (Sudret, 2008), which allows for Sobol's sensitivity indices to be trivially determined once we have a sufficiently accurate PCE. We begin the derivation with our expression for a PCE, truncated to order p, and containing $P + 1$ terms,

$$\mathcal{Y}(\boldsymbol{\xi}) = \sum_{k=0}^{P} \alpha_k \Psi_k(\boldsymbol{\xi}) \qquad P + 1 = \frac{(p+d)!}{p!d!} \tag{3.35}$$

We may expand the above as follows,

$$\begin{aligned}\mathbb{Y}(\boldsymbol{\xi}(\omega)) = a_0\Gamma_0 &+ \sum_{i_1=1}^{\infty} a_{i_1}\Gamma_1(\xi_{i_1}(\omega) \\ &+ \sum_{i_1}^{\infty}\sum_{i_2=1}^{i_1} a_{i_1 i_2}\Gamma_2(\xi_{i_1}(\omega),\xi_{i_2}(\omega)) \\ &+ \sum_{i_1}^{\infty}\sum_{i_2=1}^{i_1}\sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3}\Gamma_3(\xi_{i_1}(\omega),\xi_{i_2}(\omega),\xi_{i_3}(\omega)) + \cdots\end{aligned} \tag{3.36}$$

where we have grouped the terms in the expansion according to which factors/random variables they depend on, and therefore revert to the gamma notation of Section 3.2.1, recalling that there is a one-to-one correspondence between $\Gamma(\cdots)$ and $\Psi(\cdots)$.

The above expression is then equivalent to the Sobol decomposition of our PCE approximation $\mathbb{Y}(\boldsymbol{\xi})$ and we may write,

$$f_{i_1,\cdots,i_d}(\xi_{i_1},\cdots,\xi_{i_d}) = \sum_{i=1}^{P} \alpha_i \Psi_i(\xi_{i_1},\cdots,\xi_{i_d}) \tag{3.37}$$

Since there is a one-to-one correspondence between the terms in the truncated PCE and the Sobol decomposition of $\mathbb{Y}$ it is trivial to compute Sobol's sensitivity indices for each of the uncertain variables $\xi_i \in \boldsymbol{\xi}$. We compute the total variance as,

$$V(\mathbb{Y}) \approx \sum_{k=1}^{P} \alpha_k^2 \langle \Psi_k, \Psi_k \rangle = \sum_{k=1}^{P} \alpha_k^2 \tag{3.38}$$

with the final simplification arising from the mutual orthogonality of the terms in the PCE. The conditional variance of a term $u(x_{i_1},\cdots,x_{i_d})$ in the Sobol decomposition is,

$$V(E[\mathbb{Y}|\xi_u) \approx \sum_{k\in k_u} \alpha_k^2 \langle \Psi_k, \Psi_k \rangle \tag{3.39}$$

where k_u is the set which indexes all terms in the PCE which depend on a subset of the uncertain variables $\xi_u \subseteq \{\xi\}$. Finally, each of the Sobol indices may be determined in the usual way, by dividing the conditional variances by the complete variance,

$$S_u = \frac{\sum_{k\in k_u} \alpha_k^2 \langle \Psi_k, \Psi_k \rangle}{\sum_{k=1}^{P} \alpha_k^2} \tag{3.40}$$

Although the above expressions are intimidating, and rely on intricate indexing, the process is entirely intuitive. Starting with main order effects, to calculate the first-order Sobol sensitivity index of a given variable, say ξ_1, we find each term in the PCE which depends only on ξ_1 and take the squared sum of their coefficients. Dividing by the total variance, which we calculate as the squared sum of every coefficient in the expansion, yields the first-order Sobol index for ξ_1. The process can then be repeated for each of the other variables.

The process is analogous for higher-order Sobol indices, but instead of searching for terms which depend only on a certain variable, we accept interaction terms of a certain order. For example, calculating the second-order Sobol index for ξ_1 we find each term which depends on both ξ_1 and any of the other variables, and whose combined order is equal to two. Repeating

the process of square summing the coefficients, and dividing by the total variance, yields the second-order Sobol index of ξ_1.

The total-order index of a given variable is more straight forward, we simply square sum the coefficients of each term which contains our target variable as a factor and, again, divide by the total variance.

The above process is implemented as a python script which can be found at JeffreyDillonLyons/curse_breaking.git.

In the next section we examine the use of numerical integration to directly solve for each of the expansion coefficients. Once the coefficients have been determined then the sobol indices follow directly.

3.3 NUMERICAL INTEGRATION AND QUADRATURE

As referenced in Chapter 2, the task of expanding a random variable as a series of orthogonal polynomials reduces to determining the expansion coefficients for each of the basis vectors. Given the mutual orthogonality of the basis vectors, and their orthogonality with the output random variable, we may directly extract the expansion coefficients by taking the inner-product of the basis vectors with the random output,

$$\beta_k = \frac{\langle \Upsilon(\boldsymbol{\xi}), \Psi_k(\boldsymbol{\xi}) \rangle}{\langle \Psi_k, \Psi_k \rangle}, \qquad \forall k \tag{3.41}$$

The evaluation of the numerator in the above expression then reduces to the evaluation of $P+1$, d-dimensional integrals, one for each term in the expansion..

$$I_k \equiv \int_{\Omega^d} \Upsilon(\boldsymbol{\xi}) \Psi_k(\boldsymbol{\xi}) d\boldsymbol{\xi} \tag{3.42}$$

Since in practical applications the above integrals cannot be solved analytically, they must instead be solved using numerical methods such as quadrature rules. In general, quadrature rules attempt to approximate a given integral as a finite, weighted sum of function values at a number of specified integration points, or nodes, within the domain of integration. We may write an *n-point* quadrature rule approximation of the integral above as,

$$I_k \approx \sum_{i=1}^{n} \Upsilon(\boldsymbol{\xi}^{(i)}) \Psi_k(\boldsymbol{\xi}^{(i)}) w^{(i)} \tag{3.43}$$

Where $\Upsilon(\boldsymbol{\xi}^{(i)})$ is a vector of model evaluations at the n quadrature nodes, or abscissas, and $w^{(i)}$ are the weights. The more quadrature nodes that are taken, the better the quadrature rule approximates the integral and thus the better the approximation of the expansion coefficients. Here we a brief note on nomenclature regarding numerical integration and quadrature rules. The terms *node*, *abscissa* and *point* all refer to the same mathematical object, a location in the uncertainty space for which the function, or model, was be evaluated. A node in R^n is represented by an *n-tuple* with an entry for each dimension of the integrand, or, in the case of applying such methods to models, a value for each uncertain parameter. The terms order and level may be used interchangeably but refer to slightly different concepts in describing a quadrature rule. A quadrature rule of order 0 is the lowest-order quadrature rule possible and always produces a single node. The level of a quadrature rule is simply one more than

Quadrature grid level, l	Number of nodes, n	Nodes,x_i	Weights,w_i
1	1	0	2
2	2	$\pm\sqrt{\frac{1}{3}}$	1
3	3	0	$\frac{8}{9}$
		$\pm\sqrt{\frac{3}{5}}$	$\pm 5_{\overline{9}}$

Table 3.4: Table of nodes and weights for low-order univariate Gaussian quadrature rules

its order, therefore the lowest level quadrature rule is '1'. 'l' is a useful index since, in the case of non-nested Gaussian quadrature rules, the level l also denotes the number of nodes produced by that rule.

In order to build the theoretical foundation which is required for sparse grid methods, as well as examine the features of adaptive sparse grid methods as they pertain to PCEs, we start by considering one-dimensional Gaussian quadrature.

Uni-variate quadrature

Uni-variate quadrature rules are used to integrate function which vary in one dimension only. Gaussian quadrature rules are commonly used to integrate polynomial functions since an *n-point* Gaussian quadrature rule will integrate polynomials of degree $2n-1$ or less exactly for a suitable choice of integration points x_i, and weights w_i, for $i \in [1, 2, ..., n]$. The most common domain of integration for such a rule is [1,-1] so we may write an *n-point* Gaussian quadrature rule for an arbitrary polynomial function as,

$$\int_{-1}^{1} f(x)dx \approx \sum_{i=1}^{n} w_i f(x_i) \tag{3.44}$$

where $f(x_i)$ are the function evaluations at the nodes, x_i, and w_i are the weights. It can be shown that the nodes, x_i, for which the function is evaluated are the roots of the Legendre polynomials over the interval [1,-1], where higher-order(more points) integration rules correspond to the roots of higher-order Legendre polynomials, $P_n(x)$. If the n^{th} polynomial is normalised to give $P_n(1) = 1$ then the integration point x_i is the i^{th} root of P_n. The weights are then given by the formula,

$$w_i = \frac{2}{(1-x_i^2)[P_n'(x_i)]^2} \tag{3.45}$$

where $P^{'}$ is the first derivative of the i^{th} Legendre polynomial. Because of the relationship between the Legendre polynomials and the weights, a quadrature rule constructed as in equation 3.45 is often termed a *Gauss-Legendre* rule. The nodes and weights of an *n-point* Gaussian quadrature rule can then be defined *a priori*. A progression of low-order quadrature rules is given in Table 3.4. For an integral over an arbitrary interval $[a, b]$ a change of interval must be made to an integral over $[1, -1]$ before Gaussian quadrature can be applied.

We may illustrate the Gaussian integration process by considering a polynomial function $f(x)$ which we seek to integrate numerically. For the sake of illustration we choose a function for which the integral over [1,-1] can be also determined analytically.

Grey box 1.1

Consider the function,

$$f(x) = x^2 - x$$

a uni-variate, second-order polynomial. Analytical integration over [1,-1] yields,

$$\int_{-1}^{1} f(x) = \int_{-1}^{1} x^2 - x \; dx = \frac{2}{3}$$

By definition, the integral above is approximated exactly by a $2n - 1 = 3$ point quadrature rule, where n is the order of the polynomial to be integrated. We may therefore write,

$$\int_{-1}^{1} f(x) = \sum_{i=1}^{n} w_i f(x_i) = \int_{-1}^{1} x^2 - x \; dx = w_1 f(x_1) + w_2 f(x_2) + w_3 f(x_3)$$

From figure 3.4 we have the integration points and weights,

$$x_i = [0, \pm\sqrt{\frac{3}{5}}], \quad w_i = [\frac{8}{9}, \pm\frac{5}{9}]$$

Evaluating $f(x)$ for the given integration points yields the following result for the integral, which is exact, as required.

$$\mathbf{I} = \frac{5}{9}\mathbf{f}(\sqrt{\frac{3}{5}}) + \frac{8}{9}\mathbf{f}(0) + \frac{5}{9}\mathbf{f}(-\sqrt{\frac{3}{5}}) = \frac{2}{3}$$

In addition to the straightforward *Gauss - Legendre* quadrature there are a variety of other rules which may be applied to solve analogous integration problems. These rules are differentiated by there weight functions but follow the same basic form. These include Chebyshev quadrature and Clenshaw-Curtis quadrature rules. Clenshaw-Curtis rules are sometime used due to their fully nested nature. A nested quadrature rule is one in which a rule of level l contains all the point of the previous level $l - 1$ as well as additional nodes. Such nesting allows for more efficient sampling plan for the adaptive integration of high-dimensional problems since function evaluations can be reused when increasing the order of the quadrature rule.

Multi-variate quadrature

The above process may be extended to provide for the numerical integration of multi-variate polynomials. The integration points are then points in $\mathbb{R}^d$ rather than points on the line. For a d-dimensional integrand, an n-point quadrature rule results in n^d nodes and n^d weights. The formula for a multi-variate *Gauss-Legendre* quadrature rule is a simple extension of the univariate case outlined above. For the two-dimensional case we may write the following for a quadrature rule of level l,

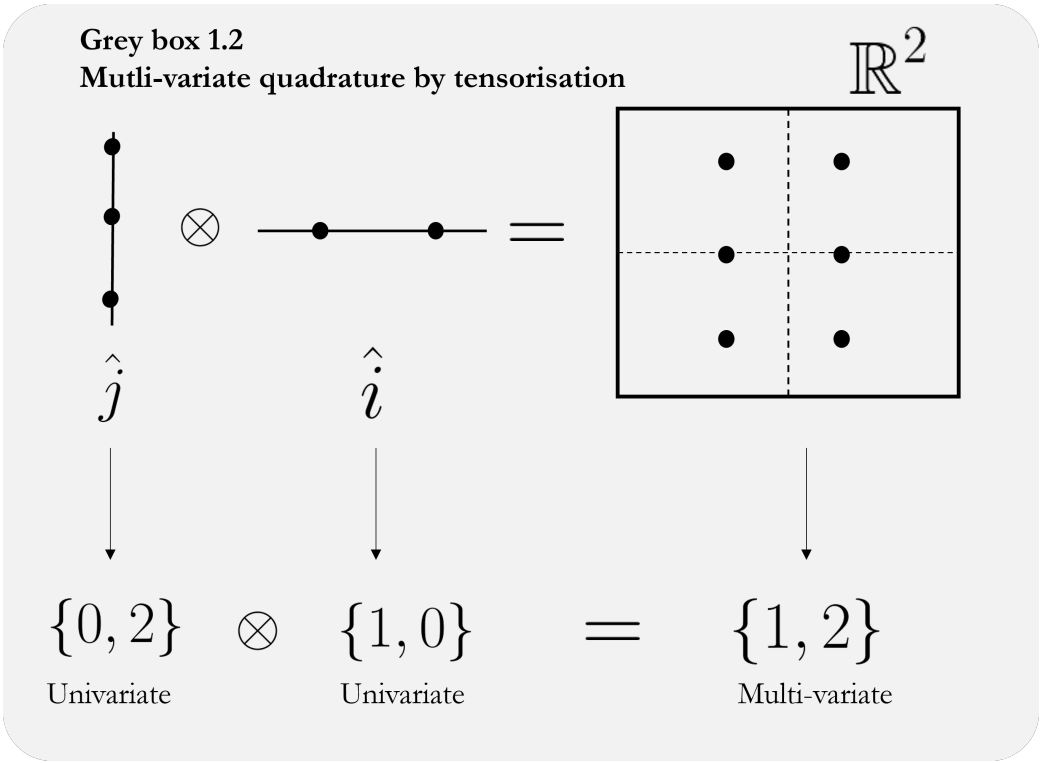


Figure 3.1: Here we demonstrate the process of *tensorisation* whereby two *univariate* Gaussian quadrature rules are used to construct a *multi-variate* quadrature rule. This is achieved by taking the tensor product of two univariate rules in orthogonal directions, indicated by the basis vectors $\hat{i}$ and $\hat{j}$. This operation results in a grid of nodes which lie in two-dimensions, as shown above.

$$\int f(x, y) = \sum_{i=1}^{l} w_x^{(i)} w_y^{(i)} f(x_i, y_i) \tag{3.46}$$

where, $w_x^{(i)}$ and $w_y^{(i)}$ are the weights for the respective dimensions, and $f(x_i, y_i)$ are the function evaluations of the multi-variate function at the quadrature nodes. To compute the integral of an arbitrary multivariate polynomial $f(x, y)$ the function must be evaluated, or sampled, at each point in the grid. For the two-dimensional case above the quadrature nodes are now points in the $\mathbb{R}^2$ plane, as pictured in Figure 1.2.

In general, for a multi-variate quadrature rule of arbitrary dimension d the quadrature points form a grid in the d-dimensional hypercube. The grid is constructed by taking the tensor or *Cartesian* product of the respective univariate quadrature rules in each of the dimensions. The number of nodes in a grid then rises exponentially with the number of dimensions. In the anisotropic case where each of the univariate quadrature rules are of order l the resultant d-dimensional grid will contain l^d nodes. Since the function, or model, must be evaluated at each of these nodes in order to compute the integral, the computational work associated with fully-tensorised grids is proportional rises with $\mathcal{O}(n^2)$. The process of tensorisation is illustrated more completely in Greybox 1.2.

The exponential scaling of the computational work required to solve such multi-variate Gaussian quadrature problems is where the so-called curse of dimensionality begins to affect the feasibility of such methods for high dimensional spaces. Above we showed how such quadra-

ture methods may be applied directly to solve for the expansion coefficients of a PCE via non-intrusive spectral projection. The computational time required to solve for the coefficients in the PCE then rises exponentially along with the number of nodes in the multi-variate quadrature grid. Therefore while surrogate-modelling methods, such as PCEs, may certainly provide for the efficient calculation of certain quantities if interest, they are not immune from the curse of dimensionality. In addition, although stochastic collocation methods do not directly utilise numerical integration to solve for the expansion coefficients, they still suffer from the curse of dimensionality via the required collocation points which are constructed in the same way.

Owing the the plurality of fields in which the solution of high-dimensional integrals is of prime importance, a number of methods have been devised to mitigate the curse of dimensionality for certain classes of functions. In the next section we will continue with a construction of sparse grids.

3.3.1 Sparse grids

We shall begin our examination of sparse grid methods by re-clarifying the problem in the context of PCEs. In order to utilise PCEs for the sensitivity analysis of high-dimensional models we must sample the uncertainty space at a certain number of points or nodes. In the case of NISP, the nodes and their corresponding function evaluations, may be used directly to solve for the coefficients of the PCE by numerical integration as outlined above, or serve as collocation points in order to solve for the coefficients using regression. In either case the feasibility of the approach is severley hampered by the exponential rise in the number of function evaluations with the dimensionality of the problem.

Sparse grids aim to mitigate this problem by reducing the number of points in the grid, *without* compromising the interpolation accuracy of the eventual PCE. Put plainly, the sparse grid approach assumes that for sufficiently smooth functions the number of function evaluations may be reduced by strategically placing the position of the grid nodes. Retaining the notation of the previous section, consider the grid defined by the linear combination of the grids indexed by the following multi-index set, $\Omega^* = \{(0,0),(0,1),(1,0)\}$. Each of the multi-indices contained in Ω^* are fully tensorised grids in their own right, created by tensorising the nodes from each of the rules, as outlined in Greybox 1.3.

It is instructive to plot each of the sub-grids according to their index in each of the constitutive dimensions. The grid described by $\Omega^* = \{(0,0),(0,1),(1,0)\}$ is said to be sparse and is constructed by taking linear combinations of the grids shown in blue in Figure 3.1. Although each of the sub-grids are fully tensorised, the result is clearly more 'sparse' than for the fully tensorised grid shown in purple (five rather than nine points). In the above case both the sparse grid and the fully tensorised grid are **isotropic** in that each dimension is sampled with the same density of points. In general, sparse grids of level L obeys the following relation,

$$||\mathbf{l}||_1 - d + 1 \leq L \tag{3.47}$$

where $||\mathbf{l}||_1 = l_1 + l_2 + \cdots + l_d$ is the 1-norm of the multi-index $\mathbf{l}$, and d is the dimensionality of the uncertainty space. Such a constraint selects the triangle of multi-indices highlighted in blue above. For the d-dimensional case they describe a 'simplex' hyperplane lying in the hypercube.

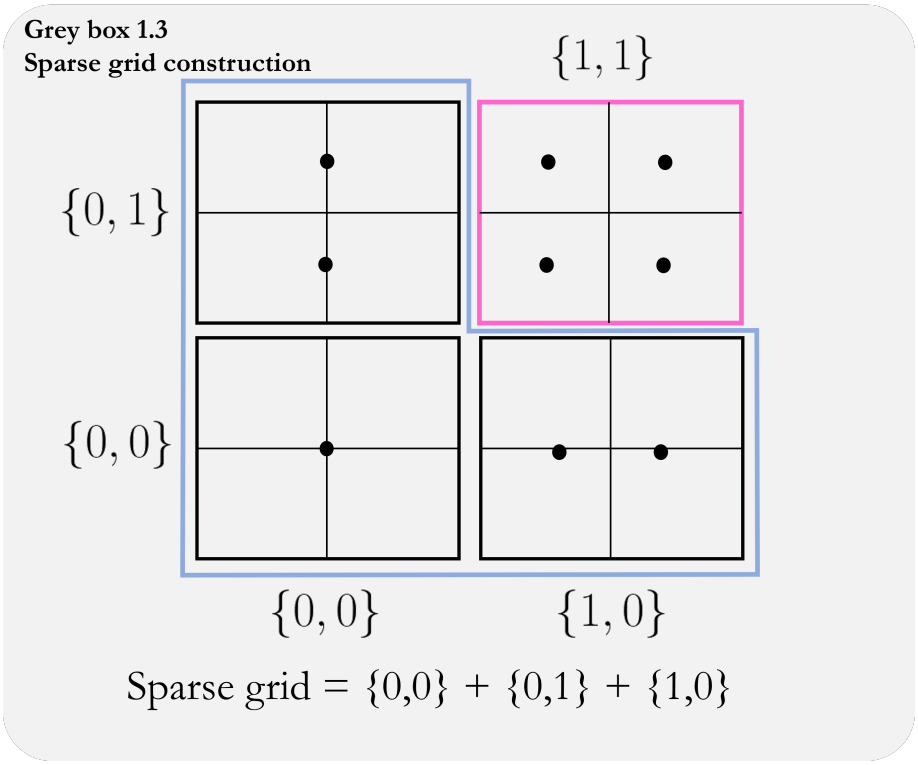


Figure 3.2: Here we illustrate the construction of a sparse grid in two dimensions. The sparse grid is composed by taking a linear combination of the fully-tensorised sub-grids, pictured in the blue box. To construct a sparse grid of level 1, we add the nodes from a sub-grid only if the l_1-norm of its index is less than 1, given the simplex shown in blue. A fully tensorised grid of level 1 is also shown, indicated by the purple box. Note that in the case of non-nested Gaussian quadrature, a sparse grid of level 1 in two-dimensions actually has more points that the corresponding full grid of level one, given by $\{1, 1\}$; however, this would cease to be the case with a higher level or higher dimension.

Although isotropic sparse-grids are demonstrably 'sparser' than their fully-tensorised counterparts, they can still become computationally infeasible due to the scaling of the number of quadrature nodes with dimension. Edeling cites $d = 10$ as a rough estimate of uncertainty spaces for which isotropic sparse grid nodes will become a computational bottleneck (Edeling, 2022).

3.3.2 Adaptive sparse grids

For higher dimensional spaces it may therefore by efficient to actively refine the sparse quadrature grid to concentrate quadrature node in those dimensions which have the largest effect on interpolation accuracy. This may be achieved through the use of so-called adaptive sparse grids where we iteratively add the constitutive sub-grids, pictured in Figure 3.2, which when added to the full grid will contribute most to reducing the error of the approximation. Such methods were first developed by (Gerstner and Griebel, 2003) for the integration of high-dimensional functions and may be used for our purposes after some alterations.

For example, taking again the quadrature grid illustrated in Figure X, if we had knowledge that a given factor, say x_2 did not significantly affect the output of the model then we would avoid refining the sparse grid in that direction, choosing instead to refine x_1. Such a sparse grid would then be anisotropic and is composed of the sub-grids given in the multi-index $\Omega^A = (0,0),(1,0)$. In the absence of foreknowledge about which factors in the model are influential we utilise a local error measure which is assumed to be a roughly effective surrogate for a global method. In (Gerstner and Griebel, 2003)'s original formulation those dimensions were chosen which contributed the greatest reduction in error once added to the grid. Such a determination is more difficult in the case of PCEs given the number of terms whose coefficients must be approximated, in contrast to the single answer produced by integrating a single functions. (Edeling et al., 2021) proposes using the the *hierarchical surplus* of a given grid as local error measure, building on the work of (Loukrezis et al., 2019). Before we provide a definition of the hierarchical surplus we first describe the main steps in the algorithm.

