

# Uncertainty propagation through an aeroelastic wind turbine model using polynomial surrogates

Juan Pablo Murcia<sup>a</sup>, Pierre-Elouan Réthoré<sup>b</sup>, Nikolay Dimitrov<sup>b</sup>, Anand Natarajan<sup>b</sup>, John Dalsgaard Sørensen<sup>c</sup>, Peter Graf<sup>d</sup>, Taeseong Kim<sup>b</sup>

<sup>a</sup>*jumu@dtu.dk, Department of Wind Energy, Technical University of Denmark*

<sup>b</sup>*Department of Wind Energy, Technical University of Denmark*

<sup>c</sup>*Department of Civil Engineering, Aalborg University*

<sup>d</sup>*National Renewable Energy Laboratory, Colorado USA*

---

## Abstract

In the present work, polynomial surrogates are used to characterize the energy production and lifetime equivalent fatigue loads for different components of the DTU 10 MW reference wind turbine under realistic atmospheric conditions. One of the contributions of the present article is to model the stochasticity caused by different turbulent structures in the inflow. This is done by creating independent surrogates for the local mean and local standard deviation of each output of the aeroelastic model. A global sensitivity analysis shows that the turbulent inflow realization has a bigger impact on the total distribution of equivalent fatigue loads than the shear coefficient or yaw miss-alignment. The methodology presented extends the deterministic power and thrust coefficient curves to uncertainty models and adds new variables like damage equivalent fatigue loads in different components. These uncertainty models can then be implemented inside other work-flows such as: estimation of the uncertainty in annual energy production due to inter yearly resource variability and/or robust wind power plant layout optimization. It can be concluded that it is possible to capture the global behavior of a modern wind turbine and its uncertainty under realistic inflow conditions using polynomial response surfaces. The surrogates are a way to obtain power and load estimation under site specific characteristics without sharing the proprietary aero-elastic design.

*Keywords:* Uncertainty quantification, aeroelasticity, wind turbine model, annual energy production, lifetime equivalent fatigue loads

---

## 1. Introduction

The wind turbine design standard IEC 61400-1 [1] provides wind climate specifications which are used as a reference for the structural design of the wind turbines. For achieving type certification of a new turbine model, the designer has to demonstrate that the structural capacity of the turbine is sufficient for withstanding the reference wind conditions over the entire lifetime of the turbine. Such a demonstration is normally given by dynamic load simulations which characterize the behavior of the turbine under the reference wind conditions. Once certification is achieved, the given turbine model can safely be installed on sites where the wind conditions are identical or more benign than the reference standard conditions. However, in many occasions one or more of the parameters describing the site environmental conditions will be outside the ranges which are sufficiently covered by the IEC reference conditions. In such cases, it is necessary to estimate the actual loads which the turbine will experience over its entire lifetime, by considering the full joint distribution of the variables that describe the turbulent inflow. If a full design load case setup similar to the IEC 61400-1 design cases is used for that purpose, the problem quickly becomes time-consuming as new dynamic simulations would be required for each site. Graf et al. showed that the number of simulations required to predict an accurate, within 1% error, lifetime equivalent fatigue loads on a floating wind turbine in which the inflow (sea/wind state) is characterized by five stochastic variables can reach up to  $3,200,000 = 20^5$  simulations using regular grid-based estimates or in the order of 50,000 using Monte-Carlo samples [2]. An approach that alleviates these issues is mapping the turbine response to different environmental inputs by means of a fast and accurate surrogate model.

Some examples of the techniques that can be used to predict the behavior of the turbine from a limited set of model evaluations are: interpolation techniques, response surface techniques, polynomial chaos expansions (PCE), Gaussian process (Kriging) and machine learning algorithms. Toft et al [3] have applied a quadratic response surface technique based on a circular central composite design, for representing the response of the wind turbine as function of five environmental input parameters. Another example of a surrogate model is presented in [4] where the probability density function (PDF) of the load components is expressed as function of the PDF of two inputs: wind shear and wind veer. Abdallah et al. used a Kriging surrogate as a function of wind speed and turbulent standard deviation to predict the 50

year extreme loads on a 5 MW wind turbine [5]. Clifton et al. presented two different regression tree surrogates as a function of wind speed, turbulence intensity and shear exponent for power production [6] and for equivalent fatigue loads [7], although these studies assumed that the response of the wind turbine model to a given condition is captured using a single turbulent realization.

Polynomial chaos expansion is a methodology used to efficiently propagate input uncertainties through a non-linear model. This methodology consists in building a polynomial response surface to capture the global dependency of the output as a function of the uncertain inputs. PCE is a technique widely used in the uncertainty quantification field because of its simplicity and fast convergence in comparison to a full Monte-Carlo simulation based on the original model [8, 9, 10, 11, 12]. Furthermore, adaptive PCE training algorithms can be used to obtain a sparse surrogate that minimizes the number of terms that have multiple variable dependency [13, 14, 15], making the surrogates extremely efficient response surface in multiple dimensions. In particular, for smooth continuous models with multiple input variables, sparse polynomial chaos expansion methodology is the most efficient technique to build the surrogates in terms of the number of model evaluations required, the number of input dimensions they can handle and the rate of convergence [13].

This article proposes the use of two different variable transformations to construct a PCE. These variable transformations are designed to simplify the polynomial response surface fitting problem, see Figure 1. The first transformation is the Rosenblatt transformation [16], which is used to de-correlate the joint input distribution,  $\text{PDF}(\mathbf{x})$ , into independent uniform distributed variables,  $\text{PDF}(\mathbf{w})$ . The second transformation is a logistic transformation, and it is used to enforce constraints to the polynomial surrogates [17]. This transformation enables the use of polynomial surrogates in problems where the output has a minimum and/or maximum value. Without the logistic transformation the polynomial surrogates will present oscillations in the regions where the model has a constant output. The power production of a turbine is an example of a variable with a strict constrain at rated power.

### *1.1. Problem statement*

One of the main difficulties in surrogating an aeroelastic wind turbine model is the fact that the turbulent inflow realization causes variations on the different wind turbine model outputs: such as power, thrust, equivalent

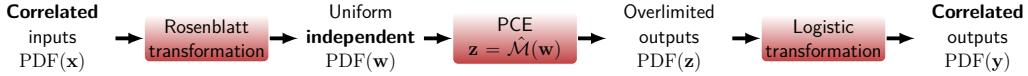


Figure 1: Model diagram.

fatigue loads and extreme loads in the different components of the turbine. Many studies have analyzed the difficulties of studying fatigue and extreme loads under different inflow turbulent inflow realizations [18, 19, 20, 5, 3]. Moriarty et al. showed that different turbulent inflow realizations can activate certain dynamics of the structure and therefore are an important source of uncertainty in the prediction of the model [18]. The high variability in the model response to certain turbulent inflow structures also showed to be problematic when Monte-Carlo estimation was used to predict lifetime averages [2].

### 1.2. Response to the problem

The aim of the present study is to demonstrate a method for building a quick and accurate surrogate model of the wind turbine dynamic response as a function of the multiple variable inputs that describe the turbulent inflow ( $\mathbf{x}$ ). The surrogate model is a set of two independent sparse PCE, which allows to predict the variability caused by different turbulent inflow fields. The distribution of the model outputs (e.g., lifetime equivalent fatigue loads, annual energy production) is predicted with the surrogate model through full Monte-Carlo simulation that covers the entire input variable space, as in Graf et al. [2] but with the possibility of running larger Monte-Carlo sample sizes. Similarly, sensitivity analysis is performed using a full Monte-Carlo simulation of the surrogate, from which the input variables can be ranked by importance in producing the model outputs variability.

The uncertainty in the wind turbine response caused by the different realizations of the turbulent inflow field is captured using two polynomial chaos expansions. The two PCE's are built independently for the first two local statistical moments of the output. A PCE surrogate for the local mean of the output:

$$\hat{y}_{\mathbb{E}}(\mathbf{x}) \approx y_{\mathbb{E}}(\mathbf{x}) = \mathbb{E}(y(\mathbf{x})) \quad (1)$$

and a PCE for the local standard deviation:

$$\hat{y}_{\mathbb{S}}(\mathbf{x}) \approx y_{\mathbb{S}}(\mathbf{x}) = \sqrt{\mathbb{V}(y(\mathbf{x}))} = \mathbb{S}(y(\mathbf{x})) \quad (2)$$

Note that the two surrogates characterize the distribution of the output at a given input point ( $\mathbf{x}$ ) resulting only from the stochasticity of the turbulent inflow field. Finally, a sample can be obtained from the local normal distribution constructed using the mean and the standard deviation surrogates in order to make a prediction of the output at a given input point:

$$\hat{y}(\mathbf{x}) \sim \text{Normal}(\hat{y}_{\mathbb{E}}(\mathbf{x}), \hat{y}_{\mathbb{S}}(\mathbf{x})) \quad (3)$$

### 1.3. Article overview

A general overview of the PCE methodology in multiple dimensions is presented in section 2. This section describes the Rossenblat transformation, the design of experiments used to define the training simulation points, the approach used to train sparse PCE and the logistic transformation used to limit the output space. In section 3, the methodology is then applied to the response of the DTU 10 MW reference wind turbine HAWC2 model [21] to turbulent inflow fields characterized by four input parameters. The four input parameters are the 10-min averaged hub height wind speed, the standard deviation of the instantaneous wind speed in the streamwise component, the shear exponent and the yaw miss-alignment angle. A study of how many independent realizations of the turbulent inflow field are required to achieve a certain error tolerance in the surrogate is presented in the section 3.8. Finally in section 3.9, the surrogates are used in an example of prediction of uncertainty in annual energy production and uncertainty of lifetime equivalent fatigue loads.

## 2. Methods

### 2.1. 1D PCE theory

Consider a model with a single uncertain input ( $x$ ) and a single output ( $y$ ). PCE consists in defining a polynomial family that is orthogonal with respect the input distribution,  $\text{PDF}(x)$ . Orthogonal polynomial families with respect the most important distributions are well known, see table 1. For details on how to define new polynomial basis to an arbitrary input distributions refer to Gautschi et al. [22].

Distribution	Polynomial Family
Uniform	Legendre
Normal	Hermite
Exponential	Laguerre

Table 1: Classical orthogonal polynomial families.

The orthogonal polynomials are used to build a polynomial approximation of the output, i.e. a polynomial response surface, see equation 4. Where,  $\phi_l(x)$  is the  $l$  order orthogonal polynomial,  $c_l$  is its correspondent coefficient and  $M$  represents the truncation order of the PCE.

$$y(x) \approx \hat{y}(x) = \sum_{l=0}^M c_l \phi_l(x) \quad (4)$$

The orthogonality property makes the PCE an useful approach to propagate uncertainty over a polynomial response surface because the statistical moments of the output can be derived directly from the coefficients:

$$\mathbb{E}(\hat{y}) = c_0 \quad \mathbb{V}(\hat{y}) = \sum_{l=1}^M c_l^2 \quad (5)$$

There are two different approaches to determine the PCE coefficients:

*Semi-Spectral projection* consists in using quadrature rules to approximate the inner product definition of the coefficient, see eq. 6. Many quadrature rules exist to approximate the integrals; but all quadrature rules give  $N_n$  nodes for model evaluation ( $x_i$ ) and their corresponding weights ( $\omega_i$ ). Gaussian quadrature rules are widely used because they are accurate for smooth function integration with respect a weight function, in this case the PDF( $x$ ), see equation 6.

$$c_l = \langle y, \phi_l \rangle = \int y(x) \phi_l(x) \text{PDF}(x) dx \approx \sum_{i=0}^{N_n} \omega_i y(x_i) \phi_l(x_i) \quad (6)$$

In general, semi-spectral projection is an efficient method for low number of input dimensions, but the number of model evaluations required grows exponentially with the number of dimensions. Additionally, quadrature rules can be unstable for heavy tailed PDFs such as the Weibull distribution [22].

