Faculty of Aerospace Engineering

Using polynomial chaos expansion for wind energy

Wouter Bailleul

June 15, 2018

Delft University of Technology

Challenge the future

Using polynomial chaos expansion for wind energy

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Wouter Bailleul

June 15, 2018

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

Delft University Of Technology
Department Of
Wind Energy and Aerodynamics

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **“Using polynomial chaos expansion for wind energy”** by **Wouter Bailleul** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: June 15, 2018

Readers:

Dr. ir. O. Morales Napoles

Dr. ir. E. R. G. Quaeghebeur

Prof. dr. S. J. Watson

Acronyms

AEP	Annual energy production
BGA	Binary genetic algorithm
CDF	Cumulative probability distribution function
GA	Genetic algorithm
GHA	Greedy heuristic algorithm
GSA	Gradient search algorithm
LCOE	Levelised cost of energy
PCE	Polynomial chaos expansion
PDF	Probability density function
PMF	Probability mass function
WFLOP	Wind farm layout optimisation problem
WT	Wind turbine

List of Symbols

Greek Symbols

α_i	Coefficient of the ith order polynomial
λ	Weibull scale parameter
ϕ_i	Polynomial of order i
θ	Wind direction [°]
$\boldsymbol{\theta}$	Vector of wind directions
$\boldsymbol{\xi}$	Model inputs

Roman Symbols

$\mathbb{E}$	Expectation (operator)
F	Cumulative probability distribution function (CDF)
f	probability density function (PDF)
$f\left(v,\theta\right)$	Joint PDF of wind speed and wind direction
$f_{\mathbf{disc}}$	Joint probability mass function (PMF) of wind speed and wind direction
f_{θ_i}	Probability occurrence of of the ith wind sector
$f\left(v;\lambda,k\right)$	Weibull PDF
$\mathbf{h_y}$	Number of hours in a year (8760)
k	Weibull shape parameter
M	Number of polynomials (order of the basis/polynomial)
m_n	nth raw moment
$m_{n,l}$	Two dimensional moment. A combination of the nth moment of the first random variable and the lth moment of the second random variable

N	Number of wind turbines
n	Wind turbine n
P	Power produced by the wind farm [W]
$\hat{P}$	Power produced by the wind farm (Surrogate model) [W]
$P\left(v,\theta\right)$	Power produced by the wind farm [W]
v	Wind speed [m/s]
$v_{\textbf{cut-in}}$	Cut in wind speed of the wind turbine [m/s]
$v_{\textbf{cut-out}}$	Cut out wind speed of the wind turbine [m/s]
$v_{\textbf{eq}}$	Equivalent wind speed at the wind turbine (wake losses taken in account) [m/s]

Contents

List of Figures

Chapter 1

Introduction

1-1 Background

Wind energy is one of the fastest growing forms of energy production in the world [1]. Often, wind turbines are built in wind farms where they are placed in close proximity to each other. The most important variables of the wind farm design that have an effect on the annual energy production are the number of turbines and their location with respect to each other. These variables influence the wake effects present in the wind farm [2]. The wake effect is the aerodynamic influence of an upstream wind turbine on all downstream turbines. Finding an optimal layout is thus critical to limit the wake effects and to maximise the annual energy production. Various models have been developed to compute the effect of shading by multiple turbines (wake effects of multiple wind turbines) and to incorporate the factor of distance between the turbines [3].

Wind farm layout optimisation is a topic of research which dates back to the nineties, when the idea of using algorithms to optimise the energy production was first introduced by Mosetti et al. [4]. Since then a lot of improvements have been made and now various optimisation algorithms, wind farm models and optimisation objectives exist. In the beginning the focus was mainly on aerodynamics and wake phenomena. Nowadays the objective function generally consists of minimising the levelised cost of energy (LCOE). This can be achieved by optimising the layout, based on the number of turbines, the wake effects [3], the terrain and electrical infrastructure, and environmental impacts (e.g. wildlife and biodiversity, visual impact, noise) [5]. A recent development is an optimisation including risk management and uncertainty [6].

One key aspect of all these optimisation processes is that a lot of expensive aerodynamic wind farm model evaluations are needed. When using computationally expensive models this means that a lot of computing time is needed. Surrogate models are used to approximate the expensive 'true' simulation codes. These models have the potential to speed up the optimisation process and can improve the optimisation of problems that are non-smooth. Additionally, they can provide an insight into the design space [7]. One technique to make surrogate models is polynomial chaos expansion (PCE). PCE approximates a model using orthogonal polynomials which are constructed based on the distribution of the uncertain stochastic variables [8]. In case of the wind farm layout optimisation problem these random variables are the wind speed and wind direction. When using a more general approach with applying PCE to wind energy, other variables can be chosen to investigate for example model uncertainty, or uncertainty in the wind resource.

Up till now, PCE has mainly been used in other fields than wind energy. It has been used a lot in the field of computational structures and also in the aerodynamics world where computationally heavy CFD codes are used. It is mainly used as an uncertainty quantification method and not as much to create a surrogate model in order to help with optimisation problems. While it is possible to use PCE with multiple stochastic variables, it is a requirement that these variables are statistically independent. As a result it is often assumed that partially correlated variables are either independent or fully correlated [9]. An additional difficulty of the application of PCE in the wind energy field is the statistical distribution of the wind direction. Most of the time, the wind direction distribution is presented on a wind rose with a range of 0° to 360°. It has been shown that the conversion to the statistical distribution needed as the input for PCE is not straightforward [10, 11].

This research project investigated the use of polynomial chaos expansion for wind energy where wind speed and wind direction will be the two variables. These variables are correlated (and hence not statistically independent) and not much is known on how to implement PCE in this case and what influence this correlation has on the technique. The project will also look into the ways in which the surrogate model can be constructed and compare them to find the optimal combination for use in wind farm layout optimisation.

1-2 Objectives and research question

Now that the problem has been explained, the research question can be introduced. In the literature review report the following research objective was established:

The research aim of this project *is to* investigate whether polynomial chaos expansion can be used in the wind energy field *by* looking into the challenges with respect to the construction of the surrogate model and correlated random variables that currently exist.

Since the research question is quite broad, it can be split up in various sub questions.

1. How does the method to obtain the orthogonal polynomials influence the construction of the surrogate model?
2. How do the various methods to solve the system of orthogonal polynomials using PCE influence the result?
3. What effects does the use of PCE have on the accuracy of the surrogate model output?
4. How can the wind direction be used in PCE?
5. Can wind speed and wind direction be used in a multivariate PCE?

1-3 Thesis outline

This chapter introduced the problem and corresponding research question. In Chapter 2, an analysis of methods to obtain the annual energy production (AEP) is performed. Polynomial chaos expansion (PCE) is introduced in Chapter 3. In this chapter all the relevant parameters to construct the surrogate model will be analysed. The wind farm layout optimisation problem (WFLOP) and its algorithms are explained in Chapter 4. The results of using PCE in layout optimisation will be presented here. Finally, a conclusion to the research is provided in Chapter 5, together with recommendations for future work.

Chapter 2

Annual energy production of a wind farm

As explained in the introduction, the layout optimisation process requires a lot of model evaluations. As will be explained in chapter 4, one of the most important parameters for this process is the calculation of the annual energy production (AEP). This chapter will show how this can be computed.

2-1 Annual energy production of one turbine

The first step towards the estimation of the AEP of a whole wind farm is to look at one turbine. The power generated by a wind turbine is a function of wind speed and is described by the power curve. An example power curve is shown in figure 2-1.

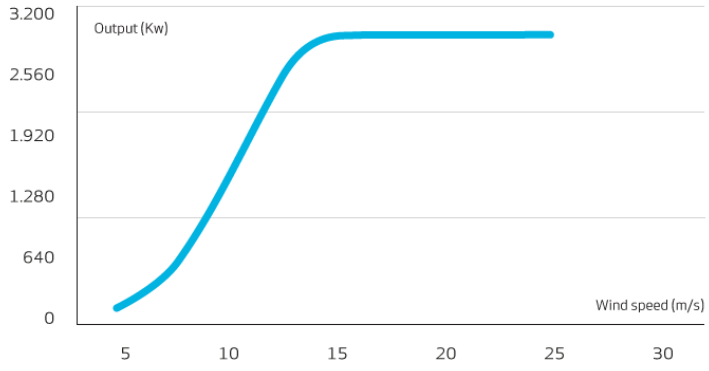


Figure 2-1: Power curve of a Vestas V90 wind turbine. Taken from [12].

Not only information about the wind turbine is needed, but also information about the wind resource. This is provided in the form of a probability density function (PDF), most often a

Weibull distribution

$$f\left(v;\lambda,k\right)=\frac{k}{\lambda}\left(\frac{v}{\lambda}\right)^{k-1}\exp\left(-\left(\frac{v}{\lambda}\right)^{k}\right)\,, \tag{2-1}$$

where v is the wind speed, λ is the Weibull scale parameter and k is the Weibull shape parameter [13]. These last two parameters are obtained by fitting a Weibull distribution to measured data. This is shown in figures 2-2 and 2-3. From these figures it can be seen that it is important to pay attention of the parameter fitting to the data. While the left figure shows a good correspondence between the data and the fitted Weibull distribution, this is certainly not the case for the right figure. Additionally it is shown that multiple methods exist to estimate the Weibull parameters (see legend of figure 2-3). The choice of estimation method and the result of the fit can all have an influence on the final AEP estimate [14, 15].

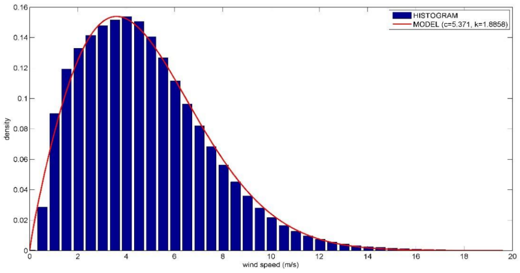


Figure 2-2: Fitted Weibull distribution and original data. Take from [16].

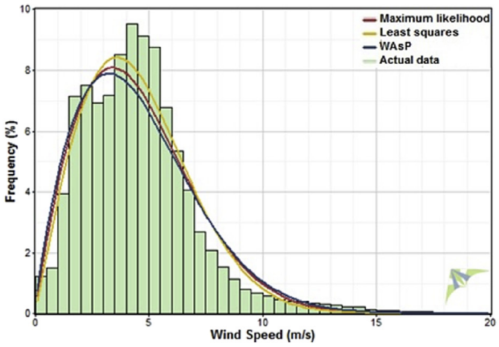


Figure 2-3: Fitted Weibull distribution and original data. Taken from [14].

Knowing the power curve of the wind turbine and the Weibull distribution of the wind climate, the AEP can be estimated as

$$\mathrm{AEP}=\mathrm{h_y}\int_{v_{\text{cut-in}}}^{v_{\text{cut-out}}}P\left(v\right)f\left(v;\lambda,k\right)\mathrm{d}v\,, \tag{2-2}$$

where $\mathrm{h_y}$ is the number of hours in a year, $v_{\text{cut-in}}$ and $v_{\text{cut-out}}$ are the cut-out and cut-in wind speed of the wind turbine and $P\left(v\right)$ is the power produced by the wind turbine at a certain wind speed (given by the power curve). When the AEP of only one wind turbine is computed, the wind direction does not have an influence on the result. In this case, the Weibull distribution is computed for all data without selecting specific wind direction sectors.

2-2 Annual energy production of a wind farm

When computing the AEP of a wind farm, things get more complicated due to the wake effects of the wind turbines in the wind farm. This means that in addition to the wind speed, also wind direction plays a role. This results in the need for more information about the wind direction distribution, which is most often represented in the form of a wind rose. A wind rose is a graphical representation of how the wind speed and wind direction are distributed. It shows the frequency of time that the wind blows from a particular wind direction sector at a particular wind speed. An example of such a wind rose is shown in Figure 2-4.

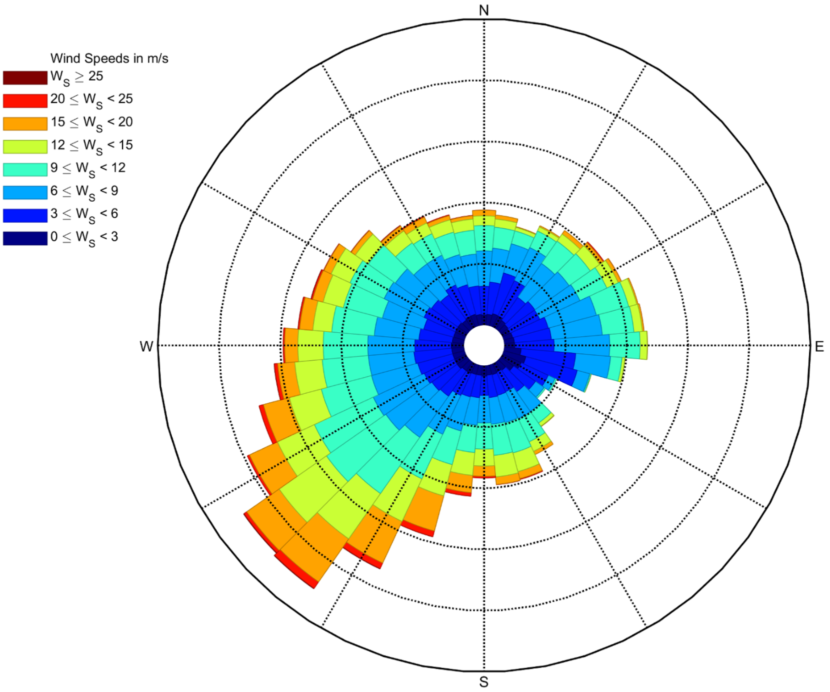


Figure 2-4: The wind rose corresponding to the data set used in this research.

The wind speed indicated by this wind rose is the undisturbed wind speed. The estimated wind speed inside the wind farm depends on the wake model that is applied. This means that for a combination of wind speed and wind direction each turbine can experience its own wind speed (v_{eq}) due to these wake effects. The AEP of the wind farm can then be computed as

$$\text{AEP} = \text{h}_\text{y} \int_{0^\circ}^{360^\circ} \int_{v_{\text{cut-in}}}^{v_{\text{cut-out}}} \left(\sum_{n=1}^{N} P\left(v_{\text{eq},n}\right) \right) f\left(v,\theta\right) \text{d}v\, \text{d}\theta \,, \tag{2-3}$$

where θ is the wind direction, N is the number of turbines in the wind farm and n represents the nth turbine. It has to be noted that $f\left(v,\theta\right)$ now represents the joint probability distribution of wind speed and wind direction.

When a wind farm wake model is used, there is no analytical solution for the total power produced by the wind farm and thus also not for the integrals in Equation (2-3). It becomes necessary to do a numerical integration. This can be done by dividing the wind direction in wind direction sectors and the wind speed in wind speed sectors. For each resulting sector (see Figure 2-4), the corresponding probability of occurrence (f_{disc}) is computed. At each combination of wind speed and wind direction the wind farm model is evaluated ($P\left(v,\theta\right)$) and multiplied by the probability of this wind event. Finally all the values are summed and multiplied by the amount of hours in a year:

$$\text{AEP} = \text{h}_\text{y} \sum_{\theta=0^\circ}^{359^\circ} \sum_{v=v_{\text{cut-in}}}^{v_{\text{cut-out}}} P\left(v,\theta\right) f_{\text{disc}}\left(v,\theta\right) \,, \tag{2-4}$$

where f_{disc} is the cumulative joint probability mass function.