(Gerstner and Griebel, 2003)'s approach mandates starting with the lowest-order grid in each dimension, and iteratively increasing the quadrature order of those dimensions which are deemed important by the local error measure, thereby keeping the majority of dimensions sampled at a low resolution reducing the computational burden. We therefore start with a quadrature grid defined by the following multi-index, $\Lambda := \{(1,\cdots,1)\}$ corresponding to a quadrature grid which contains only a single node, where Λ is the set of all multi-indices chosen for inclusion. The model is evaluated for the single node, and a PCE fit, using the model evalautions. Next we consider the *forward neighbours* of the initial multi-index, those that can be reached by adding the set of elementary basis vectors $\mathbf{e}_j$ in each of the $j-$ dimensions to each index contained in the multi-index Λ. For example, adding the basis vector $\mathbf{e}_2 = (0,1,0,\cdots,0)$ to the single multi-index in Λ would yield the multi-index $(1,2,1,\cdots,1).The forward neighbours of the initial multi-index set \Lambda$ is then the set,

$$\{\mathbf{1} + \mathbf{e}_j | 1 \leq i \leq d\} \tag{3.48}$$

Similarly, for each of the calculated *forward* neighbours we can calculate their backward neighbours by performing the reverse procedure, namely, *subtracting* each elementary basis vector from each forward neighbour. A given multi-index l is deemed admissible if each of its backward neighbours are also in Λ. Using the nomenclature introduced by (**?**), the set of

Grey box 1.4
Dimension-adaptive refinement

Step	Chosen index
0	{0,0}
1	{1,0}
2	{2,0}
3	{0,1}
4	{3,0}
5	{0,2}
6	{1,1}

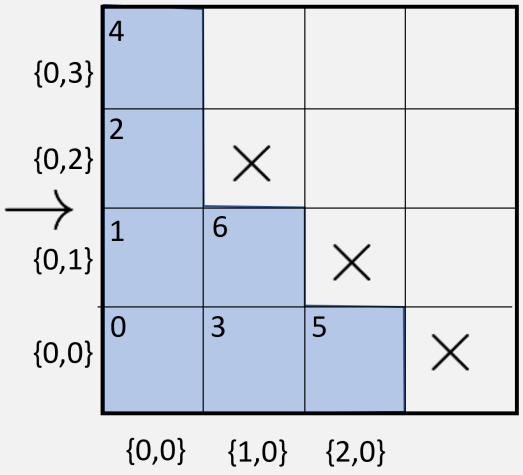


Figure 3.3: Here we illustrate possible steps of the algorithm for a 2d uncertainty space with the chosen indices (*left*) and (*right*) the path of the algorithm. For the zeroth step the polynomial is fit using zeroth-order quadrature rules for each dimension. At each step the algorithm chooses a new index to add the the 'old set' of accepted indices, which defines the grid of points over which we fit the polynomial. At step 6 the algorithm chooses to add an index which produces points in shared dimensions, indicating that at this stage in the refinement strategy it is most beneficial to resolve the interaction terms in the PCE. The admissible indices, which may be added in the seventh set, are indicated by the cross marks.

admissible forward neighbours is termed the *active set*, and represent directions in which the quadrature grid can be refined, whereas Λ is termed the *old* set and contains all those multi-indices which have been accepted.

In order to decide which of the multi-indices in the admissible set to choose, the aforementioned *hierarchical surplus* of each multi-index in the active set is calculated. The hierarchical surplus is defined as the difference between the model output $\mathbb{Y}$ and the prediction given by the PCE, $\tilde{\mathbb{Y}}_\Lambda$ evaluated at the nodes X_l defined by an admissible forward neighbour $\mathbf{l}$,

$$s(\boldsymbol{\xi}_j^l) := \mathbb{Y}(\boldsymbol{\xi}_j^l) - \tilde{\mathbb{Y}}_\Lambda(\boldsymbol{\xi}_j^l) \tag{3.49}$$

where $\boldsymbol{\xi}_j^l$ is a point in the input uncertainty space belonging to the quadrature grid produced by admissible forward neighbour l. X_l and X_Λ are the quadrature grids produced by the admissiable forward neighbour and old-set respectively, with the set difference operator indicating that we are computing the hierarchical surplus over new points only. For the surrogate PCE approximation $\tilde{\mathbb{Y}}_\Lambda$ we use the subscript Λ to indicate that it is the PCE fit using all of the points in the old set. We can now define the local error measure as follows,

$$\eta^l := \frac{1}{\#(X_l \setminus X_\Lambda)} \sum_{\boldsymbol{\xi}_j^l X_l \setminus X_\Lambda} ||s(\boldsymbol{\xi}_j^l)|| \tag{3.50}$$

The admissible index with the largest error measure is chosen and added to the old set as well as removed from the active set. A new PCE approximation is fit from the quadrature grid produced from the linear combination of each multi-index in the old-set. After this step

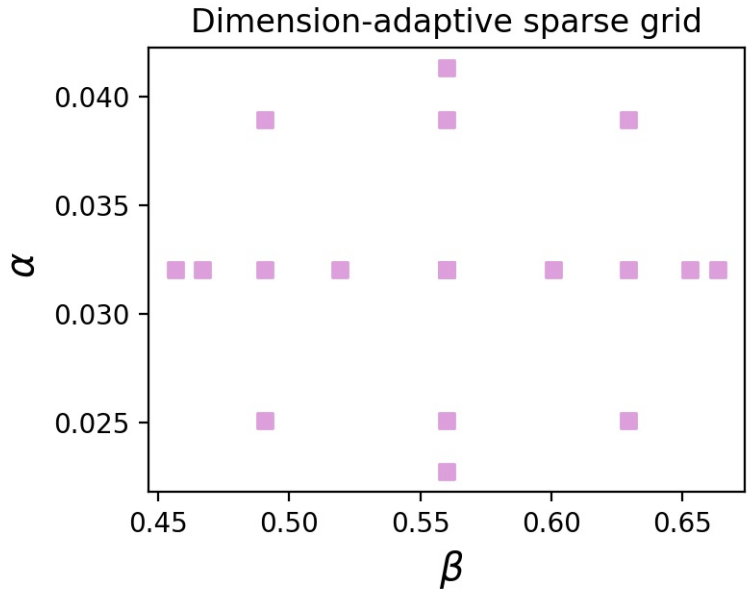


Figure 3.4: Figure showing the two-dimensional quadrature grid produced by the dimension-adaptive algorithm for the grid indices shown in Grey box 1.4. Note the characteristic sparsity of the grid, with the highest sampling density along dimensions, with a lower sampling density in the interior of the space.

the algorithm repeats, again computing new forward neighbours, checking them for admissibility and computing the local error measure. It is worth noting here that additional error measure can also be defined whose attributed may better suit different purposes, such as those based on statistical measures as given by (Perkó et al., 2014). On a practical note in order to ease the computational burden on the error calculation step it may be necessary to store the node-model evaluation pairs for later use since if a given multi-index l remains in the active set for more than one iteration of the algorithm the model will have to be evaluated for each node defined by its multi-index.

For the sake of clarity we again point out that each of the multi-indices $l = (l_1, \cdots, l_d) \in \Lambda$ are composed of one-dimensional quadrature rules which are fully tensorised. To arrive at the complete quadrature grid defined by Λ we take the linear combination of the quadrature nodes produced by each of the multi-indices in Λ. Once added together these points make up an anisotropic sparse grid. In order that the contributions from each of the multi-indices are added together correctly a coefficient, c_l is introduced which acts as a weight function and accounts for the fact that there contributions of some sub-grids cancel each other out. We define c_i as,

$$c_i = \sum_{\mathbf{e}_j \in N, i+\mathbf{e}_j \in \Lambda} (-1)^{||\mathbf{i}||_1} \tag{3.51}$$

In the case of NISP these coefficients may be readily included in the expansion by multiplying the weights produced by a given multi-index by the coefficient for that multi-index.

4 RESEARCH METHODS

4.1 MEASURES OF PERFORMANCE

4.2 EXPERIMENTAL DESIGN

In this section we specify the computational experiments which are required to answer sub-research question IV.

> **sRQ - IV:** What is the relative efficacy of the dimension-adaptive, polynomial chaos expansion method versus other *best-in-class* algorithms?

The information gained in the exploration phase now allows for the formulation of an experimental series which will contribute to answering the main research question. Whereas the previous sections consisted mostly of desk-research, and sought to provide the theoretical foundation for the project, here we will specify the computational experiments with which we will measure the performance of dimension-adaptive polynomial chaos expansions in the context of exploratory modelling.

Since the main research question concerns two separate methods which may be used in tandem, for completeness both must be first tested in isolation before their combined efficacy is assessed. Each of the algorithms will be assessed under a variety of experimental conditions according to the performance measures set out in Section 4.1. We therefore have the following main experiments.

4.2.1 Experiment I - Full quadrature grids

> ***Q:*** *What is the relative efficacy of polynomial chaos expansions in calculating Sobol sensitivity indices versus quasi-Monte Carlo methods?*

The goal of this experiment is to assess the baseline performance of PCEs versus the incumbent algorithms when applied to our selected test cases. Since PCEs are favoured for the calculation of Sobol sensitivity indices, the main output for all experiments will be the the first-order, second-order, and total-order Sobol sensitivity indices. Since the test models may not be solved analytically for the Sobol indices, an absolute error measure cannot be used to measure the level of convergence. Instead, a qMC experiment with a very large sample size will be performed to calculate the 'ground truth' for each of the sensitivity indices. The indices calculated from the fitted PCEs may then be compared against these figures and error measures calculated. Error measures may be calculated in the same manner both for the Sobol indices or for the base quantities of interest or statistical measures which we extract

from the test models.

The main factors in this experiments are the number of qMC samples, the truncation order of the PCE, and the order of the quadrature rule used to calculate the coefficients. A full-factorial design of the aforementioned factors will facilitate a full assessment of the PCEs performance by comparing the determined Sobol indices and statistical moments with groundtruth, for varying truncation order and quadrature order.

Since PCEs may find particular use in the sensitivity analysis of many-dimensional models it will be instructive to compare their performance for increasing model dimension. Such experiments have precedent in the literature albeit with analytical test functions (Crestaux et al., 2009). The dimensionality of many analytical test-functions can be chosen arbitrarily to fit the experimenter's needs which allows for a straigh-forward analysis of its effect on performance. In our case, where the analysis must be carried out on a model for which the dimensionality of the input space is fixed *a priori*, it will be necessary to artificially constrain the dimensionality of the model. This may be done by fixing the values of certain input parameters to their given default values, thereby confining the sensitivity analysis to a certain number of 'free' dimensions which remain uncertain. Usually the least influential parameters are fixed after prioritised according to their Total-order Sobol index as described in (Homma and Saltelli, 1996). The performance of the PCE in computing the Sobol indices for varying dimension may then be measured.

4.2.2 Experiment II - Sparse grids

> ***Q:*** *What is the relative efficacy of dimension-adaptive sampling in providing for faster convergence of polynomial chaos expansions versus constant and sparse cubature?*

The goal of experiment two is to examine the improvement conferred on the convergence speed of PCE by the adaptive increase of quadrature levels in certain dimensions. Since sensitivity-based order reduction of high-dimensional models is based on the presence of a lower *effective dimension* the convergence speed of PCEs may be significantly increased by concentrating function evaluations on only those dimensions which contribute to output variance. The incumbent method in this case is the sparse grid approach based on Smolyak's tensor-product space construction. It is of value to examine the relative effectiveness of both approaches under different dimensionality. Since it can be proven that both the sparse grid and dimension-adaptive approaches will always converge quicker than constant cubature it is not necessary to test the algorithm against constant cubature integration.

The PCE will be implemented using the level-order heuristic derived in **Experiment I** and applied to the toy-model(s) with the three different grid-schemes outlined above. Both constant cubature and Smolyak sparse grids may be directly implemented through Chaospy whereas dimension-adaptive sampling will have to be supported by a purpose-written procedure. As in **Experiment I** the main measure of performance will be the speed of convergence as measured in function evaluations until a halting condition is reached.

4.2.3 Experiment III - Adaptive sparse grids

> ***Q:*** *What is the relative efficacy of polynomial chaos expansions with dimension-adaptive sampling in calculating Sobol sensitivity indices versus quasi-Monte Carlo methods?*

This experiment involves comparing the performance of the combined method of PCE+DAS against the Monte Carlo and quasi-Monte Carlo methods. Although this is a separate experimental question from those outlined above, the outputs of experiments one and two provide all the data necessary for this step. As such this step is a data analysis step rather than a computational experiment. Nevertheless it will prove valuable insight with respect to the main research questions, namely, on how the combination of PCE with DAS performs on high dimensionality models.

4.3 CASE SELECTION

In order to draw conclusions as to the efficacy of the methods examined in this project a number of test-cases must be selected. The chosen test cases will be the basis of the experiments described above and should constitute good analogs of models used in exploratory modelling projects. In this section we outline the broad requirements that the chosen models must fulfill and give a brief description of their characteristics and use cases.

In order to provide for the widest applicability of our results the careful selection of cases or test-models on which to execute the experiments is of prime importance. At this stage we can delineate a number requirements which may be used to narrow the choice of models.

Given that the central problem addressed by this project arises from the exponential scaling of computational cost with model dimension, it is essential that the algorithms be tested for varying dimensions. With reference to the literature a threshold of interest can already be defined (Crestaux et al., 2009) notes that PCEs struggle to reach convergence faster than qMC methods for $d > 20$.

In this case the some models chosen must allow for us to test the algorithms either side of this threshold. Most deterministic models have a static number of input dimensions, therefore we may have to resort to factor fixing in order to vary dimension. Since this study will use Sobol sensitivity indices as its main output parameter the use of factor fixing requires a model for which the Sobol indices are known a priori or can be feasibly calculated given available computational resources.

Since approximation methods such as PCEs assume a certain regularity of the function between evaluation points, the effect of output smoothness on algorithm performance should also be probed. Whereas the regularity of analytical functions can be readily determined, this is not the case for most policy models where the propensity for highly non-linear behaviour cannot be determined by inspection.

Therefore it would be advantageous to compare algorithm performance on cases which exhibit both smooth and non-smooth output. However, since the smoothness of the output

Parameter	Interpretation
α	Exponential growth rate of prey
β	Predation rate
δ	Predator growth rate
γ	Exponential death rate of predators

Table 4.1: Table showing the physical interpretation of parameters in the Lotka-Volterra model of predator-prey population dynamics (Kinoshita, 2013)

cannot be determined by inspection model selection is not trivial.

In order to ensure that the implemented algorithms work as expected, and that the attendant processes concerning data preparation and storage are well optimised, we have chosen to stagger the complexity of each of the test cases. The algorithms and processes are first trialled with the simplest test-case before being trialled on a more complicated model. With regards to model complexity both model dimension and run-time are the two most salient factors. To ensure timely testing the simplest possible model is chosen as a starting point.

4.3.1 Lotka - Volterra Predation Models

The Lotka - Volterra equations are a pair of first-order nonlinear differential equations that model the dynamics of biological systems in which two species interact as predator and prey. Solving the coupled differential equations for a given set of boundary conditions yields the population trajectories of both species over time. The model structure is defined by four positive, real-valued parameters and may be written as,

$$\frac{dx}{dt} = \alpha x - \beta xy$$

$$\frac{dy}{dt} = \delta xy - \gamma y$$

Where $x(t)$ and $y(t)$ are the prey and predator populations at time t respectively. The boundary conditions are the initial populations of both species at t_0. The values of the parameters determine how the populations of both species by reproduction and through interaction with the opposite species in the antagonistic pair. Example interpretations of these parameters taken from (Kinoshita, 2013) are given in Table 4.1.

First proposed by Alfred J. Lotka in 1910 to describe auto-catalytic chemical reactions, the equations were later extended to include the model competitive predator-prey systems which behaved analogously to auto-catalytic systems (Lotka, 1925). In 1928, Vito Volterra modelled predator-prey population dynamics using the same set of equations in order to account for the apparent periodic nature of Adriatic Sea fish catches (Volterra, 1928).

Under a number of simplifying assumptions the equations model the interdependent evolution of predator and prey populations as they interact in the environment. For certain parameter values the equations produce periodic solutions, with the populations of each species rising and falling in response to increasing or decreasing numbers of its food source or predator. An example solution to the model for a given set of parameters and initial conditions is shown in Figure **??**.

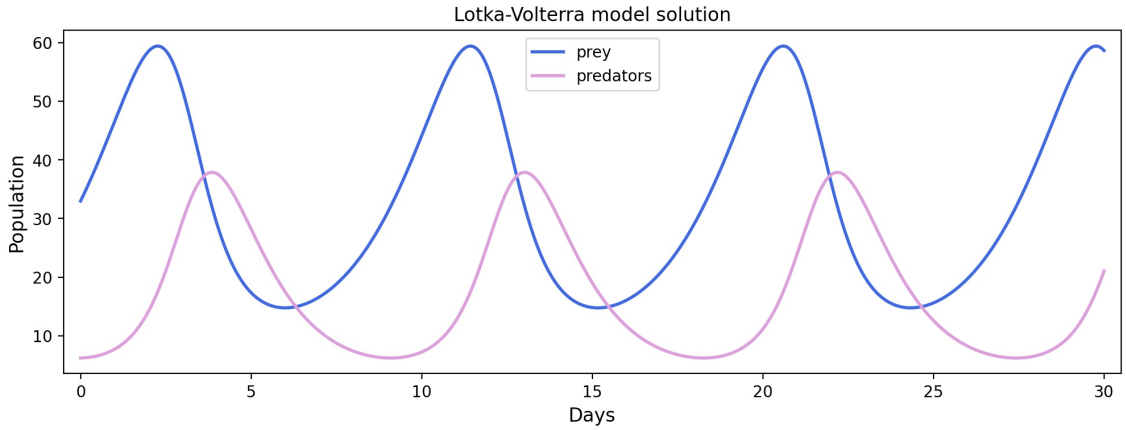


Figure 4.1: Figure showing an example solution to the Lotka-Volterra equations for an arbitrary set of parameters and initial conditions. Note the periodic nature of both predator and prey population trajectories. The peaks and troughs of both traces are slightly out of phase owing to the reponse time in population growth or decline. Note how the 'atto-fox' problem is demonstrated in the figure. Within each cycle both species are seen to recover from populations of less than one individual.

As a model of biological systems the equations suffer from a number of drawbacks which arise from the initial assumptions. The 'atto-fox' problem is the name used to describe the *unphysical* nature of the model solutions where population growth occurs continuously rather than in discrete steps corresponding to the birth, or death, of single organisms. As a consequence, species populations within the model can recover from numbers less than one, and, as such, the model does not account for the population structures that are necessary for animal reproduction.

For our purposes the accuracy of the model in describing physics systems in unimportant since we intend only to use the model as a simple test case. The Lotka-Volterra model is an appropriate choice for a test-case since it depends on only four parameters, resulting in a low-dimension input parameter uncertainty space. In addition, the equations may be solved numerically using a fourth-order Runge-Kutte method in less than a second on a personal PC. Such computational tractability allows for algorithms to be rigorously tested before being implemented on a more powerful computing cluster with a more demanding model.

The dimensionality of the input parameter space associated with the Lotka-volterra Model can be supplemented by the addition of a third species. The Three-species Lotka-Volterra Model features a seven-dimensional uncertainty space as well as high-order interaction effects and an output character that may prove difficult for surrogate modelling methods. Below, the constitutive equations of the Three-species model are presented. The increase in the number of interaction terms in the model is readily evident.

$$\frac{dx}{dt} = \alpha x - \beta xy$$

$$\frac{dy}{dt} = -\delta y + \gamma xy - ezy$$

$$\frac{dz}{dt} = -fz + hy$$

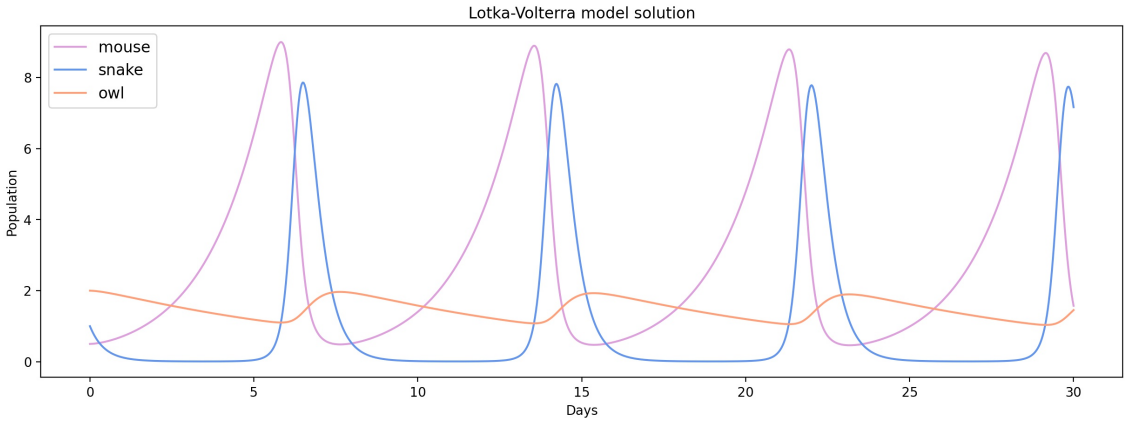


Figure 4.2: Figure showing an example solution of the Three-Species Lotka-Volterra Equations for an arbitrary set of parameters. The periodic nature of the solutions observed for the two species model is replicated here.

4.3.2 Energy Transition Model

The EU Energy Transition Model is a System Dynamics model designed to investigate plausible future states of the EU energy system as it transitions away from fossil fuels as well as find adaptive policy pathways which might help in aiding its stated transition goals. Developed by (Loonen et al., 2013) of TU Delft, it has found particular use in helping researchers assess how parametric and structural uncertainties may affect the bloc's energy transition. In one study, Loonen *et al.* employ exploratory modelling and analysis methods to propose *plausible* models of the EU energy system, identify the most important uncertainties, and generate possible future states of the system as it transitions.

The model represents distinguishes seven distinct regions and models the deployment and use of nine different generation technologies. The mode is sufficiently detailed to represent such system mechanisms as competition between generation technologies, market behaviour, and interconnection between regions (Hamarat et al., 2014).

Figure 4.3 illustrates the model's main elements and the direction of their influence on other sub-mechanisms of the model. These sub-models are realised as a series of 33 ordinary differential equations and 499 auxiliary equations in 632 variables, some of which are pictured in Figure **??**. In (Hamarat et al., 2014), Hamarat *et al.* focus their analysis on how a set of 46 uncertain parameters affect three quantities of interest, namely, the fraction of renewable technologies in operation, the fraction of carbon emissions reductions realised, and the average total costs of electricity production.

A model composed of such a quantity of interdependent equations is capable of hosting influential, high-order interaction. The strength of the System Dynamics formalism is that it provides an intuitive, and tractable, method of composing a complex model with diverse interactions. Such a model can present difficulties to modellers owing to the large number of free parameters, the potential for non-linear, and chaos-like behaviour.