*Point collocation* consists in fitting the polynomial basis to a small sample of model evaluations. Traditionally, this fit can be done using least squares algorithm, but some other optimization algorithms can be used to obtain PCE approximations that minimize the number of terms in the surrogate [13, 14, 15]. These techniques are explained in the section 2.5. In general, point collocation is robust and the advanced optimization algorithms are designed to handle large number of dimensions, to avoid over-fitting and to achieve sparsity in the final surrogate. The present study focuses only in the point collocation techniques since the number of model evaluations required to fit a multiple dimensional PCE is smaller [13] than in other methods.

## 2.2. Rosenblatt transformation

To build the PCE of a model with multiple correlated inputs ( $\mathbf{x}$ ), it is required to initially transform the correlated input space into an uncorrelated space ( $\mathbf{w} = R^{-1}(\mathbf{x})$ ). In this article, the Rosenblatt transformation is used because the input distribution of the turbulent inflow field parameters are usually defined in a sequence of conditional relationships. Refer to Dimitrov et al. [23] and Graf et al. [2] for examples of such distributions used for offshore and floating wind turbine fatigue and extreme load analysis.

The sequential dependency means that the distribution of each of the input variables is given by a parametric  $\text{PDF}_i$ , whose parameters  $\boldsymbol{\theta}_i$  are themselves functions of the previous input variables:

$$x_i \sim \text{PDF}_i(x_i | \boldsymbol{\theta}_i = \boldsymbol{\theta}_i(x_0, x_1, \dots, x_{i-1})) \quad (7)$$

The Rosenblatt transformation consists in using the cumulative density function of each individual variable  $\text{CDF}_i$  in sequence to transform the variable into an uncorrelated uniform variable [16]:

$$w_i = R^{-1}(x_i) = \text{CDF}_i(x_i | \boldsymbol{\theta}_i = \boldsymbol{\theta}_i(x_0, x_1, \dots, x_{i-1})) \quad (8)$$

or to use the inverse  $\text{CDF}_i^{-1}$  to transform the uniform variable into the correspondent input:

$$x_i = R(w_i) = \text{CDF}_i^{-1}(w_i | \boldsymbol{\theta}_i = \boldsymbol{\theta}_i(x_0, x_1, \dots, x_{i-1})) \quad (9)$$

Since all the variables are transformed into uncorrelated unitary uniform variables then the PCE only requires the use of the Legendre polynomials:  $y(\mathbf{x}) = y(R(\mathbf{w})) \approx \hat{y}(\mathbf{w})$ . See appendix Appendix A for the equations of the Legendre polynomials for an unitary uniform variable.

### 2.3. Multi-dimensional PCE

A multi-dimensional polynomial is constructed as the sum of the product between individual one dimensional polynomials for each of the  $D$  uniform input variables,  $\mathbf{w} = [w_0, \dots, w_{D-1}]$ . The  $D$ -dimensional surrogate is written using a set of multiple indexes  $\mathcal{I} \subset \mathbb{N}^D$ . An element  $J \in \mathcal{I}$  contains the order of the polynomial in each dimension:  $J = [l_0, \dots, l_{D-1}]$ . Additionally, the multiple indexes are enumerated,  $J \leftrightarrow j \in \mathbb{N}$ . A surrogate that contains  $N_c$  terms can be written as:

$$y(\mathbf{x}) = y(R(\mathbf{w})) \approx \sum_{j=0}^{N_c-1} c_j \phi_j(\mathbf{w}) \quad (10)$$

where an element in the multidimensional polynomial basis is given as:

$$\phi_j(\mathbf{w}) = \phi_{l_0}(w_0) \times \cdots \times \phi_{l_{D-1}}(w_{D-1}) \quad (11)$$

An example of a 3-dimensional polynomial response surface with 4 terms is given in table 2. In this example the multi-dimensional polynomial is given in eq. 12. This equation can be evaluated using the Legendre polynomials,  $\phi_l(w)$ . Note that  $\phi_0(w) = 1$ .

$j$	$l_0$	$l_1$	$l_2$	$c_j$
0	0	0	0	0.5
1	1	0	0	7.5
2	2	2	0	-1.4
3	3	1	2	21.1

Table 2: Example of the notation of a 3-dimensional polynomial response surface.

$$\hat{y}(\mathbf{w}) = 0.5 + 7.5 \phi_1(w_0) - 1.4 \phi_2(w_0) \phi_2(w_1) + 21.1 \phi_3(w_0) \phi_1(w_1) \phi_2(w_2) \quad (12)$$

### 2.4. Training point selection

The Rosenblatt transformation enables the use of multiple variance reduction Monte-Carlo sampling techniques to define the training points of a surrogate [24]. Latin hypercube sampling [25], Sobol sequence [26] and Hammersley sequence [27] are some examples of such techniques. These techniques are designed to sample from the unitary hypercube of  $D$  dimensions, i.e. the uniform distributed variables:  $\mathbf{w}_i \sim \text{PDF}(\mathbf{w})$ . Finally, the

Rosenblatt transformation is used to transform this sample into the correlated input space,  $\mathbf{x}_i = R(\mathbf{w}_i) \sim \text{PDF}(\mathbf{x})$ .

If the total polynomial order of the PCE, i.e. the maximum sum of the one dimensional orders on each dimension, is truncated to be smaller than  $M$ , then the number of unknown coefficients is given by the following combination:

$$N_c = \binom{M+D}{M} = \frac{(M+D)!}{M! D!} \quad (13)$$

Blatman et al. suggest that the number of model evaluations should be between 2 or 3 times the number of unknowns in order to have extra data to test the accuracy of the surrogate and to implement strategies to avoid over-fitting [13]. Note that the maximum order is only used to estimate the number of model evaluations. Advanced regression techniques allow to explore higher order terms [15, 13]. The maximum order  $M$  can be increased in order to achieve higher accuracy PCE surrogates but at the cost of having more model evaluations and the requirement of assuring that there is not over-fitting.

### 2.5. LASSO problem

The least absolute shrinkage and selection operator (LASSO) problem is a modified least squares optimization problem that adds a term that penalizes the amount of active terms in the surrogate (terms with non zero coefficients). LASSO is used to achieve sparsity and to avoid over fitting in the surrogate. Additionally, the number of model evaluations required for solving the LASSO problem is smaller in comparison to a least squares regression that has the same maximum polynomial order  $M$  [13].

A LASSO problem can be described as finding the set of coefficients  $c_j$  that minimizes the sum of squared errors plus the sum of the absolute values of all coefficients ( $\ell_1$  norm regularization term) [15]:

$$\min_{c_j} \sum_{i=0}^{N-1} \left[ \sum_{j=0}^{N_c-1} c_j \phi_j(\mathbf{w}_i) - y(\mathbf{x}_i) \right]^2 + \alpha \sum_{j=0}^{N_c-1} |c_j| \quad (14)$$

where the number of model/surrogate evaluation points  $N$  is fixed. Note that the input and surrogate evaluation points are related by the Rosenblatt transformation  $\mathbf{x}_i = R(\mathbf{w}_i)$ . The maximum number of possible terms of the

surrogate  $N_c$  is also fixed by selecting a maximum total multi-dimensional polynomial order.

The regularization coefficient  $\alpha$  controls the amount of active terms in the final solution. Smaller values allow to have more active terms while larger values will prefer final surrogates with few active terms. A sparse surrogate has the advantage of making the evaluation of the multi-dimensional surrogate faster in comparison to the full least squares solution; this advantage becomes critical in high number of input dimensions.

There are two algorithms widely used to solve the LASSO problem: coordinate descent [15] and least angle regression (LAR) [13]. Coordinate decent is used in the present work because it tends to be more stable for high dimensional problems [14]. The reason for this is that coordinate descent operates on a given regularization coefficient instead of exploring the full space of  $\alpha$ 's as in LAR algorithm.

Cross-validation is used to select the regularization coefficient  $\alpha$  that minimizes over fitting of the data. A k-fold cross-validation consists in splitting the dataset into k groups of data. All the points in k-1 groups are used for training and the remaining group is used for cross-validation. This means that the surrogate fitted using k-1 groups is used to predict the output in each of the elements of the remaining group. The mean squared error of the prediction of the surrogate is then computed. This process is repeated leaving out each individual fold and for multiple regularization parameters. The regularization parameter that gives the lowest mean cross-validation mean squared errors is then selected to train the whole dataset. This translates as selecting the sparse model that performs the best by predicting missing data, i.e. that has less over-fitting. Figure 2 shows an example in which an over fitted model has a larger cross validation prediction error than a sparse surrogate.

## 2.6. Logistic transformation

A logistic transformation is applied to an output of the model in order to avoid oscillations in the regions where the model is constant. In practice this transformation is used to impose strict restrictions to the polynomial surrogates. The logistic transformation uses the logit function:

$$L(p) = \ln \left( \frac{p}{1-p} \right) \quad (15)$$

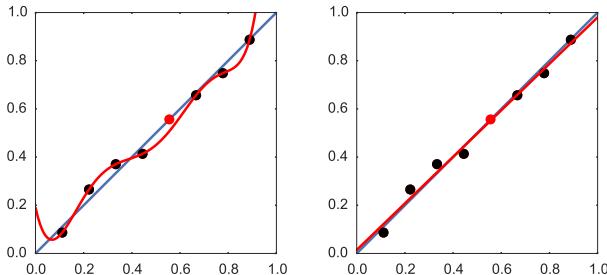


Figure 2: Simple example of over fitting and cross validation. The red point is only used for cross validation. (Left) Over fitted surrogate. (Right) Optimal surrogate.

and its inverse, the logistic function:

$$L^{-1}(q) = \frac{1}{1 + e^{-q}} \quad (16)$$

The transformation consists in applying the logit function to the model output at the training points  $y_i = y(\mathbf{x}_i)$  to define the training points in the possibly over-shooting variable space  $z_i$  [17], see equation 17. The surrogates are then fitted to the  $z_i$  outputs.

$$z_i = L(a_1 y_i + a_0) \quad (17)$$

Finally, each time the surrogate is evaluated, the prediction of the surrogate  $\hat{z}$  is transformed back to the original output space  $\hat{y}$ , see equation 18. The constants of the transformation ( $a_0$  and  $a_1$ ) are calibrated in order to impose the constraints of the output and to avoid numerical instabilities that are inherent to the logit function.

$$\hat{y} = \frac{L^{-1}(\hat{z}) - a_0}{a_1} \quad (18)$$

### 2.7. Global sensitivity analysis

Global sensitivity analysis (SA) is a methodology to determine how important each input is to explain the variance of the output. SA can be obtained with a Sobol variance decomposition [28]. In this technique, the variance of the output is explained into the different terms of variance of each of the inputs, in a process similar to the analysis of the variance of experiments (ANOVA) [29]. Total effect Sobol indices are widely used as

measures of how much of the variance of a given output is explained by the variance of an input, including possible interactions with other variables. This method is the most recognized method for global sensitivity analysis because it accounts for non-linear dependencies and for interactions between variables [30].