In most papers on wind farm layout optimisation, the width of these sectors and intervals is not indicated. However, Feng et al [16] compared various sector and interval sizes and recommend to use values of $\Delta v = 1\,\mathrm{m/s}$ and $\Delta\theta = 1°$ for use in wind farm layout optimisation. Since the cut-in speed of a wind turbine is most often 3 m/s and the cut-out speed 25 m/s, this means that in this case $23 \cdot 360 = 8280$ model evaluations are needed to compute the AEP of the wind farm. For other applications, a wind sector width of 30° could be used which corresponds to 272 model evaluations. It has to be noted that when the wind speed is below the cut-in speed or above the cut-out speed the model returns a zero power output without the need for evaluation of the expensive wake model.

Additionally, it needs to be stated that in a lot of papers, the authors use a constant wind speed (e.g. 8 m/s) with a varying wind direction in order to perform the layout optimisation process. As an example, Grady et al [17] used three cases to test their optimisation process:

- $\theta = 0°$ and $v = 12\,\mathrm{m/s}$
- $\theta = 0°,\ 10°,\ 20° \ldots 360°$ and $v = 12\,\mathrm{m/s}$
- $\theta = 0°,\ 10°,\ 20° \ldots 360°$ and $v = [8,\ 12,\ 17]\,\mathrm{m/s}$

All three cases limited the amount of model evaluations needed to compute the AEP while using very simple input distributions. This makes it easier and less computationally expensive to obtain results. It is the aim of this research to implement polynomial chaos expansion (PCE) in the optimisation process. As a result, data will be used with a real 2D distribution of wind directions and wind speeds without being limited to a set of wind speeds and directions. The use of all available wind speeds and directions should have a positive effect on the accuracy of the AEP estimation.

An alternative to the division of the wind data into sectors for wind speed and wind directions is to fit a Weibull distribution per wind direction sector. Provided that there is enough data available, this sectorwise Weibull distribution can also be a valid fit to the data. This approach is shown in Equation (2-5).

$$\mathrm{AEP} = \mathrm{h_y} \sum_{i=1}^{I-1} f_{\theta_i} \int_{v_{\text{cut-in}}}^{v_{\text{cut-out}}} P\left(v, \frac{\boldsymbol{\theta}_i + \boldsymbol{\theta}_{i+1}}{2}\right) f\left(v; \lambda_i, k_i\right) \mathrm{d}v \tag{2-5}$$

$$\text{with}$$

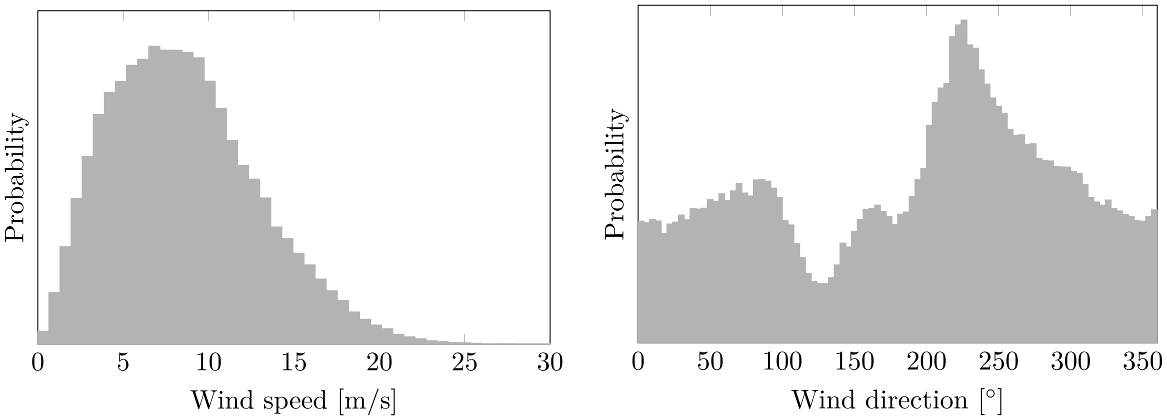
$$\boldsymbol{\theta} = \left\{0°, \frac{360°}{I-1}, \ldots, 360°\right\} \qquad \text{with } I \text{ elements}$$

In this equation, λ_i and k_i represent the parameters of the Weibull fit to the dataset for $\boldsymbol{\theta}_i \leq \theta < \boldsymbol{\theta}_{i+1}$ and f_{θ_i} is the probability of occurrence of each wind sector i. The accuracy of this method depends on the amount of available data and whether a good fit with the Weibull distribution is possible for each sector.

2-3 Wind Data

In this research a time series of wind speeds and wind directions from a met mast at the 'Offshore Windpark Egmond aan Zee' (OWEZ) [18] will be used. This wind farm was the first big offshore wind farm in The Netherlands and started producing electricity in April 2007, but a met mast was built before that. The timeseries starts on the first of July 2005 and ends on the first of January 2011. It consists of 10 minute averaged values for a total of 276 840 valid measurements. The measurements at a height of 70 m are extracted and used as the basis for the wind speed and wind direction data in this research. In Figure 2-4 a wind rose of this dataset is shown. It is clear that the most important wind direction is South-West, which is in agreement with other measurement campaigns in the North Sea [19, 20].

In Figure 2-5, the histograms for the wind speed and wind direction can be found. It seems that a Weibull fit for the wind speed (Figure 2-5(a)) would be a very good option. However there does not seem to be a 'standard' statistical distribution that could fit the wind direction (Figure 2-5(b)). This will be further elaborated in the next chapter.



(a) Histogram of the wind speed data.

(b) Histogram of the wind direction data.

Figure 2-5: Histogram for the full data set of wind speeds and wind directions at OWEZ at a height of 70 m.

In Figure 2-6, a histogram of the wind speed is shown for each wind sector with a width of 20°. The wind sector is indicated for each subplot, together with the probability that the wind blows from this sector given in percent. The probability on the y-axis of each graph is the probability for that specific event in the wind sector. It is thus not a probability for the wind speed and wind direction sector in the general data set. This last number could be obtained by multiplying the number on the y-axis with the probability of the wind direction sector from the title of the subplot.

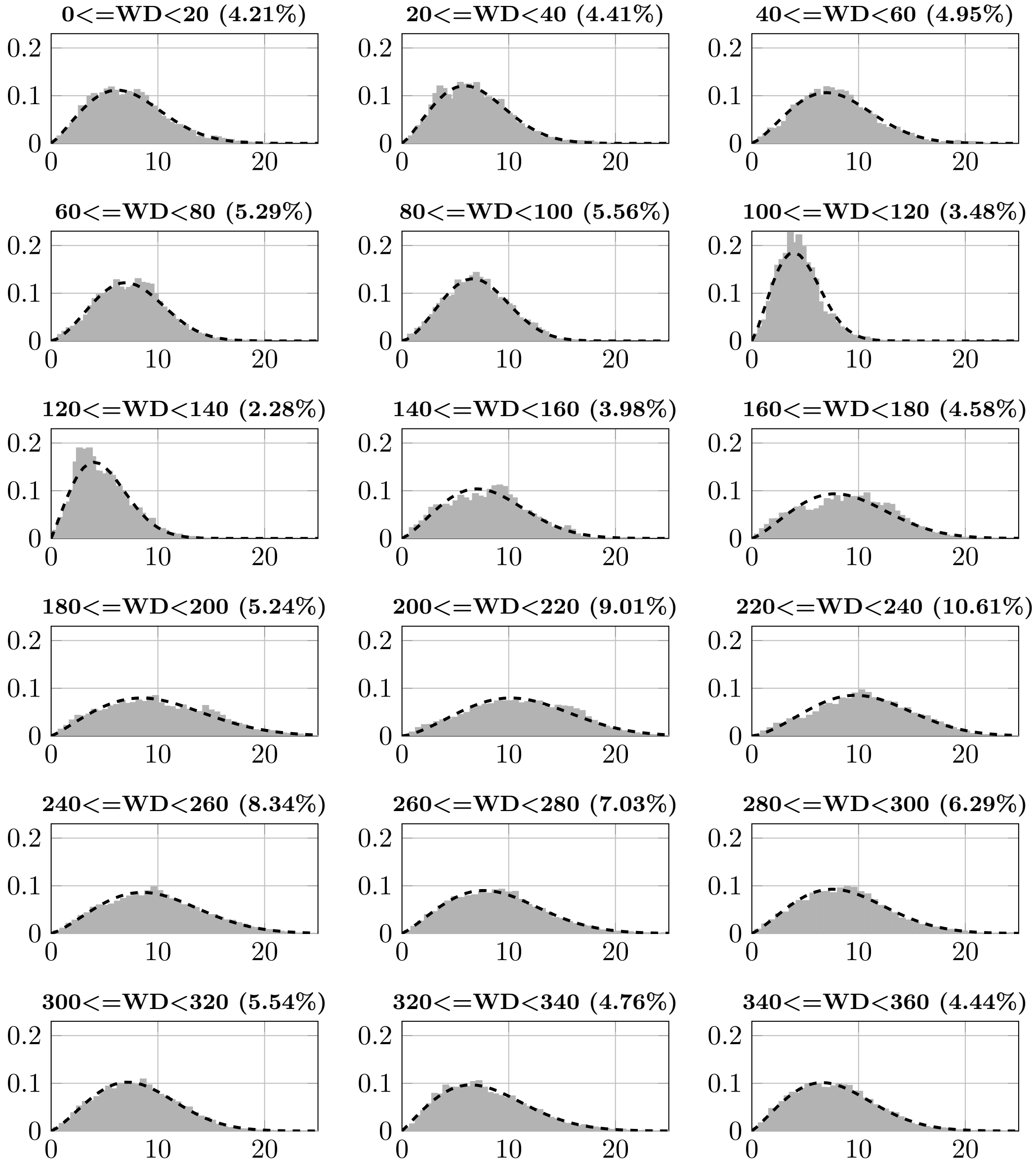


Figure 2-6: Histogram and Weibull fit of wind speed per wind direction sector.

2-4 Wind farm model

The wind farm model used in the research project is the model developed by Bo Hu during his master thesis at the Delft University of Technology [21]. It is an extension of the Bastankhah and Porté-Agel model (BP model). During the research presented in this document, the model was treated as a black box. No research was done on the model itself. It is merely treated as a function which computes the output of the wind farm when some inputs are provided (wind speed, wind direction, location of the turbines and their properties).

Theory of the wind farm model

Bo Hu stated three reasons to use this BP model [21]:

- It is a kinematic model which means the BP model is less computationally expensive than a lot of other wind farm models.
- The BP model has a very high accuracy with respect to the modelling of the velocity deficits in the wind farm.
- The model is easy to implement since it offers a closed-from solution.

The model is derived from the simplified momentum equation

$$F_T = \int_{A_w} \rho U \left(U_0 - U\right) \mathrm{d}A_w \, , \tag{2-6}$$

where F_T is the rotor thrust force, A_w is the cross-sectional area of the wake, ρ is the air density, U_0 is the ambient wind speed and U is the wind speed in the wake. The BP model does not assume a uniform velocity profile in the wake and uses an axis-symmetric Gaussian distribution for the velocity deficit in the wake. The deficit can be normalised and is defined as

$$\frac{\Delta U}{U_0} = C\left(x\right) \exp\left(\frac{-r^2}{2\sigma_u^2}\right) \, , \tag{2-7}$$

where $C\left(x\right)$ is the maximum normalised velocity deficit at the centre of the wake at each downwind location, r is the radial distance from this location to the centre of the wake and σ_u is the standard deviation of the Gaussian velocity deficit profile.

The thrust force can also be expressed in terms of a thrust coefficient

$$F_T = \frac{1}{2} C_T \rho A U_0^2 \, , \tag{2-8}$$

where A is the rotor area.

Equations (2-6) to (2-8) can be combined and an integration from 0 to ∞ is performed. The following result is obtained

$$8\left(\frac{\sigma_u}{D}\right)^2 C\left(x\right)^2 - 16\left(\frac{\sigma_u}{D}\right)^2 C\left(x\right) + C_T = 0 \, , \tag{2-9}$$

where D is the rotor diameter. From this equation, an expression for $C\left(x\right)$ can be obtained:

$$C\left(x\right) = 1 - \sqrt{1 - \frac{C_T}{8\left(\frac{\sigma_u}{D}\right)^2}} \, . \tag{2-10}$$

The BP model uses a linear expansion of the wake with a wake radius of $2\sigma_u$.

$$\frac{\sigma_u}{D} = k^* \frac{x}{D} + \epsilon \, , \tag{2-11}$$

where k^* is the wake growth rate and ϵ is an empirical correction factor. Equation (2-10) and Equation (2-11) can be substituted in Equation (2-7) with the following result:

$$\frac{\Delta U}{U_0} = \left(1 - \sqrt{1 - \frac{C_T}{8\left(\frac{k^* x}{D} + \epsilon\right)^2}}\right) \exp\left(\frac{-1}{2\left(\frac{k^* x}{D} + \epsilon\right)^2}\left(\frac{r}{D}\right)^2\right) . \tag{2-12}$$

The BP model was compared with the Frandsen model, which is derived from the same set of equations, and ϵ was determined to be equal to

$$\epsilon = 0.2\sqrt{\frac{1+\sqrt{1-C_T}}{2\sqrt{1-C_T}}} . \tag{2-13}$$

Based on an analysis by Niayifar and Porté-Agel, the following empirical relation for k^* can be used

$$k^* = 0.3837 I + 0.003678 , \tag{2-14}$$

in the range $(0.065 < I_0 < 0.15)$ where I_0 is the ambient turbulence intensity and I is the turbulence intensity at the rotor.

For more information about this wake model and its derivation, the reader is referred to the MSc thesis of Bo Hu[21] and the papers by Bastankhah, Porté-Agel and Niayifar [22, 23].

Changes to the model

The code of the wind farm model was developed by Bo Hu during his master thesis. Some small changes were made in order to adapt the code in this research.

- At the beginning of the wind farm model, a check was implemented to verify whether the requested wind speed was between the cut-in and cut-out wind speed of the wind turbine. If this was not the case, the code immediately returned zero power without the need for further evaluation of the code.
- One of the most used functions in the wind farm model was not optimised. By using the expression $(\sin^2(\theta) + \cos^2(\theta) = 1)$, the required computational time of a test case of the wind farm model could be reduced by 11 %.
- The definition of the wind direction angles as implemented by Bo Hu was not the conventional definition that is used in a wind rose (North at 0° and East at 90°). The code was adapted to work with this conventional definition of angles.

2-5 Wind farm layouts

Since this thesis is on the topic of wind farm optimisation, it is important to discuss which layouts will be used in the various phases. The initial construction of the PCE algorithm will be tested on the power curve of one wind turbine. During the expansion to 2D, the Horns Rev wind farm will be used. Since the wind farm model is computationally expensive when used with large wind farm layouts, the optimisation process will be performed on a three by three wind farm, thus consisting of nine wind turbines.

The wind farm that will be the subject of the final optimisation algorithm is the Horns Rev I wind farm in Denmark. It is chosen because research into this wind farm has already been done, and values for the optimised AEP can be found in literature. The Horns Rev wind farm is located on the west of Denmark, 14 km to 20 km offshore. The layout of the wind farm can be seen in Figure 2-7 where the markers indicate a turbine and the dashdotted line indicates the boundary of the wind farm. The shape of the wind farm is trapezoid and consists of eight rows of ten turbines, for a total of 80 turbines in the wind farm. The hub height of the turbines is 70 m with a rotor diameter of 80 m. The distance between the turbines is approximately seven rotor diameters (560 m) in both south-west and north-south directions [24].