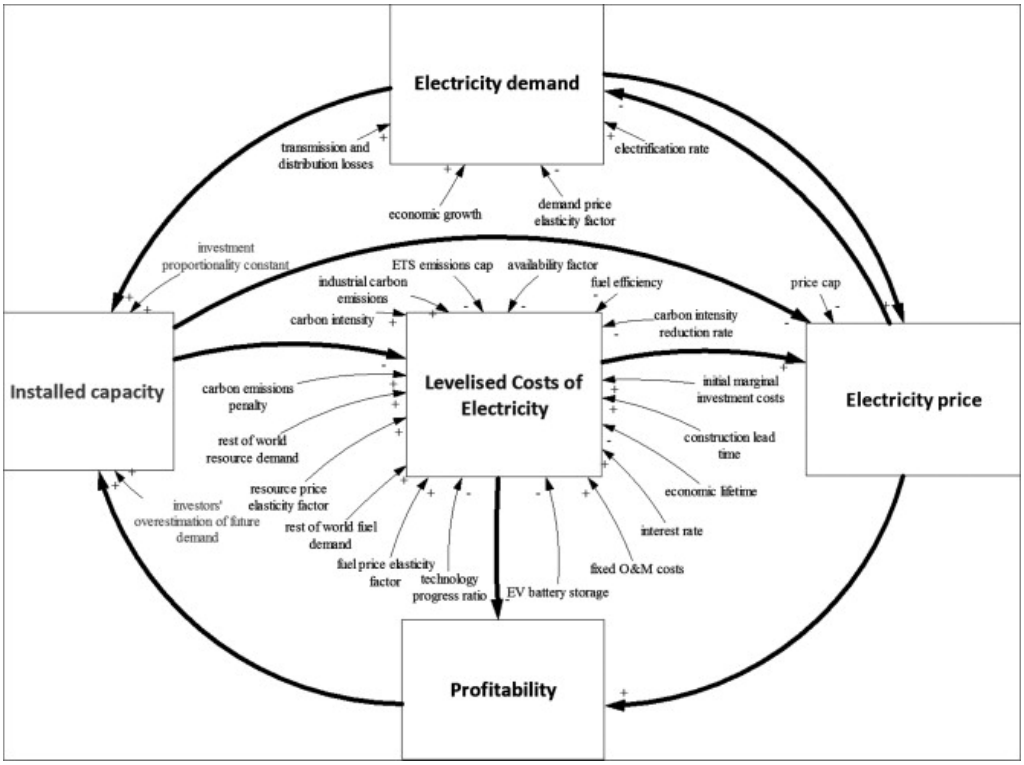


Figure 4.3: Figure showing the main sub-models of the EU Energy Transition model as described in (Loonen et al., 2013).

4.4 MEASURES OF PERFORMANCE

The algorithms investigated in this study will be judged both on their speed in converging as well as the accuracy of the result upon convergence. In order to draw concrete conclusions as to both of the aforementioned measures of performance the result will be compared with the 'so-called' ground truth values established via direct sampling methods. 'Speed' in this context is measured as the number of model evaluations required to reach convergence and is analogous to the computational cost associated with a given algorithm.

In this case of Sobol sensitivity indices we define a vector $\mathbf{S}_x$ in which we store the given Sobol index for each parameter. The Sobol indices may then be directly compared with the ground truth values by computing the following error metric,

$$\epsilon = \frac{||\mathbf{S}_x - \mathbf{S}_{GT}||}{||\mathbf{S}_{GT}||}$$

where x is the order of the Sobol index in question, and $||\cdots||$ is L_2-norm operator. In this study we primarily focus on the total-order Sobol indices since they fully capture inter-action effects of all orders and may be directly applied as the basis for model order reduction by factor fixing.

4.5 SOFTWARE

With the exception of Vensim, in which the Energy Transition Model is realised, all of the software utilised over the course of this study is open-source and freely available for use. All

of the packages used, again with the exception of Vensim, lie within the Python ecosystem.

The majority of the calculations concerning polynomials and polynomial chaos expansions were completed using Chaospy, which provides a well documented API for uncertainty quantification using PCEs and other direct sampling methods.

Additional inspiration was drawn from EasyVVUQ, a python library for uncertainty quantification in high performance computing (Richardson et al., 2020).

Jan Kwakkel's Exploratory Modelling Workbench was primarily used in order to interface with Vensim and run the Energy Transition Model. The in-built multi-processing functionality facilitated running the experiments in parallel on a high-performance cluster, significantly reducing the required computational time.

SALib, a multi-purpose Python library for the sensitivity analysis methods was used to compute Sobol sensitivity indices during the quasi Monte Carlo experiments.
All of the code written in fulfillment of this thesis project is available on Github and free to redistribute and reuse. The code base is not neatly composed as a package and, as such, some parts may be poorly commented and opaque to the reader. However, the main functional subunits of the code are easy to reach and readily understandable. Please refer to the README at https://github.com/JeffreyDillonLyons/curse$_b$*reaking.git* for more information on accessing and making use of the code base.

5 RESULTS

5.1 TWO SPECIES LOTKA – VOLTERRA MODEL

5.1.1 Model characterisation

As noted in Chapter 4, solution of the coupled Lotka-Volterra equations numerically yields two periodic traces. Each trace charts the population level of the given species as a function of time. An example solution is illustrated in Figure 4.1. The population trajectories oscillate in tandem, with the response of the predator population lagging by a certain phase factor. Analytical solution of the equations yields traces which oscillates infinitely, so the numerical results behaves as expected .

The solutions pictured in Figure 4.1 correspond to a single point in uncertainty space which specifies the values of all four parameters. Together with the boundary conditions they specify a unique solution to the coupled equations. The surrogate models which we are attempting to fit are intended to represent the response of the two-species Lotka-Volterra equations over the entire uncertainty space. As a quantity of interest (QoI), we extract the population level of both species at a particular time-step, $t = 910$. The choice of quantities of interest is arbitrary, and depends on the requirements of the modeller. In this case, we might equally have chosen any other relevant statistical measure or property as our QoI; however, it should here be noted that a fitted PCE, and all statistical moments that are derived from it, are specific to the chosen QoI. By extracting a single value for our QoI from each model run we are creating an ensemble of point-model evaluation pairs which will be used to fit the interpolating PCEs and approximate a response-surface for the model. The response surfaces in this case are four-dimensional surfaces which map the four input uncertainty values to the population value at $t = 910$. In the sequel a ´point´ refers to a point in the n-D uncertainty space which specifies a value for each of the four uncertain input parameters.

In order to characterise the model behaviour over the entire uncertainty space an ensemble run was executed by sampling the uncertainty space at 327,680 points and recording the population values for both species. The points were chosen using Saltelli's extension of Sobol's low-discrepancy sequence (Saltelli, 2002). As such, it constitutes a quasi Monte Carlo experimental design. The results of this experiment are shown in Figure **??** and Table **??**.

The statistical moments listed in Table 5.1 confirm what can be noted by inspection of Figure 4.1, namely, that the distribution of predator population levels over the uncertainty space is much sharper than that of prey, featuring a dense concentration of observations to the left of the sample mean. Both distributions feature long tails, though it is the prey population which features the largest standard deviation and variance. As noted in Chapter 3, the convergence speed of interpolative approximation methods depend on the smoothness of the target function, with less-smooth functions requiring higher resolution sampling schemes, and higher order expansions, to fully capture the variability of the function over a given domain. Where

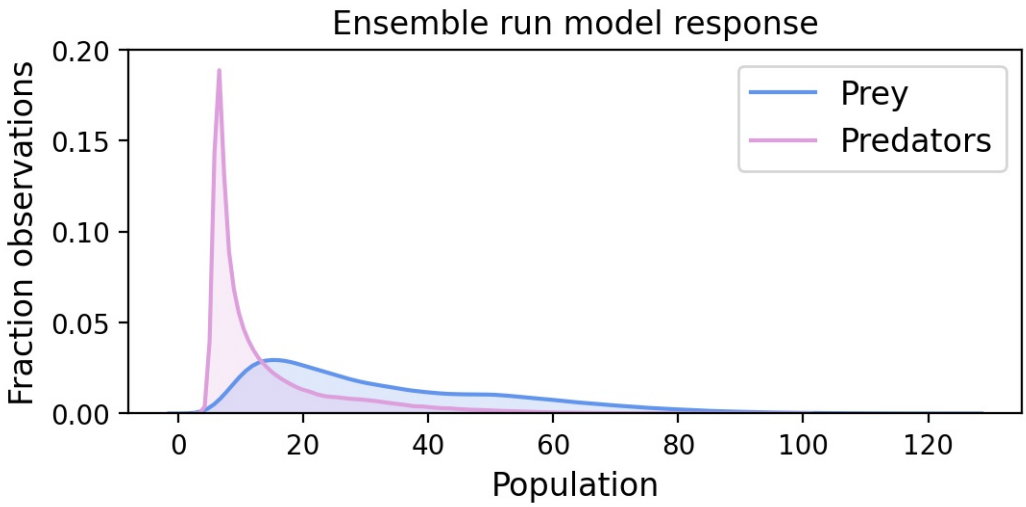


Figure 5.1: Figure showing the output distributions produced by the model for each of the species. The samples were drawn using a Sobol-Saltelli sequence. Note the characteristic sharpness of the predator distribution (*pink*)

Moment	**Prey**	**Predator**
Mean	32.72	13.78
StDev (normalised)	0.6	0.8
Variance (normalised)	11.91	8.92
Skew	0.93	2.24
Kurtosis	0.25	5.89

Table 5.1: Table showing the statistical moments calculated for both predators and prey for the Two-Species Lotka-Volterra model. These are some of the figures which we will attempt to reproduce using the fitted PCEs.

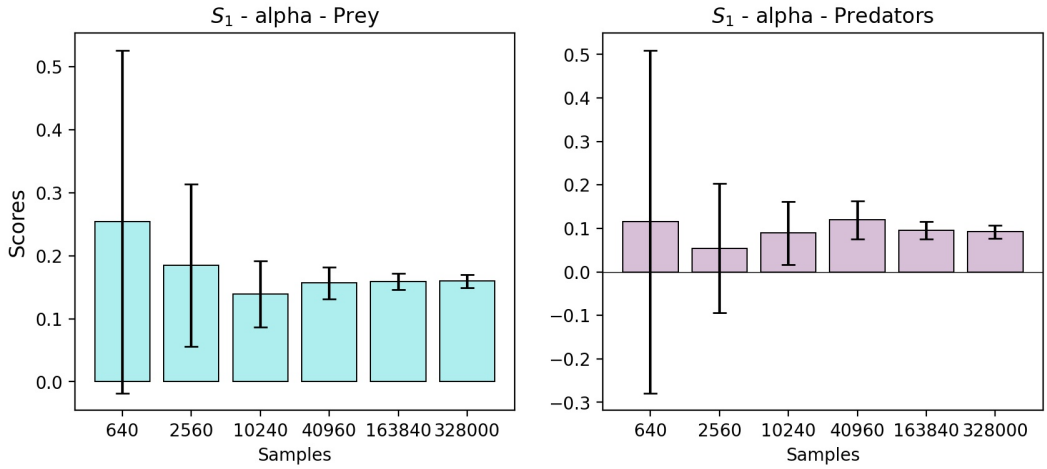


Figure 5.2: Figure showing the convergence of the qMC experiment for the first-order Sobol sensitivity index of a single model paramter, α.

the two-species Lotka-Volterra model is concerned, the prey population distribution, with its characteristic smoothness and low skewness, should be easier for the PCEs to approximate. The predator population distribution however, is sharp and highly skewed, exhibiting a step-like character about its mode.

Once the PCEs are fitted to both response surfaces they will be capable of reproducing the above statistical moments to a certain degree of accuracy. In addition to the statistical moments we also calculate the first, second, and total-order Sobol Indices. These values, taken together, form the 'ground truth' against which the accuracy of the PCEs are judged. Figure 5.2 shows the first-order Sobol index, S_1, for a single input parameter, α, as a function of the number of qMC samples used in its calculation. The value is observed to quickly converge, with little change in the determined figures after 40,960 samples. The error bars are calculated by the SALib package using a bootstrap re-sampling procedure and illustrate the 95% confidence interval. The relatively small re-sampling errors, of 3.2% for prey, and 5.7% for predators, with 327,680 samples at the 95% confidence interval indicates that this is an acceptable sample size for establishing the ground truth approximations of the Sobol indices.

These results demonstrate that, accounting for first-order interactions alone, the parameters α and δ have a larger impact on the variance of the model output than β and γ.

Figure 5.3 shows the ground truth S_1, S_2, and S_t values for both species. Comparison of the first and total-order indices indicates the presence of significant higher order effects, an observation confirmed by the calculated second-order indices. The second-order effects are observed to mostly account for the difference between S_1 and S_t across the parameters. The remaining difference which is not accounted for demonstrates the presence of effects with an order higher than two; however, the successive contribution of higher order terms to the variance of the output is quickly diminishing, as demonstrated in the figure. It is instructive at this point to attempt to replicate the findings of (Box and Meyer, 1986), namely, that the influence of higher order effects follows a Pareto distribution. Taking prey as an example, we find that first and second-order effects together account for 79.3% of the model output variance, with the remaining 20.7% due to higher order effects, in near perfect agreement with the 80:20 heuristic. While we might have expected first-order effects alone to account for around 80%

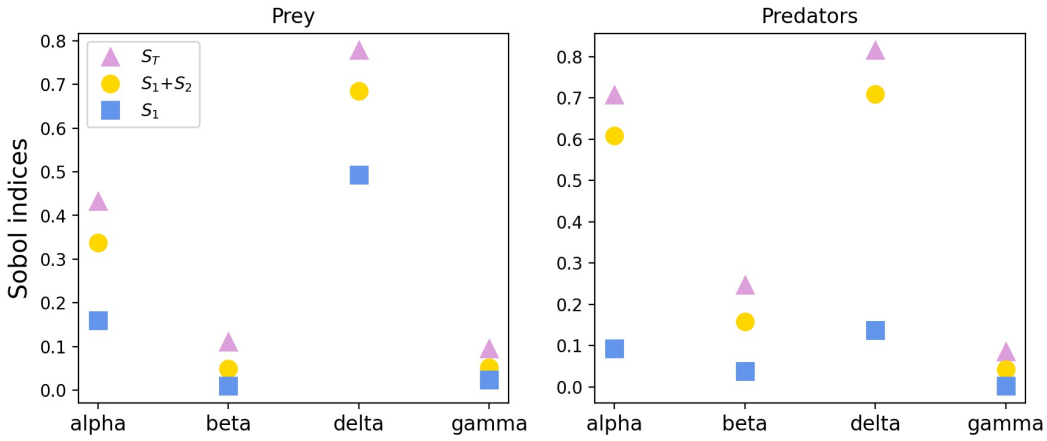


Figure 5.3: Figure displaying the first, second, and total-order Sobol sensitivity indices for each parameter and both species. The stacked nature of the plot well displays the summative nature of Sobol indices.

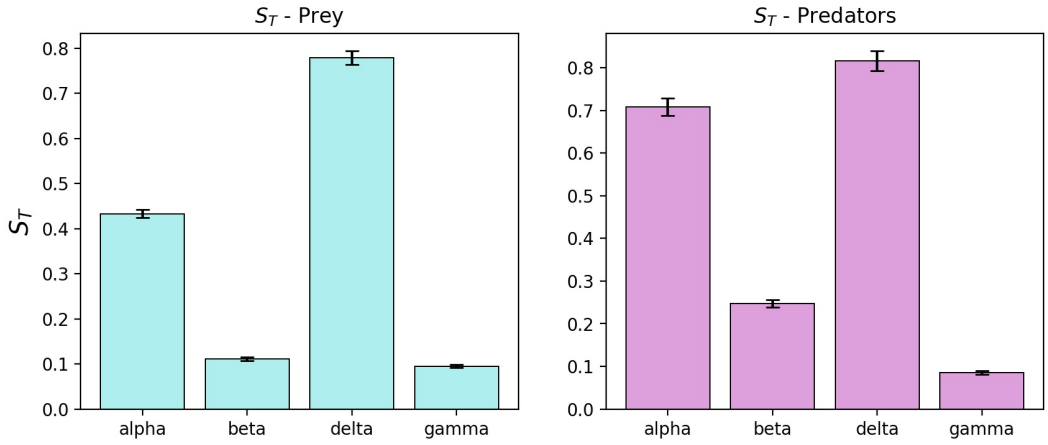


Figure 5.4: Figure illustrating the sensitivity landscape which we must attempt to approximate with PCEs.

of the model variance that is simply not possible for a model with highly-correlated inputs. Nevertheless, our findings are a reasonably good fit and support the findings of (Box and Meyer, 1986).

Such data provides vital information for the later evaluation of the PCEs. In this case, the observed extent of higher order effects implies that a third-order PCE will be required to capture the majority of the behaviour produced by the model, whereas a second-order polynomial would serve as a mostly accurate ($\sim$ 80%) and ´cheaper´ alternative.

Figure 5.4 contains a clearer illustration of the total-order indices which will serve as the main object of comparison is assessing the performance of PCEs versus the traditional qMC methods explained above. A more complete elaboration of the results from this section is available in Table 5.2

For the sake of brevity, in the sequel we will only make use of the total-order Sobol indices in order to compare the performance of the PCEs.

5.1.2 Experiment I - Full quadrature grids

We now proceed with the results for Experiment I, where we compare the statistical moments and sensitivity measures calculated from the fitted PCEs against those presented above. While the figures may be compared directly against ground truth in order to gauge the relative accuracy of the methods, comparison of convergence speed requires more nuance. While we required 327,680 model calls to determine ground truth, we could have saved a large amount of computational effort for settling for slightly higher error. An advantage of quasi Monte Carlo methods is that their convergence speed is only weakly dependent on the dimension of the sample space they are drawing from (Gerstner and Griebel, 2003). While this still renders them too slow for use with high dimensional problems, they can approach convergence quite quickly for spaces with lower dimensionality. The graph in Figure 5.2 illustrates the diminishing returns inherent in such quasi-random methods, with 40,960 samples giving a re-sampling error of 9.6% and 11.7% for prey and predators, respectively. Furthermore, the calculated total-order index values at 40,960 samples are within 5% of the established ground-truth values for both predators and prey. The extent of the diminishing returns means that comparing the PCEs against ground truth in terms of convergence speed or function calls may be misleading. We will therefore make some effort to compare the PCE convergence rates with the figures determined for 40,960 samples.

Since the presence of higher order effects in the model has already been confirmed by the Sobol-Saltelli sensitivity analysis, we expect to require a PCE which contains higher order terms to adequately fit the model output. A second-order PCE is then the minimum order surrogate model capable of capturing higher order effects. Figure 5.5 shows results for both a first and second-order PCE in approximating the model output, with the number of nodes in the fully-tensorized integration grid used to fit the PCEs given on the x-axis. For a fully tensorized grid, Ω_o^d, of order o, and dimension d, the number of nodes is related to the grid order by,

$$o = \sqrt[d]{N}$$

where d is the dimensionality of the uncertainty space, and N is the number of nodes in the grid.

Since we are mainly interested in the calculation of Sobol sensitivity indices, the error measure is determined with respect to the ground truth values for the total-order indices calculated in the previous section. We compose the sensitivity indices for each model parameter into a vector and calculate the relative error in the vector's l_2-norm with respect to the ground truth values as described. The relative error metric is then,

$$\epsilon = \frac{||\mathbf{S_T^{PCE}} - \mathbf{S_T^{GT}}||}{||\mathbf{S_T^{GT}}||}$$

Figure 5.5 (a) demonstrates the poor fit achieved by the first-order PCE, with no gain in approximation accuracy with an increasing number of samples. By contrast, the second-order PCE is observed to converge quickly, with the accuracy of approximation plateauing after 256 samples. Both expansion orders show large errors. These results confirm our intuition that the resolution of the first-order PCE is too low to fully capture the present interaction effects, and even when fully converged is only a poor surrogate of the model output. Similarly, the second-order PCE is only capable of approximating second-order interaction effects, and will not represent model behaviour arising from third-order effects or higher, nevertheless it is

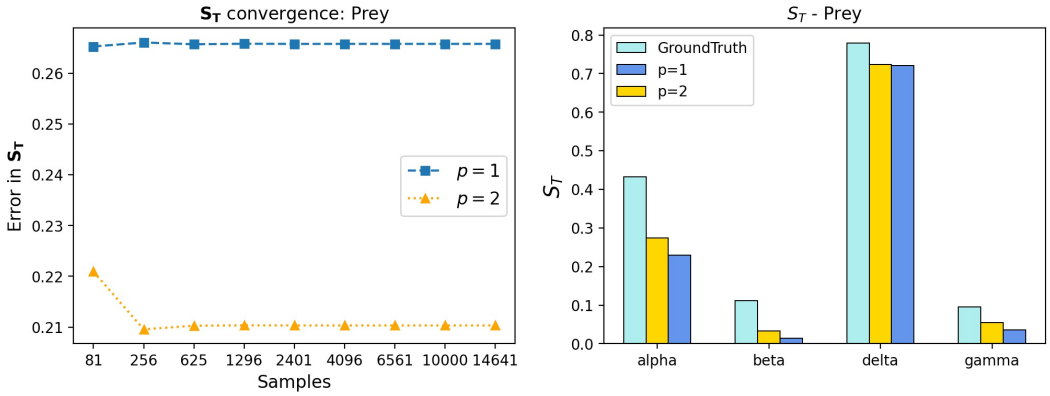


Figure 5.5: Figure (*left*) showing the error in the total-order Sobol indices for a first and second-order PCE, as well as the calculated indices for each parameter (*right*). Note the relatively small increase in accuracy (5%) conferred by increasing the order of the PCE, this is indicative of potentially influential higher-order effects.

observed to be a much more accurate approximation. Once again, these results follow the generally diminishing returns which are expected when including higher order interaction terms as suggested by the sparsity of effects principle (Box and Meyer, 1986). The relative error of the second-order PCE versus ground-truth is $\sim$ 20% because it cannot approximate interaction effects of order higher than three which we determined in Section 5.1.1 to account for $\sim$ 20% of the model variance.

The comparison of total-order indices for all parameters, shown in Figure 5.5, well illustrates the increase in approximation accuracy gained by increasing the truncation order of the PCE. For parameters with small higher order interaction effects, such as β and γ, the second-order PCE confers no increase in accuracy. The results for δ are of interest, since the expected gain in approximation accuracy with a higher order PCE was not measured. This is likely due to approximation error inherent in the interpolation itself, rather than arising solely for the inclusion of interaction effects. For brevity we shall only show the results for prey as the results for the predators are similar and differ only due to the presence of larger second-order effects for alpha and delta, as shown in Figure **??**. A full reporting of the results for both predators and prey can be found in Table 5.2.

We may repeat the above analysis for even higher expansion orders, aiming to more completely capture the behaviour of the model. Since the ground truth values are know *a priori*, it is straightforward to measure the diminishing returns with increasing order. In situations where ground truth values are not known *a priori* it may be instructive to inspect the accuracy of the model visually, as is shown in Figure 5.7. Although far from rigorous, evaluation of the approximation error of the PCE, visual inspection is useful for quickly ascertaining how well the surrogate model is reproducing the model values.