Variance decomposition can be expressed as the sum of the variance of the marginal expected value of a subset of input variables, see eq. 19. Note that this decomposition is not an infinite series expansion, it is truncated to the maximum number of variable interactions.

$$\begin{aligned} \mathbb{V}(y) &= \sum_{k=0}^{D-1} \mathbb{V}_k + \sum_{k=0}^{D-1} \sum_{l>k}^{D-1} \mathbb{V}_{kl} + \sum_{k=0}^{D-1} \sum_{l>k}^{D-1} \sum_{m>l}^{D-1} \mathbb{V}_{klm} + \dots + \mathbb{V}_{0\dots D-1} \\ \mathbb{V}_k &= \mathbb{V}(\mathbb{E}_{\forall n \neq k}(\mathcal{M}(\mathbf{x}|x_k))) \\ \mathbb{V}_{kl} &= \mathbb{V}(\mathbb{E}_{\forall n \neq k,l}(\mathcal{M}(\mathbf{x}|x_k, x_l))) \\ \mathbb{V}_{klm} &= \mathbb{V}(\mathbb{E}_{\forall n \neq k,l,m}(\mathcal{M}(\mathbf{x}|x_k, x_l, x_m))) \end{aligned} \quad (19)$$

The global sensitivity measure is defined by normalizing eq. 19 with the total variance of the output  $\mathbb{V}(y)$ . From this normalization one can define the Sobol index of a given degree of interaction between input variables as:

$$S_k = \frac{\mathbb{V}_k}{\mathbb{V}(y)} \quad S_{kl} = \frac{\mathbb{V}_{kl}}{\mathbb{V}(y)} \quad S_{klm} = \frac{\mathbb{V}_{klm}}{\mathbb{V}(y)} \quad \dots \quad (20)$$

The total effect Sobol index of an input variable  $x_i$  is then the sum of all the Sobol indices that include the variable in any interaction:

$$S_{\text{total } x_i} = S_i + \sum_{\substack{k=0 \\ k \neq i}}^{D-1} S_{ik} + \dots \quad (21)$$

The sensitivity analysis of the response of the turbine should consider the effect of having different turbulent inflow realizations. The turbulent inflow is modeled with the two independent PCE for the local mean and local standard deviation. Therefore, even though the Sobol indexes could be computed directly from the PCE coefficients, see Sudret et al. [31], they would not include the effect of the turbulence inflow realization. To solve this limitation, the approximate method proposed in Saltelli et. al [30] is used to compute the total effect Sobol indexes. This approach estimates the total effect Sobol indexes from a large Monte-Carlo simulation.

### 3. Results

#### 3.1. Implementation

Several open source implementations of PCE methods are available such as: Chaospy [24], Dakota [32], UQLab [33] and OpenTurns [34]. In the present work we use Chaospy because of its implementation of the Rosenblatt transformation. Additionally, the present work uses the LASSO problem solvers [15] and the cross-validation capabilities available in the open source library Scikit-learn [14]. These capabilities are used inside of Chaospy for general users and are used externally in the present study to gain control over the different stages of the cross-validation.

#### 3.2. Case description

The model consist of the DTU 10 MW reference wind turbine HAWC2 model [35, 21] with Mann turbulent inflow generation [36]. The turbulent inflow conditions are defined using the four variables described in table 3.

Input	Variable	Distribution	Parameters	
10-min mean hub height wind speed	$x_0 = \text{WS}$	Rayleigh	$\mathbb{E}(\text{WS}) = 10 \text{ m/s}$	
Std. of the inst. wind speed in the streamwise direction during the 10-min simulation	$x_1 = \sigma_1$	Lognormal	$\mu_{\sigma_1}(\text{WS})$	$\sigma_{\sigma_1}(\text{WS})$
10-min mean shear exponent	$x_2 = \alpha$	Normal	$\mu_\alpha(\text{WS})$	$\sigma_\alpha(\text{WS})$
10-min mean yaw miss-align.	$x_3 = \gamma$	Normal	$\mu_\gamma = 0$	$\sigma_\gamma = 5 \text{ deg.}$

Table 3: Wind turbine model inputs.

The dependency between WS and  $\sigma_1$  is defined in the Normal Turbulence Model described in the IEC 61400-1 [1]. The present case uses a reference ambient turbulence intensity of a site Class I-A:  $\text{TI}_{\text{ref}} = 0.16$ . This dependency is given by the local statistical moments of  $\sigma_1$  as:

$$\begin{aligned}\mathbb{E}(\sigma_1|\text{WS}) &= \text{TI}_{\text{ref}} (0.75\text{WS} + 3.8) \\ \mathbb{V}(\sigma_1|\text{WS}) &= (1.4 \text{TI}_{\text{ref}})^2\end{aligned}\tag{22}$$

which translates into a dependency of the local Lognormal distribution parameters of  $\sigma_1$  as:

$$\begin{aligned}\sigma_{\sigma_1} &= \left( \ln \left( \frac{\mathbb{V}(\sigma_1|WS)}{\mathbb{E}^2(\sigma_1|WS)} + 1 \right) \right)^{1/2} = \left( \ln \left( \frac{1.4^2}{(0.75WS + 3.8)^2} + 1 \right) \right)^{1/2} \\ \mu_{\sigma_1} &= \ln (\mathbb{E}(\sigma_1|WS)) - \frac{\sigma_{\sigma_1}^2}{2} = \ln (\text{TI}_{\text{ref}} (0.75WS + 3.8)) - \frac{\sigma_{\sigma_1}^2}{2}\end{aligned}\quad (23)$$

The correlation between  $\alpha$  and WS is based on the simplified joint distribution defined by Dimitrov et al. [23]:

$$\begin{aligned}\mu_{\alpha} &= 0.088(\ln(WS) - 1) \\ \sigma_{\alpha} &= 1/WS\end{aligned}\quad (24)$$

Seven different model outputs are considered ( $\mathbf{y}$ ), see table 4. The damage equivalent fatigue loads (EFL) are computed using a rainflow counting algorithm to determine the number of load cycles  $n_i$  with their corresponding load range  $S_i$  in the 10-min time series of turbine response. These two variables are then weighted using different material's Möhler coefficient  $m$ , see equation 25 by Miner et al [37]. The equivalent fatigue loads are given as a 1 Hz cyclic load acting during the 10 minutes, therefore the reference number of load cycles is  $N_{\text{ref}} = 600$ .

$$S_{\text{eq}} = \left( \frac{\sum n_i S_i^m}{N_{\text{ref}}} \right)^{\frac{1}{m}} \quad (25)$$

Output	$m$	Variable
10 minute mean power production	-	$P$
10 minute mean thrust coefficient	-	$CT$
EFL blade root flapwise bending moment	12	BRF
EFL tower bottom fore-aft bending moment	4	TBF
EFL tower bottom sidewise bending moment	4	TBS
EFL tower top tilt bending moment	4	TTT
EFL tower top yaw bending moment	4	TTY

Table 4: Wind turbine model outputs.

### 3.3. Training points

In this article, the number of model evaluations are set to be  $N = 2N_c$ , and the maximum order of the polynomial is expected to be  $M = 4$ . This leads to a number of model evaluations of 140 for a 4-dimensional case, see equation 13. A Hammersley sequence [27] is preferred over other variance reduction methods to generate the training sample in the uniform space as it is a sequence that can be extended to contain larger sample size without changing the previous points [24, 38]. The uniform sample is then transformed into the correlated variables using the Rossenblat transformation. The training input sample is shown in figure 3 as well as a large Monte-Carlo sample used for comparison. It can be observed that the training points are more densely packed in the regions of higher probability of the inputs. This means that the surrogate is better trained in the most likely regions of the input space. Note that the full training dataset includes 100 different turbulent inflow realization for each input point which are used to estimate the uncertainty in the turbine performance. This makes a full training sample size  $140 \times 100$  HAWC2 10 minutes simulations.

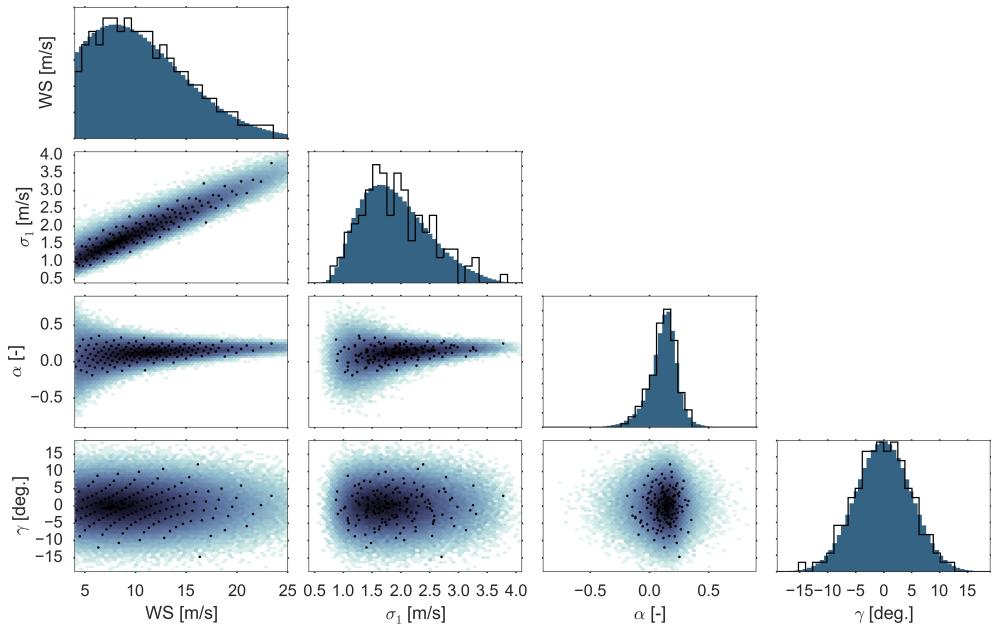


Figure 3: (Black points) Training dataset in the inputs: 140 Hammersley sequence sample of input joint distribution. (Histogram colored hex-bins) 80000 Hammersley sequence sample Monte-Carlo sample.

### 3.4. Model selection example

Several number of folds are tested in the k-fold cross-validation selection of the proper level of sparsity in each of the outputs. 20-fold and 140-fold (maximum number of folds possible or leave-one-out) cross-validation results are shown in figure 4. This figure shows the cross validation mean squared error for some of the outputs for all the possible training subsets. It can be seen that there is a large deviation in the errors when different folds are left for comparison. In particular, the extreme points in the input sample are important for the training process as they will contain the only information about the model response in regions with low probability. The error of a surrogate is larger if the training set does not include such extreme points. For the final results the 20-fold validation is selected because it gives a good balance between computational effort and accuracy in the estimation of the regularization coefficient obtained from the leave-one-out.

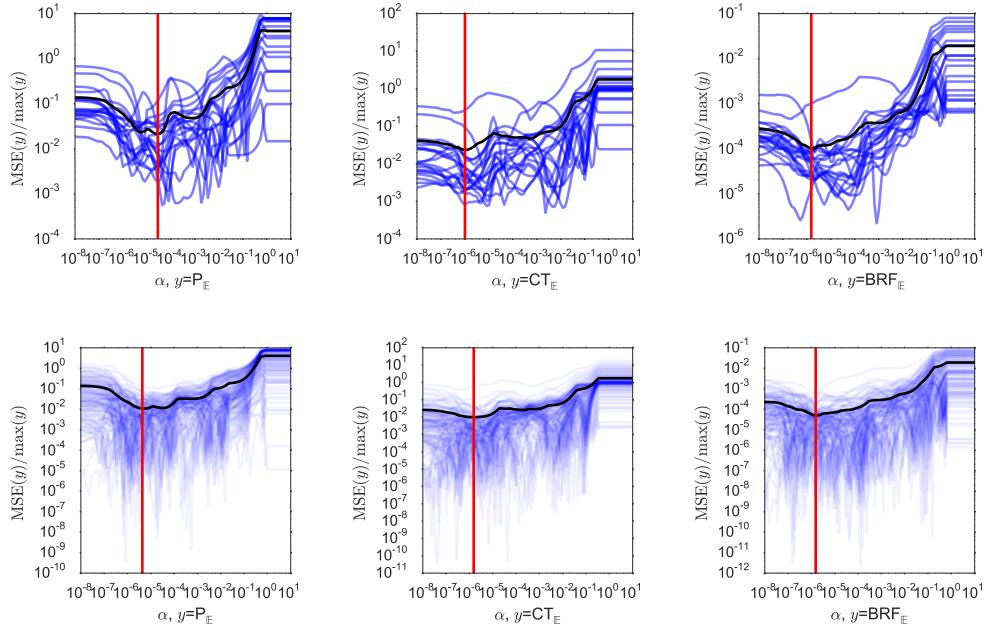


Figure 4: Examples of the cross validation mean squared error (MSE) for the optimization of regularization coefficient. (top row) 20-Fold. (bottom row) 140-fold. (Blue lines) Cross-validation mean square error when each different fold is used as validation dataset. (Black line) Mean of the k-fold cross-validation MSE. (Red line) Selected regularization parameter.