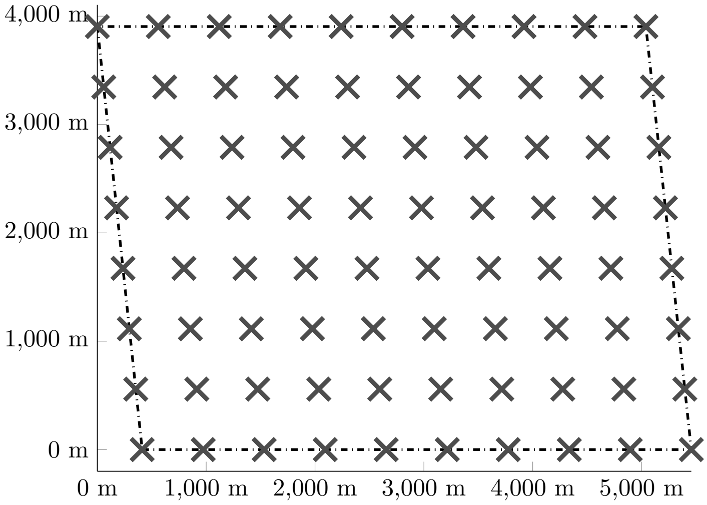


Figure 2-7: Layout of the Horns Rev wind farm.

2-6 Conclusion and outlook

In this chapter the wind farm model and the ways to compute the AEP were introduced. It can be concluded that, depending on the chosen discretisation, a lot of model evaluations are required when computing the AEP of a wind farm using a conventional approach. Instead of using a sectorwise or 'Weibull per wind direction sector' method, a new computational technique will be explored in this research project. Starting in the next chapter, polynomial chaos expansion will be introduced. It opens up the possibility of reducing the number of model evaluations that are needed to compute the AEP of a wind farm.

Chapter 3

Polynomial Chaos Expansion

In this chapter, polynomial chaos expansion (PCE) will be introduced. In the first section a general explanation will be presented and the mathematics behind the technique will be explained. In the second section, a one-dimensional surrogate model will be constructed. The various parameters that are important in PCE will be evaluated in the third section and a choice for a specific set of options will be made. The expansion to a two-dimensional PCE will be explained in section 4. In section 5 a better fitting surrogate model will be constructed using sectorwise PCE. In the final section of this chapter, conclusions on the analysis will be presented.

3-1 Introduction

3-1-1 Surrogate models

In engineering problems, it is often the case that a full and detailed model is too computationally expensive to evaluate. In order to be able to work with a certain model and to investigate a phenomenon on which the model is based, a surrogate model can be used. A surrogate model is an approximation of the computationally expensive model that can either be constructed with or without specific knowledge of the full computationally expensive model. In case the surrogate model is constructed with this knowledge, it is often the case that assumptions are made or that empirical factors are used in order to come up with a model that still represents the phenomenon, but less detailed than before. An example could be the decision to neglect an aerodynamic effect (that normally is very computationally expensive and has limited influence on the overall outcome of the model) in order to obtain a model that still gives good results but is now a lot faster. If no knowledge is used of the expensive model, the model gets treated as a black box. In this case, the surrogate model is constructed based on a statistical method (e.g. PCE, Guassian Processes ...) and on a limited set of evaluations of the expensive model. When a surrogate model is constructed, it can be used to investigate the phenomenon at a much lower computational cost.

Since surrogate models are an approximation of the computationally expensive model, it is important to check whether this approximation is accurate enough to use in for example an optimisation process. Next to the accuracy, it is important to take the speed of computation of the surrogate model into account. If the construction of the surrogate model takes too much time, the advantage to using it may be limited with respect to the original model.

3-1-2 History of PCE

PCE is a method that can be used to construct surrogate models. It was first presented by Wiener in 1938 in a paper on homogeneous chaos theory [25]. The method was based on Gaussian random variables. However, for non-Gaussian random variables, PCE was very slow, which led to a decrease of interest in the method [9]. PCE was then extended into generalised PCE (gPCE) by Xiu and Karniadakis [26]. This method includes more statistical distributions which can be used.

After many extensions, arbitrary PCE (aPCE) was developed in order to extend the scope. Using aPCE, any statistical distribution can form the basis of a PCE surrogate model. To obtain the orthogonal polynomials from these arbitrary distributions, a data-driven approach [27] or the Gram-Schmidt process [28] can for example be used. These newer techniques show better convergence rates than the old gPCE technique. However the remark is still made that the random variables have to be statistically independent. There may exist a linear correlation between the variables, which can be removed by applying linear transformations [27].

PCE has mainly been used in other fields than wind energy. The theory has been developed by mathematicians and now more and more researchers try to apply it to their field of interest. Most of the time it has been used as an uncertainty quantification method or to test the sensitivity of certain parameters on the final result.

3-1-3 Principles of PCE

PCE is a technique that uses orthogonal polynomials in order to construct a surrogate model. By doing so, a surrogate is created in the form of a polynomial function:

$$P(\boldsymbol{\xi}) \approx \hat{P}(\boldsymbol{\xi}) = \sum_{i=0}^{M} \alpha_i \phi_i(\boldsymbol{\xi}) \ , \tag{3-1}$$

where P is the original, computationally expensive model with $\boldsymbol{\xi}$ as its model inputs (wind speed and wind direction) and $\hat{P}$ is the surrogate model which consists of a basis (of order M) of polynomials ϕ_i with coefficients α_i.

The method can be explained as follows: the variables that are the input of the computationally expensive model (e.g. wind speed and wind direction) are expressed in moments. These moments are used to determine a set of orthonormal polynomials (with respect to these random variables). The polynomial basis is then fitted to the model and the coefficients of this fit can be used to compute statistics of the model. Each step will be explained in the overview below.

1. **Identification of the input parameters** Take a look at the model that needs to be approximated. What are the input parameters? What are the random variables and what are ‘constant’ inputs of the model? In case of a wind farm model, a time series of wind speeds and wind directions are the variables of interest and for example the rotor radius, the number of turbines and the layout are the constant input parameters. Instead of a time series the input variables can also be expressed as a statistical distribution (for example the Weibull distribution for the wind speed).
2. **Compute the moments of the input variables** For each time series or statistical distribution, the statistical (raw) moments can be determined. They are a representation of where the ‘weight’ lies in a distribution. A choice can be made to compute the moments straight from the data (if available) or from a fitted statistical distribution. Most datasets (time series, histogram etc) can be represented by a statistical distribution but this proves to not always be as accurate as required (as could already be seen in the previous chapter). An example is the distribution of the wind direction. It is not really clear which distribution is the best choice so for this variable it is better to compute the moments directly from a time series. Wind direction is circular ($0° = 360°$) so the influence of the phase shift needs to be investigated.
3. **Construction of the orthonormal polynomials** With the moments from the random input variables, the orthonormal polynomials can be constructed. Different methods exist, so also in this step a choice needs to be made. The first method that will be used is the Gram-Schmidt process. It is an iterative process which starts with the polynomial of the lowest order. The other method that is considered in this report is the Oladyshkin data-driven method in which a system of linear equations will be solved in order to produce the orthonormal polynomials. Although PCE only requires orthogonal polynomials, using orthonormal ones makes processing the end result (the coefficients) much easier since no scaling factors are required.
4. **Fitting the polynomials to the model** In this phase the basis of orthogonal polynomials is fitted onto the results from the computationally expensive model. Again various options exist. First of all, a choice needs to be made between an intrusive or non-intrusive method. The intrusive method uses the equations of the expensive model to obtain analytical expressions for some of the parameters. Since in this research the wind farm model is treated as a black box, a non-intrusive approach is used. Next, various options exist in order to decide on collocation points (where the expensive model will be evaluated) and on methods to fit the basis on the model. More information will be given in the next section. The result of this step is N coefficients for each of the orthonormal polynomials in the basis of rank N. With these coefficients the surrogate model can be constructed: all polynomials in the basis are multiplied with their respective coefficient and summed. The result is one polynomial which is a surrogate of the computationally expensive model.
5. **Using the coefficients** The coefficients that were computed in the previous step can now be used to compute the expected value and variance of the output of the model, based on the distribution of the random variables.

The result of this process is a polynomial (the surrogate model) in function of the input variables. A polynomial is normally much quicker to evaluate than the original computationally expensive model.

3-2 Construction of a 1D surrogate model

First, we will look into the construction of a one-dimensional surrogate model. In this section, all the techniques to construct such a model will be explained.

3-2-1 Calculation of statistical moments

As explained in the previous section, two methods exist to compute the statistical moments of a random variable. The first method uses the statistical distribution of the random variable. In the case of the wind climate, the wind speed is often represented by a Weibull distribution with a shape and a scale parameter. When a wind resource assessment is performed, these are the variables that are communicated with the (future) wind farm owner. Sometimes, no time series of the actual measurements is available. The moments of most distributions can be found in the literature (e.g. [8, 29, 30]). Specifically for the Weibull distributions, the nth moment is given by:

$$m_n = E\left[X^n\right] = \lambda^n \Gamma\left(1 + \frac{n}{k}\right) , \tag{3-2}$$

where m_n is the nth moment, X is the random variable and λ and k are the scale parameter and shape parameter of the Weibull distribution. Γ represents the gamma function [13].

The second method to construct the moments consists of a data-driven approach. If the random variable cannot accurately be represented using a known probability density function (PDF) type, it is often better to compute the moments directly from the data. This is the case for the wind direction. Even when a known analytical expression exists, it is no guarantee that one can find an equation for the corresponding moments. In these cases, the moments can be computed directly from the data using [31]

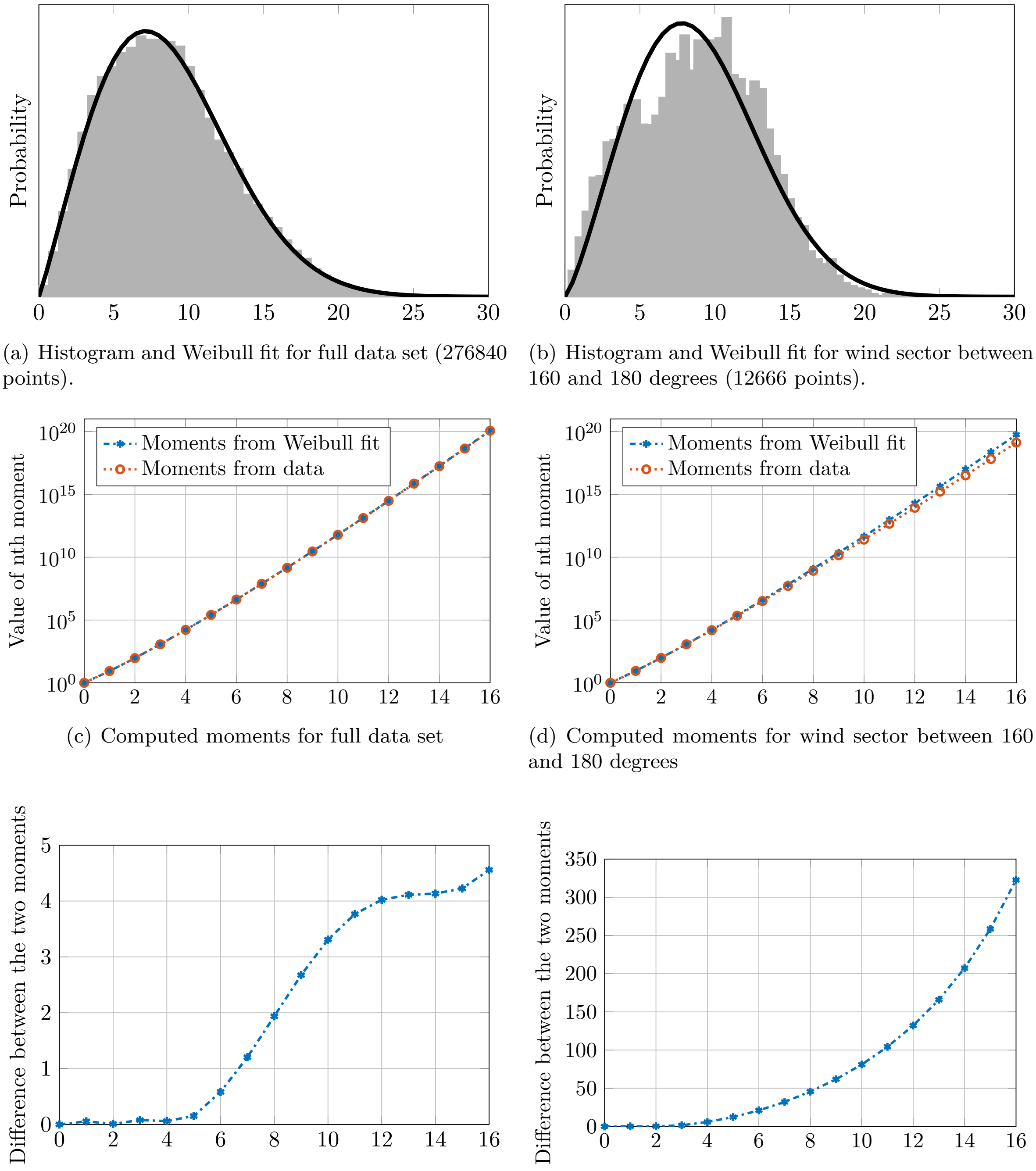
$$m_n = \mathbb{E}\left[X^n\right] = \int_{-\infty}^{\infty} x^n \, \mathrm{d}F\left(x\right) = \int_{-\infty}^{\infty} x^n f\left(x\right) \mathrm{d}x \, . \tag{3-3}$$

In case of a discrete dataset (where this method will be used for), the moments can be calculated with

$$m_n = \mathbb{E}\left[X^n\right] = \frac{1}{|X|} \sum_{i=1}^{|X|} x_i^n \, , \tag{3-4}$$

where $|X|$ represent the length of the dataset and x is one entry from the dataset. In the above equations, m_n is the nth moment, F is the cumulative probability distribution function (CDF) and f is the PDF.

As can be seen in Figure 3-1, the computed moments for the Weibull fit and directly from the data are similar when the Weibull distribution fits well with the data set (see on the left, Figures 3-1(a), 3-1(c) and 3-1(e)). The figures on the right were constructed using a limited set of wind speeds and as seen in Figure 3-1(d), the Weibull distribution is not such a good fit. The differences between the two sets of moments can be observed in Figure 3-1(f). The consequence of this difference will be discussed in Section 3-3. This result was also observed by Wang et al [32] when they investigated both methods. The influence of this difference on the annual energy production (AEP) calculation will be investigated in Section 3-3.



(a) Histogram and Weibull fit for full data set (276840 points).

(b) Histogram and Weibull fit for wind sector between 160 and 180 degrees (12666 points).