In Figure 5.6, the results for the entire series of truncation orders are presented. The sharply diminishing returns in accuracy are evident in the graph. There is a large gain in accuracy for $p = 3$, with the error from GT dropping to below 10% upon convergence. For higher truncation orders however the marginal gain in efficiency is significantly lower, as the successively smaller contributions to model variance by successively higher order interaction affects are captured by the PCE. Similarly, once converged there is no gain in accuracy with fitting an ex-

	Prey			Pred		
Observation	$p=3$	$p=6$	GT	$p=3$	$p=6$	GT
Nodes / Function evaluations	256	2401	327,680	256	2401	327,680
Mean	32.72	32.72	32.72	13.78	13.78	13.78
StDev (normalised)	0.58	0.58	0.6	0.77	0.8	0.8
Variance (normalised)	11	11	11.91	8.1	8.76	8.92
Skew	0.41	-5482	0.93	2.09	6624	2.24
Kurtosis	-0.64	-	0.25	4.87	-	5.89
$S_T-\alpha$	0.39	0.42	0.43	0.69	0.7	0.71
$S_T-\beta$	0.08	0.1	0.11	0.2	0.23	0.25
$S_T-\delta$	0.76	0.78	0.78	0.8	0.81	0.82
$S_T-\gamma$	0.08	0.09	0.1	0.05	0.08	0.09

Table 5.2: Table showing a full breakdown of results for Experiment II. Both the statistical moments and and total-order sensitivity indices which were extracted from the fitted PCEs are compared directly against those determined by direct-sampling methods. Note the marginal gain in accuracy achieved by increasing the truncation order of the expansion for three to six.

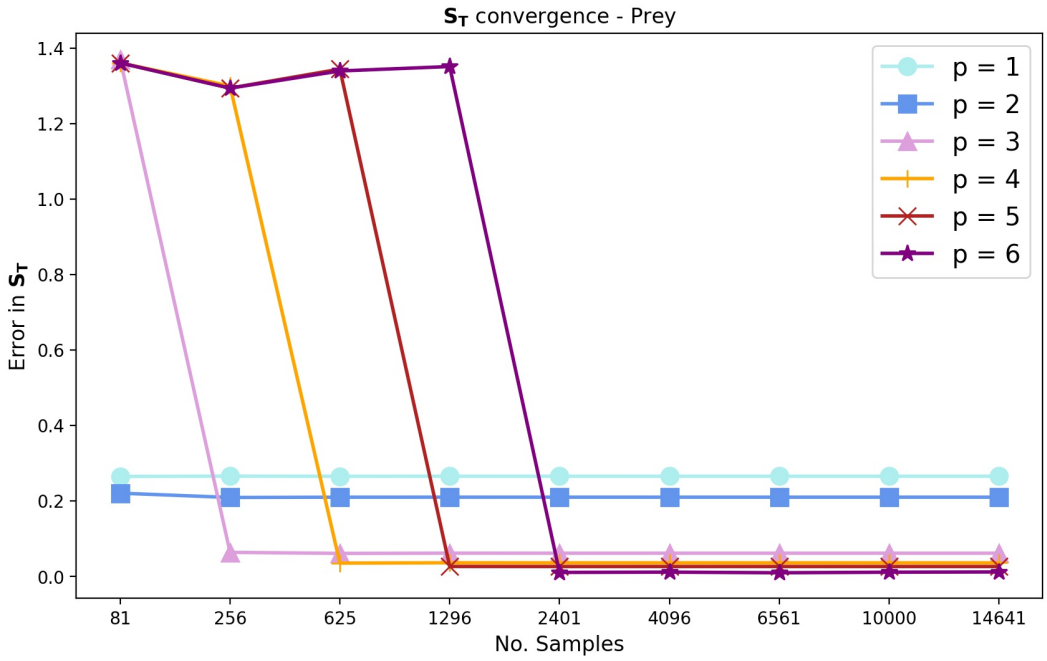


Figure 5.6: Figures showing the approximation accuracy for each PCE in Experiment I. PCEs of truncation order from 1-6 to six were fit using fully tensorised Gaussian quadrature grids from levels 1-10. Once converged, the PCE with truncation order $p = 3$ is observed to confer a large increase in approximation accuracy. This is due to third-order effects present in model being captured by the third-order terms in the PCE. The sharply diminishing returns for higher truncation orders are evident in the plot. Note the characteristic convergence behaviour with each PCE converging to its maximum accuracy when fit by a quadrature grid of level $l = p$.

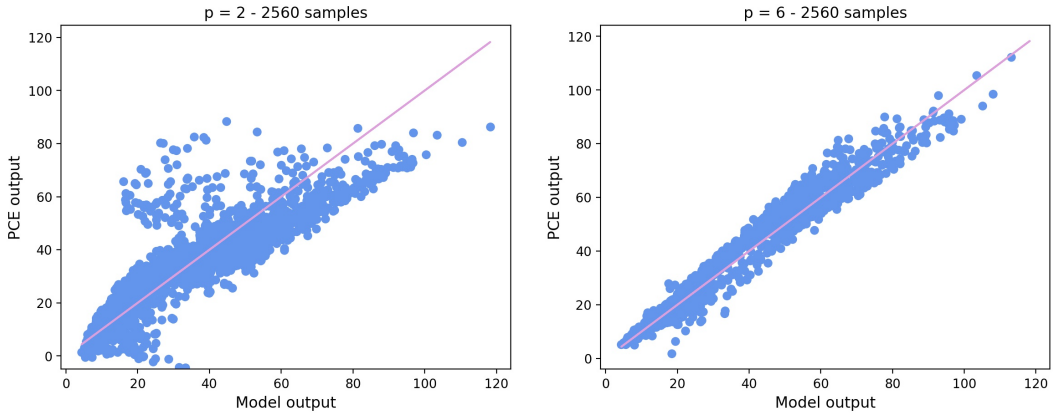


Figure 5.7: Figure showing the accuracy of the PCE surrogate models for two truncation orders as illustrated by a scatter plot. A 'perfect' surrogate model would be represented by a *one-to-one* correspondence between the the actual model and PCE evaluations, lying along the line shown in pink. The increase in truncation order from two to six is observed to confer a large increase in the accuracy of the surrogate model, although this is not reflected in a corresponding increase in the accuracy of the calculated statistical measures.

pansion of a given order with a higher resolution grid. This marginal gain in approximation accuracy must be balanced against the computational cost inherent in fitting the higher order models. Although certainly a more accurate surrogate model, as illustrated in Figure 5.7, the corresponding gain in approximation for the Sobol indices is minimal. The third-order PCE converged for 256 samples and determined the Sobol indices to within 10% error. By comparison, the sixth-order PCE, which required on the order of 10 times more function evaluations to fit, conferred only an additional 5% in approximation accuracy. As evidenced by the clear trend in the Figure 5.6, each higher order PCE required a larger number of samples to converge to its maximum accuracy.

5.1.3 Experiment II - Sparse grids

We now proceed with the results from Experiment II, testing the effect of sparse grid techniques on the accuracy and convergence properties of PCEs. The characteristic sparsity of sparse grids reduces the number of quadrature nodes used to approximate a given function, leveraging the smoothness of the integrand to derive accurate results with less computational cost. For high-dimensional models, where the curse of dimensionality begins to severely limit the order of quadrature rules which may be feasibly employed, sparse grids delay the onset of the curse until higher dimensions.

Figure 5.9 illustrates the convergence properties of the sparse grids when applied to determine the expansion coefficients of the PCE and thereby determine the Sobol indices for prey. The data follows the same trend found in Experiment I with the fully tensorised grids, that a PCE of a given order has a maximum attainable accuracy, and once converged, gains little approximation accuracy with an increase in grid size. We also observe the same increase in accuracy with increasing PCE truncation order, albeit with the same sharply diminishing returns in accuracy after $p = 3$.

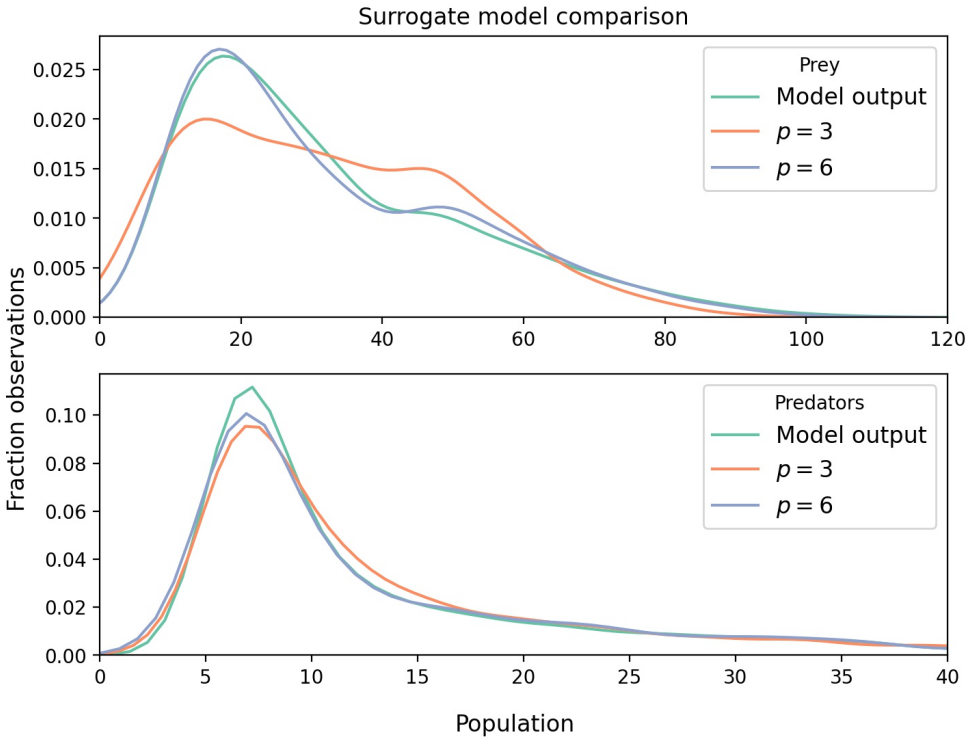


Figure 5.8: Figure showing the output population distributions from both the actual model and surrogate PCEs of various truncation orders.

The most significant qualitative difference observed is the more gradual convergence of the PCEs for successively higher-order sparse quadrature rules. Whereas, in the case of fully tensorised grids, the PCEs converged completely once a certain threshold had been passed, the application of sparse grids leads to a steadier approach toward convergence. This arises from the preference of the sparse grid method for certain dimensions over others. As illustrated by the diagram in Grey Box 1.3, sparse grids have the highest sampling density along axes of the sample space, with fewer samples of shared dimensions. Consequently, terms in the expansion of mixed order are integrated with lower order quadrature rules and, as such, their coefficients are less accurately determined.

This is the central strength of the sparse grid method, that it dose not 'waste' model evaluations on terms of mixed order, which are assumed to contribute only marginally to the accuracy of the result. Indeed, it is analogous to the sparsity of effects principle, albeit applied more generally to the high-dimensional numerical methods.

The results for prey in Figure 5.9 show excellent convergence for expansions of order higher than three, converging to within 5% of the ground truth values. Comparison with fully tensorised results in Figure 5.6 shows that in actuality, the sparse grid method was costlier, with larger number of model evaluations required for convergence. This is well exemplified for the PCE of truncation order 6 which was integrated exactly with only 2,401 model evaluations, required 46,721 to converge to a comparably low error. This is in direct contradiction to the supposed aim of sparse grid methods. This phenomenon arises from the dimensionality of the uncertainty space. For low-dimensional models, such as that investigated here, the sparse grid construction, being based on linear combinations of low order quadrature grids leads to more sampling points than a fully-tensorised grid of the same order. However, for high

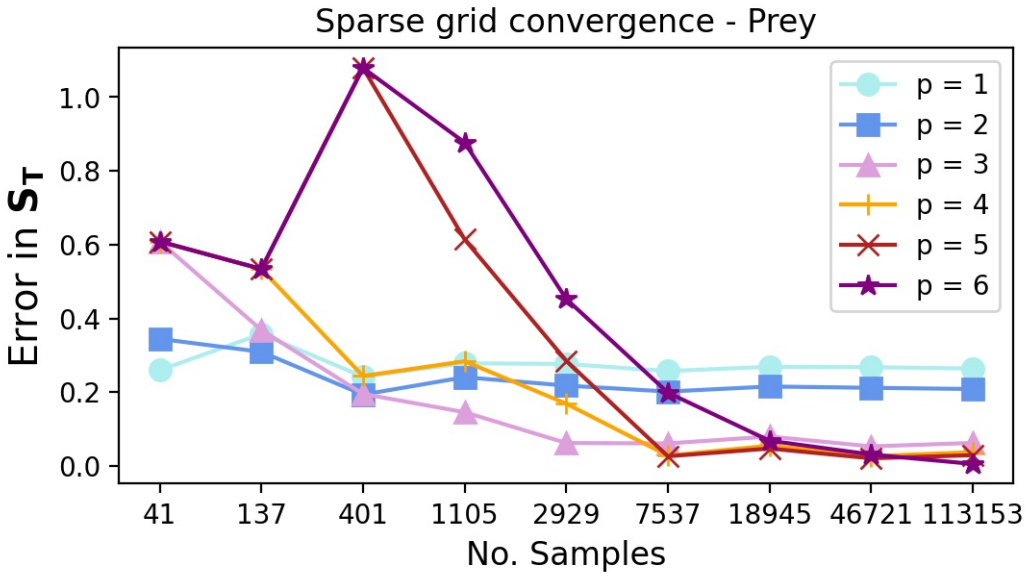


Figure 5.9: Figures showing the approximation accuracy for each PCE in Experiment II. PCEs of truncation order from 1-6 to six were fit using sparse quadrature grids of levels 1-10. Note the smoother approach toward convergence when compared to the spectral convergence observed in 5.6. For the Two-Species model sparse grids are observed to be relatively more expensive than the fully-tensorised grids.

dimensional uncertainty spaces, while the fully tensorised grids feature an exponential rise in the number of nodes, rendering them quickly unfeasibly, the size of sparse grids rises more slowly.

5.1.4 Experiment III - Adaptive sparse grids

Through the use of the adapted Gerstner's algorithm, explained in detail in Chapter 3, we attempt to iteratively choose those dimensions which of the uncertainty space (parameters) for which the hierarchical surplus is the greatest (Gerstner and Griebel, 2003). The hierarchical surplus method, as defined in Chapter 3, provides a local error metric which can serve as a surrogate metric for the actual global error which can be very costly to compute (Edeling et al., 2021) (Loukrezis et al., 2019). This error metric may then be be used, over the course of the iterative procedure, to refine the sampling plan in those dimensions which have the the greatest impact on the accuracy of the PCE.

Owing to the correspondence between PCE coeffecients and the Sobol indices for a given model, the adaptive algorithm should refine those dimensions which have the largest Sobol indices. Therefore the progress of the algorithm can be tracked by contrasting its refinement directions against the ground truth values.
Since the quadrature grids produced by the algorithm are sparse, the results from Experiment II provide useful upper bounds on the maximum accuracy which may be attained.

As an example we seek to replicate the 'best-value' expansion from Experiment II, that for $p = 3$, which converged to an error of 6.2% with a sparse grid containing 2,929 points. The dimension-adaptive algorithm should be able to replicate the accuracy of this results albeit

with far fewer function evaluations.

Figure 5.10, shows the results for the dimension adaptive algorithm applied to a a third-order expansion with the prey population at time $t = 910$ as its target. The tile plot in Figure 5.10 illustrates each step in the algorithms iterative refinement strategy. At the zeroth step the grid is constructed from a linear combination of level 1 quadrature rules in each dimension, represented as $\{1, 1, 1, 1\}$ in multi-index notation. This initial grid contains $2^4 = 16$ points and produces the first fitted PCE from which hierarchical surpluses are calculated. In the second iteration, δ is chosen as the dimension to refine. After the points from the chosen index $\{1, 1, 2, 1\}$ were added to the old set a new PCE is fitted and local errors calculated for each admissable index set as outlined in Chapter 3.

In the next iteration, α is chosen for refinement and the points from the grid $\{2, 1, 1, 1\}$ are added to the old set. The final iteration sees the inclusion of a grid with terms from a shared dimension, $\{1, 1, 2, 2\}$. The algorithm halts after the third iteration. The final sparse grid, constructed from all the indices in the old set contains, 116 nodes. The Sobol indices calculated from the resultant PCE were within 4% of the ground truth values. This implies that the dimension-adaptive algorithm required 25.25 times fewer function evaluations than the sixth order sparse grid used to fit a third order PCE in Experiment II. In fact, the dimension adaptive algorithm produced results which were more accurate than the comparable result in Experiment 2 ($p = 3, \Omega_6^s$). This is in direct contradiction with the assertion made above that the results from the complete sparse grids would provide upper bounds on the maximum attainable accuracy by the algorithm. We hypothesis that the discrepancy is due to errors introduced by the large number of nodes used in determining the figure in Experiment II.

Note the correspondence between the refinements chosen by the algorithm and the total-order Sobol indices determined in the model characterisation step. The most influential factor, as measured by its total-order Sobol index, δ, was chosen first, followed by α, the second most influential variable. This result indicates that the local error measure based on the hierarchical surplus is effective in directing the algorithm toward dimensions which are influential. The final refinement sees the addition of points from the dimension shared between δ and γ, which suggests that higher order interaction terms involving these two uncertain variables were more important to reducing the error of the eventual PCE versus the other grids under consideration.

In this run of the algorithm the maximum quadrature order of each dimension is 2.

To illustrate the action of the algorithm on a higher order PCE we take the an expansion for $p = 3$, the highest order expansion investigated in Experiment II. In both Experiment I and II we concluded that although the 6^{th} order PCE was the most accurate, the marginal gains in accuracy did not justify the large cost of fully fitting the PCE. It is of interest, therefore, to investigate whether the dimension-adaptive algorithm can cheaply fit such a high order polynomial, dedicating function evaluations to important dimensions while ignoring the rest.

The results for the 6^{th} order PCE are shown in Figure 5.11. The increased complexity of the path taken by the algorithm when compared to that for $p = 3$ is readily seen.

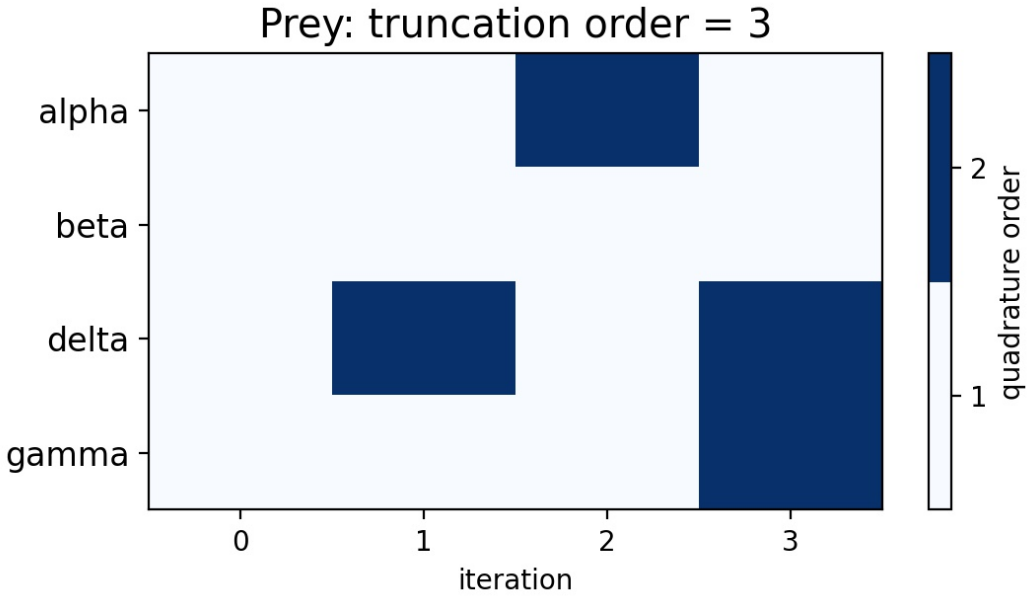


Figure 5.10: Figure showing the steps in the adaptive refinement strategy employed by the algorithm for truncation order $p = 3$. Note that the algorithm was able to converge to sufficient accuracy with a maximum quadrature order of two.

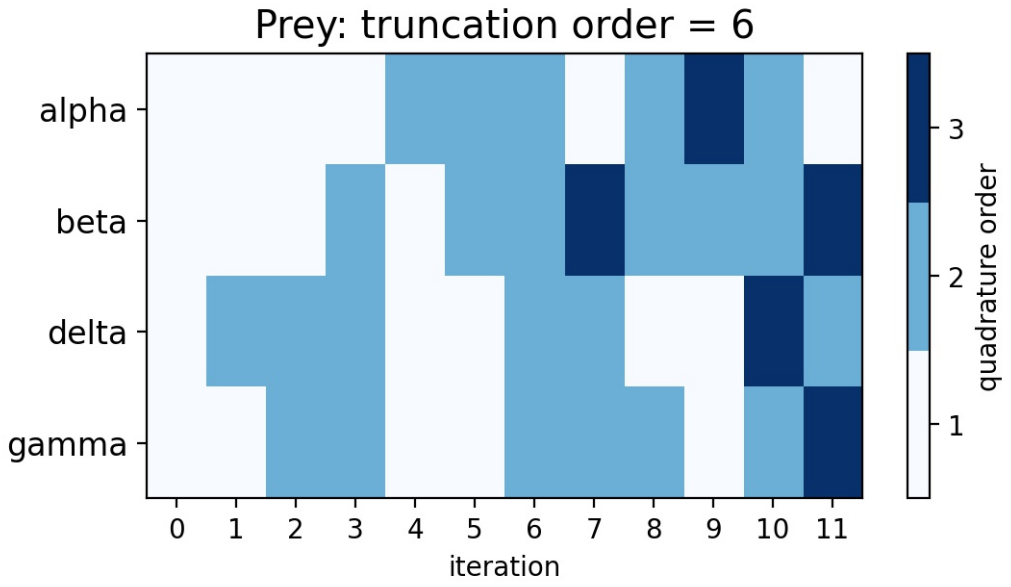


Figure 5.11: Figure showing the adaptive refinements used by the algorithm to approximate a sixth-order PCE for the two-species Lotka-Volterra model.