### 3.5. Example of PCE surrogates for individual statistical moments

Some examples of the surrogates for  $\mathbf{y}_E$  and  $\mathbf{y}_S$ <sup>1</sup> are shown in figures 5 and 6. In general, the surrogates accurately capture the global behavior of the model with respect the 4 input variables. The surrogates of the local standard deviations present larger errors, although the errors are small in comparison to the total scale of the output. These local errors cause an inability to fully capture the overall distribution of the local standard deviations, see the plots in the fifth column in figures 5 and 6. The distribution of the errors of the surrogate and its impact in the final prediction are quantified in section 3.8. These errors can be reduced up to a tolerance level selected by the user by adding more training points (input points with their turbulent inflow realizations).

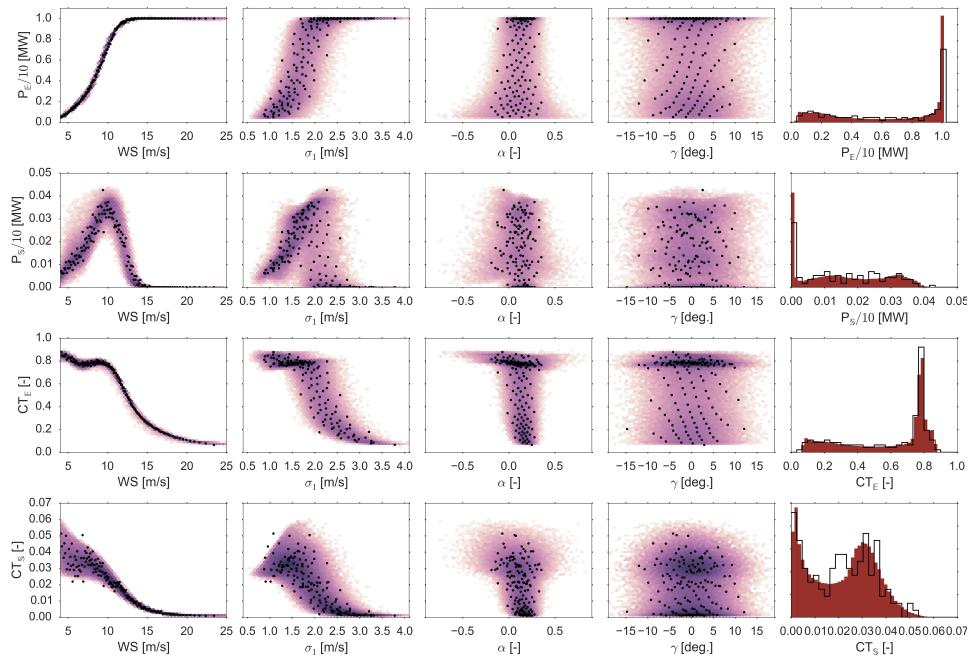


Figure 5: Example surrogates for individual local statistical moments. (Black points) 140 training dataset. (Histogram colored hex-bins, lighter pink bins have lower probability) 80000 Monte-Carlo simulation on the surrogate.

<sup>1</sup> $P_S$  represents the standard deviation of 100 different realizations of the 10-min averaged power; this variable should not be confused with the standard deviation of the instantaneous power during the 10 minutes of simulation.

The surrogates are robust enough to predict the frequency of occurrence of extreme values such as the outputs resulting from the input point with largest  $\sigma_1$ , see first and third row in figure 6. This point seems to be outside the main trend in WS in figure 6 because it has a large  $\sigma_1$  and  $\alpha$  given its WS, see figure 3.

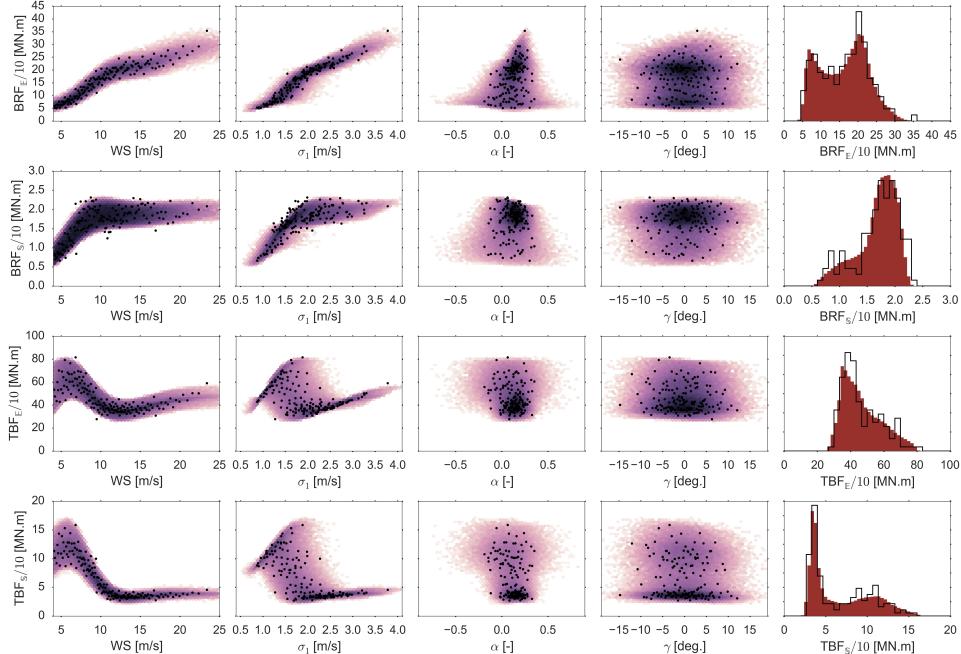


Figure 6: Example surrogates for individual local statistical moments. (Black points) 140 training dataset. (Histogram colored hex-bins, lighter pink bins have lower probability) 80000 Monte-Carlo simulation on the surrogate.

### 3.6. Final surrogate predictions

The final surrogates of the DTU 10 MW RWT obtained are presented in figure 7. The amount of local output variation due to the turbulent inflow realization varies between outputs and depends on the region of the input space. In the figure 7, each see-through black cross represents an individual realization of the turbulence. The effect of the turbulent inflow realization is more important for the fatigue loads. The final surrogate covers the space of model outputs and predicts the general details of the overall distributions of the outputs, see column 5 in figure 7.

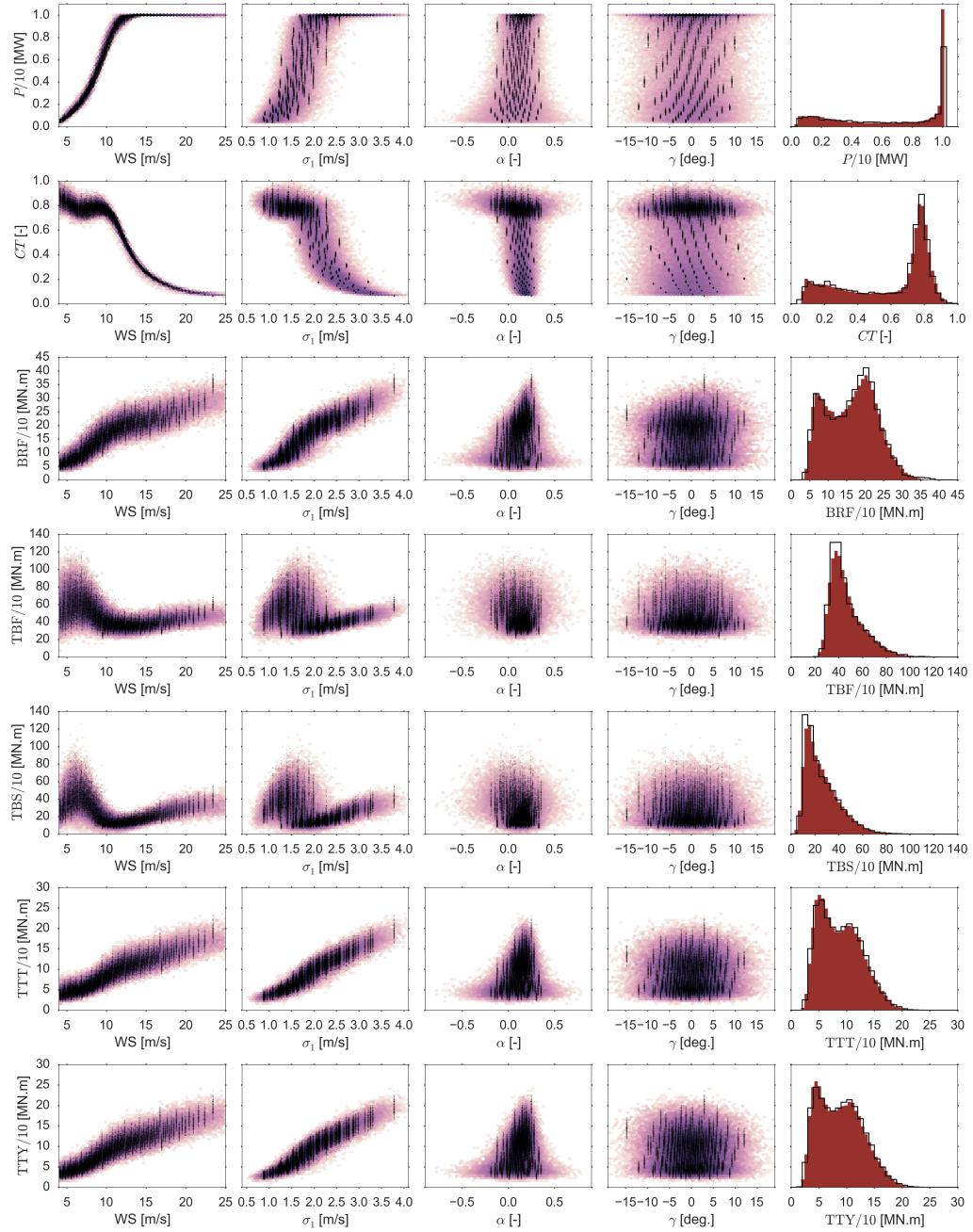


Figure 7: (Black points) Training dataset in the outputs: 140 Hammersley sequence sample  $x$  100 turbulent inflow realizations. (Histogram colored hex-bins, lighter pink bins have lower probability) 80000 Monte-Carlo simulation of the surrogate.

### 3.7. Sensitivity analysis

The global sensitivity analysis (SA) for the outputs are presented in table 5. The total effect Sobol indexes are computed using the approximation presented by Saltelli et al. [30]. The total effect Sobol indexes represents the non-linear influence of the input variable in the total variance of the output. Most of the outputs have a large total Sobol index for the wind speed. WS is clearly the main variable to explain the power and loads in a wind turbine. The SA shows that the local expected power and thrust coefficient can be explained almost fully with WS. This corroborates the use of power and thrust coefficient curves by the wind industry.

The variance introduced by the turbulent inflow realization is an important component for all the outputs, it has a higher influence than  $\sigma_1$  for most outputs. This counterintuitive result is due to the large amount of correlation between WS and  $\sigma_1$ , therefore most of the variance due to  $\sigma_1$  is already explained by WS. The shear and yaw have reduced effects over most output variables. The yaw miss alignment has reduced total effect because its distribution is centered around zero. The shear exponent becomes important only for capturing the fatigue at the tower tip tilt and yaw bending moments (TTT, TTY); while the yaw miss alignment becomes important for modeling the fatigue due at the tower bottom fore-aft moment (TBF).