(c) Computed moments for full data set

(d) Computed moments for wind sector between 160 and 180 degrees

(e) Difference (%) in moments for full data set

(f) Difference (%) in moments for wind sector between 160 and 180 degrees

Figure 3-1: Influence of quality of Weibull fit

3-2-2 Construction of the orthonormal polynomials

When the moments of the random variables are determined, the orthonormal polynomials can be constructed. Two methods were found in the literature: The Gram-Schmidt process and the Data-Driven aPCE method of Oladyshkin, which uses a linear system of equations [27, 33]. In essence the Gram-Schmidt process can also be part of a Data-Driven aPCE method, thus in order to prevent confusion in the rest of the document, the methods will be labelled ‘Gram-Schmidt Process’ (or ‘GS’) and ‘data-driven Oladyshkin’ (or ‘DD-Oladyshkin’) respectively.

The Gram-Schmidt process

The Gram-Schmidt process produces the orthonormal polynomials with an iterative process. Each newly constructed polynomial (each time of higher rank), is made orthogonal with respect to all previous polynomials. This is done by subtracting the projection of the new initialisation vector on the existing polynomials for each new polynomial (see Equation (3-8)).

A sequence of polynomials $\{\phi_n(x)\}_{n=0}^{\infty}$ of order n is called orthogonal on an interval (a, b) with respect to a weight function $w(x)$ if

$$\int_a^b w(x)\,\phi_m(x)\,\phi_n(x)\,\mathrm{d}x = h_n\delta_{mn} \quad \text{with} \quad \delta_{mn} := \begin{cases} 0 & m \neq n \\ 1 & m = n \end{cases}. \tag{3-5}$$

When h_n is 1, the polynomials are orthonormal.

Define the inner product of the polynomials c and d as

$$\langle c, d\rangle := \int_a^b w(x)\,c(x)\,d(x)\,\mathrm{d}x\,. \tag{3-6}$$

Since we want to make polynomials based on the moments of the random variables, $w(x)$ will be equal to the PDF (f) of random variable x.

$$\langle c, d\rangle := \int_a^b f(x)\,c(x)\,d(x)\,\mathrm{d}x\,. \tag{3-7}$$

Equation (3-3) stated that

$$m_n = \int x^n f(x)\,dx\,.$$

Combining Equations (3-3) and (3-7) gives the ability to compute the inner product of the polynomials by multiplying them and using m_n for each power x^n to compute the inner product of polynomials c and d.

EXAMPLE 3.1

Use the functions $c(x) = c_2x^2 + c_1x + c_0$ and $d(x) = d_1x + d_0$ and a set of moments m_n. The inner product $\langle c, d\rangle$ is then equal to:

$$\langle c, d\rangle = \int f(x)\,c(x)\,d(x)\,\mathrm{d}x$$

Substitute the functions $c(x)$ and $d(x)$.

$$= \int f(x)\left(c_2x^2 + c_1x + c_0\right)(d_1x + d_0)\,\mathrm{d}x$$

$$= \int f(x)\left(c_2d_1x^3 + c_2d_0x^2 + c_1d_1x^2 + c_1d_0x + c_0d_1x + c_0d_0\right)\mathrm{d}x$$

Group the constants per power of x and move them outside the intergration.

$$= c_2 d_1 \int f(x)\, x^3\, \mathrm{d}x + (c_2 d_0 + c_1 d_1) \int f(x)\, x^2\, \mathrm{d}x +$$
$$(c_1 d_0 + c_0 d_1) \int f(x)\, x\, \mathrm{d}x + c_0 d_0 \int f(x)\, x^0\, \mathrm{d}x$$

By applying Equation (3-3) to each integral we end up with the following result for the inner product:

$$= c_2 d_1 m_3 + (c_2 d_0 + c_1 d_1)\, m_2 + (c_1 d_0 + c_0 d_1)\, m_1 + c_0 d_0 m_0$$

This process is repeated for every moment m_n.

Now, the Gram-Schmidt process can be executed as follows: start with a sequence $v = \{1, x, x^2, ...\}$. $p_0(x) = 1$ per definition. The other orthogonal polynomials (p_n) are then computed as

$$p_n(x) = v_n - \sum_{i=0}^{n-1} \frac{\langle v_n, p_i(x) \rangle}{\langle p_i(x), p_i(x) \rangle} p_i(x) \ , \tag{3-8}$$

The procedure will be shown for the first three polynomials in the following example.

EXAMPLE 3.2

As per the definition we start with:

$$p_0(x) = 1 \tag{3-9}$$

The polynomial of order 1 can then be computed by applying Equation (3-8).

$$p_1(x) = x - \frac{\langle x, p_0 \rangle}{\langle p_0, p_0 \rangle} p_0 = x - \frac{\langle x, 1 \rangle}{\langle 1, 1 \rangle} 1 = x - \frac{m_1}{m_0}$$

The second order polynomial is computed in the same way, but here an extra term is present due to the summation over all polynomials of a lower order.

$$\begin{aligned}
p_2(x) &= x^2 - \frac{\langle x^2, p_0 \rangle}{\langle p_0, p_0 \rangle} p_0 - \frac{\langle x^2, p_1 \rangle}{\langle p_1, p_1 \rangle} p_1 \\
&= x^2 - \frac{\langle x^2, 1 \rangle}{\langle 1, 1 \rangle} 1 - \frac{\langle x^2, x - \frac{m_1}{m_0} \rangle}{\langle x - \frac{m_1}{m_0}, x - \frac{m_1}{m_0} \rangle} \left(x - \frac{m_1}{m_0} \right) \\
&= x^2 - \frac{m_2}{m_0} - \frac{m_3 - \frac{m_1 m_2}{m_0}}{m_2 - 2\frac{m_1^2}{m_0} + \frac{m_1^2}{m_0}} \left(x - \frac{m_1}{m_0} \right) \\
&= x^2 - \frac{m_2}{m_0} - \frac{m_0 m_3 - m_1 m_2}{m_0 m_2 - m_1^2} \left(x - \frac{m_1}{m_0} \right) \\
&= x^2 - \frac{m_0 m_3 - m_1 m_2}{m_0 m_2 - m_1^2} x + \frac{m_1 m_3 - m_2^2}{m_0 m_2 - m_1^2}
\end{aligned}$$

Finally the polynomials can be made orthonormal by dividing them by their norm.

$$\phi_i = \frac{p_i}{||p_i||} = \frac{p_i}{\langle p_i, p_i \rangle} \ . \tag{3-10}$$

Since the inner product of each polynomial needs to be computed, we can see from Example 3.1 that there is a need for $2n$ moments in order to compute n polynomials.

Data-driven Oladyshkin method

The data-driven Oladyshkin method uses the orthogonality equations to derive a system of linear equations that can be solved in order to produce the coefficients of the polynomial basis. The following derivation is explained further by Oladyshkin [27]. The derivation starts with Equation (3-11).

$$\int_a^b w(x)\,\phi_m(x)\,\phi_n(x)\,\mathrm{d}x = 0 \qquad \forall m \neq n \tag{3-11}$$

Instead of the normality condition, an extra condition is applied by requiring that the first coefficient q_n^n of each polynomial is equal to 1. The n subscript denotes the nth polynomial, while the superscript represents the nth coefficient (corresponding to x^n) in that polynomial. The above equations implies that $q_0^0 = 1$. The orthogonality expressions for ϕ_1 are now:

$$\begin{aligned}
&\int_a^b w(x)\, q_0^0 \left[\sum_{i=0}^{1} q_1^i x^i\right] \mathrm{d}x = 0\\
&q_1^1 = 1
\end{aligned}$$

This set of equations can be constructed for all polynomials to obtain a polynomial basis. The system of equations to obtain a polynomial basis of order n can be written as

$$\begin{aligned}
&\int_a^b w(x)\, q_0^0 \left[\sum_{i=0}^{n} q_n^i x^i\right] \mathrm{d}x = 0\\
&\int_a^b w(x) \left[\sum_{j=0}^{1} q_1^j x^j\right] \left[\sum_{i=0}^{n} q_n^i x^i\right] \mathrm{d}x = 0\\
&\vdots\\
&\int_a^b w(x) \left[\sum_{j=0}^{n-1} q_{n-1}^j x^j\right] \left[\sum_{i=0}^{n} q_n^i x^i\right] \mathrm{d}x = 0\\
&q_n^n = 1
\end{aligned}$$

This system of equations is closed and defines all unknown coefficients of ϕ_n. It can be seen that ϕ_n uses information of all polynomials of a lower order. When substituting the first equation in the second, the first and second equation in the third and so on, the following relations are obtained

$$\begin{aligned}
&\int_a^b w(x) \sum_{i=0}^{n} q_n^i x^i \,\mathrm{d}x = 0\\
&\int_a^b w(x) \sum_{i=0}^{n} q_n^i x^{i+1} \,\mathrm{d}x = 0\\
&\vdots\\
&\int_a^b w(x) \sum_{i=0}^{n} q_n^i x^{i+n-1} \,\mathrm{d}x = 0\\
&q_n^n = 1
\end{aligned}$$

In Equation (3-3) it was stated that the nth raw moment can be computed using

$$m_n = \int x^n f(x)\, dx\,.$$

Using this expressions, the above system of equations can be rewritten to

$$\begin{aligned}
&\sum_{i=0}^{n} q_n^i m_i = 0 \\
&\sum_{i=0}^{n} q_n^i m_{i+1} = 0 \\
&\vdots \\
&\sum_{i=0}^{n} q_n^i m_{i+k-1} = 0 \\
&q_n^n = 1
\end{aligned}$$

Alternatively this can be written in a matrix form

$$\begin{bmatrix} m_0 & m_1 & \dots & m_n \\ m_1 & m_2 & \dots & m_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n-1} & m_n & \dots & m_{2n-1} \\ 0 & 0 & \dots & 1 \end{bmatrix} \begin{bmatrix} q_n^0 \\ q_n^1 \\ \vdots \\ q_n^{n-1} \\ q_n^n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}\,. \tag{3-12}$$

Using this matrix equation, all required coefficients of the polynomial basis can be computed [27].

Comparison

Both methods can be compared by looking at the inner product of each polynomial with itself with respect to the moments. In theory, each inner product should be equal to 1, but as can be seen in Figure 3-1 the values of the moments can get really high. Due to this, machine errors (rounding errors) are introduced in the computations which may have an influence on the quality of the constructed polynomial basis. In Figure 3-2, the result of the inner product of each polynomial with itself is shown, subtracted by 1. This was done in order to be able to present the data on a logarithmic axis. The polynomials were constructed from data-driven moments from the full wind speed series.

For inner products where the result was exactly 1 (polynomials 0 and 1 in the Figure 3-2), the value used in the figure is `eps`, the machine precision in MATLAB. From the results, it can be seen that the Gram-Schmidt process gives slightly better accuracy, but that problems start to rise when the polynomial basis becomes larger. When the rank is increased to 12, the difference between the highest and lowest moments gets so big that computational errors can not be prevented and the polynomial basis can not be computed. In Section 3-3 the choice for moments from data or via a Weibull fit will be combined with the choice for the construction method of the polynomial basis.

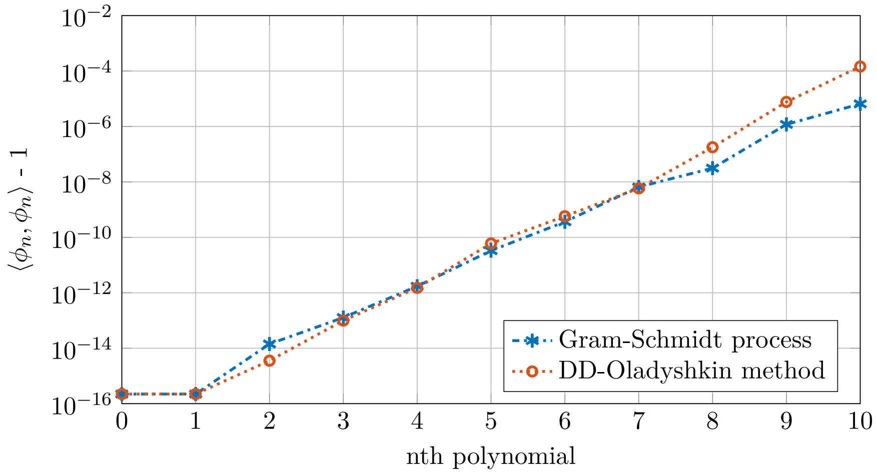


Figure 3-2: Inner product of each polynomial from the orthonormal basis with itself.

3-2-3 Fitting the polynomials to the data and using the PCE coefficients

Now that a polynomial basis is constructed, it needs to be fitted to the model. In order to do so, a set of so called collocation points needs to be constructed at which the (expensive) model will be evaluated. With these evaluations, the polynomials will be fitted to the model. The most straightforward method to do so is least squares regression. It consists of the following steps:

- Construct a polynomial basis.
- Choose the points which will be used to fit the polynomial basis to the model and evaluate the model at these points.
- Make a matrix so that each polynomial (on the rows) is evaluated at all the selected points (and put the result in the columns).
- Solve the system of equations using a least-squares approach to compute the coefficient of each polynomial.
- Use the coefficients to compute the expected value and standard deviation of the power production of the wind farm or wind turbine.
- Use these values to compute the AEP.

Using the coefficients

Before we can discuss the importance of how and which points to select, the method on which this choice will be evaluated needs to be introduced. Once the polynomials are fitted to the model, a coefficient for each of the polynomials in the orthonormal basis is computed.

With these coefficients, the mean (μ_P) and variance (σ_P^2) of the output distribution can be determined as

$$\mu_{\hat{P}} = \alpha_0 \, ||\phi_0|| \qquad \text{and} \qquad \sigma_{\hat{P}}^2 = \sum_{i=1}^{M} \alpha_i^2 \, ||\phi_i||^2 \ . \tag{3-13}$$

When the polynomials are orthonormal, these properties can conveniently be written as

$$\mu_{\hat{P}} = \alpha_0 \qquad \text{and} \qquad \sigma_{\hat{P}}^2 = \sum_{i=1}^{P} \alpha_i^2 \, . \tag{3-14}$$

Similar equations can be derived for the higher order moments (e.g. skewness and kurtosis) and can be found in literature (e.g. [34]).

Selection of collocation points

The selection of collocation points is important as it can influence the result in an important way. In Figure 3-3 a polynomial basis of varying order, constructed with the Gram-Schmidt process from data-driven moments was used to compute the AEP. Least squares regression was used to fit the collocation points to the PV curve of a 2 MW wind turbine. The following selections of collocation points were used:

- **Fit 1:** The roots of all polynomials that are present in the orthonormal basis.
- **Fit 2:** The roots of the polynomial of the highest order in the polynomial basis.
- **Fit 3:** A linear spacing starting at zero and ending at 25 with steps of 2.5 m/s.
- **Fit 4:** A linear spacing starting at zero and ending at 25 with steps of 1 m/s.

The first two fits are based on literature where it is suggested that the roots of the polynomial basis give the best results compared to other methods [35, 36]. Note that the first two selections of collocations points have a growing number of evaluation points when the order of the polynomial basis grows. The last two selections have a constant number of points.

The results in the following graphs are obtained by computing the $\mathbb{E}\left(P\right)$, multiplying by $\mathrm{h_y}$, the number of hours in a year, and dividing by the theoretical result of the time series. This time series was introduced in Section 2-3. For each wind speed value, the corresponding power was computed. The expected power was then determined by calculating the mean of this computed power data.

From the figure it can be seen that the more complex methods (fit 1 and 2) to select the collocation points do not give an advantage over the simpler methods (fit 3 and 4). These simple methods for the choice of collocation points also give the possibility to use the same points across different polynomial bases. In the analysis a set of 10 and 25 linearly spaced points were used.