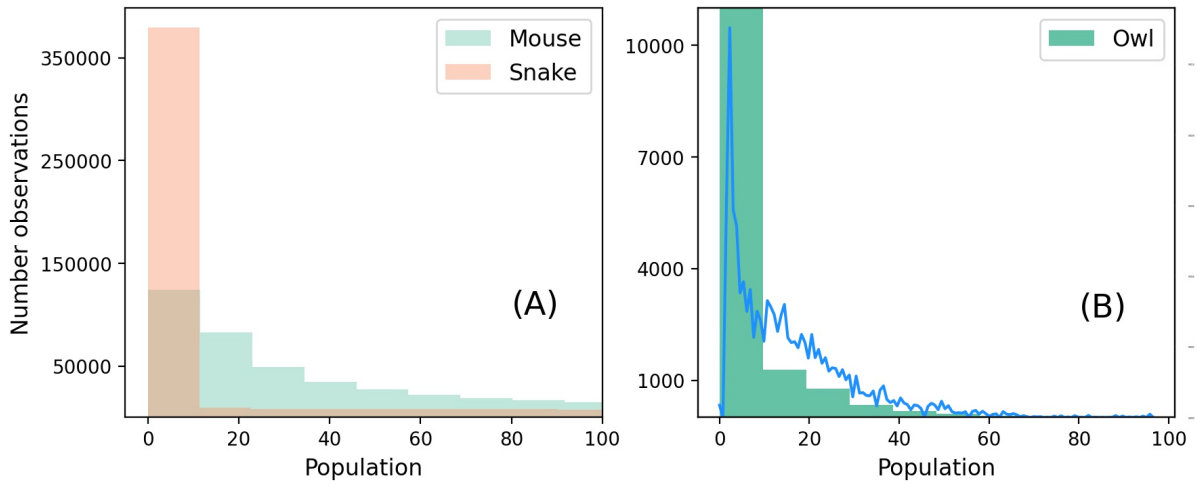


Figure 5.12: Figure showing the population distributions for each trophic level produced by the Three-Species Lotka-Volterra Model when sampled over the entire uncertainty space. Note the characteristic increase in the sharpness and skew of the distributions as the trophic level increases. This behaviour is comparable to that observed in the two-species Lotka-Volterra model albeit more severe. The owl distribution is so sharp that it frustrates attempts to clearly illustrate it, with over 99.5% of observations occurring in the bin for values between zero and ten.

5.2 THREE SPECIES LOTKA–VOLTERRA MODEL

5.2.1 Model characterisation

The inclusion of a third species is achieved by adding an additional equation to the standard Lotka-Volterra construction investigated above. The model, now composed of three interacting species, produces three quantities of interest which we store in a response vector. As above, we take the population of each species at time $t = 910$ as the QoI. The equations are specified by seven parameters, giving a seven-dimensional input uncertainty space from which samples are taken. The model is then a map $\mathbb{R}^7 \to \mathbb{R}^3$.

Numerical solution of this equation system by a third-order Runge-Kutta method produces population traces comparable to that of the two-species model, pictured in Figure 4.1 . The solutions are again periodic, with the peaks and troughs of each species separated by a phase difference from its respective prey species. The traces for the three species model, for a specific set of parameter values, are pictured in Figure 4.2. The boundary conditions for the new parameters were chosen arbitrarily and set to values which produced sufficiently well behaved solutions over the specified sampling ranges.

The plot of the model response in Figure 5.12 over the complete sampling space reveals the change in behaviour produced by the inclusion of a third species. While the distributions for the lower level species are comparable to that for the two species model, the apex predator population distribution is observed to be highly positively skewed, with the majority of its values occurring below its mean value, with a small subset of outliers with values far from the mean. This is comparable to the relationship between predator and prey observed in Figure 5.1, albeit more severe. In order to provide for interpretability the axis in Figure 5.12 has been clipped prematurely. In actuality, the bin containing the most values, that for a

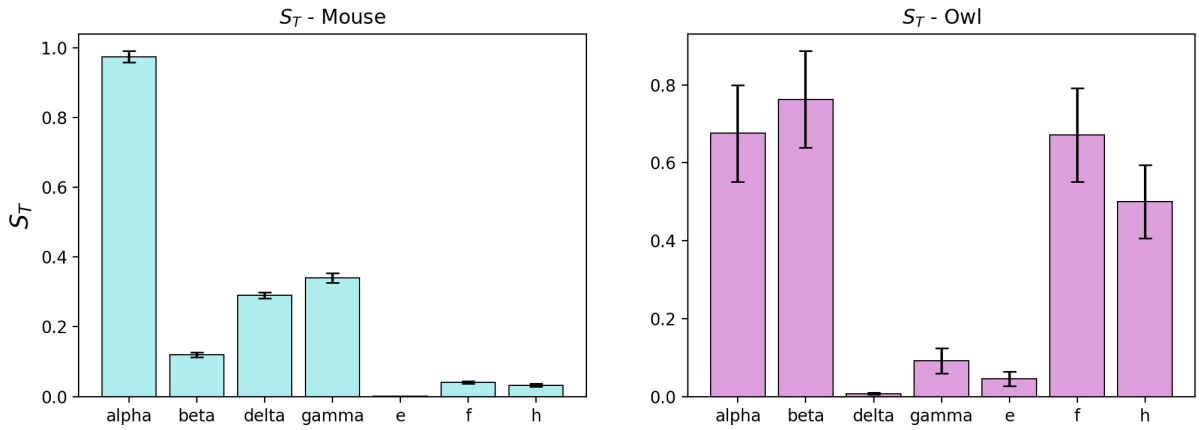


Figure 5.13: Figure showing the total-order sensitivity indices for the Three-Species Lotka-Volterra Model's lowest and highest trophic levels.

population between zero and ten, contains 99.5% of outcomes for a sample size of 524,288. The kernel-density estimation plot, pictured in blue in Figure 5.12 (B), well demonstrates the actual sharpness of the owl population distribution, with the vast majority of values lying close to zero. Note the error on the kernel-density plot, an artefact of the smoothing algorithm used in its determination. The error introduced by attempting to interpolate data with such severe steps is analogous to the issues which may arise trying to approximate the owl distribution using PCEs.

The three-species model was chosen specifically for this behaviour. The highly skewed nature of the apex predator distribution, coupled with the sharpness of its peak may prove to challenging for the smooth polynomial interpolants investigated here. Furthermore, the use of sparse grids in order to fit the given PCEs may introduce additional error. As mentioned previously, sparse grid methods leverage the assumed smoothness of the target function within the sample space, to compute a high accuracy solution with fewer samples, and function evaluations, required. In situations where this assumption does not hold, sparse grids may fail to assign enough samples to areas with large variability, thereby reducing the accuracy of the eventual interpolation.

Figure 5.13 shows the calculated total-order indices for the lowest and highest trophic levels, determined though a quasi Monte Carlo experiment with 524,288 samples taken from the 7D uncertainty space. A larger sample was required in this case in order to reduce the estimated error at the 95% confidence interval as calculated by the SAlib package. This is again indicative of both the distribution skew, sharpness, and the presence of outliers, all of which contribute to large deviations in the calculated values upon resampling. No change in the calculated confidence intervals was observed with a large number of resamples. For brevity the indices for the middle trophic level, that of the snake, have been left out of Figure 5.13.

In addition to the aforementioned sensitivity measures, a number of statistical moments characterising the population distribution of each species were calculated. Note the outsized skewness and kurtosis of the owl distribution, which indicate a sharp, asymmetric distribution with influential outliers which lie far from the mean. The mean of the owl distribution is another demonstration of the 'atto-fox' problem in biological competition models. Owl population outcomes which are less than unity are clearly un-physical but do not interfere

Moment	Mouse	Snake	Owl
Mean	68.77	32.54	0.14
StDev (normalised)	1.1	1.97	13.51
Variance (normalised)	83.3	126.47	16.47
Skew	1.33	2.11	18.99
Kurtosis	0.97	3.86	444.15

Table 5.3: Ground truth statistical moments for Three-Species Lotka-Volterra Model. Note the outsized skew, kurtosis, and stdev for the Owl.

with the structures and methods used in this study.

These figures constitute the outcomes of interest which we shall try and replicate through the use of PCEs. The three species Lotka-Volterra equations then present challenges to surrogate modelling methods on two fronts. In the first instance, the output distributions for animals at successively higher trophic levels should be more difficult to approximate accurately with PCEs. As such the more varied nature of responses produced by this model allows for a more thorough examination of PCE performance under different experimental contexts. Secondly, the progressively 'rough' nature of the output distributions may lead to a lower approximation accuracy when using sparse grids though the dimension adaptive algorithm investigated in Experiment III should be able to identify dimensions which require a higher sampling density.

5.2.2 Experiment I - Full quadrature grids

As above, in this section we only visualise the results for the lowest and highest trophic levels, those of the *mouse* and *owl*, as these lie at either end of the difficulty spectrum and, as such, allow for the best comparison.

The addition of an extra trophic level and three additional parameters increased computational times significantly. This was expected and for fulls grids follows directly from the the *curse of dimensionality*. In addition, the number of terms in each expansion, though only weakly dependent on dimensionality, also significantly slowed the computation of the Sobol indices. In light of the results from Section 5.1.2, which clearly show the extent of diminishing returns for PCEs of increasing truncation order, in this experiment we investigate expansions with truncation order less than four. Expansions of order four or less should well represent the range of mode behaviour and approximation ability. In executing the experiments we used the results from Section 5.1.2 as a heuristic to decide the maximum quadrature order required to fully fit an expansion of each order.

Figure 5.14 shows the results for the three species Lotka-Volterra model with the PCEs fit using a fully tensorised Gaussian quadrature grid. The grid levels vary from two to six with the resultant number of samples in each grid shown on the x-axis. We observe similar convergence behaviour as for the two species model, albeit with a very significant increase in the number of function evaluations required to fit each expansion. Again, the PCEs are observed to converge to maximum accuracy when fit with a quadrature rule of order $p+1$, with sharply diminishing returns in accuracy after $p = 2$. This is in agreement with the results of the Sobol-Saltelli SA which show that the majority of the variance on the mouse distribution

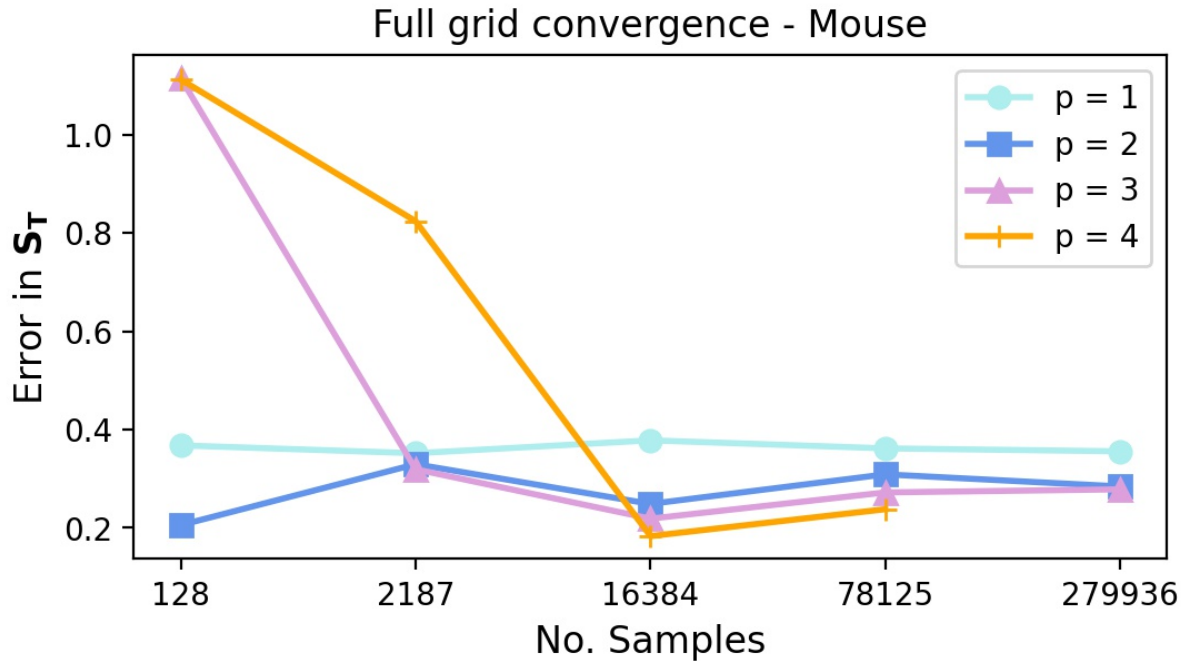


Figure 5.14: Figure showing the error for PCEs of order 1-4 fit using fully-tensorised quadrature grids of levels 2-6. Note the absence both of higher truncation order PCEs and higher quadrature grid levels. For a 7-D input uncertainty space the *curse of dimensionality* begins to limits the computations which are feasible given the available computational resources.

can be accounted for with first and second-order effects alone. The addition of higher order terms to the expansion offers little increase in approximation accuracy. It possible that the residual error upon convergence is caused by very higher order effects. While this in contrast with the Pareto heuristic verified for the two species Lotka-Volterra model it is not unlikely with a model such as this which is known for demonstrating near-chaotic behaviour.

The results for the highest trophic level show comparable convergence behaviour albeit with larger errors for expansions with lower orders. The large error for the PCE with $p = 3$ is indicative of influential higher order effects, not observed in the case of the mouse. The successive large gains in accuracy with truncation order may then be explained by the inclusion of higher order terms which capture the variability of the higher order effects. The gain in accuracy for the PCE with $p = 4$ shows that there was also significant effects of order higher than three, contributing 10% to the approximation for $p = 3$. These results are supported by the data from the Sobol-Saltelli sensitivity analysis campaign which indicate the presence of influential higher-order effects, which in the case of the owl, contributed significantly to the total-order indices. The difference between the character of the mouse and owl output distributions with respect to the influence of higher order effects is evident in the plot in Figure 5.15. The Sobol indices for the owl distribution, also pictured in Figure 5.15., are especially illustrative. For a given parameter the Pareto heuristic suggests that higher order terms account for a successively smaller amount of the total variance attributed to variation in that parameter. For the parameters α and β, the reverse appears to be true, with second-order effects accounting for only a small portion of the variance as measure by the total-order index, with higher order terms accounting for the rest. While there is no reason that a higher order expansion should not be able to capture these effects, the *curse of dimensionality* renders such endeavours infeasible. If, as we have observed, an expansion truncated to order p requires a

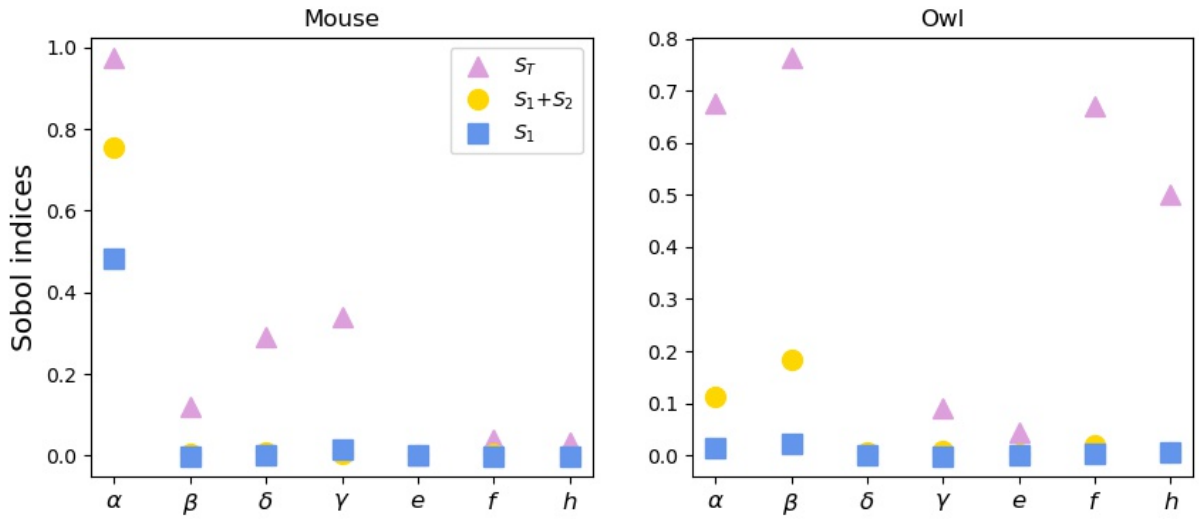


Figure 5.15: Figure showing the first, second, and total-order indices for the lowest and highest trophic levels of the Three-Species Lotka-Volterra model.

Gaussian quadrature rule of order $p+1$ to converge fully, then there is a firm practical limit on the maximum expansion order.

Such difficulty is compounded further by the scaling of the number of terms. In the case of non-intrusive spectral projection methods, where each coefficient is solved by straight forward numerical integration, the integration step is a O(n) process and therefore is insignificant with respect to the model evaluation step. In contrast, in the case of the stochastic collocation the coefficients are solved by a regression step on P equations, where P is the number of terms in the given expansion. The resulting matrix is then of size $P \times P$ which requiresO(n^2) work to solve.
Comparison with the results for the two species Lotka-Volterra model illustrates the expected difficulty in approximation caused by the characteristic 'roughness' of the owl population distribution. Although the predator population distribution showed a similarly peaked nature the sharpness was not great enough to frustrate the accuracy of the high order PCEs. In contrast, in the case of the owl for the three species model the error for the converged fourth-order expansion is larger, and does not fall below 10%.
The number of required function evaluations for the three species model is a good demonstration of the difficulties caused by the *curse of dimensionality*. The addition of only four parameters to the input uncertainty space resulted in a very sharp increase in the number of function evaluations with grid order. Although the exponential dependence of grid size on dimension is straightforward to comprehend for a mathematical standpoint is is often hard to appreciate just how quickly the number of computations required can rise. In the case of a simple differential equation model such an increase is not detrimental since the runtime is less than a second.

5.2.3 Experiment II - Sparse grids

The increase in dimensionality introduced by the three-species Lotka-Volterra model, along with the corresponding sharp increase in the number of function evaluations required pro-

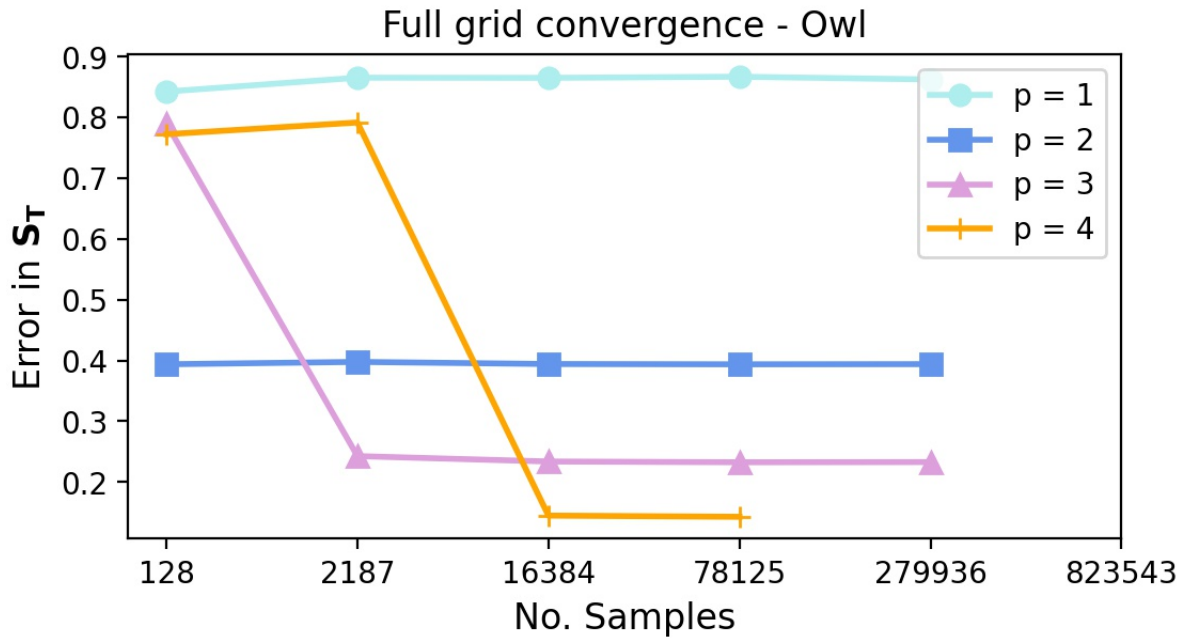


Figure 5.16: Figure showing the error for PCEs of order 1-4 fit using fully-tensorised quadrature grids of levels 2-6. The large increases in approximation accuracy gained by increasing the truncation order of the interpolating PCE are evident in the plot. This is consistent with the results displayed in 5.13 which imply the presence of significant higher-order effects.

vides an appropriate use-case for sparse grid methods. Following the logic of the *sparsity of effects* principle we can expect that the majority of the model's output variance may be explained by the variance of only a few parameters, and additionally, that lower order effects in all parameters contribute more to the output variance that higher-order effects. Sparse grid methods should then allow for faster convergence to sufficiently accurate solutions, as the points in the grid sample along dimensions with a higher density. As explained earlier, sparse grid methods rely on the assumption that that the functions to be approximated is sufficiently 'well-behaved', or smooth, in the interior of the sampling space.

Inspection of the model output in Section 5.2.1 has revealed certain characteristics that may make it difficult for sparse grid methods to yield sufficiently accurate PCEs. The highly skewed and sharp nature of the owl distribution in particular suggests *chaos-like* behaviour, with certain parameter values leading to outlier values very far from the mean. This indicates that there are 'pockets' of parameter values within the uncertainty space to which the model output, specifically, the QoI that we investigate here, is very sensitive to. Such behaviour reduces the validity of the regularity conditions assumed by sparse grid methods.

The results for the highest trophic level are illustrative of the convergence behaviour expected with sparse grids. Each of the expansion orders, illustrated in Figure 5.17, demonstrate a gradual and linear convergence with increasing grid order. Note also the relatively grouped convergence behaviour, with each expansion order above $p = 3$ converging in tandem with comparable errors. This is in contrast to the case with the full Gaussian grids investigate above, with each expansion order requiring a higher order quadrature rule to converge fully. In the this case, the sparse grids are observed to provide quite accurately fitted PCEs, with higher order expansions converging to within 20% of the ground truth values. Our comparison at this stage is hampered by the large resampling error found when calculating the ground

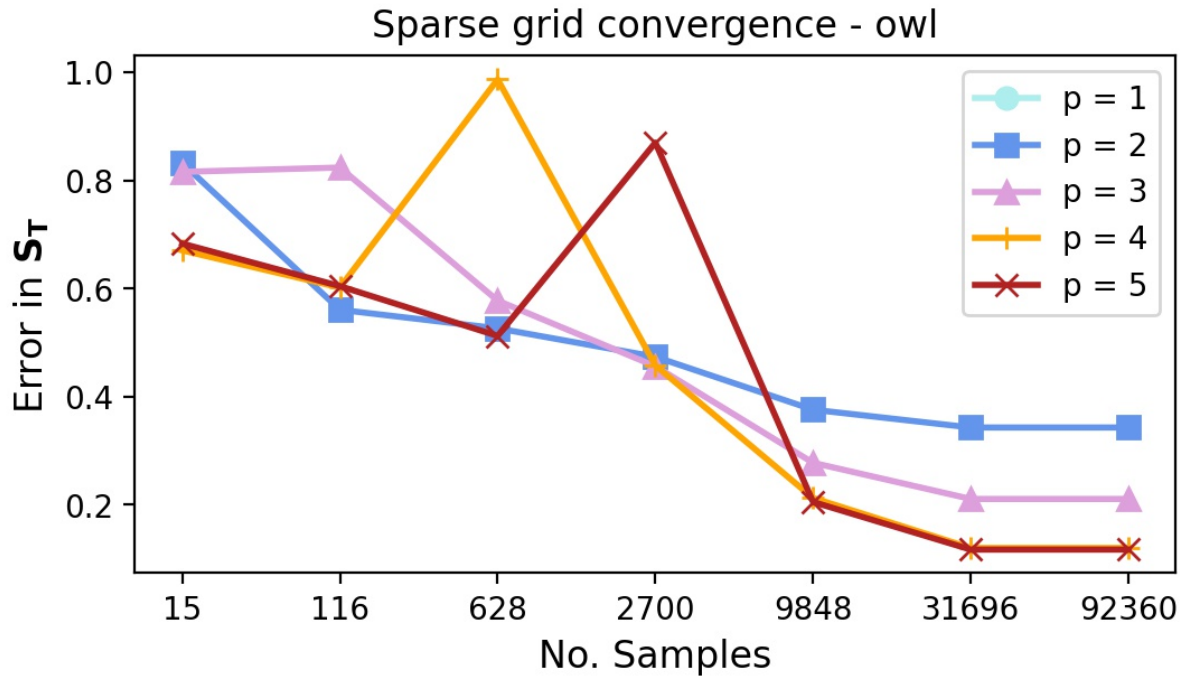


Figure 5.17: Figure showing the approximation accuracy for each PCE in Experiment II. PCEs of truncation order from 1-5 to six were fit using sparse quadrature grids of levels 1-8. Note the smoother approach toward convergence when compared to the spectral convergence observed in 5.16. Compared to the Two-Species case, the sparse quadrature grids now contain far fewer points than the fully tensorised grids, allowing higher truncation order PCEs to be fit with higher level quadrature grids. The relatively large error upon convergence was expected given the character of the owl output distribution.

truth Sobol indices. Although there is a significant envelope of uncertainty around the actual values, we nevertheless observe convergence to within the midpoint of the determined confidence intervals. There is also a clear gain in approximation accuracy for high-order PCEs, for example, $p = 4$ and $p = 3$, consistent with the observation from the model characterisation step that owl distribution is affected by significant higher-order effects. This is in contrast to the full grid where only a marginal increase in accuracy was conferred by very high orders. Comparison of the number of evaluation points produced by the sparse quadrature rules with those for the full grids in Figure 5.16 well demonstrates the eponymous sparsity of the the grids, with significantly fewer points required to attain accurate solutions.