Output	WS	$\sigma_1$	$\alpha$	$\gamma$	Turb_realization
P	1	$3 \times 10^{-4}$	$3 \times 10^{-4}$	$1 \times 10^{-4}$	$4 \times 10^{-3}$
	1st	3rd-4th	3rd-4th	5th	2nd
CT	1	$1 \times 10^{-3}$	$1 \times 10^{-3}$	$7 \times 10^{-4}$	$1 \times 10^{-2}$
	1st	3rd-4th	3rd-4th	5th	2nd
BRF	$9 \times 10^{-1}$	$5 \times 10^{-2}$	$1 \times 10^{-2}$	$3 \times 10^{-3}$	$7 \times 10^{-2}$
	1st	3rd	4th	5th	2nd
TBF	$6 \times 10^{-1}$	$2 \times 10^{-1}$	$3 \times 10^{-4}$	$1 \times 10^{-3}$	$3 \times 10^{-1}$
	1st	3rd	5th	4th	2nd
TBS	$7 \times 10^{-1}$	$7 \times 10^{-2}$	$2 \times 10^{-3}$	$2 \times 10^{-4}$	$3 \times 10^{-1}$
	1st	3rd	4th-5th	4th-5th	2nd
TTT	$9 \times 10^{-1}$	$7 \times 10^{-2}$	$3 \times 10^{-4}$	$6 \times 10^{-4}$	$8 \times 10^{-2}$
	1st	2nd-3rd	4th	5th	2nd-3rd
TTY	$9 \times 10^{-1}$	$7 \times 10^{-2}$	$2 \times 10^{-4}$	$1 \times 10^{-3}$	$7 \times 10^{-2}$
	1st	3rd-4th	3rd-4th	5th	2nd

Table 5: Total sensitivity indexes (Total influence Sobol indexes).

### 3.8. Convergence

A leave-one-out cross-validation (LOO) is done to estimate the distribution of the prediction error of each surrogate as a function of the number of independent turbulent seeds. A LOO is equivalent to a k-fold validation in which the number of folds is the number of data points. This means that the surrogate is trained using 139 input points with a different number of turbulent inflow realizations per inputs point. Then, the local statistical moments of the output predicted by the surrogates at the missing point are compared against the statistics computed using the full 100 realizations. In this article, the prediction errors are normalized with respect the maximum scale of the output variable, which means that the errors represent the fraction of the total scale that should be considered as extra uncertainty due to the inadequacy of the surrogate. The prediction error for the local mean surrogate is defined as:

$$\epsilon_{y\mathbb{E}} = \frac{y_{\mathbb{E}}(\mathbf{x}_{LO}) - \hat{y}_{\mathbb{E}}(\mathbf{x}_{LO})}{\max(y)} \quad (26)$$

while the prediction error for the local standard deviation surrogate is defined as:

$$\epsilon_{y\mathbb{S}} = \frac{y_{\mathbb{S}}(\mathbf{x}_{LO}) - \hat{y}_{\mathbb{S}}(\mathbf{x}_{LO})}{\max(y)} \quad (27)$$

The convergence of the prediction error of the statistical moments are shown in figure 8. It can be seen that all the prediction errors tend to be distributed around zero and its standard deviation converges as the number of turbulent inflow realizations per input are increased. The errors converge to the distribution of the errors to the current surrogate. New input points need to be added to the training dataset in order to reduce the converged distribution of surrogate errors. In this figure the outliers are the extreme cases of selecting seeds with similar outputs, therefore, they are those cases that have large errors in the statistical moments. Finally, the converged distribution can be used to estimate the uncertainty in the final prediction of the output as:

$$\hat{y}(\mathbf{x}) \sim \text{Normal}(\hat{y}_{\mathbb{E}}(\mathbf{x}) + \epsilon_{y\mathbb{E}} \max(y), \hat{y}_{\mathbb{S}}(\mathbf{x}) + \epsilon_{y\mathbb{S}} \max(y)) \quad (28)$$

where the errors of the surrogates can be sampled from the distribution predicted using LOO cross validation, see figure 8:

$$\epsilon_{y\mathbb{E}} \sim \text{Normal}(\mathbb{E}(\epsilon_{y\mathbb{E}}), \mathbb{S}(\epsilon_{y\mathbb{E}})) \quad \epsilon_{y\mathbb{S}} \sim \text{Normal}(\mathbb{E}(\epsilon_{y\mathbb{S}}), \mathbb{S}(\epsilon_{y\mathbb{S}})) \quad (29)$$

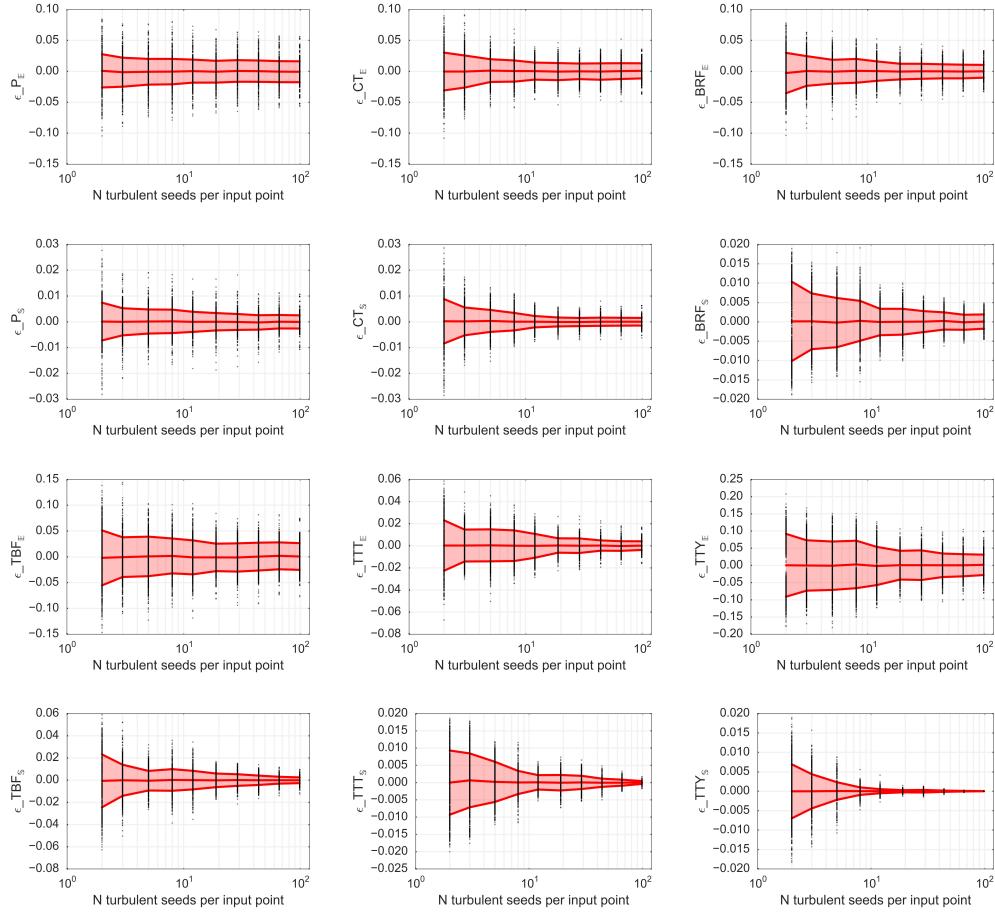


Figure 8: Convergence of the LOO cross-validation prediction error as a function of the number of turbulent seeds per input point used in PCE training. (Pink area) One standard deviation confidence interval around the mean  $\mathbb{E}(\epsilon) \pm \mathbb{S}(\epsilon)$ .

### 3.9. Example of using the surrogates for the estimation of the uncertainty in annual energy production and lifetime equivalent fatigue loads

This section presents an example to illustrate the use of the surrogates of the DTU 10 MW RWT to estimate the uncertainty in the yearly distribution of annual energy production and of equivalent fatigue loads. A single turbine is planned to operate in a location from which the yearly variability of wind resources has been studied during a long term (more than 20 years) campaign. The distribution of the variability of the wind resources is presented in the table 6. The main difference with the distribution used for training the surrogates is the fact that the WS follows a Weibull distribution with uncertain shape and scale parameters. This distribution of the Weibull parameters is used to characterize the year-to-year variability in the wind resources. Subsequently, the conditional distributions of  $\sigma_1$ ,  $\alpha$  and  $\gamma$  follow the same dependency described in the table 3.

Variable	Distribution	Parameters	
$A$	Normal	$\mu_A = 9$	$\sigma = 0.5$ m/s
$k$	Normal	$\mu_A = 2$	$\sigma = 0.1$
$x_0 = \text{WS}$	Weibull	scale= $A$	shape= $k$
$x_1 = \sigma_1$	Lognormal	$\mu_{\sigma_1}(\text{WS})$	$\sigma_{\sigma_1}(\text{WS})$
$x_2 = \alpha$	Normal	$\mu_\alpha(\text{WS})$	$\sigma_\alpha(\text{WS})$
$x_3 = \gamma$	Normal	$\mu_\gamma = 0$	$\sigma_\gamma = 5$ deg.

Table 6: Uncertainty in wind resources.

The propagation of uncertainty of this case is done in two steps. Initially a PCE with Gaussian quadrature is used to determine the nodes and weights of the distributions of the Weibull parameters  $A$  and  $k$ , see equation 6. Each of these nodes represent the Weibull parameters in a given year. For each of these nodes, a large sample of the inputs of the surrogate,  $\mathbf{x} = [\text{WS}, \sigma_1, \alpha, \gamma]$ , is generated. The size of the sample is the number of 10-min cases in a year:  $365 \times 24 \times 6 = 52,560$  cases. The power and EFL are evaluated using the surrogate and the mean power and mean EFL for that year are calculated. Each individual surrogate evaluation has its own realization of the local distribution of the outputs due to the turbulence inflow realization,

see equation 3. Additionally, the effect of the errors of the surrogate can also be considered, by sampling the distribution of the errors for each evaluation of the outputs, see equation 28. The results of the yearly distribution of capacity factor and of yearly equivalent loads are presented in figure 9. There is no differences in the distributions of interannual capacity factor or mean EFL obtained using the surrogate or the ones obtained including the uncertainty of the surrogate. The distribution in the capacity factor shows a negative skewness or a heavier tail for smaller values, which means that the p90 is smaller than the one predicted using a normal distribution. On the contrary, the interannual distribution of mean TBF shows a positive skewness, which means that in some years the fatigue in the tower was larger than expected. The distribution of mean BRF is symmetric, but has heavier tails than a normal distribution.

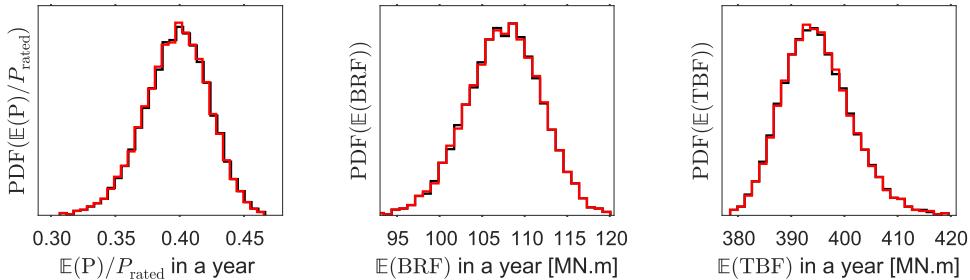


Figure 9: Distribution of yearly distribution of capacity factor and of yearly BRF and TBF equivalent loads. (Black) Surrogate. (Red) Surrogate and surrogate prediction errors.