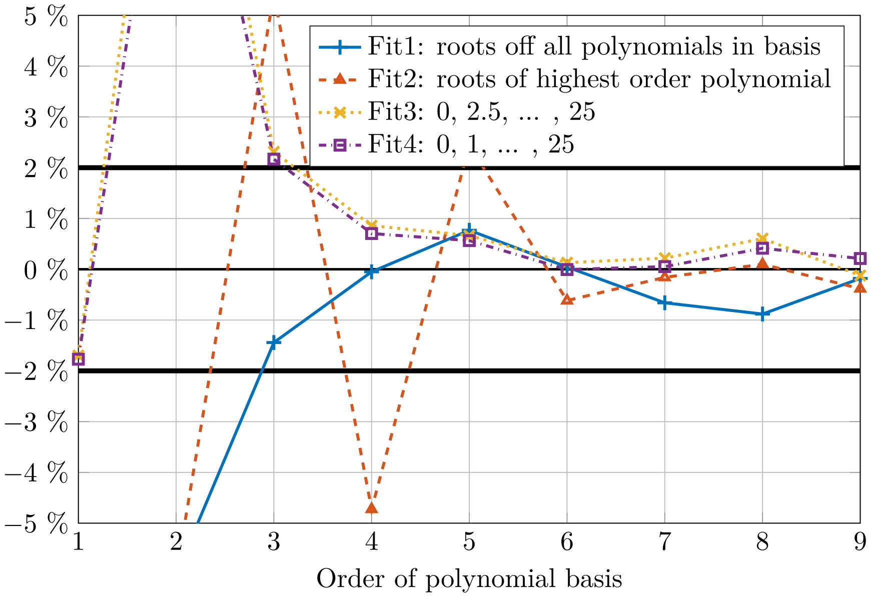


Figure 3-3: Difference (%) of the AEP between the estimate by PCE and the exact solution (7.786×10^6 MW h) for various sets of collocation points.

Figure 3-4 shows a fit for a polynomial basis of order 1, 4, 6 and 9 for a set of linearly spaced collocation points. It can be seen that the solution converges for an increasing number of collocation points. This means that from a certain amount of points, it is of no use to select more of them to perform the fit.

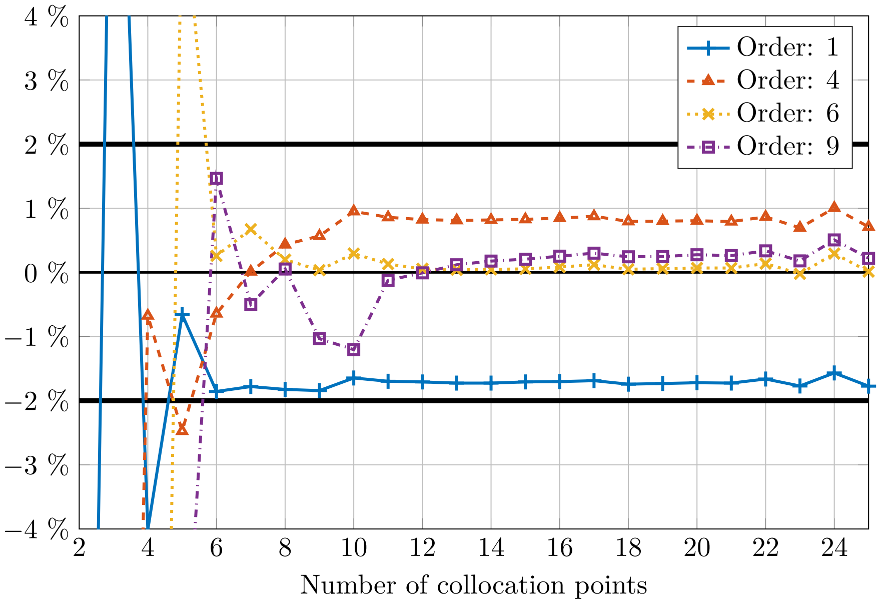


Figure 3-4: Difference (%) of the AEP between the estimate by PCE and the exact solution (7.786×10^6 MW h) for a varying number of collocation points.

As expected, the higher the order of the polynomial basis, the better the accuracy of the approximation. However, this is not the case for the last order (9). This could already be seen in Figure 3-3 where the result of the 9th order approximation for Fit 4 is higher than the 6th order of Fit 3 and 4. It is not really clear why this is the case. It is possibly due to the numerical inaccuracies with respect to the determination of the moments from the input variables and thus the quality of the polynomial basis. Each of the different polynomial bases have a disruption around the number of points which corresponds to the order of the polynomial basis (this is especially the case for the basis of order 9). In the current example (approximation of the 2 MW power curve) it seems that the sixth order polynomial basis performs the best and gives good results when having around 8 linearly spaced collocation points. This result needs to be validated when using a full wind farm model in the next sections.

Computation of the standard deviation

The same plots can also be made for the standard deviation. It has to be noted that this is not the standard deviation on the AEP (which is energy), but the standard deviation on the expected power. The computed standard deviation can be used to have an idea of how the power varies in function of the input variables (wind speed and wind direction).

In Figure 3-5, the difference between the standard deviation computed by PCE and the time series is shown for the same sets of collocation points as discussed on page 25. It can be seen that none of the sets of collocation points gives a really accurate result for the standard deviation. The results of the first set (Fit 1) are acceptable, but require a lot of collocation points. Fit 2 is not nearly as accurate. Again, the more simple methods seem to be the better choice (although it has to be noted that the results are far from perfect). This choice gives the same advantage as discussed in the previous section: it gives the possibility to compare over different polynomial bases.

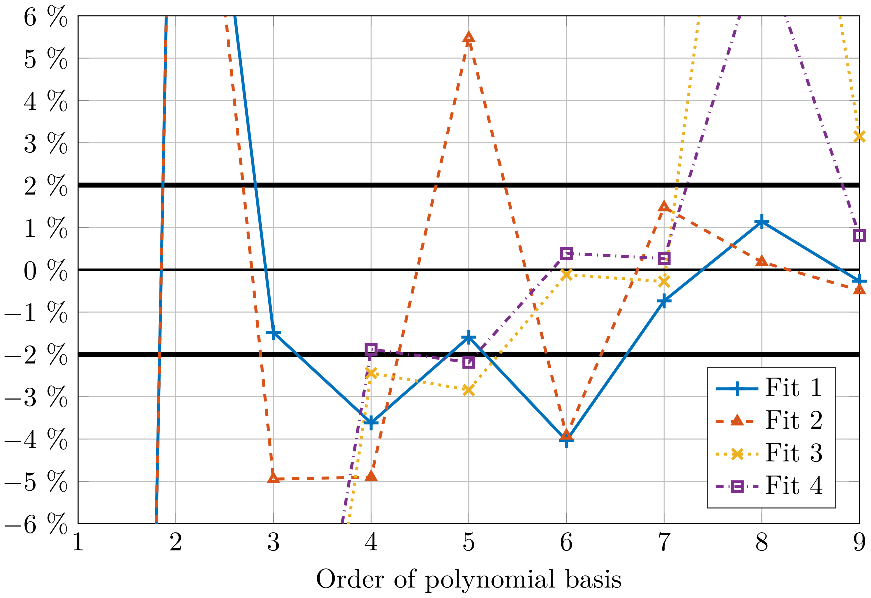


Figure 3-5: Difference (%) of the standard deviation of the expected power between PCE and the exact solution for various sets of collocation points.

Figure 3-6 shows results for the standard deviation for the 1st, 4th, 6th and 9th order for different number of linearly spaced collocation points. The first order results are not visible since they converge to -40%. The 6th and 9th order polynomials give the best results but it should be noted that a very large error is seen for the 9th order polynomial basis when using 10 collocation points. This behaviour can also be observed for the other bases on the points $M + 1$ where M is their order.

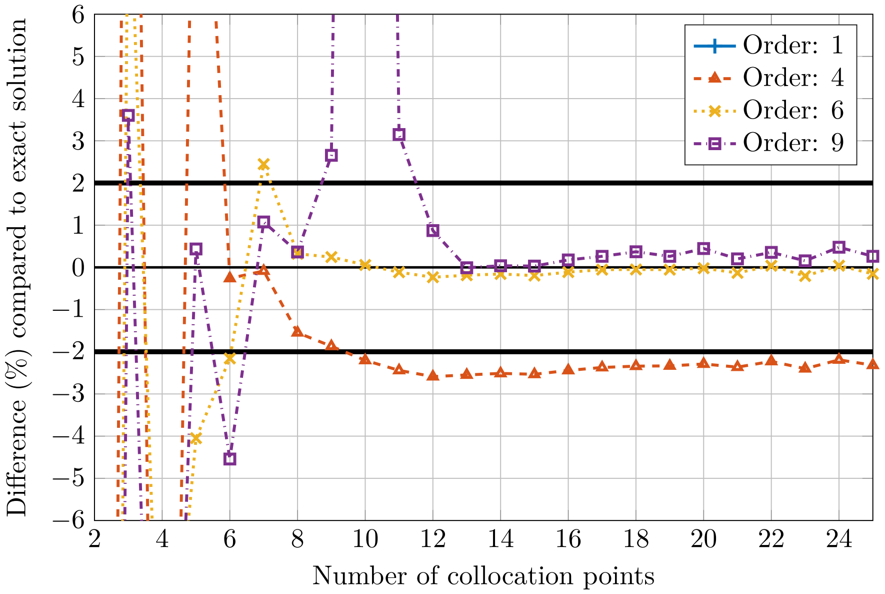


Figure 3-6: Difference (%) of the standard deviation of the expected power between PCE and the exact solution for a varying number of collocation points.

Visual representation of the surrogate model

The end result after all of the above steps is a polynomial surrogate model. The results of the first two moments of the surrogate model (the mean and standard deviation) have been discussed above but the real surrogate model has not been shown yet. Figure 3-7 shows the result of making a surrogate model for the PV curve of a wind turbine for various orders of the polynomial basis. 25 linearly spaced collocation points were used with a data-driven Gram-Schmidt process. It can be seen that the zeroth order expansion is not a good representation of the real model (the PV curve). The higher the order gets, the better the approximation of the surrogate model to the PV curve. Since the PV curve can not be represented by polynomials, there will never be a perfect surrogate model. The fits will get better and better, so the error on the expected value for the power (and thus AEP) will approach zero (in the ideal case when no numerical issues are present).

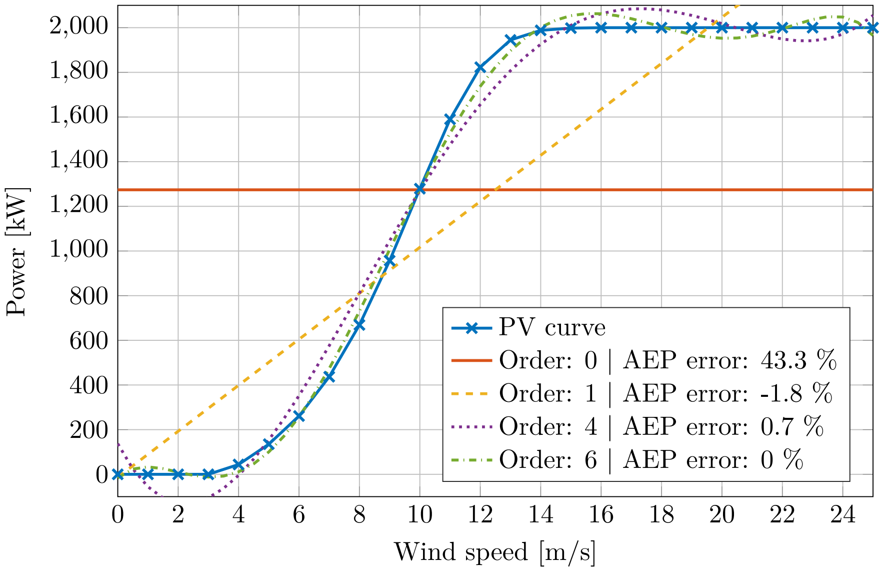


Figure 3-7: The PV curve of a 2 MW wind turbine with surrogate models of varying order.

Verification of the method on a simple function

In order to verify that the PCE method (and the code) functions well, a polynomial function will be used as the model that needs to be approximated. The evaluated polynomial is $y = -0.5x^3 + 12x^2 + 60x + 0$. These coefficients were chosen so that the resulting polynomial looks like it could be a PV curve of a wind turbine. It has the same range in y-values as a real 2 MW PV curve.

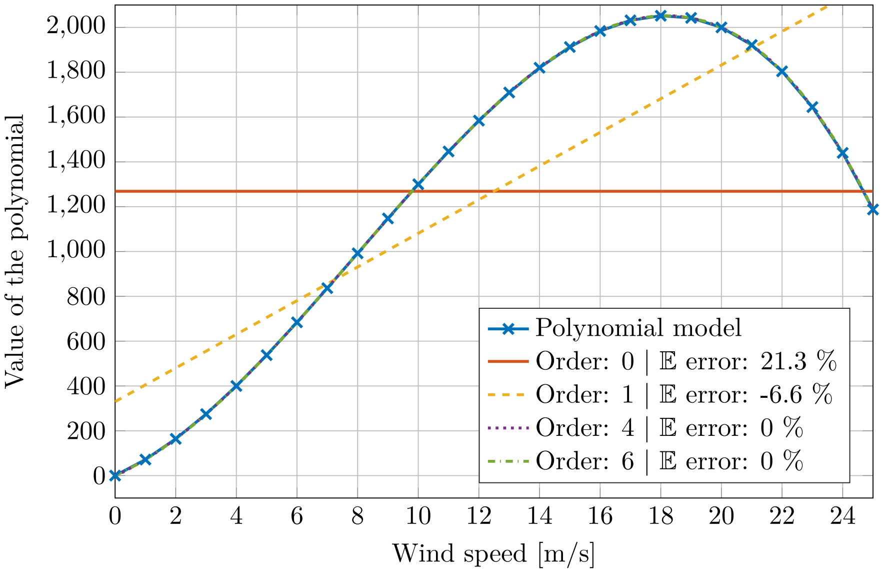


Figure 3-8: Approximation of a third order polynomial by PCE surrogate models of varying order.

It can be seen in Figure 3-8 that the PCE surrogate model is a perfect match with the polynomial model, as long as its order is at least three. This is an expected result since the function that was approximated is a polynomial of the third order. The purple and green lines for the fourth and sixth order PCE surrogate are perfectly on top of the polynomial model (blue).

3-3 Influences of construction methods on the AEP

Now that the construction of the surrogate model is explained, we can again have a look at the first choice we had to make: the calculation of the moments and the construction method for the polynomial basis. The choices will be re-evaluated with the full wind speed data series and the same limited set as in Figure 3-1.

3-3-1 Influence of moments on polynomial construction

Figure 3-9 shows the inner product of the polynomials with themselves (with respect to their corresponding weighting function) for the full wind speed series. Four different construction methods are being used: a combination of moments from the Weibull distribution or data-driven and the Gram-Schmidt process or data-driven Oladyshkin method. No big differences between the various construction methods can be observed. It just seems that the polynomial basis from the data-driven moments is slightly less precise than the basis constructed from the Weibull moments.