The result for that of the lowest trophic level, that of the mouse, are consistent with that which is expected for a more 'well behaved' output distribution where higher order effects play a less influential role. The convergence behaviour is again typical of that expected from sparse grids, with a more linear convergence toward the error minima. The best performing PCE is that for $p = 2$ which converges to within 10% of the ground truth values with only 116 nodes, a significant improvement over the full grids employed in Experiment I, which required 16,384 function evaluations for the best performing PCE, which returned an error of 20%. The increased cost of fitting PCEs of too high an order for the effects which are present is also evident in Figure 5.18, with higher order expansion such as that for $p = 4$ requiring 92,360 function evaluations but converging to a similar error as that for $p = 2$.

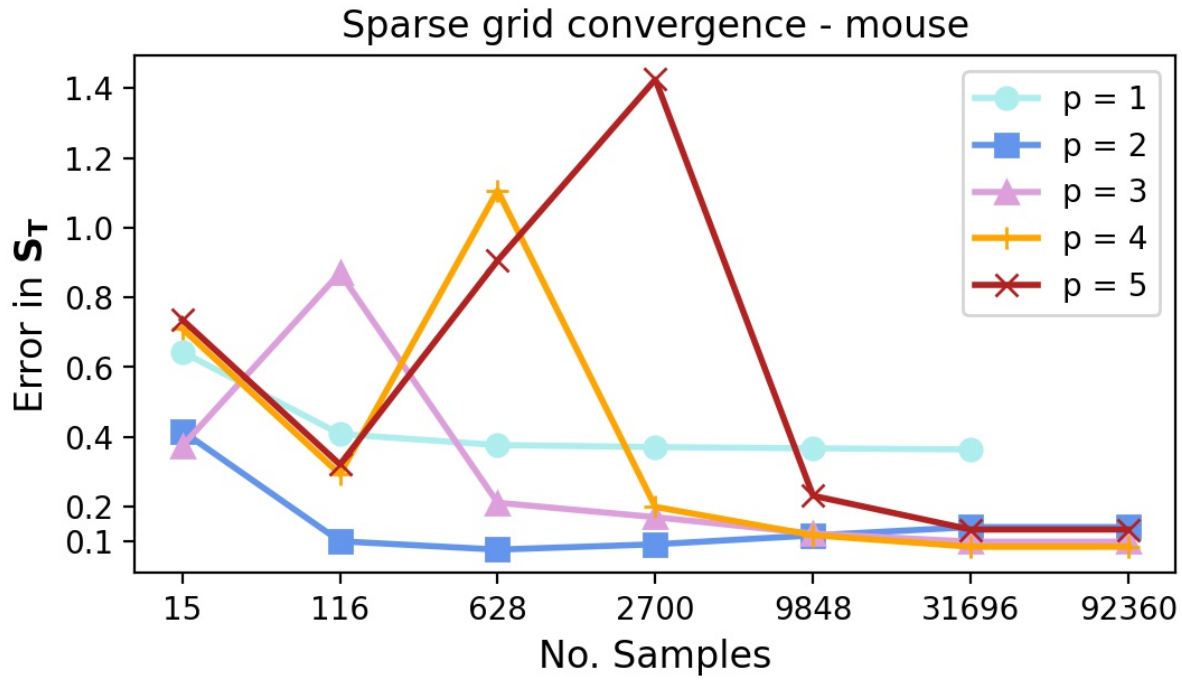


Figure 5.18: Figure showing the approximation accuracy for each PCE in Experiment II. PCEs of truncation order from 1-5 to six were fit using sparse quadrature grids of levels 1-8. The sparse grids afforded very quick convergence for PCE approximating the mouse population distribution due to its smooth character and the absence of high order effects.

5.2.4 Experiment III - Adaptive sparse grids

The results for the traditional isotropic sparse grids outlined above give some indication of the performance expected from the adaptive algorithm. For the mouse, where high-order effects are observed not to play a large role in determining the output variance, we expect relatively quick convergence to similar levels of accuracy as those shown in Figure 5.18, hopefully with fewer function evaluations. However it is with the highest trophic level, that of the owl, where the anisotropic nature of the adaptive algorithm is expected to improve performance the most. The results in Figure 5.17 show that sparse grids are capable of producing an accurate PCE for large grids. Therefore it is possible that the adaptive algorithm will provide similar accuracy with fewer function evaluations.

In both cases the dimension adaptive algorithm failed to converge for each of the investigated PCE truncation orders, and each of the trophic levels. In each case the error of the Sobol indices produced at each step did not change as the algorithm progressed and adaptively refined the sparse grid. Such behaviour was expected in the case of the highest trophic level, where influential high-order effects may not be captured by the the low order quadrature rules present in the the initial steps in the algorithm; however, the lowest-trophic level also displayed the same behaviour. To illustrate this behaviour we examine the results produced by the algorithm for the lowest trophic level with $p = 2$. Figure 5.19 shows each of the refinements taken by the algorithm, with some dimensions approximated with very high-order quadrature rules. The algorithm completed 94 refinement steps before it was manually halted. Although the quadrature grid grew to include over 2,500 nodes, the approximation error never fell below 88.7% and was essentially unchanged throughout the course of the run. An expansion of $p = 2$ fitted to the mouse population output was specifically chosen owing

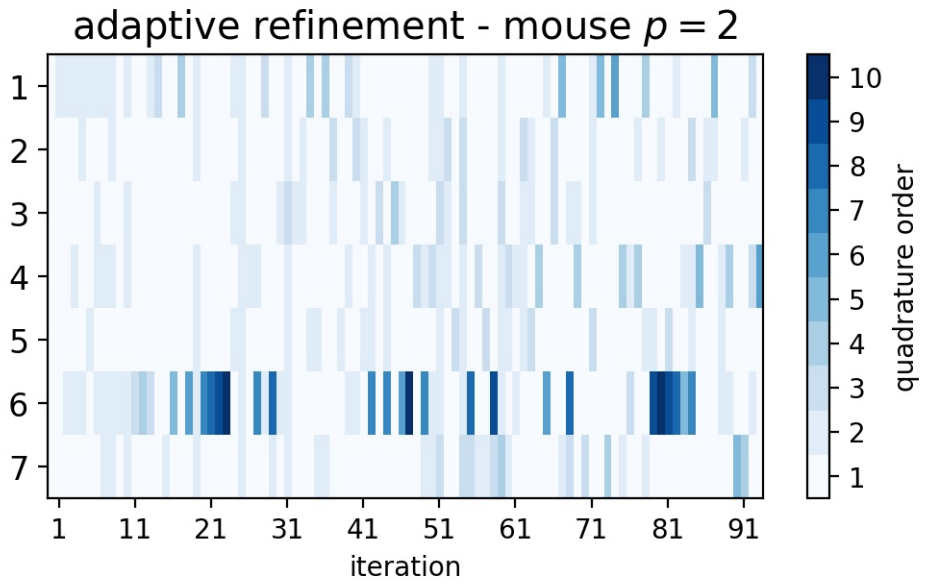


Figure 5.19: Figure showing the iterative refinement strategy taken by the algorithm in attempting to fit a PCE truncated to second-order for the lowest trophic level of the three-species Lotka-Volterra model.

to the exemplary results from Experiment II, with error of less than 10% with fewer than 200 nodes. However, in this case the result could not be replicated even with far more nodes. The same behaviour was noted for each of the other expansion orders as well as for the other trophic levels.

These results constitute a complete failure of the algorithm to accurately approximate the output of the three-species Lotka-Volterra model over the timescales afforded by the scope of this study. In each case the algorithm had to be prematurely halted as it continued to progress after a period of twelve hours with no discernible change in the error of the approximation. At time of writing no error has been found in the code which can explain such behaviour.

5.3 ENERGY TRANSITION MODEL

5.3.1 Model characterisation

For the Energy Transition Model we extract only a single quantity of interest, that of the fraction of renewables deployed at the end of the simulated period. However, this model differs from the two investigated above in that we may dynamically alter the dimensionality of the input uncertainty space by arbitrarily fixing a given number of factors. The sensitivity analysis and surrogate modelling campaigns are then carried out over the remaining variables allowing us to evaluate the effect of varying dimensionality on the performance of PCEs with a single model. The algorithms were tested for five, eight, and ten dimensional input uncertainty spaces. For the remainder of this section we use the term 'dimensions' to refer to the dimensionality of the uncertainty space rather than referencing any one dimension in particular. In each case the input space is considered to be 'high-dimensional' with computational time featuring as a significant bottleneck. Figure 5.20 shows the output distributions of the

model for each of the aforementioned input spaces, visualised using a kernel-density estimate plot.

Each of the distributions pictured in Figure 5.20 appear 'well-behaved', with low skew and low kurtosis. The inclusion of more factors in the sensitivity analysis campaign does not greatly affect the character of the output distribution aside from a slight widening the distribution. The statistical moments of each distribution are comparable, with the output mean remaining in the region of 75% renewables deployment, as illustrated by the box plot, also in Figure 5.20. Using the results from the three-species Lotka-Volterra campaign to inform our intuition, we would expect the energy-transition model output to be easier to approximate.

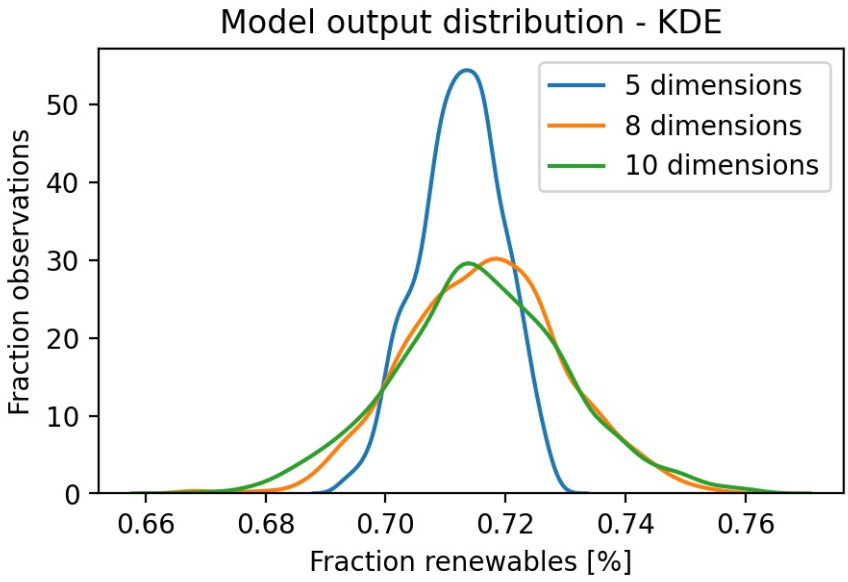
The process of factor fixing completely changes the nature of the sensitivity analysis campaign as well as the its results, therefore it is necessary to carry out individual SA campaigns on each uncertainty space. It may also be the case, due to the presence of higher-order effects between variables, that the ranking of influential factors is not preserved as we progressively fix more factors. We note here that the factors were chosen arbitrarily so as not to bias the experiment. Although a rudimentary sensitivity analysis was carried out in (Loonen et al., 2013), the results did not figure in the selection of factors for this study. The only choice made in the selection was to avoid the inclusion of categorical variables.

Figure 5.22 shows the total-order indices for each of the uncertainty spaces. The chosen uncertainty spaces are nested, in that each space contains the factors of the spaces with lower dimension as a subset. A table giving the full names of each of the factors is available in the Appendix for reference. Each of the spaces shows the distribution of effects among the factors. While some factors have an outsized influence of the model output variance, others do not contribute at all. In a situation where the modellers sought to reduce the order of the ETM, these factors, for example, 'pr_igcc', and 'pr_ngcc' for $\mathbb{R}^5$ would be fixed to their default values and excluded from further analyses.

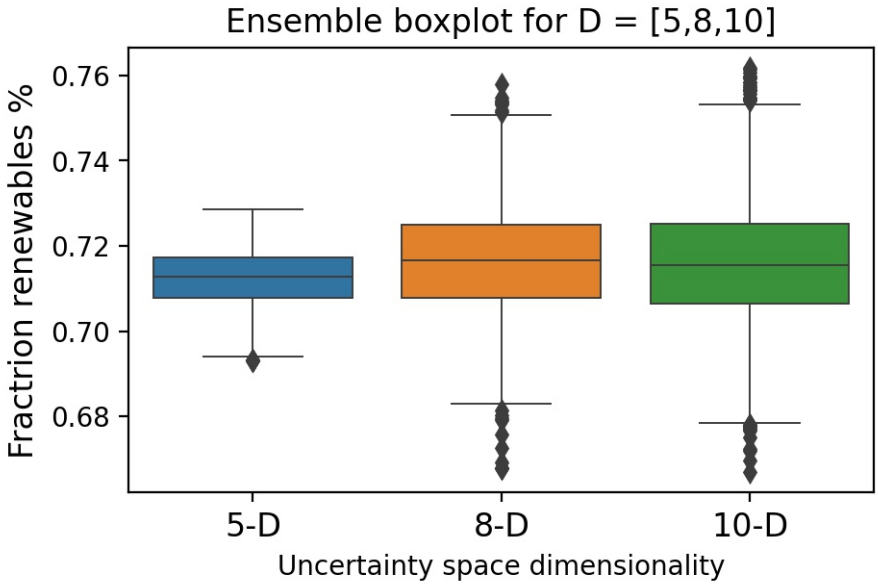
A comparison of both the first and total-order effects for each of the factors shows that few of the factors contribute to higher-order effects. In Figure 5.23, the coincidence of the purple triangles, representing the total-order indices, and the blue square, representing the first-order indices, demonstrate the near complete absence of second-order effects or higher. Only some factors, such as 'pr_nuke' in $\mathbb{R}^{10}$ and 'pr_ngcc' show marginal higher order effects. Such a result goes some way toward explaining the model's behaviour around our chosen quantity of interest.

5.3.2 Experiment I - Full quadrature grids

As expected, the output distribution of the ETM is well approximated by PCEs providing for the expedient calculation of the Sobol indices. Figure 5.24 shows the results for the uncertainty space of dimension 5. These results display the now familiar behaviour of initially high approximation error followed by convergence with expansions of order p requiring full-grid quadrature rules of level l in order to converge fully. There is observed to be effectively no increase in approximation order with increasing polynomial order. The results of the quasi Monte Carlo analysis lend support to these results. The absence of significant higher-order effects removes the benefit conferred from using higher-order expansions to approximate



(a)



(b)

Figure 5.20

Figure 5.21: Figures showing the output distributions of the ETM. The chosen quantitiy of interest is the fraction of renewables deployed at the end of the simulation period. Figure (a) shows the change in the output distribution caused by the addition of factors. Figure (b) shows the comparable means, inter-quartile ranges, minima, and maxima of the distributions produced by the model during the model characterisation step.

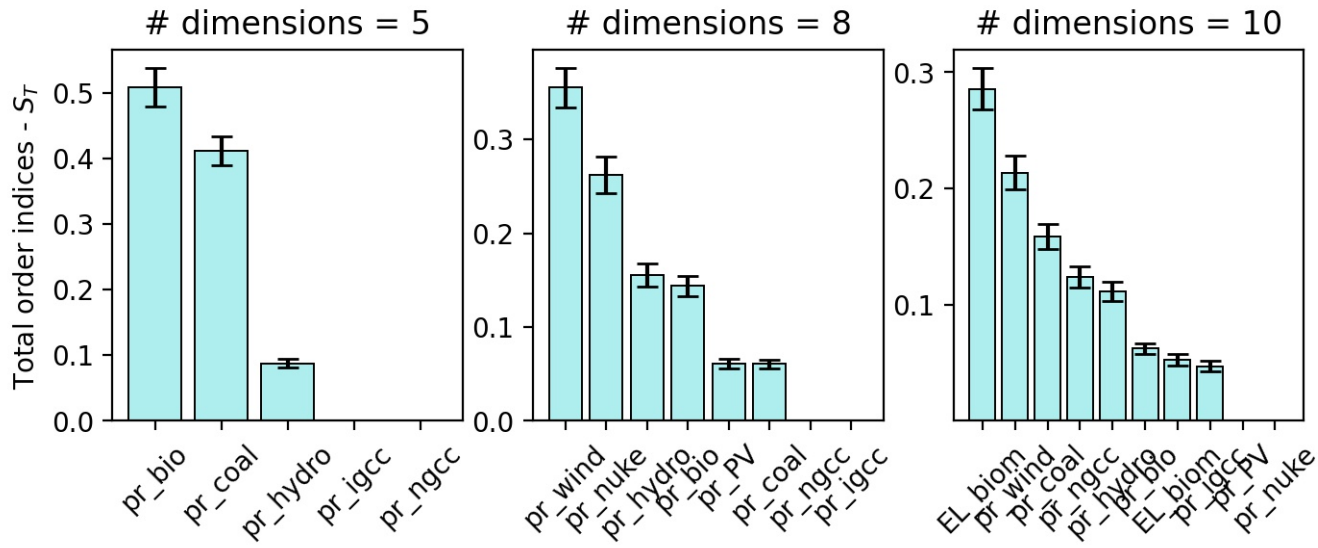


Figure 5.22: Figure showing the total-order indices determed by direct sampling for each of the investigated uncertainty spaces. Note the expected distribution of 'influence' among the parameters, with come factors contributing significantly more to the variance of the output than others. Note also how the ranking of parameters is not conserved as more factors are added to the set.

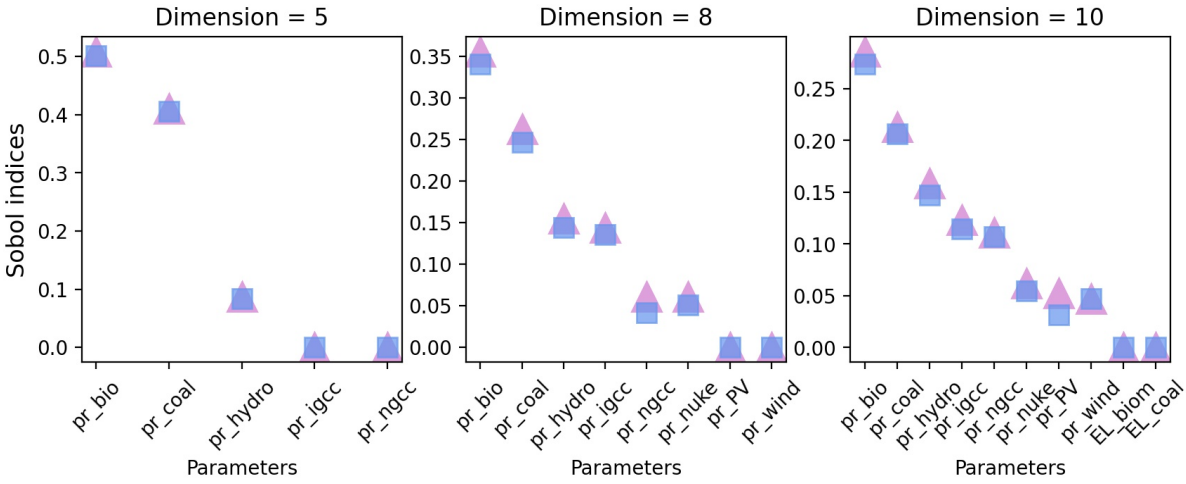


Figure 5.23: First and total-order Sobol sensitivity indices for each parameter in each of the investigated ETM uncertainty spaces. Note the coincidence of the purple triangles and blue squares, indicating the absence of higher-order effects.

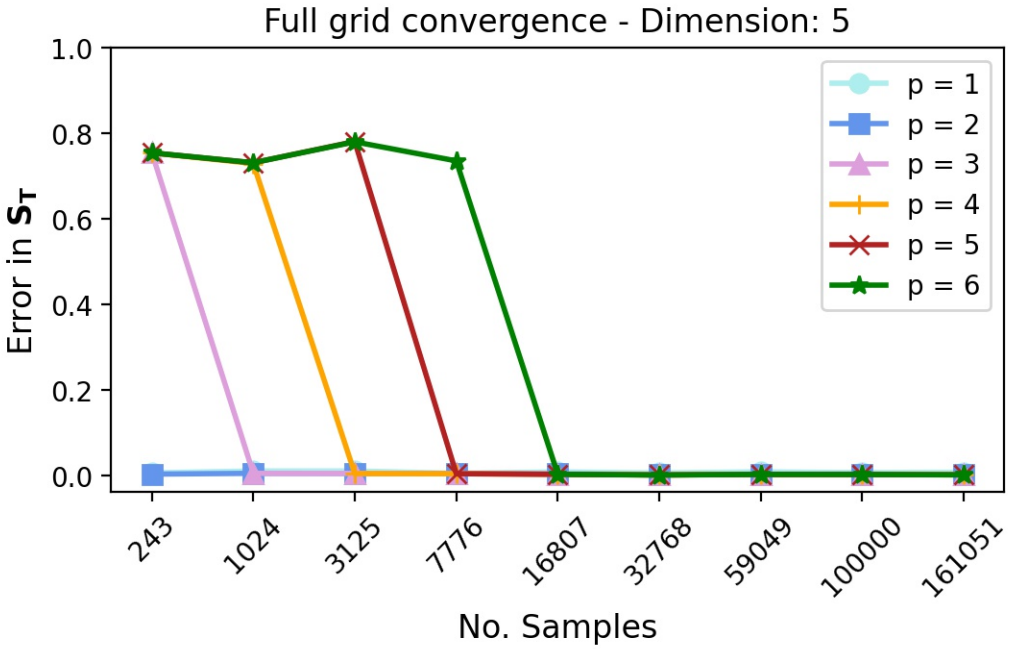


Figure 5.24: Figure showing the results from PCEs of trunction order 1-6 fitted with fully-tensorised grids of level 2-10. Each PCE is observed to converge for a quadrature grid of level $l = p$. The absence of higher-order effects means that a PCe with $p = 1$ fit using a grid of only 243 points provides a perfect approximation according to the total-order indices.

higher-order terms. For a model such as the ETM an expansion of truncation order $p = 1$ is capable of accurately modelling the response. This result well demonstrates the power of surrogate modelling techniques. Contingent on some assumptions of function regularity, such interpolative methods can approximate the full-behaviour space of a model with very few function evaluations.