#### 4. Discussion

The present article presents a methodology to implement sparse polynomial surrogates for aeroelastic wind turbine models. PCE are widely used in the uncertainty quantification field due to their efficiency to compute the statistical properties of the output and because the sensitivity analysis is obtained without any additional effort. The main two limitations in the use of PCE for wind energy are: (1) The input distribution of atmospheric parameters are usually jointly distributed with several layers of dependency (2) Some of the output have discontinuities and/or are restricted to certain values (e.g. only positive). The present article has shown how to solve this

two problems: the implementation of an iso-probabilistic transformation to de-correlate the inputs, and the use of a logistic transformation to implement restrictions on the outputs. The benefits of using the logistic transformation can be seen in figure 5, note that the PCE do not present oscillations in the constant regions, as it can be seen for  $P_E$ , for  $P_S$  and  $CT_S$ .

The final surrogate can be used to generate an output sample that covers the full output space, and that will predict the general details of the distributions of the outputs. One of the main limitations of the present surrogates is that the local distribution of the output is assumed to be normal, this is not the case for the region close to rated wind speed, but since this assumption only affects the turbulent inflow realization then it is considered to be a good enough approximation. The local distribution of most outputs are not normal in reality, because the wind turbine controller has different goals in each region, which creates skewness in the local distributions.

The results presented in this article show that there are several correlations between the input variables and between inputs and outputs. Such complicated inter dependencies are difficult to capture when applying other methods such as interpolation or Gaussian processes. For example, advanced interpolation methods such as radial basis functions will fail to understand the likelihood of an extreme training point and will generate trends that always pass through all the model observations. This behavior penalizes the capacity of the surrogate to generalize and to predict the output in new conditions. The sparse PCE are ideal for this sort of problem because the k-fold cross validation is a step inside the training. Additionally, the correlations between the outputs are fully captured when using the presented surrogates; this occurs because each of the outputs has a dependency on the inputs. The full pair plot of the training dataset and the resulting surrogate for all inputs and outputs is presented in the acompa.

The final results shows a promising new approach to communicate the performance of a wind turbine between the turbine manufacturers and the project operators. It is common practice in the wind energy industry to not share the aeroelastic information of a wind turbine because it will conflict with the manufacturer's intellectual property. Because of this, the wind project planners and operators do not have enough information about the expected performance of a turbine at the site they are developing. On top of this, there is no model for the uncertainty of the turbine performance. We propose that the multiple polynomial surrogate should be fitted by the manufacturer and then, the surrogate can be shared with all their users and clients.

In this way, the turbine manufacturers can keep the information about their turbine and controller design classified. By receiving the surrogate of the wind turbine performance the developers will be able to perform uncertainty estimations on their projects which will lead to better estimations of the AEP and the uncertainty in AEP. As an additional benefit, the project developers will obtain the information about the fatigue damage of the turbine, which can lead to wind plant optimization that considers the damage that the turbines experiences through the years of operation. Fatigue information can lead to better operation and maintenance models which will improve the over all estimation of leveled cost of energy and of its uncertainty.

To obtain the  $\text{PDF}(P)$  and  $\text{PDF}(EFL)$  is useful as they can be used for uncertainty estimation of the leveled cost of energy on a yearly basis. The surrogates can be evaluated on a long time series of the local wind resources (in multiple variables) such as the ones predicted by wind resource forecast (WRF) models without considerable extra computational effort. The power surrogate can then predict the inter year variation of energy production while the EFL can be used to estimate the operation and maintenance costs. Such a probabilistic output can be considered to be the input for a decision making tool.

## 5. Conclusions

The presented approach is a promising replacement of the traditional techniques to explore the global behavior of an aeroelastic wind turbine model under multiple uncertain turbulent inflow parameters. A surrogate of the DTU 10 MW RWT under a 4 dimensional turbulent inflow parameters can be built using only 140 input cases (with multiple turbulent inflow realizations per case) and can be used to predict the distribution of the power, thrust coefficient and equivalent fatigue loads on the turbine. In contrast, traditional approaches require in the order of  $20^4$  gridsearch/interpolation (full factorial design with 20 points per dimension) or  $10^5 - 10^6$  variance reduction Monte-Carlo sample of the inputs [23]. Furthermore, the present approach enables to build an uncertainty model around the 10 minutes performance of the turbine that captures the effect of the turbulent inflow realization.

The combined PCE surrogate approach can be used to improve traditional conservative designs in which a worst case scenario for shear and turbulence intensity is considered. The improvement can be to perform reliability based

designs, in which the probability of failure can be estimated and with the advantage of having better control of the effect of the safety factors used.

The present approach can also be used to improve the estimation of the annual energy production and of the lifetime equivalent fatigue load based on site specific characteristics. The combined PCE surrogates can be understood as an extension of the power curve and can be a way to communicate the aero-elastic performance of a turbine between the manufacturer and the operators without compromising intellectual property. The final surrogate is an uncertainty model that predicts the performance of the turbine as a probability distribution for a given inflow condition. This uncertainty model of turbine performance can be easily implemented in different workflows such as: (1) Uncertainty quantification of the energy production inside a wind plant. (2) Robust wind plant layout optimization problem based on AEP and EFL and their corresponding uncertainties.

## 6. Acknowledgments

This work was supported by the International Collaborative Energy Technology R&D Program of the Korea Institute of Energy Technology Evaluation and Planning (KETEP), granted financial resource from the Ministry of Trade, Industry & Energy, Republic of Korea. (No. 20138520021140). The authors will like to thank Michael McWilliam for the suggestion of the use of logistic transformations to enforce strict restrictions to the polynomial surrogates.

## References

- [1] I. E. Commission, et al., Iec 61400-1: Wind turbines part 1: Design requirements, International Electrotechnical Commission.
- [2] P. A. Graf, G. Stewart, M. Lackner, K. Dykes, P. Veers, High-throughput computation and the applicability of monte carlo integration in fatigue load estimation of floating offshore wind turbines, *Wind Energy*.
- [3] H. S. Toft, L. Svenningsen, W. Moser, J. D. Sørensen, M. L. Thøgersen, Assessment of wind turbine structural integrity using response surface methodology, *Engineering Structures* 106 (2016) 471–483.
- [4] A. Natarajan, N. Dimitrov, P. H. Madsen, J. Berg, M. C. Kelly, G. C. Larsen, J. Mann, D. R. Velerst, J. Dalgaard Srensen, H. Toft, I. Abdallah, N.-J.

Tarp-Johansen, T. Krogh, J. Strdahl, C. Eriksson, E. Jrgensen, F. Klint, L. Thesbjerg, Demonstration of a basis for tall wind turbine design, EUDP project final report, Tech. Rep. E-0108, Technical University of Denmark, Department of Wind Energy, Roskilde, Denmark (2016).

- [5] I. Abdallah, A. Natarajan, J. D. Sørensen, Influence of the control system on wind turbine loads during power production in extreme turbulence: Structural reliability, *Renewable Energy* 87 (2016) 464–477.
- [6] A. Clifton, L. Kilcher, J. Lundquist, P. Fleming, Using machine learning to predict wind turbine power output, *Environmental research letters* 8 (2) (2013) 024009.
- [7] A. Clifton, M. Daniels, M. Lehning, Effect of winds in a mountain pass on turbine performance, *Wind Energy* 17 (10) (2014) 1543–1562.
- [8] C. Soize, R. Ghanem, Physical systems with random uncertainties: chaos representations with arbitrary probability measure, *SIAM Journal on Scientific Computing* 26 (2) (2004) 395–410.
- [9] O. Le Maître, O. Knio, H. Najm, R. Ghanem, Uncertainty propagation using wiener–haar expansions, *Journal of computational Physics* 197 (1) (2004) 28–57.
- [10] S.-K. Choi, R. V. Grandhi, R. A. Canfield, C. L. Pettit, Polynomial chaos expansion with latin hypercube sampling for estimating response variability, *AIAA journal* 42 (6) (2004) 1191–1198.
- [11] M. Berveiller, B. Sudret, M. Lemaire, Stochastic finite element: a non intrusive approach by regression, *European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique* 15 (1-3) (2006) 81–92.
- [12] D. Xiu, J. S. Hesthaven, High-order collocation methods for differential equations with random inputs, *SIAM Journal on Scientific Computing* 27 (3) (2005) 1118–1139.
- [13] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, *Journal of Computational Physics* 230 (6) (2011) 2345–2367.
- [14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in python, *The Journal of Machine Learning Research* 12 (2011) 2825–2830.

- [15] R. Tibshirani, Regression shrinkage and selection via the lasso, *Journal of the Royal Statistical Society. Series B (Methodological)* (1996) 267–288.
- [16] M. Rosenblatt, Remarks on a multivariate transformation, *The annals of mathematical statistics* 23 (3) (1952) 470–472.
- [17] P. Y. Simard, Y. A. LeCun, J. S. Denker, B. Victorri, Transformation invariance in pattern recognition tangent distance and tangent propagation, in: *Neural networks: tricks of the trade*, Springer, 1998, pp. 239–274.
- [18] P. Moriarty, Database for validation of design load extrapolation techniques, *Wind Energy* 11 (6) (2008) 559.
- [19] A. Natarajan, D. R. Verelst, Outlier robustness for wind turbine extrapolated extreme loads, *Wind Energy* 15 (5) (2012) 679–697.
- [20] C. Tibaldi, L. C. Henriksen, C. Bak, Investigation of the dependency of wind turbine loads on the simulation time, in: European Wind Energy Conference & Exhibition 2014, 2014.
- [21] C. Bak, R. Bitsche, A. Yde, T. Kim, M. H. Hansen, F. Zahle, M. Gaunaa, J. P. A. A. Blasques, M. Døssing, J.-J. Wedel Heinen, et al., Light rotor: The 10-mw reference wind turbine, in: EWEA 2012-European Wind Energy Conference & Exhibition, 2012.
- [22] W. Gautschi, Algorithm 726: Orthpol—a package of routines for generating orthogonal polynomials and gauss-type quadrature rules, *ACM Transactions on Mathematical Software (TOMS)* 20 (1) (1994) 21–62.
- [23] N. Dimitrov, A. Natarajan, M. Kelly, Model of wind shear conditional on turbulence and its impact on wind turbine loads, *Wind Energy* 18 (11) (2015) 1917–1931. doi:10.1002/we.1797.
- [24] J. Feinberg, H. P. Langtangen, Chaospy: An open source tool for designing methods of uncertainty quantification, *Journal of Computational Science* 11 (2015) 46–57.
- [25] M. D. McKay, R. J. Beckman, W. J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics* 42 (1) (2000) 55–61.
- [26] I. M. Sobol', On the distribution of points in a cube and the approximate evaluation of integrals, *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki* 7 (4) (1967) 784–802.

- [27] J. M. Hammersley, Monte carlo methods for solving multivariable problems, *Annals of the New York Academy of Sciences* 86 (3) (1960) 844–874.
- [28] I. M. Sobol, Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates, *Mathematics and computers in simulation* 55 (1) (2001) 271–280.
- [29] S. E. Maxwell, H. D. Delaney, Designing experiments and analyzing data: A model comparison perspective, Vol. 1, Psychology Press, 2004.
- [30] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, S. Tarantola, Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index, *Computer Physics Communications* 181 (2) (2010) 259–270.
- [31] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, *Reliability Engineering & System Safety* 93 (7) (2008) 964–979.
- [32] M. S. Eldred, A. A. Giunta, B. G. van Bloemen Waanders, S. F. Wojtkiewicz, W. E. Hart, M. P. Alleva, DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 4.1 reference manual, Citeseer, 2007.
- [33] S. Marelli, B. Sudret, UQLab: a framework for uncertainty quantification in MATLAB, ETH-Zürich, 2014.
- [34] G. Andrianov, S. Burriel, S. Cambier, A. Dutfoy, I. Dutka-Malen, E. De Rocquigny, B. Sudret, P. Benjamin, R. Lebrun, F. Mangeant, et al., Open turns, an open source initiative to treat uncertainties, risks n statistics in a structured industrial approach, in: *Proceedings of ESREL*, 2007.
- [35] T. J. Larsen, A. M. Hansen, How 2 HAWC2, the user's manual, Risø National Laboratory, 2007.
- [36] J. Mann, Wind field simulation, *Probabilistic Engineering Mechanics* 13 (4) (1998) 269 – 282. doi:10.1016/S0266-8920(97)00036-2.
- [37] M. A. Miner, et al., Cumulative damage in fatigue, *Journal of applied mechanics* 12 (3) (1945) 159–164.