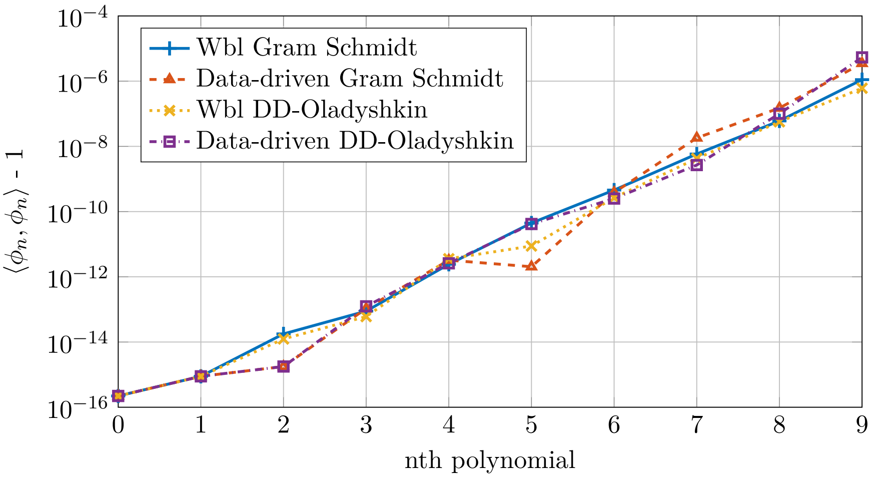


Figure 3-9: Inner product for polynomials constructed in four different ways from the full wind speed series.

Figure 3-10 shows the same graph for the limited data set of wind speeds where the wind direction is between 160° and 180° (as defined in Figure 3-1). The same behaviour can be observed here. Near the end, a clear difference is visible between the polynomials that are constructed from the Weibull fit moments or from the data-driven moments. It has to be noted that the inner product is still very close to 1, so no real problems are to be expected from the slightly less precise polynomial bases.

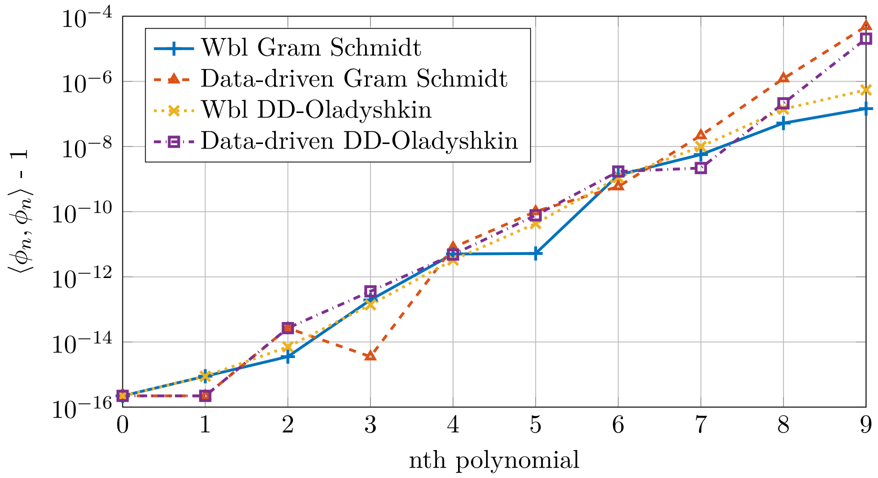


Figure 3-10: Inner product for polynomials constructed in four different ways from the limited wind speed series.

3-3-2 Influence of construction method on AEP calculation

It is important to note that the accuracy of the polynomial basis is not the most important thing to end up with a good approximation for the expected value. The most important thing is that the polynomials are a good representation of the random variable. In Figure 3-11 the four different methods to construct the polynomial basis are used to approximate the AEP of the full wind speed series. Based on the results from Figure 3-4, 25 collocation points were used.

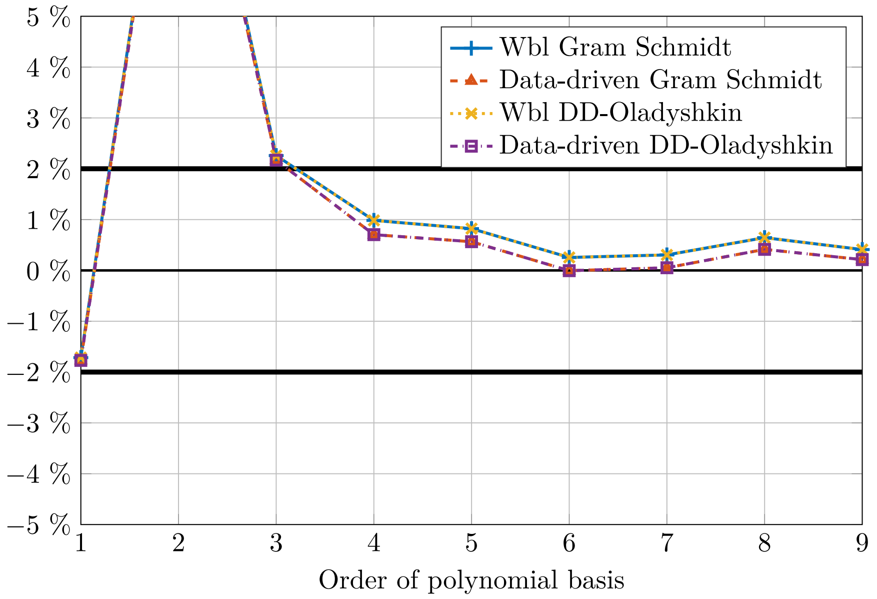


Figure 3-11: Difference (%) of the AEP between the estimate by PCE and the exact solution (7.786×10^6 MW h) for varying construction methods for the full data set.

In the figure, it can be seen that there is no difference in results between the Gram-Schmidt and DD-Oladyshkin construction method. The data-driven polynomial bases give a slightly more accurate result than the bases which are constructed from the Weibull fit.

Figure 3-12 shows the same approximation but based on the limited data set of wind speeds. Again, the Gram-Schmidt and data-driven Oladyshkin method give the same results, but now a bigger difference between the data-driven and Weibull fit polynomial bases can be observed. It can thus be concluded that even though the quality of the polynomial basis can be slightly lower when using data-driven moments (see for example Figure 3-10), they definitely result in a better approximation of the AEP.

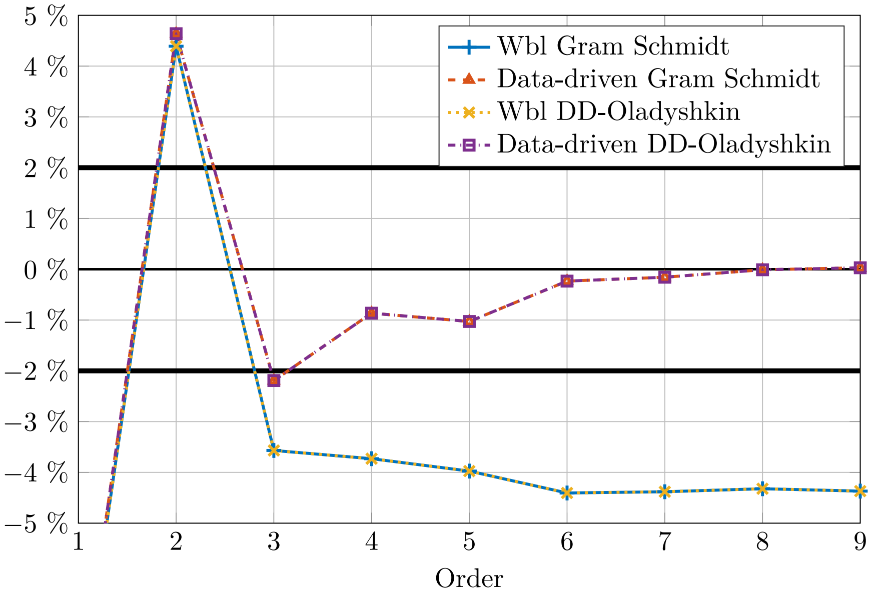
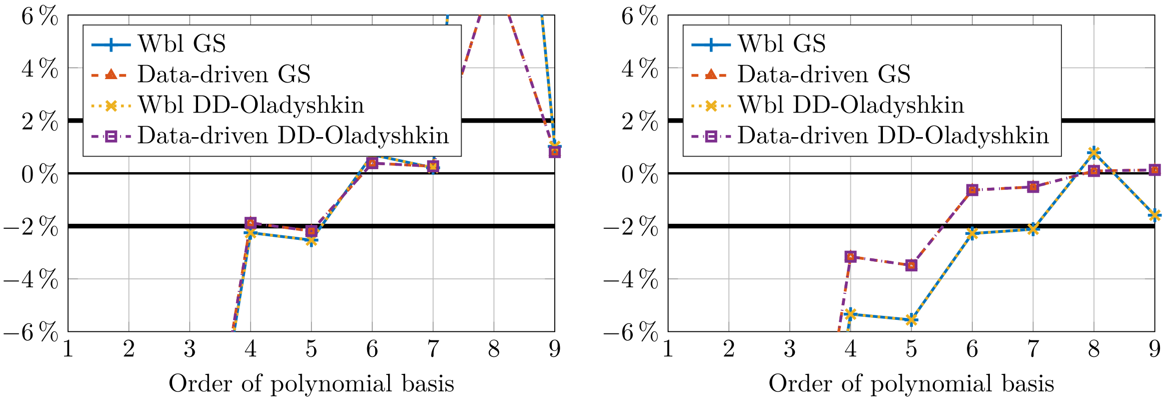


Figure 3-12: Difference (%) of the AEP between the estimate by PCE and the exact solution (8.673×10^6 MW h) for varying construction methods for the limited data set.

3-3-3 Influence of construction method on standard deviation

The same fits as in the previous section were also used to compute the standard deviation of the expected power. As can be seen in Figure 3-13, the polynomial basis construction method does again have no influence on the result, however the way the moments are calculated does. This is most visible in Figure 3-13(b). Some strange behaviour can be seen in 3-13(a) for the polynomial basis of the eight order. It is unclear what its cause is.



(a) Full wind speed series. $\sigma = 742.4\,\mathrm{W}$.

(b) Limited wind speed series. $\sigma = 749.3\,\mathrm{W}$.

Figure 3-13: Difference (%) of the standard deviation of the expected power between PCE and the exact solution for varying construction methods.

3-3-4 Conclusion

In this intermediate conclusion we can decide which construction methods to use when constructing a one-dimensional PCE surrogate model. It can be seen that a data-driven approach gives the best results for an accurate estimation of the AEP and standard deviation. The polynomial basis construction method does not have a big influence, but the choice is made to continue with the Gram-Schmidt process since it offers the most insights in the construction process. Due to this insight, it also offers an easier expansion in two dimensions than the DD-Oladyshkin method. Furthermore, the method's iterative nature ensures that numerical inaccuracies in the calculation of the higher order polynomials (due to large difference in value between zero and higher order moments) have no influence on the polynomials of a lower order.

3-4 Construction of a 2D surrogate model

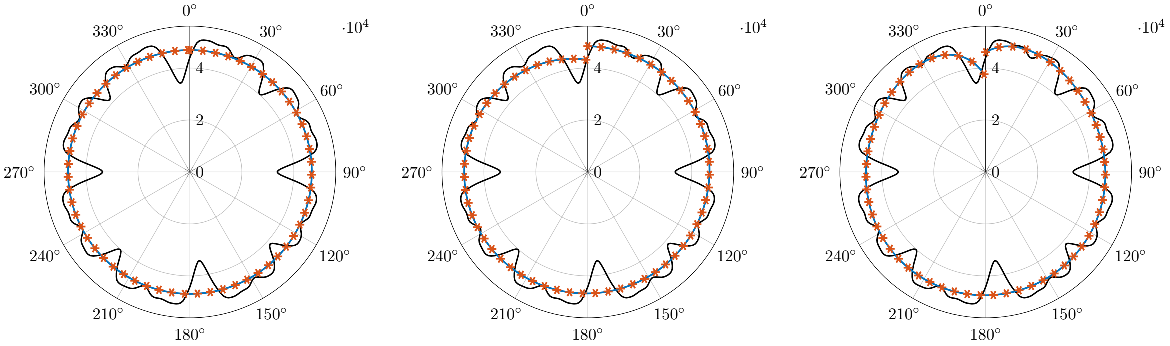
In the previous section, a one-dimensional polynomial chaos expansion surrogate model was developed and the various choice in the construction process were evaluated. When the wind turbines are placed in a wind farm, also the wind direction is important and for that reason, the construction of a two-dimensional PCE surrogate model will be introduced in this section. Instead of using one wind turbine and varying the wind speed, this section will use the Horns Rev wind farm layout with a fixed wind speed and vary the wind direction.

3-4-1 PCE and wind direction

The first step in the process of making the two-dimensional PCE is looking at the wind direction in one dimension. This can be done by using a fixed wind speed and by varying the wind direction around the wind farm. In the following analysis a fixed wind speed of 8 m/s will be used. It is between the cut-in wind speed and rated wind speed of the wind turbines so that wake effects are clearly present in the wind farm. In this section, the Horns Rev wind farm layout is used in the analysis.

It can be seen in Figure 2-5(b) on page 9 that the wind direction has an irregular histogram. As a result, it is really difficult to fit a particular parametric distribution on this data. Since the data-driven moments have proven their worth with the one-dimensional wind speed analysis, they will also be used in the case for the wind direction. Wind direction data is circular in nature, so a check will need to be performed to investigate the influence of the phase of the dataset. When using a dataset with a phase shift, one also has to account for the new direction and has to rotate the wind farm over the same angle to compensate for the rotation of the reference axis.

Figure 3-14 shows the surrogate models for for the Horns Rev wind farm with a constant wind speed of 8 m/s and 60 collocation points. Each of the three subfigures shows a surrogate model of a different order, respectively a zeroth, fourth and eight order surrogate model. As can be seen in the figures, not one of the surrogate models is an accurate approximation of the output of the expensive wind farm model. However, the approximation of the AEP is still good. The respective deviations from the exact solution are −0.02 %, 0.2 % and 0.09 % while the AEP is 411 × 10^3 GW h. When used like this, the surrogate model for the wind direction is practically useless in all cases except for wind farm layout optimisation where only the AEP is needed. Since the surrogate model is (visually) not a good approximation of the expensive model, all other (higher) coefficients will be bad and as a result, the standard deviation will be wrong. This will be shown in Figure 3-16.



(a) Surrogate of zeroth order

(b) Surrogate of fourth order

(c) Surrogate of eight order

Figure 3-14: Surrogate models for the Horns Rev wind farm. The black line is the power output of the wind farm, the blue line is the surrogate model and the yellow markers indicate the points that were used to construct this surrogate.

Using the conclusion from the wind speed surrogate model analysis, it was decided to use again a linearly spaced set of points to perform the least square fit. Figure 3-15 shows that the approximation of the AEP is really unstable when not enough points are selected. Starting from ±20 collocation points, acceptable results are obtained, while the response is stable with 30 points or more. The order of the surrogate model is not that important for the accuracy of the AEP estimation. This is due to the fact that the surrogate model is not a close fit to the expensive wind farm model.