Taking the expansion for $p = 1$ as an example, the algorithm converged for a quadrature rule of level $l = 2$, corresponding to a quadrature grid of $3^5 = 243$ nodes; however, the heuristic repeatedly observed throughout this study, that a PCE will converge for a quadrature grid of level $l = p$, suggests that the first-order expansion would have converged with a grid of only $2^5 = 32$ points. Nevertheless the PCE fit with only 243 function evaluations represents a very cheap method of replicating the results of the traditional qMC approach which required 24,576 function evaluations. The surrogate modelling approach has the added benefit of being able to use the fitted PCE as a cheaper surrogate for the model when carrying out new function evaluations. The severe increase in grid size with dimension as the potential to seriously limit the order of the PCEs which can reasonably be fit with finite computational resources. In the case of the ETM, where there are no higher order effects, such a finding is of little concern, but may serve to reduce the accuracy of surrogate methods for models with significant higher-order effects.

The results for the eight and ten dimensional spaces display a similar character to those above. For $d = 8$, each of the PCEs converged to a reasonably low error. Note however the residual error for first-order PCE, which is suggestive of higher order effects. The perfect convergence of the second-order PCE further suggest that these were second-order effects. Both the PCEs for $p = 2$ and $p = 3$ achieve effectively perfect convergence. Note also the marked increase

in the size of the low-order quadrature grids when compared to the case for $d = 5$. Although the reader is likely accustomed to such an increase by now, it is instructive once again to consider how the curse of dimensionality affects such methods as these. Although the low-order PCEs converged remarkably quick for $d = 5$, upon the addition of just three extra uncertain factors the cost of fitting even low-order PCEs has risen by orders of magnitude, reducing the attractiveness of such methods. What may have been a feasible approach in the former case, with respect to computational cost, is quickly being rendered unfeasible by the curse in the case of the former. For example, fitting a third-order PCE for $d = 8$ cost almost twice as much as the qMC campaign at 65,536 versus 36,864.

The number of terms in the respective expansions was also found to be an issue with increasing dimension. In the case of $d = 8$ the model was evaluated for nodes in quadrature grids of levels $2 - 7$, providing enough data to potentially fit a 6-th order PCE. However, during the actual PCE fitting step the logical processors involved in the computations raised memory errors in response to the size of the array involved in the computation. While the scaling of the number of terms which must be computed is technically a direct result of increasing dimensionality, via $P + 1 = \binom{d+p}{p}$, it is the vectorised nature of the software packages used in this study that resulted in the errors. Vectorised array operations, such as those supported by Python's NumPy library, can easily result in arrays which require over 5 GB of storage, which can be challenging for a single logical processor in a cluster or a single personal computer. Such problems could be easily mitigated by evaluating the coefficients of the expansion on a *term-by-term* basis rather than all at one.

The results for $d = 10$ further illustrate this trend with only the first and second-order PCEs converging for the given number of function evaluations. The errors upon convergence were also significantly higher for $d = 10$, with both the first and second-order PCEs only converging to within 70% of the ground truth values. Such high errors would normally be attributable to the presence of higher-order effects but in this case the results of the qMc campaign exclude that possibility. This result indicates that there is an additional factor that limits the ability of the PCEs to approximate high-dimensional response functions.

5.3.3 Experiment II - Sparse grids

We now proceed with the results from Experiment II, which sees sparse grids applied to the ETM. The ETM presents the perfect use-case for sparse grids. The combination of a smooth model output distribution, coupled with a high dimensional input uncertainty space should allow sparse grids to provide satisfactory approximations with fewer function evaluations.

The results from the lowest dimension uncertainty space, that for $d = 5$, show that the Sobol indices produced by evaluating the model at sparse nodes are consistently very accurate and demonstrate a similar convergence behaviour than that seen with the full grids employed above. In Figure 5.26 the results for $d = 5$ are illustrated. The lowest order expansion in the series is observed to perform best with an error of 5.34% when approximated using only 116 nodes. Despite an temporary decrease in approximation accuracy for higher quadrature levels, each of the expansions converges to within the region of 5% of the ground truth values. The absence of any marginal gain in accuracy with an increase in expansion order is consistent with the distribution of effects observed in the model characterisation step. The

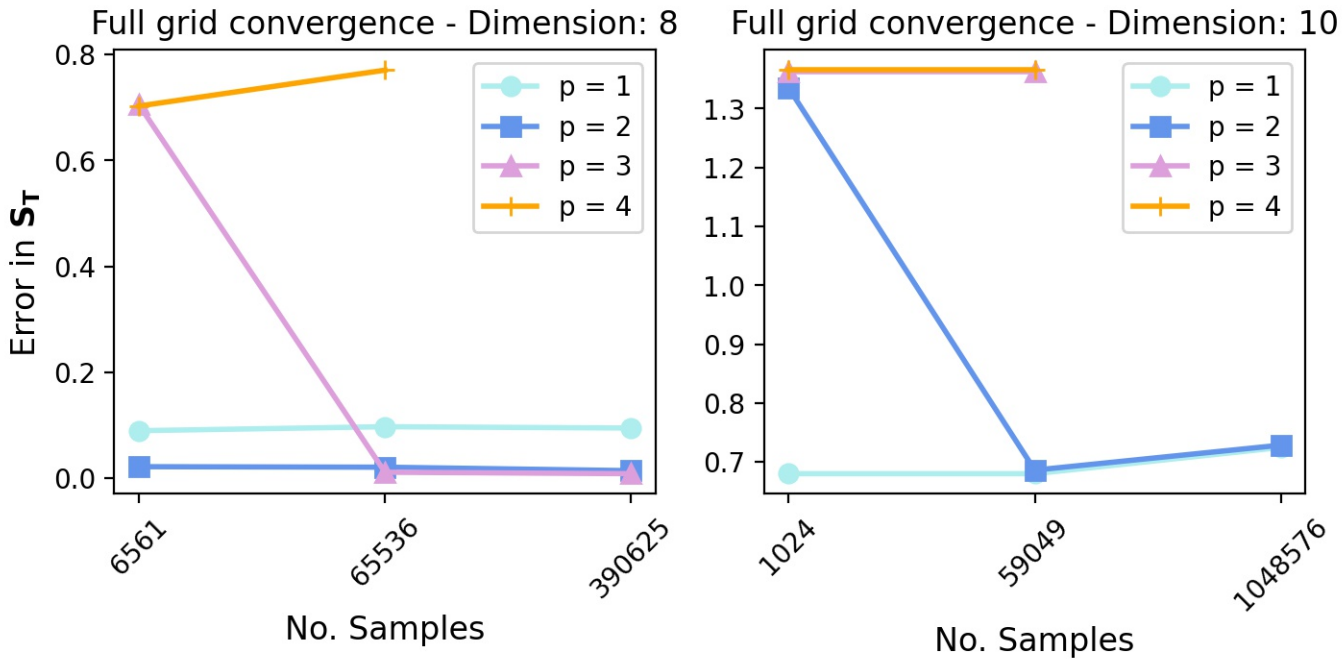


Figure 5.25: Figure showing the results from Experiment II for the 8d and 10d ETM uncertainty spaces. Not the effect of increasing dimension on the maximum truncation order and quadrature level which can feasibly be used. Both time and memory constraints served to limit the extent of the experiment.

smoothness and regularity of the model output for $d = 5$ lends itself well to approximation by sparse grids. In addition the absence of effects other than main effects implies that the characteristic sparsity of sparse grids in shared dimensions has no effect on the accuracy of the calculated Sobol indices.

The results for the higher-dimensional uncertainty spaces mirror those above, and are pictured in Figure 5.27. Each of the expansions in the series return errors within 10% of the ground truth values. The 'best value' expansion is clearly that for $p = 2$ which fully converged for only 17 nodes and returned an error of 9.15%. The lowest error in the series was measured for the third-order polynomial fit with 239,657 nodes and returned Sobol indices which were within 5.6% of the ground truth values. Although the most accurate, the marginal gain in accuracy is not justified by the huge increase in computational cost. Again the smoothness and regularity of the model output distribution is observed to be the perfect environment for sparse grids to provide savings in the required function evaluations and the results truly demonstrate the power of surrogate modelling methods.

The errors for the 10 dimensional uncertainty space are higher than expected given the success of the sparse grid methods with each of the lower dimension spaces. Every expansion in the series converges to the same residual error for the largest quadrature grid in the series containing 52,940 nodes. The minimal gain in accuracy with increasing PCE order implies that the higher error at 75% is unlikely to be improved with higher order expansions.

Both of the data sets picture in Figure 5.27 show the same increase in errors as the quadrature level of the underlying sparse grid is increased as well as a large error even for the best performing expansion in the series. This increase in error cannot be accounted for by higher-

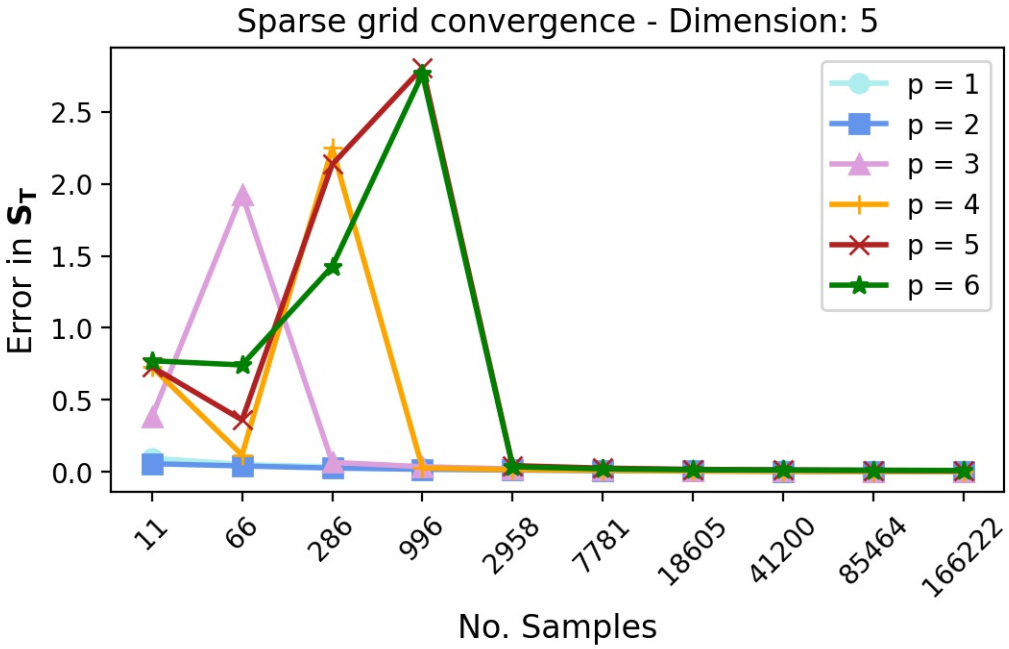


Figure 5.26: Figure showing the results from the application of sparse grids applied to the $5d$ ETM. Again, the absence of higher-order effects means that the PCE for $p = 1$ is by far the computationally cheapest option, converging with an error of 5.34% with only 11 points. Tese results are a good illustration of the circumstances under which sparse quadrature grids excel, the regularity of the model output allowed for us to replicate the accuracy of the full grid approximation with fewer points.

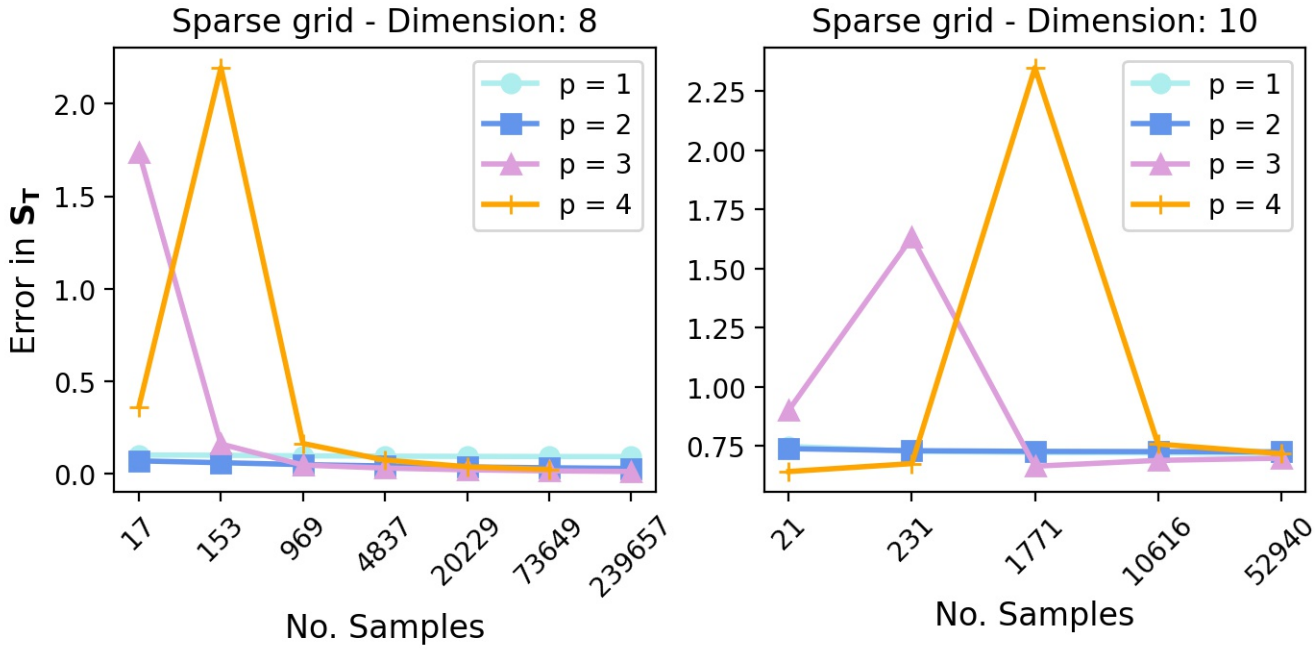


Figure 5.27: Figure showing the results from the application of sparse grids to the higher dimensional ETM uncertainty spaces. These results mirror those pictured in Figure 5.25, with a high error returned for the $10d$ uncertainty space.

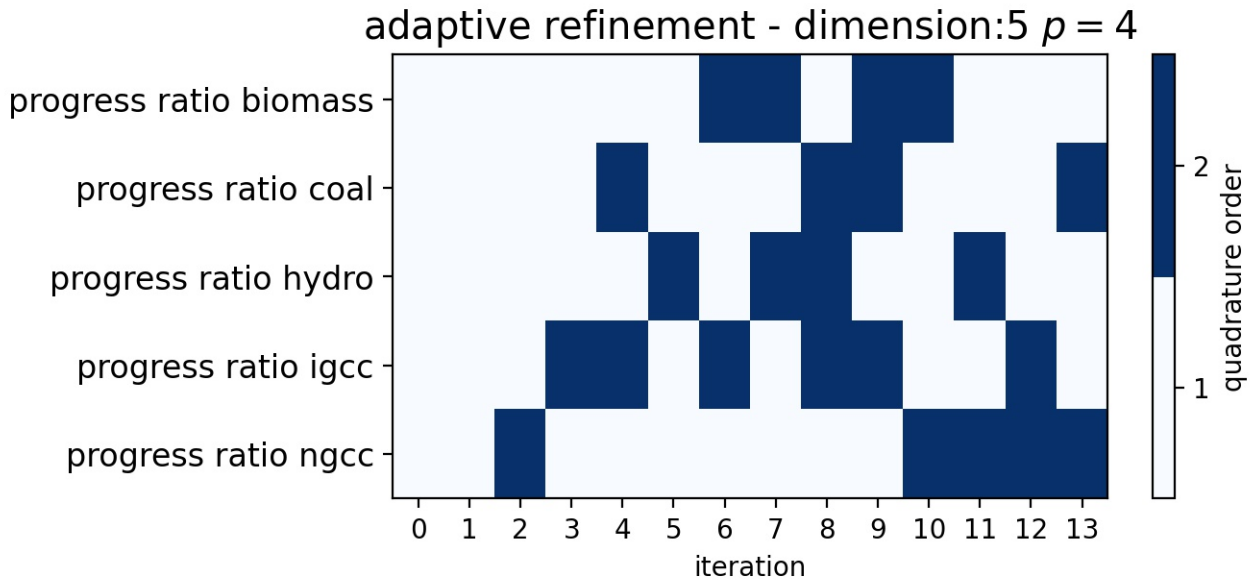


Figure 5.28: Figure showing the iterative refinement strategy employed by the algorithm in attempting to parsimoniously fit a 4th order PCE over the $5d$ uncertainty space.

order effects or by the presence of residual error from high-order expansion terms. If the trend in the graphs is global then it is unlikely that lower errors could be achieved with even higher expansion orders.

5.3.4 Experiment III - Adaptive sparse grids

The results for the dimension-adaptive sparse grid algorithm are in agreement with those illustrated above for conventional sparse grids. Although the quadrature grids constructed by the adaptive algorithm are sparse, the iterative nature of the algorithm, with refinement guided by the hierarchical surplus error function, allows it to increase the order of the grid in those dimensions for which the error is greatest. While this does result in an mostly accurate approximation with fewer nodes, given that the adaptive sparse grids are in actuality 'sparser', the results obtained for the conventional sparse grids should serve as natural upper bounds for accuracy obtainable by the dimension adaptive algorithm. However, the PCEs produced by the dimension-adaptive algorithm are significantly more accurate with far fewer nodes in the corresponding quadrature grid. Figure 5.28 displays the iterative strategy employed by the algorithm to fit a fourth-order PCE for $d = 5$. The stopping criterion was set at 20% error versus the ground truth figures. The algorithm halted after thirteen iterations with an error of only 1.2%, computed using 928 nodes. This can be considered perfect convergence and is directly comparable to the results obtained by the most accurate full grids in Experiment I at a far lower computational cost. This is also comparable to the best conventional sparse grid PCE from Experiment II, pictured in Figure 5.26, which returned an error of 1.3% with 85,464 nodes. The dimension adaptive algorithm is therefore provides the most accurate and least computationally expensive method of fitting the PCEs and thereby calculating the Sobol indices.

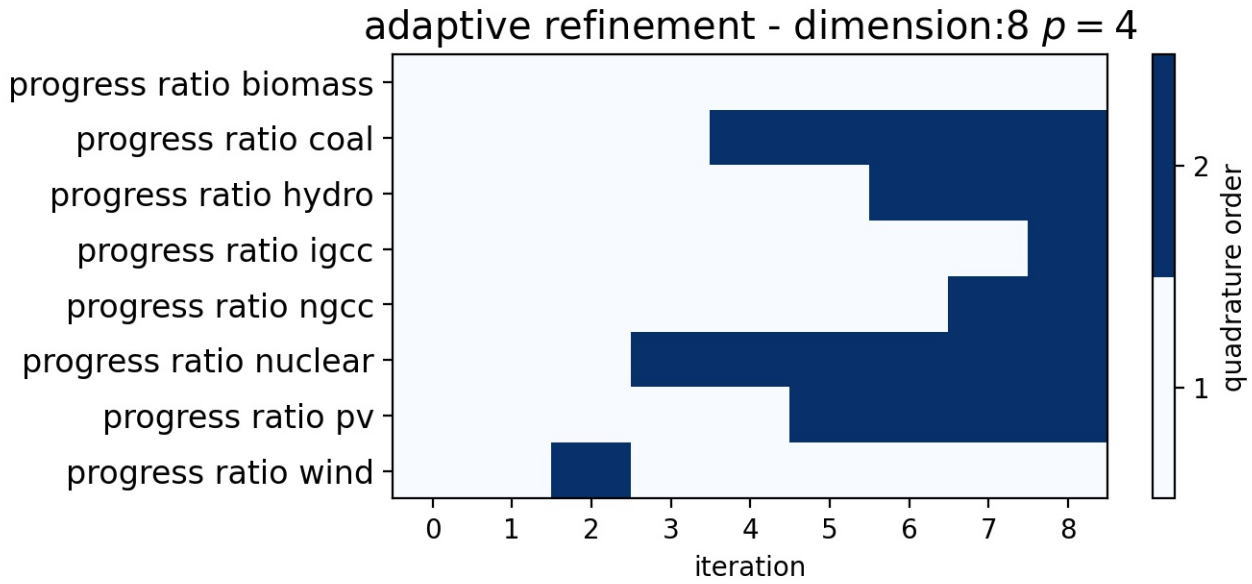


Figure 5.29: Figure showing the refinement strategy undertaken by the dimension-adapative algorithm when attempting to fit a fourth-order PCE for the eight-dimension Energy Transition Model

We cannot display the iteration plot for the PCEs of order $1, 2$, and 3 as in all three instances the algorithm halted after the zeroth step for the lowest order grid corresponding to the index $\{1, 1, 1, 1, 1\}$. For these truncation orders the algorithm returned errors in the total-order Sobol indices of between 2.1% and 2.2%. Although slightly less accurate than the result for $p = 4$ they were achieved at a substantially reduced computational cost at only 32 nodes.

The algorithm also halted at the zeroth step for truncation order $1, 2$, and 3 with $d = 8$ although the errors upon convergence were slightly higher than for $d = 5$ at between 6% and 10%. In the case of $d = 8$ the lowest order quadrature grid, again corresponding to an index of the form $\{1, 1, 1, \cdots, 1\} \in \mathbb{R}^8$ contains 256 nodes. Again the results returned by the adaptive algorithm far outperform the results from the failed conventional sparse grid campaign. Figure 5.29 shows the refinement strategy employed by the algorithm to fit the PCE for $p = 4$, for which the algorithm had to be manually halted due to time constraints. Given the extremely low errors returned by the algorithm for the lower expansion orders it is highly likely that the algorithm would eventually meet the halting condition and return a low error. Nevertheless in the context of this study, where computational time was limited, it had to be halted prematurely.