- [38] S. Hosder, R. W. Walters, M. Balch, Efficient sampling for non-intrusive polynomial chaos applications with multiple uncertain input variables, in: Proceedings of the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, AIAA paper, Vol. 1939, 2007.

## Appendix A. Legendre polynomials

$$\begin{aligned}
 \phi_0(w) &= 1 \\
 \phi_1(w) &= w - 0.5 \\
 \phi_2(w) &= w^2 - w + 0.166666666667 \\
 \phi_3(w) &= w^3 + -1.5w^2 + 0.6w - 0.05 \\
 \phi_4(w) &= w^4 - 2.0w^3 + 1.28571428571w^2 - 0.285714285714w + 0.0142857142857 \\
 \phi_5(w) &= w^5 - 2.5w^4 + 2.22222222222w^3 - 0.83333333333w^2 + 0.119047619048w \\
 &\quad - 0.00396825396825 \\
 \phi_6(w) &= w^6 - 3.0w^5 + 3.40909090909w^4 - 1.81818181818w^3 + 0.454545454545w^2 \\
 &\quad - 0.0454545454545w + 0.00108225108225
 \end{aligned} \tag{A.1}$$

## Appendix B. DTU 10 MW RWT surrogates

Output	$a_1$	$a_0$
$P_E$	$9.5 \times 10^{-1}$	0
$CT_E$	1	$-5 \times 10^{-2}$
$BRF_E$	$2.2 \times 10^{-2}$	0
$TBF_E$	$1 \times 10^{-2}$	0
$TBS_E$	$1.4 \times 10^{-2}$	0
$TTT_E$	$3.3 \times 10^{-2}$	0
$TTY_E$	$3.3 \times 10^{-2}$	0
$P_S$	22	$1 \times 10^{-4}$
$CT_S$	5	$1 \times 10^{-4}$
$BRF_S$	$1.7 \times 10^{-1}$	$1 \times 10^{-4}$
$TBF_S$	$2.5 \times 10^{-2}$	$1 \times 10^{-4}$
$TBS_S$	$2.5 \times 10^{-2}$	$1 \times 10^{-4}$
$TTT_S$	$1.7 \times 10^{-1}$	$1 \times 10^{-4}$
$TTY_S$	$1.7 \times 10^{-1}$	$1 \times 10^{-4}$

Table B.7: Logistic transformation constants.

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	1	0	0	0	17.824827
1	2	0	0	0	-78.633549
2	3	0	0	0	219.315687
3	4	0	0	0	-243.351228
4	5	0	0	0	90.916652
5	3	1	0	0	2.182398
6	0	1	0	0	-0.236267
7	1	1	0	0	0.674163
8	2	1	0	0	-2.880281
9	1	2	0	0	-0.065094
10	0	2	0	0	0.407897
11	0	3	0	0	-0.201789
12	1	0	1	0	0.062216
13	2	0	1	0	0.419585
14	0	1	1	0	0.390699
15	0	2	1	0	-0.661779
16	0	3	1	0	0.441186
17	0	0	1	0	-0.920436
18	1	0	2	0	-0.412364
19	0	1	2	0	0.107086
20	0	0	2	0	1.012321
21	0	0	3	0	-0.356882
22	1	0	1	0	-0.139125
23	2	0	1	0	0.085514
24	1	1	0	0	0.123286
25	0	1	0	1	0.685970
26	0	2	0	1	-0.145333
27	0	0	1	1	1.956024
28	1	0	1	1	-0.012298
29	2	0	1	1	-0.279339
30	0	1	1	1	-0.432207
31	0	0	2	1	-1.986752
32	0	0	3	1	0.134547
33	0	0	0	1	-0.453163
34	1	0	2	1	0.413360
35	0	1	0	2	-0.798297
36	0	0	1	2	-1.339587
37	0	0	0	2	0.384846
38	0	0	2	2	1.779210
39	0	1	0	3	0.520667
40	0	0	1	3	-0.401148
41	0	0	0	0	-3.033096
42	0	0	0	3	-0.069222

Table B.8:  $P_E$  PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	2	0	0	0	-134.127728
1	4	0	0	0	-783.293694
2	5	0	0	0	379.323723
3	3	1	0	0	92.588283
4	4	1	0	0	-54.384443
5	1	2	0	0	21.703299
6	0	2	0	0	-20.471705
7	0	3	0	0	14.812553
8	1	3	0	0	-13.073937
9	2	0	1	0	9.071876
10	3	0	1	0	-6.264682
11	0	2	1	0	16.698677
12	0	3	1	0	-16.131545
13	0	2	2	0	6.641359
14	1	0	0	1	8.684951
15	2	1	0	1	-0.366464
16	0	2	0	1	0.258849
17	1	1	1	1	13.799775
18	0	0	1	1	10.511176
19	1	0	1	1	-36.179227
20	2	0	1	1	-0.366464
21	0	1	1	1	-30.322117
22	1	1	2	1	-1.887029
23	1	0	0	2	1.465741
24	1	0	1	2	1.603893
25	0	1	1	2	32.674897
26	0	0	1	2	1.698385
27	0	1	2	2	-38.553710
28	0	0	0	2	0.061389
29	0	0	2	2	9.876207
30	1	1	0	3	5.859210
31	0	1	0	3	6.117555
32	0	1	0	3	3.098249
33	1	0	0	0	16.459288
34	3	0	0	0	513.554006
35	0	1	0	0	6.573813
36	1	1	0	0	4.075511
37	2	1	0	0	-47.681732
38	2	2	0	0	-1.833545
39	1	0	1	0	8.983489
40	0	1	1	0	-6.126051
41	1	1	1	0	-5.698179
42	0	0	1	0	-1.208320
43	2	1	1	0	0.650296
44	1	1	0	2	-0.851140
45	1	0	2	0	-2.751493
46	0	1	2	0	4.006299
47	0	0	2	0	-0.357406
48	1	1	2	0	0.943515
49	1	0	3	0	-8.754990
50	0	1	3	0	-6.986286
51	1	1	0	1	-14.029346
52	1	2	0	1	-0.517698
53	2	1	1	1	0.732929
54	0	0	2	1	-15.338654
55	1	0	0	1	0.9230817
56	2	0	0	1	-57.123284
57	3	0	0	1	360.129037
58	1	1	0	0	6.551412
59	2	1	0	0	-13.178827
60	1	2	0	0	-10.882394
61	2	2	0	0	3.012151
62	3	2	0	0	2.228364
63	1	3	0	0	11.486608
64	2	3	0	0	-4.152913
65	4	0	1	0	-5.319179
66	3	1	1	0	0.662732
67	0	1	1	0	8.984505
68	0	2	1	0	-9.350190
69	0	3	1	0	3.926594
70	0	0	1	0	-3.893998

Table B.9:  $P_S$  PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
71	1	0	2	0	-3.635569
72	2	0	2	0	-6.470444
73	3	0	2	0	6.374993
74	1	1	2	0	15.183301
75	2	1	2	0	-3.865114
76	1	2	2	0	-3.898113
77	0	1	3	0	5.761567
78	0	2	3	0	-1.239554
79	0	0	3	0	-10.324073
80	1	0	0	1	0.319789
81	4	0	0	1	-1.499582
82	3	1	0	1	0.421050
83	0	1	0	1	3.787639
84	1	2	0	1	7.826755
85	1	3	0	1	-5.966679
86	0	3	0	1	16.844501
87	0	4	0	1	-5.764745
88	1	0	1	1	-2.111665
89	1	1	1	1	-0.441080
90	0	1	1	1	-5.536196
91	1	0	2	1	2.840954
92	0	1	2	1	10.786568
93	0	1	3	1	-4.628595
94	0	1	0	2	3.239449
95	1	0	1	2	1.621912
96	0	0	0	2	-4.623801
97	0	2	0	2	-0.600983
98	1	0	2	2	-2.640482
99	0	0	0	3	6.205667
100	1	0	0	3	-9.102199
101	0	2	0	3	0.343968
102	0	0	1	3	-1.377748
103	0	1	1	3	0.458134
104	0	0	2	3	0.040432
105	1	0	0	4	2.802899
106	0	1	0	4	0.800728
107	0	0	1	4	0.696161

Table B.10:  $CT_E$  PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	1	0	0	0	-1.588617
1	2	0	0	0	6.960288
2	3	0	0	0	-13.07366
3	4	0	0	0	3.658774
4	0	1	0	0	1.268715
5	1	1	0	0	-1.307230
6	2	1	0	0	0.893358
7	0	2	0	0	-1.584460
8	0	3	0	0	1.124566
9	1	0	1	0	0.096560
10	0	1	1	0	0.570601
11	0	2	1	0	-0.279196
12	0	0	1	0	-0.193227
13	0	1	2	0	-0.003906
14	0	0	2	0	-0.065941
15	1	0	0	1	0.015114
16	2	0	0	1	-0.124400
17	0	1	0	1	0.231847
18	0	0	1	1	0.127705
19	0	1	1	1	-0.575000
20	0	0	2	1	0.017503
21	0	0	0	1	0.037185
22	1	0	0	2	0.172970
23	0	0	1	2	0.207437
24	0	0	0	2	-0.249519
25	0	0	0	0	-1.697421

Table B.11:  $CT_S$  PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	4	0	0	0	62.694878
1	5	0	0	0	-62.819279
2	6	0	0	0	26.414390
3	3	1	0	0	5.073870
4	4	1	0	0	-1.010211
5	0	1	0	0	-1.232956
6	0	2	0	0	3.032444
7	0	3	0	0	-1.380097
8	0	4	0	0	-2.786755
9	0	5	0	0	2.762271
10	1	0	1	0	1.593698
11	2	0	1	0	1.306560
12	3	0	1	0	-1.289262
13	1	1	1	0	-1.960311
14	2	1	1	0	0.462671
15	1	2	1	0	1.284869
16	0	1	2	0	2.583334
17	0	2	2	0	0.429339
18	0	0	2	0	-0.068526
19	1	0	3	0	-1.296834
20	2	0	3	0	1.644857
21	1	1	3	0	1.386674
22	0	1	4	0	1.237224
23	0	0	4	0	-0.374613
24	2	0	0	1	-6.958940
25	3	0	0	1	5.302168
26	1	1	0	1	-0.544489
27	2	1	0	1	-0.096270
28	0	2	0	1	-9.320285
29	0	0	1	1	-1.387413
30	1	1	2	1	2.055372
31	0	2	2	1	0.274349
32	0	0	2	1	2.407023
33	0	0	4	1	1.505527
34	0	0	0	1	-0.481728
35	0	2	1	1	-0.274349
36	0	0	3	1	-3.385077
37	1	0	0	2	-2.410568
38	2	0	0	2	5.058386
39	3	0	0	2	-3.556947
40	1	1	0	2	0.679389
41	0	1	1	2	-0.627927
42	0	0	1	2	-0.107242
43	0	0	2	2	2.167533
44	0	1	0	3	3.995469
45	1	0	1	3	0.335097
46	0	0	0	0	-1.621127
47	1	0	0	0	-1.236436
48	2	0	0	0	16.592903
49	3	0	0	0	-40.158224
50	1	1	0	0	4.115606
51	2	1	0	0	-6.819207
52	1	2	0	0	-2.273269
53	2	2	0	0	1.655239
54	3	2	0	0	-0.811717
55	1	3	0	0	0.445370
56	4	0	1	0	0.589926
57	3	1	1	0	0.218822
58	0	1	1	0	2.121804
59	0	2	1	0	-7.253108
60	0	3	1	0	8.743509
61	0	4	1	0	-4.116381
62	0	0	1	0	-0.998033
63	1	0	2	0	1.097308
64	2	0	2	0	-2.685648
65	1	1	2	0	-1.639455
66	0	1	3	0	-3.600087
67	0	0	3	0	1.081131
68	1	0	0	1	2.818352
69	0	1	0	1	4.830761
70	1	2	0	1	0.135315