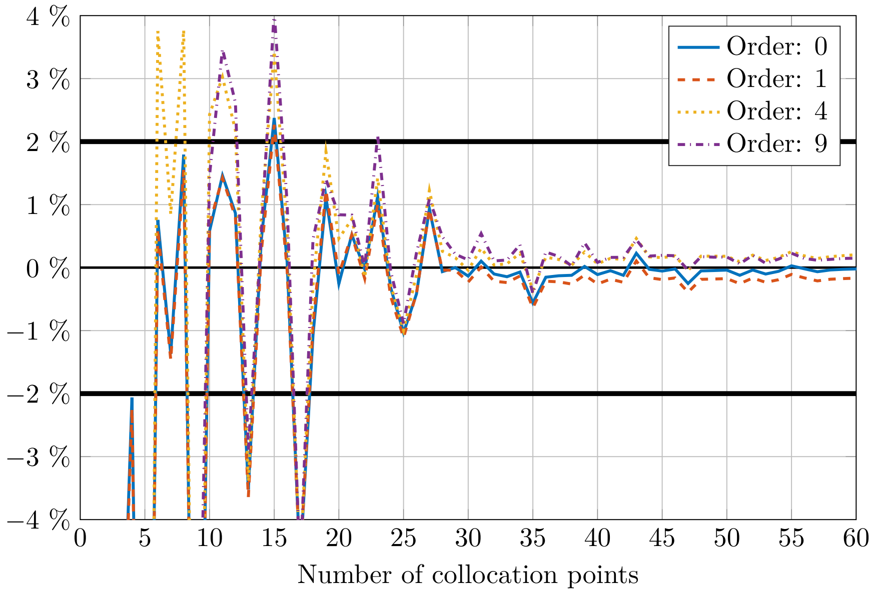


Figure 3-15: Difference (%) of the AEP for a wind speed of 8 m/s between the estimate by PCE and the exact solution for varying number of collocation points.

While the AEP may give good results for the wind direction even though the surrogate model is not so accurate, this is certainly not the case for the standard deviation, as can be seen in Figure 3-16. The deviation with the exact solution is enormous and the results can definitely not be used.

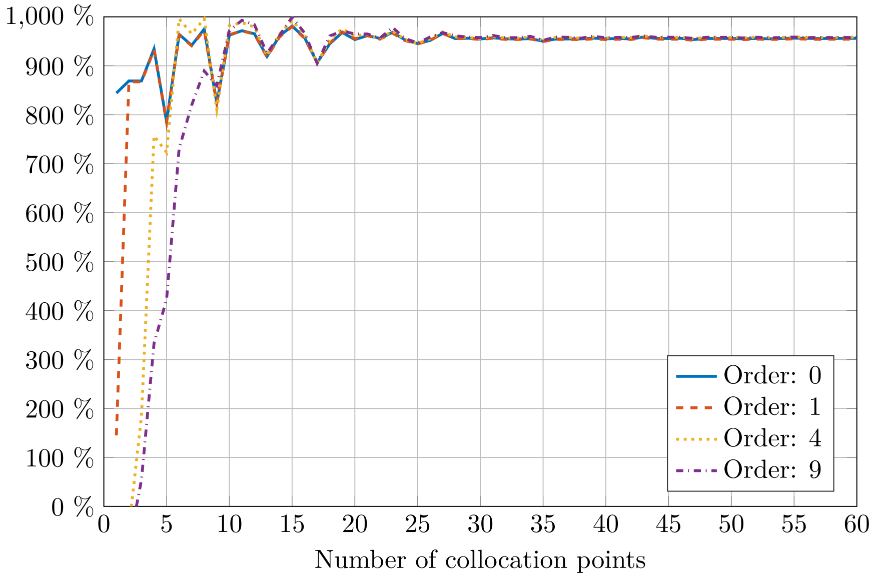


Figure 3-16: Difference (%) of the standard deviation for a wind speed of 8 m/s between the estimate by PCE and the exact solution for varying number of collocation points.

From Figure 3-14 it is clear that the surrogate model is not a good representation of the computationally expensive model. It can be seen that the surrogate model can not represent the wake effects that are present in the wind farm. A possible solution for this could be the use of multiple PCE surrogates, one per wind direction sector, instead of using one for the whole wind direction domain. This will be investigated in Section 3-5.

Since wind direction is circular, it is investigated what the result of a phase shift is on the accuracy of the surrogate model. In Figure 3-17, it is clear that applying a phase shift does not have a significant influence on the estimation of the AEP.

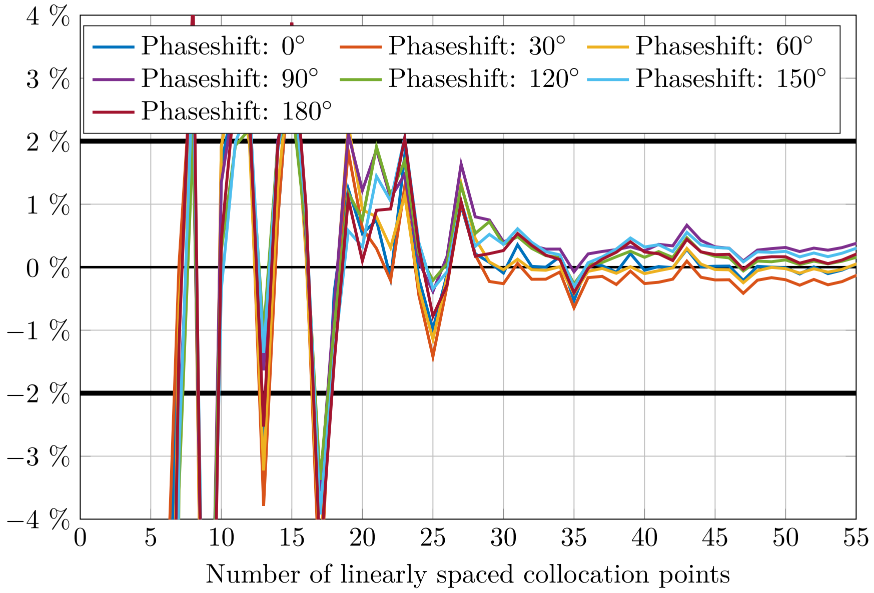


Figure 3-17: Difference (%) of the AEP for a wind speed of $8\,\mathrm{m/s}$ between the estimate by sixth order PCE and the exact solution with a varying phase shift.

3-4-2 Multivariate polynomials

In the previous subsection, a one-dimensional PCE surrogate model was constructed based on the wind direction. The goal of this research is to develop a two-dimensional surrogate model based on wind speed and wind direction. In order to do so, the methods from Section 3-2 needs to be extended in an extra dimension. The main problem with the construction of the two-dimensional surrogate model is that wind speed and wind direction are not independent of each other, which is a requirement of the PCE method. It will need to be investigated how the correlation affects the construction process and the final result of the surrogate model.

Researchers have already explored ways to develop a multivariate PCE. Oladyshkin and Novak [27] state more research is needed to develop a multivariate polynomial basis of dependent random variables. Jimenez et al [9] did already try to obtain this multivariate basis but experienced some problems on the way. Most importantly their approach results in a polynomial basis which is not unique and depends on the initial ordering of the random variables during the construction process of the orthogonal polynomials. This is not the case for the univariate techniques. Additionally they make the remark that due to the use of a polynomial basis based on correlated variables, it is also not really clear any more how each variable influences the obtained polynomial and that more research is needed in order to clear this up [9].

Since the method which is used to construct the multivariate polynomial basis has apparently an influence on the final result of the surrogate model, it was decided to develop two techniques and to compare them. Method one combines two existing one-dimensional polynomial bases. Method two uses the iterative Gram-Schmidt process with two-dimensional data-driven moments.

Combining two monovariate polynomial bases

A multivariate basis can be constructed with the use of a tensor product. At first, two sets (in case of two input variables) of orthonormal polynomials ϕ^1 and ϕ^2 need to be constructed using the methods from Section 3-2. Next, the maximum order of the expansion M is determined and for each order 1 to M a combination of the individual polynomials is made by multiplying them.

$$\phi_{m,n} = \phi^1_m \phi^2_n \quad \text{with} \quad m + n \leq M \tag{3-15}$$

As an example the different polynomials as a result of a fourth order expansion of a two-dimensional polynomial basis is given by the following table.

Table 3-1: Orders of each variable in the bivariate polynomial basis of the fourth order.

order of expansion	0	1		2			3				4				
order of ϕ^1	0	1	0	2	1	0	3	2	1	0	4	3	2	1	0
order of ϕ^2	0	0	1	0	1	2	0	1	2	3	0	1	2	3	4

The number of terms in this multivariate basis is equal to

$$\frac{(M+d)!}{M!d!}, \tag{3-16}$$

where M is the highest order of the polynomials and d is the dimension of the basis [37].

This method does not take into account the correlated nature of the two random variables. It is however the only method to use when information on the (individual) distribution of the random variables is known, but no dependence information is available.

Basis using two-dimensional data-driven moments

A multivariate basis can also be constructed using a direct data-driven approach. By doing so, the requirement for independent variables can be circumvented and the obtained polynomials would provide the best fit to the data. During the literature review no examples of this approach could however be found. Yang et al. started on this path but remarked that the combination of univariate polynomials (as described above) proves to be much easier and they decided to continue that way [38].

Despite the problems of Yang et al., it was decided to try this approach since the correlated moments can directly be computed from the data. This suggest that a better fit with the random variables can be obtained than the first two-dimensional method. The correlated moments can be computed by altering Equations (3-3) to (3-4) into

$$m_{n,l} = \mathbb{E}\left[X^n Y^l\right] = \int_{-\infty}^{\infty} x^n y^l \, \mathrm{d}F\left(x,y\right) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^n y^l f\left(x,y\right) \mathrm{d}x \, \mathrm{d}y \,, \tag{3-17}$$

and

$$m_{n,l} = \mathbb{E}\left[X^n Y^l\right] = \frac{1}{|X|}\sum_{i=1}^{|X|} x_i^n y_i^l \,. \tag{3-18}$$

The result of this adapted data-driven moment equation is a matrix of moments with in one dimension the X random variable and in the other the Y random variable.

The Gram-Schmidt process is still executed in the same way as before, but instead of using the initial sequence $v = \{1, x, x^2, ...\}$, a new sequence will be used, based on Table 3-1.

$$v = 1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3, ... \tag{3-19}$$

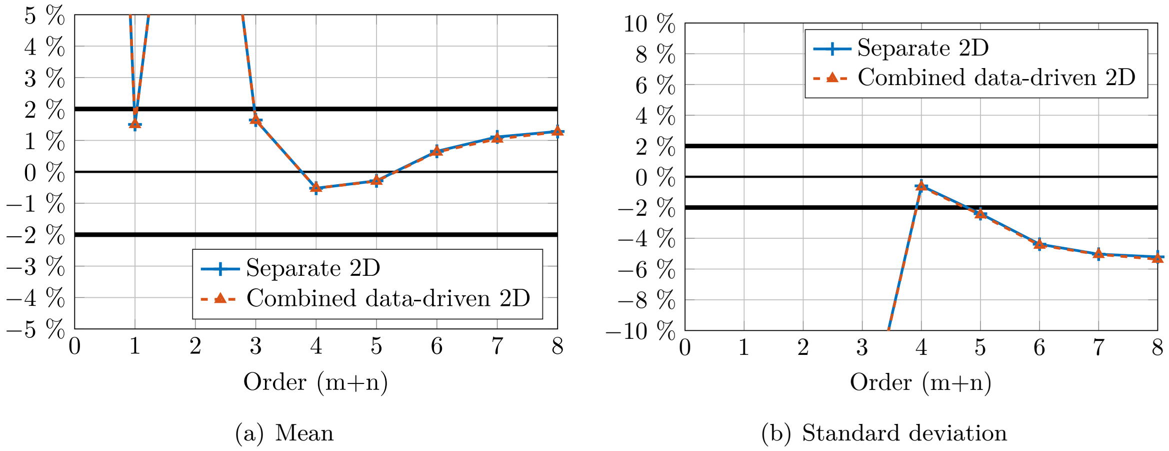
Equation (3-8) can still be used to compute the orthogonal polynomials, but it needs to be slightly altered to account for the extra random variable:

$$p_n\left(x,y\right) = v_n - \sum_{i=0}^{n-1} \frac{\langle v_n, p_i\left(x,y\right)\rangle}{\langle p_i\left(x,y\right), p_i\left(x,y\right)\rangle} p_i\left(x,y\right) \,. \tag{3-20}$$

The process to compute the polynomials is exactly the same as before, except for the fact that everything is now in two dimensions. The procedure to compute the inner products (see Example 3.1) is the same although the integration now happens over the double integral in x and y which is exchanged for the respective moment $m_{n,l}$.

Comparing the two two-dimensional construction methods

As stated above, the two-dimensional polynomial basis can be constructed in two ways: the combination of two one-dimensional polynomial bases or the modified Gram-Schmidt process. In Figure 3-18, a comparison of these two methods is shown for both the mean and standard deviation. It can be seen that there is almost no difference between the two methods. This was not entirely expected since the second method used moments for both random variables and should in theory thus give better results. However, from Section 3-4-1 it could be concluded that using wind direction as a random variable results in a bad surrogate model and this could hence also influence the results presented below.



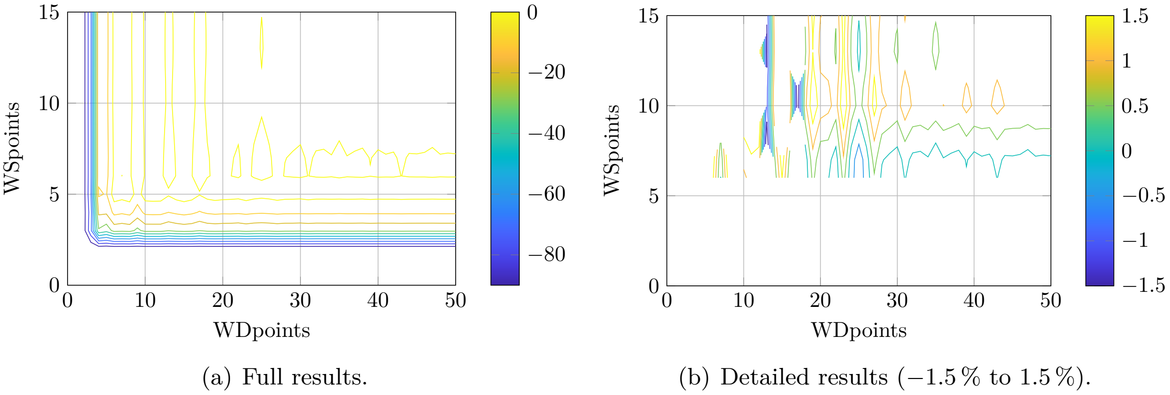
(a) Mean

(b) Standard deviation

Figure 3-18: Difference (%) between exact solution and surrogate model for varying order and for the two two-dimensional methods.

Number of collocation points

Although the wind direction does not result in a perfect surrogate model, it still performs well for an estimation of the AEP. Figure 3-19 shows a contour plot of the difference (%) between the approximation of the AEP and the exact solution for varying number of collocation points for the wind speed (y-axis) and wind direction (x-axis). The surrogate model was constructed using a polynomial basis of the fifth order constructed using the 2D data-driven Gram-Schmidt process. Figure 3-19(b) shows a detail of Figure 3-19(a). It can be seen that 6 wind speed collocation points perform the best while the number of wind direction collocation points is less important, as long as it is above 30.