We may attribute the poor performance of the algorithm in this case to two factors. First, the *curse of dimensionality* naturally increases the number of computations required at certain steps in the algorithm, thereby significantly increasing the time for each iteration. Second, the computational cost of the look-ahead step, which calculates the local error measure, and chooses the dimensions to refine, starts to become an issue for high dimensions. For the quadrature grids arising from the indices in the Active Index Set, each new node must be evaluated. Due again to the *curse* this step can involve more computations than there are nodes in the Old Index Set. While these nodes do not have to be evaluated again they nevertheless add to the computational time and make even the algorithm unfeasible when computational resources are limited. This fact has no impact on the general applicability of such algorithms since it is

straightforward to parallelise the process.

For the 10-dimensional uncertainty space the algorithm also met the halting condition after the zeroth step and converged with an error of 7.85% for 1024 nodes and a PCE of $p = 1$, a very significant improvement over the results from Experiment II. The computational difficulties arising from the large number of terms in each expansion were naturally more severe with a 10 dimensional uncertainty space, and as such the adaptive algorithm had to be artificially halted for $p = 2$ after it took over 8 hours to reach the fourth iteration.

5.4 CONCLUSIONS

From the experiments carried out over the course of this study, and explained in detail above, a number of conclusions can be drawn as to efficacy of polynomial chaos expansions in the determination of Sobol sensitivity indices.

For each of the models investigated here, polynomial chaos expansions were found to significantly outperform the incumbent, quasi Monte Carlo direct sampling methods by determining the total-order Sobol indices to a sufficiently high degree of accuracy versus the established ground truth values. Since the models used in this study do not admit analytical solution, no conclusion can be made as to the absolute accuracy of the solutions provided by the PCEs; however, when judged against the values determined by qMC methods they were found to yield acceptable solutions with far fewer function evaluations.
In the case of the two-species Lotka Volterra model, the PCEs were observed to converge when fit using a quadrature grid of level $l = p$ and demonstrated sharply diminishing returns in accuracy with the increase of expansion order. The link between the effects present in the model and the order of the terms in the PCE was established for this model, with the marginal increases in accuracy with increasing expansion order found to closely follow the distribution of higher order effects in the model itself. This result illustrates a perfect correspondence between the surrogate modelling methods and the qMC methods for the two-species model. In the case of fully tensorised quadrature grids, a clear trade-off was identified between approximation accuracy and computational cost. While higher order PCEs are demonstrably more accurate than those of lower order, the marginal gain in accuracy does not justify the increased computational cost. The Pareto heuristic espoused by (Box and Meyer, 1986) was found to be appropriate for determining the expansion order which delivered the highest accuracy for the lowest computational cost. A third-order PCE fit converged for only 256 function evaluation and returned Sobol indices within 10% of those determined by the qMC campaign at the significantly higher cost of 327,680. This result represents a decrease in computational cost of three orders of magnitude and well demonstrates the power of surrogate modelling methods for lower dimensional problems. Sparse grids were also observed to perform well for the two-species model and behaved as expected, with a more linear and smoother progression toward convergence. They were also observed to be relatively more expensive than the fully tensorised grids in determining accurate solutions although this arose from the low-dimensionality of the uncertainty space for the two-species model with allowed for extremely fast convergence of the full grids.

The three-species Lotka Volterra model was chosen for its replication of '*chaos-like*' sensitivity to conditions found in natural predator-prey systems. The inclusion of a third species and the corresponding increase in the dimensionality of the input uncertainty space presented a

more challenging test problem for the methods investigate here. The three-body interactions in the model produced very sharp and skewed output distributions which are known to cause issues for sparse grid methods in particular. The fully tensorised grids performed similarly albeit with higher errors upon convergence. The *curse of dimensionality* also began to affect the tractability of the problems, both in terms of the function evaluations required and the computational burden of evaluating so many expansion terms. Despite the higher errors, the full grids were still found to confer a large advantage in speed over the qMC methods which returned results with a high resampling error. The sparse grids were observed to outperform the full grids in providing more accurate solutions at a lower computational cost, the dimensionality of the uncertainty space now sufficient to make them relatively much more 'sparse' than the fully tensorised grids.

The Energy Transition model was chosen to represent the kind of policy models with which surrogate modelling methods may find use in the field of Exploratory Modelling. It further offered an opportunity to test the performance of the methods for much higher dimension spaces. The early identification of absent higher order effects contributing to the variance of the model output suggested that the ETM was a perfect candidate for surrogate modelling. Both the fully tensorised and sparse grids provided near perfectly accurate Sobol indices for $d = 5$ and $d = 8$, converging to within 10% of the ground truth values with fewer than 20 function evaluations. The accuracy of the PCEs was found to suffer for $d = 10$ with unexpectedly high errors considering the success of the methods on the lower dimension space.

Finally the adaptive sparse grid algorithm, adapted from the work of (Gerstner and Griebel, 2003) was found to provide both remarkable increases in accuracy and speed in some of the model context investigate within this study. Since conventional sparse grids provide rough upper bounds on the accuracy attainable by the 'sparser' adaptive grids, the adaptive algorithm performed well in those instances where conventional sparse grids were able to provide accurate solutions.

6 DISCUSSION AND CONCLUSION

6.1 ANSWERING THE RESEARCH QUESTIONS

In the following section we provide a brief summation of the project by assessing the extent to which we have answered the research question initially posed in Chapter 1.

> **MRQ:** How can polynomial chaos expansions with dimension adaptive sampling be applied to decrease computational cost in exploratory modelling projects?

We further decomposed the main research question into four sub-questions which we will no address briefly in turn.

> **1.** What are the relevant features of polynomial chaos expansions and how may they be employed in exploratory modelling projects?

Polynomial chaos expansions are capable of approximating any second-order random variable as a series expansion of orthogonal basis polynomials in normalised random variables. In this respect, they can be used to approximate the output of a stochastic model where the output distribution itself may be treated as a random variable. In the field of exploratory modelling, where deterministic models are treated as stochastic due the presence of irreducible uncertainty around input parameters, polynomial chaos expansions therefore find use as surrogate models which are capable of reproducing the behaviour of a given model without having to run the actual model for a given parameter set. Such a use-case is important in reducing the computational intensity of exploratory modelling projects. Additionally, if the uncertain model parameters themselves are used as as an orthogonal basis over which to decompose the model output, then the polynomial chaos expansion is equivalent to Sobol's decomposition and, as such, all of the Sobol indices for the model can be calculated in a post-processing step. Since polynomial chaos methods are an interpolation method, they can be fit to a high degree of accuracy with few model evaluatations. PCEs are therefore an efficient way of executing sensitivity analysis campaigns on models for which there is parametric or deep uncertainty.

> **2.** How can dimension-adaptive sampling be used to facilitate faster convergence of polynomial chaos expansions?

Although polynomial chaos expansions provide for the more expedient calculation of statistical moments and Sobol sensitivity indices than direct-sampling methods, they are not immune from the *curse of dimensionality*. Sparse grids may be used to reduce the computational cost of fitting PCEs to high dimensional models though sparse grids may also become prohibitively expensive for high-dimensional uncertainty spaces. Dimension adaptive sparse

grids, first developed by Gerstner *et al.* for the evaluation of high dimensional integrands, may be used to further reduce the exponential increase in computational cost imposed by the *curse of dimensionality*. This is achieved by increasing the density of the sampling plan in only those dimensions which contribute significantly to increasing the accuracy of the PCE through the use of a local error measure.

3. Which measures of performance may be used to evaluate the efficacy of these methods in exploratory modelling projects?

The efficacy of the methods described above depends primarily on two factors: their speed and their accuracy. They can only be used in place of traditional sensitivity analysis methods if they are capable of attaining reasonable accuracy with diverse models and they must also determine the sensitivity metrics with fewer function evaluations to justify their use. In reality, deciding on the appropriate expansion order of the PCEs, and the level of the quadrature grid required to fit them, introduces a trade-off between accuracy and computational cost which must be addressed.

4. What is the relative efficacy of these methods versus other *best-in-class* algorithms?

Polynomial chaos expansions were found to converge significantly faster than traditional quasi Monte Carlo methods, producing accurate solutions with far fewer function evaluations, often converging to within 10% of the ground truth values with orders of magnitude fewer function evaluations. Nevertheless the *curse of dimensionality* makes the use of fully tensorised quadrature grids infeasible for higher dimensional spaces. In these instances sparse grids were found to provide for the expedient fitting of the PCEs with fewer function evaluations for higher dimensional spaces owing to their characteristic sparsity. The dimension adaptive sparse grid algorithm was found to confer yet further gains in both approximation speed and accuracy in those situations where conventional sparse grids also provided reasonable approximations albeit with larger grids. Although the dimension adaptive algorithm is the most parsimonious methods of fitting such PCEs for sensitivity analysis it is still subject to the curse of dimensionality which can limit it application for very high-dimensional spaces ($d > 10$) or where computational resources are limited to a single logical processor.

6.2 POLYNOMIAL CHAOS EXPANSION

The results presented in this thesis conclusively show that polynomial chaos expansions are capable of significantly reducing the computational cost associated with exploratory modelling projects, under certain contexts. In Chapter 3 we derived the correspondence between the coefficients of a PCE and Sobol's decomposition of model variance, thereby showing how PCEs could directly be used to execute sensitivity analyses, a vital sub-method of Exploratory Modelling. In addition, as a surrogate modelling technique, PCEs were theoretically capable of determining Sobol sensitivity indices', and other measures of interest, much faster than tradition Monte-Carlo or quasi Monte-Carlo experimental designs.

Over a set of three test-cases, each with significantly different behaviour, we tested the assumed ability of PCEs to support sensitivity analyses by computing Sobol indices, and compared their accuracy against results produced by quasi Monte-Carlo methods. Furthermore,

we tested the approximation accuracy of varying truncation orders when fit using quadrature-based sampling plans of varying size. We found that, in general, PCEs, fit using fully tensorised quadrature grids, were capable of accurately reproducing Sobol sensitivity indices at a fraction of the computational cost associated with qMC methods; however, their performance was affected by a number of factors.

In cases where only low-order effects had a significant impact on the variance of the model output, PCEs of low truncation orders were capable of fully reproducing the output behaviour of the model. Given the finding that a PCE of truncation order p requires a fully tensorised Gaussian quadrature grid of level $l + 1$ in order to converge, low truncation order PCEs are significantly 'cheaper' to fit than those of higher truncation orders. This is exemplified by the case of the 5d Energy Transition Model, whose output was observed in the model characterisation step to be determined mainly by first-order effects. As such, the variability of the model output could be captured completely by a first-order PCE which, furthermore, would converge for a quadrature grid containing $2^5 = 32$ nodes, three orders of magnitude fewer function evaluations than required for the qMC experiments. Increasing the truncation order to $p = 2$ yielded an even better approximation, with an error versus the ground truth values of 0.4% with only 243 function evaluations, a sure testament to the power of surrogate modelling techniques for models with low order effects. In cases for which higher order effects played a large role in determining the output variance, increasing the truncation order of the interpolating PCE resulted in successively more accurate approximations as successively higher-order interaction effects were captured by the PCE. However the gains in accuracy were, in most cases, observed to be quickly diminishing, with the gains in accuracy often not justifying the necessary increase in function evaluations required to fit the higher truncation order PCE. This is a general consequence of the heuristic introduced by (Box and Meyer, 1986) which posits that the influence of higher-order effects approximately follows a Pareto distribution. Therefore we found PCEs of truncation order $p = 3$ sufficient to address this trade-off between accuracy and computational cost, capturing the majority of the effects while minimising computational costs. A closer analysis of the distribution of effects for the Two-Species Lotka-Volterra Model showed they exactly followed a Pareto distribution, with effects up to order two sufficient to explain 80% of the model variance, and the inclusion of third-order effects sufficient to explain 96% of the model output variance. This result was replicated almost exactly by analysing the approximation error of PCEs of varying truncation order, with a third-order PCE returning an error of 6%, thereby replicating 94% of the model variance.

Although extremely efficient for models with relatively low-dimensional input uncertainty spaces ($\approx d = 5$) the *curse of dimensionality* begins to affect the computational feasibility of PCEs for higher dimensional spaces, especially for those models with influential higher-order interaction effects. The difficulty was observed on two fronts. Firstly, the requirement of a quadrature grid of level $l = p$ to fit a PCE of truncation order p introduces an exponential increase in required function evaluations. Secondly, the fully vectorised nature of the software packages used in this study lead to problems with memory allocation, with the intermediate data structures routinely becoming larger than 5 GB.

6.3 SPARSE GRIDS

Sparse quadrature grids were investigated as a method by which to reduce the number of function evaluations required to fit a given PCE, hopefully while retaining approximation accuracy. Although sparse grids help reduce the effect of the *curse of dimensionality* they do not escape it completely, but merely postpone it to higher dimensions. The convergence ability of sparse grid methods is highly dependent on the assumption of regularity of the model output. The test cases in this study were specifically chosen since they exhibited behaviour which do not meet the criterion of function smoothness required by sparse grid methods. This is exemplified by the case of the Owl output distribution in the Three-Species Lotka-Volterra Model, whose sharp, and highly skewed output distribution, with many outliers far from the mean, provided a perfect test-case.

The results from the sparse grid experiments fully agree with the our expectations. In the case of the aforementioned Owl output distribution, the sparse grids were observed to converge slower and with higher residual errors upon convergence than the more well-behaved mouse output distribution. Nevertheless the low errors achieved exceeded expectations given the form of the distribution, with the PCE for $p = 4$ converging to within 10% of the ground truth values.

The convergence behaviour of the sparse grids was also as expected, with a more gradual, linear progression toward convergence observed when compared with fully tensorised grids. The slower convergence of sparse grids introduces some nuance as to the decision about whether to use full or sparse grids to approximate a given model. If a model can be well approximated by a low-order PCE, and is relatively low-dimensional ($\approx d = 5$), then using a fully tensorised quadrature grid will be significantly quicker and produce a result about which the error is assured (for $l = p + 1$). However as the dimensionality of the input uncertainty space increases, the weaker dependence of sparse grid size on dimension allows them to outperform full grid methods, yielding fairly accurate solutions with far fewer function evaluations. This is exemplified by the three-species Lotka-Volterra equations, where the use of sparse grids resulted in an order of magnitude reduction in computational cost and a doubling of accuracy in the case of the mouse.

Naturally, the above considerations on whether to choose a full or sparse grid to serve as the basis of interpolating a model's output is highly dependent on external factors, such as whether their is sufficient knowledge of the regularity of the model output, and the computational resources available to the project.

6.4 DIMENSION-ADAPTIVE SPARSE GRIDS

Our results show that adaptive sparse grids, constructed using a modified version of (Gerstner and Griebel, 2003)'s original algorithm, have the potential to significantly reduce the computational cost associated with PCE methods. In situations where the regularity assumptions of sparse grid methods hold, the adaptive refinement strategy of the algorithm will produce accurate solutions at a fraction of the computational cost; however, they are not a panacea, and are still affected by the *curse of dimensionality*. The *curse* is particularly present in the 'look-ahead' step during which the algorithm calculates and assigns the local error measure to each multi-index in the active set. For models where the run time of each model

instance is not a significant bottle neck, the look-ahead step of the algorithm may actually be slower than computing model evaluations in parallel for a one-shot sparse grid experiment. This was observed in the case of the three-species Lotka Volterra model, where the presence of significant higher order effects necessitated the sampling of shared dimensions. This implies that arriving at an accurate approximation will require many iterations of the algorithm, with a very large active set for which errors must be calculated. In contrast, for models with significant run times, the overhead of the look-ahead step will be likely be negligible compared to the cost evaluating the model.

While we were able to directly compare the efficacy of dimension-adaptive methods against both sparse and full quadrature grids, we were unable to field the algorithm in the situations where it can confer the largest benefit, for those models with very high dimensional input uncertainty spaces $\approx d \geq 10$. We found that the look ahead step on which the algorithm relies was especially sensitive to the increases in dimensionality. While the iterative nature of the algorithm, with all dimensions initially sampled with low-order rules, is definitely more efficient than full or sparse grid methods, it may still require a computing cluster to complete in a feasible amount of time.

6.5 RECOMMENDATIONS FOR FUTURE RESEARCH

To better leverage the potential benefits of dimension-adaptive sparse grids for use with PCEs, more research should undertake to evaluate its performance with large sensitivity analysis campaigns. In order to execute such a study it would be necessary to further refine the form of the dimension adaptive algorithm to employ parallel computing wherever possible. While it would be unfortunate to conclude that the use of these methods is out of reach for those without access to HPC resources, perhaps the algorithm could be refined to allow for its use on modestly sized clusters.

Additionally, since the performance of the algorithm is dictated in large part by its error function, more research is required on novel error measures which are suited to the problems encountered in exploratory modelling. Novel error measures may come with the added benefit of reducing the computational burden associated with the look-ahead step and provide for more expeditious computations.

BIBLIOGRAPHY

Bankes, S. (1993). Exploratory modeling for policy analysis. *Operations research*, 41(3):435–449.

Box, G. E. and Meyer, R. D. (1986). An analysis for unreplicated fractional factorials. *Technometrics*, 28(1):11–18.

Brumm, J. and Scheidegger, S. (2017). Using adaptive sparse grids to solve high-dimensional dynamic models. *Econometrica*, 85(5):1575–1612.

Bungartz, H.-J. and Dirnstorfer, S. (2003). Multivariate quadrature on adaptive sparse grids. *Computing*, 71(1):89–114.

Bungartz, H.-J. and Griebel, M. (2004). Sparse grids. *Acta numerica*, 13:147–269.

Crestaux, T., Le Maıtre, O., and Martinez, J.-M. (2009). Polynomial chaos expansion for sensitivity analysis. *Reliability Engineering & System Safety*, 94(7):1161–1172.

Dwight, R. P., Desmedt, S. G., and Omrani, P. S. (2016). Sobol indices for dimension adaptivity in sparse grids. In *Simulation-Driven Modeling and Optimization*, pages 371–395. Springer.

Edeling, W. (2022). Adaptive sparse-grid tutorial.

Edeling, W., Arabnejad, H., Sinclair, R., Suleimenova, D., Gopalakrishnan, K., Bosak, B., Groen, D., Mahmood, I., Crommelin, D., and Coveney, P. V. (2021). The impact of uncertainty on predictions of the covidsim epidemiological code. *Nature Computational Science*, 1(2):128–135.

Fang, H., Gong, C., Su, H., Zhang, Y., Li, C., and Da Ronch, A. (2019). A gradient-based uncertainty optimization framework utilizing dimensional adaptive polynomial chaos expansion. *Structural and Multidisciplinary Optimization*, 59(4):1199–1219.

Gerstner, T. and Griebel, M. (2003). Dimension–adaptive tensor–product quadrature. *Computing*, 71(1):65–87.

Hamarat, C., Kwakkel, J. H., Pruyt, E., and Loonen, E. T. (2014). An exploratory approach for adaptive policymaking by using multi-objective robust optimization. *Simulation Modelling Practice and Theory*, 46:25–39.

Homma, T. and Saltelli, A. (1996). Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1):1–17.

Kinoshita, S. (2013). Introduction to nonequilibrium phenomena. In *Pattern Formations and Oscillatory Phenomena*, pages 1–59. Elsevier.

Koekoek, R. and Swarttouw, R. F. (1996). The askey-scheme of hypergeometric orthogonal polynomials and its q-analogue. *arXiv preprint math/9602214*.

Kwakkel, J. H. and Pruyt, E. (2013). Exploratory modeling and analysis, an approach for model-based foresight under deep uncertainty. *Technological Forecasting and Social Change*, 80(3):419–431.

Li, Y., Li, H., and Wei, G. (2019). Dimension-adaptive algorithm-based pce for models with many model parameters. *Engineering Computations*.

Loonen, E., Pruyt, E., and Hamarat, C. (2013). Exploring carbon futures in the eu power sector: Using exploratory system dynamics modelling and analysis to explore policy regimes under deep uncertainty. In *Proceedings of the 31st International Conference of the System Dynamics Society, Cambridge, USA, 21-25 July 2013*. Citeseer.

Lotka, A. J. (1925). *Elements of physical biology*. Williams & Wilkins.

Loukrezis, D., Römer, U., and De Gersem, H. (2019). Assessing the performance of leja and clenshaw-curtis collocation for computational electromagnetics with random input data. *International Journal for Uncertainty Quantification*, 9(1).

Perkó, Z., Gilli, L., Lathouwers, D., and Kloosterman, J. L. (2014). Grid and basis adaptive polynomial chaos techniques for sensitivity and uncertainty analysis. *Journal of Computational Physics*, 260:54–84.

Pruyt, E. and Kwakkel, J. H. (2014). Radicalization under deep uncertainty: a multi-model exploration of activism, extremism, and terrorism. *System Dynamics Review*, 30(1-2):1–28.

Razavi, S., Jakeman, A., Saltelli, A., Prieur, C., Iooss, B., Borgonovo, E., Plischke, E., Piano, S. L., Iwanaga, T., Becker, W., et al. (2021). The future of sensitivity analysis: an essential discipline for systems modeling and policy support. *Environmental Modelling & Software*, 137:104954.

Richardson, R. A., Wright, D. W., Edeling, W., Jancauskas, V., Lakhlili, J., and Coveney, P. V. (2020). Easyvvuq: a library for verification, validation and uncertainty quantification in high performance computing. *Journal of open research software*, 8(1).

Saltelli, A. (2002). Making best use of model evaluations to compute sensitivity indices. *Computer physics communications*, 145(2):280–297.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008). *Global sensitivity analysis: the primer*. John Wiley & Sons.

Saltelli, A., Tarantola, S., and Campolongo, F. (2000). Sensitivity analysis as an ingredient of modeling. *Statistical Science*, pages 377–395.

Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). *Sensitivity analysis in practice: a guide to assessing scientific models*, volume 1. Wiley Online Library.

Santner, T. J., Williams, B. J., Notz, W. I., and Williams, B. J. (2003). *The design and analysis of computer experiments*, volume 1. Springer.

Smolyak, S. A. (1963). Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Doklady Akademii Nauk*, volume 148, pages 1042–1045. Russian Academy of Sciences.

Sobester, A., Forrester, A., and Keane, A. (2008). *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons.

Sobol, I. M. (1993). Sensitivity analysis for non-linear mathematical models. *Mathematical modelling and computational experiment*, 1:407–414.

Sudret, B. (2008). Global sensitivity analysis using polynomial chaos expansions. *Reliability engineering & system safety*, 93(7):964–979.

Tang, J., Ni, F., Ponci, F., and Monti, A. (2015). Dimension-adaptive sparse grid interpolation for uncertainty quantification in modern power systems: probabilistic power flow. *IEEE Transactions on Power Systems*, 31(2):907–919.

Tarantola, S., Gatelli, D., Kucherenko, S., Mauntz, W., et al. (2007). Estimating the approximation error when fixing unessential factors in global sensitivity analysis. *Reliability engineering & system safety*, 92(7):957–960.

Verleysen, M. and François, D. (2005). The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*, pages 758–770. Springer.

Volterra, V. (1928). Variations and fluctuations of the number of individuals in animal species living together. *ICES Journal of Marine Science*, 3(1):3–51.

Wiener, N. (1938). The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936.

Xiu, D. (2010). *Numerical methods for stochastic computations*. Princeton university press.

Xiu, D. and Karniadakis, G. E. (2002). The wiener–askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, 24(2):619–644.

COLOPHON

This document was typeset using LaTeX. The document layout was generated using the `arsclassica` package by Lorenzo Pantieri, which is an adaption of the original `classicthesis` package from André Miede.