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
71	0	3	0	1	9.122104
72	0	4	0	1	-4.121466
73	1	0	1	1	0.121521
74	2	0	1	1	0.453509
75	1	1	1	1	-1.363471
76	0	1	1	1	2.146573
77	1	0	2	1	-1.027686
78	0	1	2	1	-2.642740
79	0	1	3	1	0.864604
80	0	1	0	2	-6.806370
81	0	2	0	2	6.654701
82	1	0	1	2	0.234386
83	0	0	0	2	0.156466
84	0	3	0	2	-0.879173
85	0	0	0	3	1.039756
86	1	0	0	3	-0.165016
87	0	2	0	3	-3.557294
88	0	0	1	3	-0.112527
89	0	0	2	3	-1.312607
90	0	0	0	4	-0.680624

Table B.12:  $\text{BRF}_{\mathbb{E}}$  PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	1	0	0	0	3.265241
1	2	0	0	0	-8.920657
2	3	0	0	0	-17.700217
3	4	0	0	0	44.803873
4	5	0	0	0	-22.380164
5	0	1	0	0	1.010208
6	1	1	0	0	-0.840001
7	2	1	0	0	0.216462
8	1	2	0	0	0.010459
9	0	2	0	0	-1.764214
10	0	3	0	0	1.537057
11	1	0	1	0	0.187639
12	2	0	1	0	-0.218385
13	0	1	1	0	1.742232
14	1	1	1	0	-0.032924
15	0	2	1	0	-1.036063
16	0	0	1	0	-0.158630
17	1	0	2	0	0.063194
18	0	1	2	0	-0.290302
19	0	0	2	0	-0.311188
20	0	0	3	0	0.012734
21	1	0	0	1	-0.721490
22	2	0	0	1	1.069445
23	3	0	0	1	-0.471808
24	1	1	0	1	0.701628
25	2	1	0	1	-0.432925
26	0	1	0	1	1.772373
27	0	2	0	1	-0.906239
28	0	0	1	1	0.394469
29	0	1	1	1	-4.783365
30	0	0	2	1	1.489605
31	0	1	2	1	1.288322
32	0	0	3	1	-0.484483
33	0	0	0	1	-0.529695
34	0	2	1	1	1.787082
35	0	1	0	2	-0.623935
36	0	1	1	2	1.362023
37	0	0	1	2	-0.495626
38	0	0	0	2	0.592109
39	0	0	2	2	-0.893638
40	0	0	1	3	0.488628
41	0	0	0	3	-0.174881
42	0	0	4	0	-0.196241
43	0	0	0	0	-1.328826

Table B.13:  $\text{BRF}_{\mathbb{S}}$  PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	1	0	0	0	7.492604
1	2	0	0	0	-30.048707
2	3	0	0	0	36.485874
3	4	0	0	0	-13.644361
4	0	1	0	0	2.494525
5	1	1	0	0	-2.510374
6	2	1	0	0	1.552327
7	1	2	0	0	-0.404614
8	0	2	0	0	-2.573943
9	0	3	0	0	2.080285
10	1	0	1	0	-0.000183
11	0	1	1	0	-0.031200
12	1	1	1	0	0.062401
13	0	0	1	0	0.024172
14	1	0	0	1	-0.007061
15	1	1	0	1	0.363697
16	0	1	0	1	0.122950
17	0	2	0	1	-0.300063
18	0	0	1	1	0.013684
19	0	0	0	1	-0.086136
20	1	0	0	2	-0.066080
21	0	0	0	2	0.029179
22	0	0	0	0	-0.720555
23	0	0	0	1	0.135315
24	1	0	1	2	0.111038
25	2	0	1	2	-0.318772
26	0	1	1	2	0.024201
27	1	1	2	1	0.24201
28	0	0	2	1	0.897428
29	1	0	0	2	-0.227329
30	2	1	0	2	-0.628790
31	1	0	1	2	0.111038
32	0	1	1	2	-0.318772
33	0	0	1	2	0.024201
34	0	0	0	2	-0.643182
35	1	0	0	3	-0.313404
36	0	1	0	3	-0.106257
37	0	0	1	3	-0.053350
38	0	0	0	0	-2.444218
39	3	0	0	0	34.800840
40	0	1	0	0	2.392428

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	1	0	0	0	12.057073
1	2	0	0	0	-50.660200
2	3	0	0	0	59.790826
3	4	0	0	0	-21.176517
4	0	1	0	0	1.430674
5	1	1	0	0	-0.310681
6	2	1	0	0	-0.223647
7	0	2	0	0	-1.456880
8	0	3	0	0	0.942868
9	1	0	1	0	0.801769
10	2	0	1	0	-0.398060
11	0	1	1	0	-0.634770
12	0	2	1	0	-0.185424
13	0	0	1	0	0.336571
14	0	1	2	0	1.058729
15	0	0	2	0	-0.824320
16	1	0	0	1	0.000604
17	0	1	0	1	0.356178
18	0	0	1	1	-0.309698
19	0	1	1	1	-0.539143
20	0	0	2	1	0.551080
21	0</td				

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
31	0	0	0	1	1.764704
32	0	0	1	1	-1.301562
33	0	3	1	1	-1.403386
34	1	1	2	1	1.639571
35	0	0	3	1	0.668712
36	1	0	0	2	6.707797
37	2	0	0	2	-2.833838
38	3	0	0	2	-0.204807
39	1	1	0	2	-1.884884
40	0	1	1	2	2.118892
41	0	0	1	2	0.781327
42	0	1	2	2	-0.2068479
43	0	0	2	2	0.856648
44	2	0	0	3	2.071349
45	1	1	0	3	1.072299
46	0	1	0	3	0.171253
47	1	0	1	3	1.986781
48	0	0	0	4	-1.504521
49	0	0	1	4	0.100263
50	0	0	0	0	-3.144030
51	1	0	0	0	6.599387
52	2	0	0	0	-45.341285
53	3	0	0	0	172.955502
54	1	1	0	0	-4.733793
55	2	1	0	0	4.476663
56	1	2	0	0	4.081142
57	2	2	0	0	-0.812557
58	3	2	0	0	0.246905
59	1	3	0	0	-4.342735
60	3	1	1	0	1.005514
61	0	1	1	0	-0.858742
62	0	2	1	0	-1.332080
63	0	3	1	0	0.521297
64	0	4	1	0	-0.142228
65	0	0	1	0	1.247676
66	1	0	2	0	4.465628
67	2	0	2	0	-5.263632
68	3	0	2	0	2.304830
69	1	1	2	0	-1.701401
70	2	1	2	0	0.849900
71	0	1	3	0	-0.772282
72	0	0	3	0	-0.931977
73	1	0	4	0	-0.164284
74	1	0	0	1	-4.312543
75	4	0	0	1	-2.452739
76	3	1	0	1	0.570028
77	0	1	0	1	-1.126498
78	1	2	0	1	-1.177847
79	0	3	0	1	1.278235
80	1	0	1	1	6.296783
81	2	0	1	1	-1.522185
82	1	1	1	1	-5.723955
83	0	1	1	1	-0.900966
84	2	1	1	1	1.808171
85	1	0	2	1	-2.307984
86	0	1	2	1	1.433679
87	0	1	0	2	0.185031
88	0	2	0	2	0.144689
89	1	0	1	2	-3.853572
90	0	0	0	2	-2.831547
91	2	0	1	2	0.278039
92	1	1	1	2	0.443955
93	1	0	2	2	0.355184
94	0	0	0	3	2.855987
95	1	0	0	3	-4.437864
96	0	2	0	3	-0.560837
97	0	0	1	3	-1.193917
98	0	0	0	5	0.568050

Table B.19: TTT<sub>S</sub> PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
-----	-------	-------	-------	-------	-------

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	4	0	0	0	-296.844378
1	5	0	0	0	232.852778
2	6	0	0	0	-68.220514
3	3	1	0	0	-3.652611
4	4	1	0	0	1.308967
5	0	1	0	0	3.367015
6	0	2	0	0	-7.865412
7	0	3	0	0	14.720705
8	0	4	0	0	-13.465583
9	1	4	0	0	2.100920
10	0	5	0	0	4.661932
11	1	0	1	0	-5.645130
12	2	0	1	0	6.548212
13	3	0	1	0	-3.027013
14	1	1	1	0	4.614054
15	2	1	1	0	-3.818255
16	1	2	1	0	0.393950
17	0	1	2	0	1.240415
18	0	2	2	0	0.028452
19	0	0	2	0	-0.378514
20	1	0	3	0	-0.565743
21	2	0	3	0	0.506597
22	1	1	3	0	0.021144
23	0	0	4	0	0.647403
24	2	0	0	1	-0.384179
25	3	0	0	1	4.306313
26	1	1	0	1	4.567803
27	2	1	0	1	-1.656415
28	0	2	0	1	-0.809267
29	0	2	1	1	2.049309
30	0	0	2	1	-0.738809

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	4	0	0	0	-26.193393
1	5	0	0	0	14.668184
2	3	1	0	0	0.701493
3	0	1	0	0	2.398323
4	0	2	0	0	-8.320975
5	0	3	0	0	17.868359
6	0	4	0	0	-18.466403
7	0	5	0	0	7.456144
8	1	0	1	0	-0.984337
9	2	0	1	0	1.137439
10	3	0	1	0	-0.836179
11	1	1	1	0	1.129655
12	2	1	1	0	0.447307
13	1	2	1	0	-1.247363
14	1	3	1	0	1.085180
15	0	1	2	0	2.010482
16	0	2	2	0	-0.425747
17	0	0	2	0	-0.375596
18	1	0	3	0	-0.652325
19	2	0	3	0	0.377566
20	1	1	3	0	0.964014
21	0	1	4	0	0.323884
22	0	0	4	0	0.340476
23	3	0	0	1	0.944969
24	1	1	0	1	-0.055687
25	2	1	0	1	0.416371
26	0	2	0	1	-0.366531
27	0	0	1	3	0.034960
28	0	0	2	3	-0.180979

Table B.20: TTY<sub>E</sub> PCE

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
-----	-------	-------	-------	-------	-------

$j$	$l_0$	$l_1$	$l_2$	$l_3$	$c_j$
0	1	0	0	0	-0.110113
1	2	0	0	0	8.259241
2	3	0	0	0	-13.260383
3	4	0	0	0	6.281290
4	0	1	0	0	1.221769
5	1	1	0	0	-0.130064
6	2	1	0	0	-0.260403
7	0	2	0	0	-1.677454
8	0	3	0	0	1.212444
9	1	0	1	0	0.021723
10	0	1	1	0	0.101919
11	0	2	1	0	-0.092649
12	0	0	1	0	-0.011172
13	1	0	2	0	-0.024707
14	0	0	2	0	-0.025082
15	1	0	0	1	0.084859
16	2	0	0	1	-0.068179
17	1	1	0	1	0.002320
18	0	1	0	1	0.241057
19	0	2	0	1	-0.189773
20	0	0	1	1	-0.030629
21	1	0	1	1	0.005969
22	0	0	2	1	0.012245
23	0	0	0	1	0.054018
24	1	0	0	2	-0.059899
25	0	1	0	2	-0.021154
26	0	0	1	2	0.025697
27	0	0	0	2	-0.089572
28	0	0	0	0	-2.558245

Table B.21: TTY<sub>S</sub> PCE