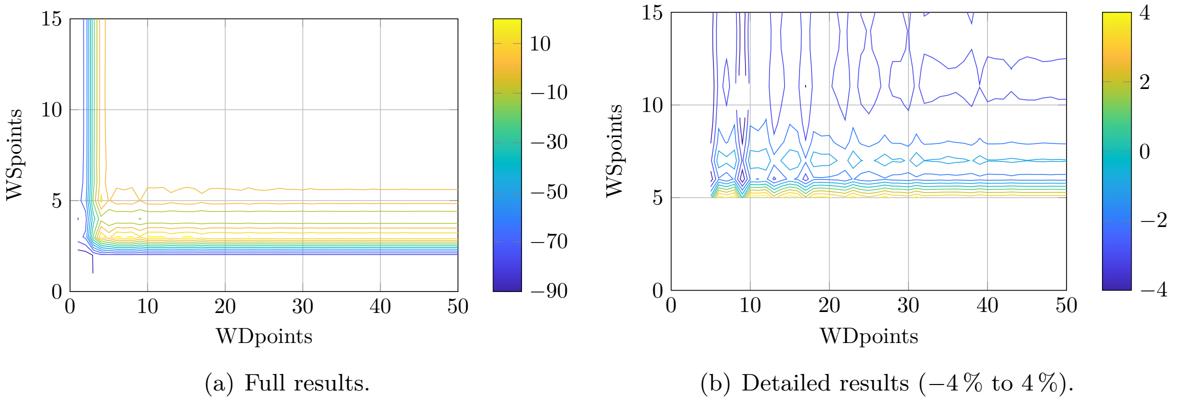


(a) Full results.

(b) Detailed results (−1.5 % to 1.5 %).

Figure 3-19: Difference (%) of AEP approximation with surrogate model and exact solution.

The same analysis can be done for the standard deviation and is shown in Figure 3-20. Also in this case, the best results are obtained when 6 wind speed collocation points are used.



(a) Full results.

(b) Detailed results (−4 % to 4 %).

Figure 3-20: Difference (%) of standard deviation approximation with surrogate model and exact solution.

3-4-3 Reduced polynomial basis

From the previous section, it was clear that using the wind direction does not result in a very good surrogate model. It may be interesting to see what happens when the highest order of the two random variables is not the same, or more extreme, when the surrogate model is constructed using only one random variable (wind speed). A reduced polynomial basis can be construction by adapting Equation (3-19) (page 38) to only include the powers that are wanted in the final surrogate model. In Figure 3-21 the a surrogate model is constructed using a fixed order (fifth) for wind speed, while the order of the wind direction can vary from 0 to 5 (on the x-axis). From this analysis it can be concluded that a minimum amount of collocation points for wind speed are necessary in order to obtain accurate results.

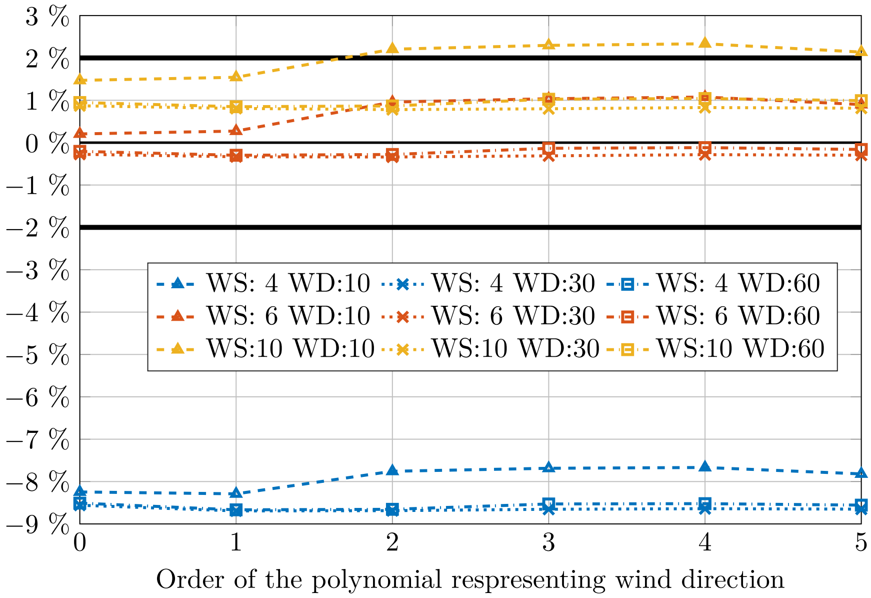


Figure 3-21: Difference (%) of the AEP from the reduced surrogate model and the exact solution for multiple combination of collocation points.

The result from Figure 3-19(b) is confirmed as it can be seen that 6 wind speed collocation points perform the best. In fact, 6 wind speed points perform better than 10. This is an unexpected result and its cause is unclear. For the wind direction it is important to have enough collocation points. Even though all options (10, 30 and 60 points) return acceptable results, it can be seen that the surrogate models with 10 collocation points for the wind direction show a bump when going from a first order to a second order polynomial for wind direction.

The same analysis was done for the standard deviation. of which the results are shown in Figure 3-22. The results for 4 collocation points for the wind speed are very low ($-19\,\%$) were omitted from the graph in order to increase the visibility of the other results. It was already established that it is difficult for the surrogate model to approximate the standard deviation, but in the figure it can be seen that acceptable results are obtained for 6 collocation points of the wind speed.

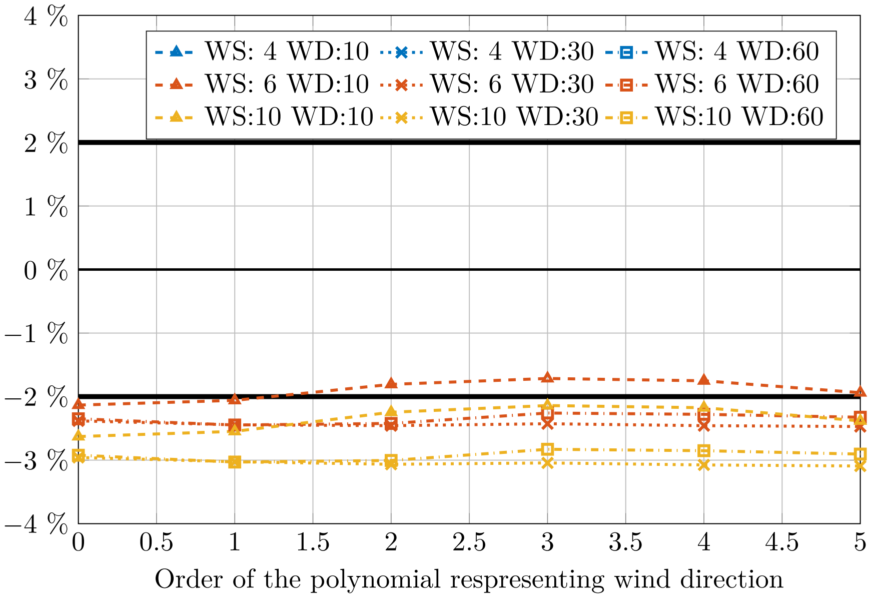


Figure 3-22: Difference (%) of the standard deviation from the reduced surrogate model and the exact solution for multiple combination of collocation points.

In Figures 3-21 and 3-22 it became clear that even a zeroth order polynomial for the wind direction results in an acceptable surrogate model, as long as the expensive model is still evaluated at a sufficient number of wind direction collocation points. In Figure 3-23, the order of the one-dimensional polynomial basis for wind speed was varied, and various sets of collocation points were used. From the graph, it can be concluded that a fourth or fifth order expansion with 6 wind speed and 30 wind direction collocation points give very good results for the approximation of the AEP.

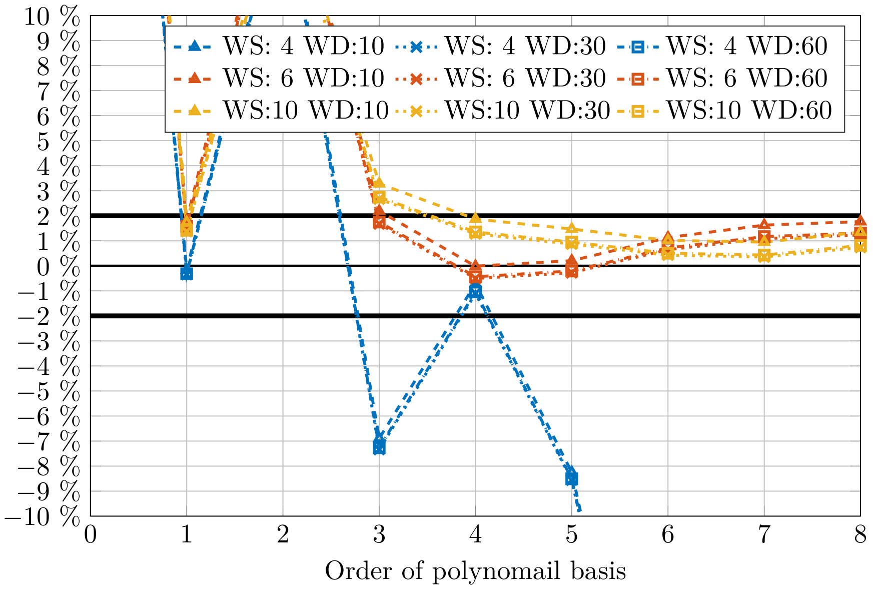


Figure 3-23: Difference (%) of the AEP from the reduced (zero expansion for wind directin) surrogate model and the exact solution for multiple combination of collocation points.

3-4-4 Varying the layout of the wind farm

All of the above analyses were performed on the layout of the Horns Rev wind farm.[1] In this section, a few alternative layouts will be analysed to see whether PCE is also able to make a good approximation of the AEP for these wind farms or not. In Figure 3-24, the difference between the computed and exact AEP is shown in percent. Since the wind farms have a varying number of installed wind turbines, this comparison was done for each layout individually. All layouts were evaluated using a surrogate model constructed by the data-driven 2D Gram-Schmidt process while using 6 wind speed collocation points and 30 wind direction collocation points, as described in the previous section. It can be seen that the approximations of the AEP are good, but that there exist some differences between the various layouts. It is unclear why two of the layouts have a lower error at an order of seven and eight while the error grows for the other layouts.

[1] To be more precise: since the layout of the wind farm under investigation didn't change, the power produced by the wind farm for each 1 m/s and each 1° were saved in a matrix and loaded in for each analysis. When needed, a 2D interpolation was used to retrieve the produced power of the wind farm at a specific wind speed and wind direction.

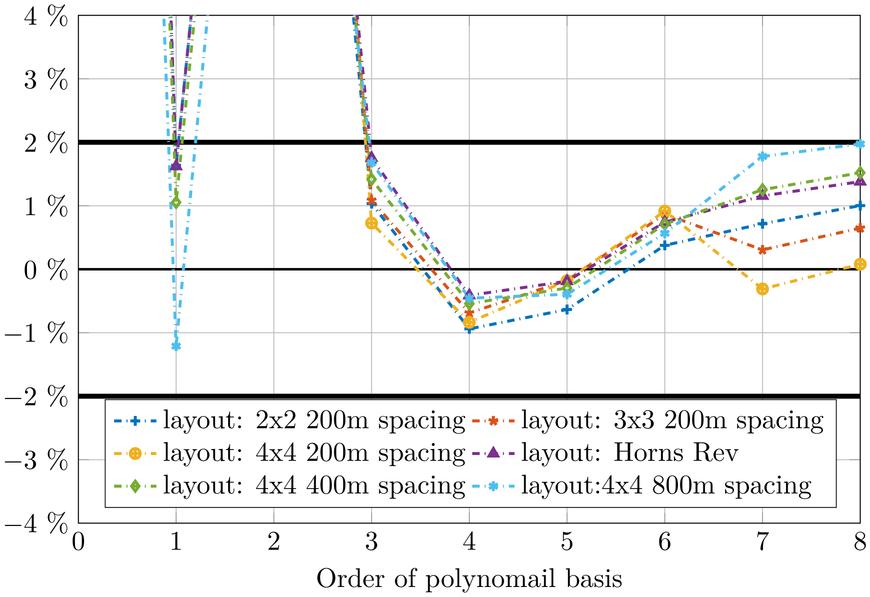


Figure 3-24: Difference (%) of the AEP from the surrogate model and the exact solution for various wind farm layouts.

3-5 PCE surrogate per wind direction sector

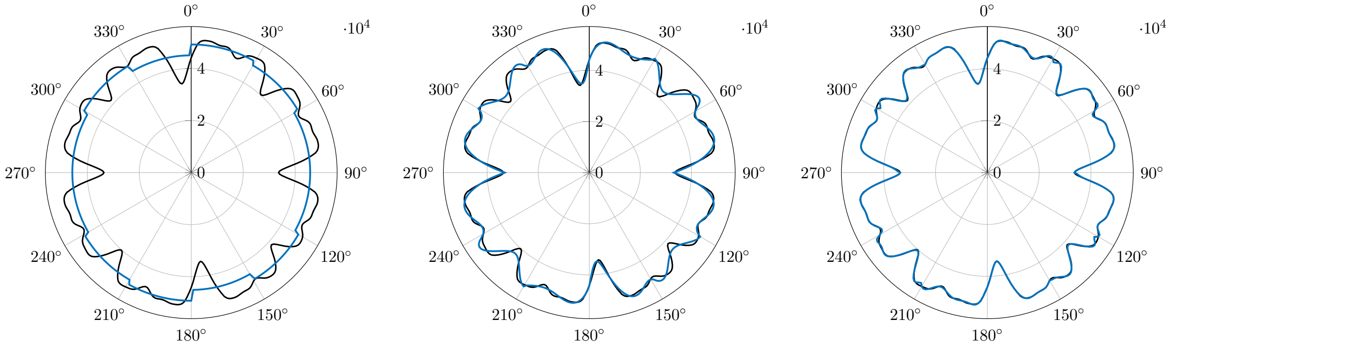
In the previous sections, it became clear that while a surrogate model based on PCE may give good results for the estimation of the AEP, this is definitely not always the case for the standard deviation, especially when considering the wind direction. In this section a PCE surrogate per wind direction sector will be used in order to improve the final surrogate model. While the ‘normal’ PCE approach uses only one parameter to indicate the number of collocation points for the wind direction, this new approach uses two: The number of wind direction sectors and the number of collocation points per wind direction sector.

3-5-1 One-dimensional sectorwise PCE

For the one-dimensional sectorwise PCE, the domain of the wind direction is divided in sectors. For each sector, a polynomial basis is constructed, collocation points are determined and evaluated on the computationally expensive model and a local surrogate model is constructed. All the local surrogates are then combined to finally end up with one global surrogate model. For the following analysis, the Horns Rev wind farm layout and a constant wind speed of 8 m/s are used.

In general it is a good idea to use data-driven moments (as shown before). However there is a slight complication when dividing the data in per sector and computing the moments from this select data set. As an example, consider the wind direction sector from 300° to 330°. When only this sector is considered and all other data is neglected, the moments are huge since they are calculated with respect to 0°. When considering a full data set, there is a lot of information between 0° and 300° which results in an overall lower moment. It has been shown that high moments are the cause of numerical inaccuracies and in this particular case, the moments of wind direction sectors far from 0° were so high that a polynomial basis could not be constructed. A solution for this problem was found by dividing the full data set in sectors and shifting the data into the range 0° - wind sector width (e.g. 30°). The polynomial basis gets constructed on this domain, and gets fitted to the results of the computational model on the original domain (300° - 330°). The fitted surrogate model is then transferred back to its original wind direction sector.

The result of this analysis for a wind sector width of 30° with 10 wind direction collocation points per wind sector is shown in Figure 3-25. It can be seen from these graphs that a much better fit to the wind farm model is obtained.



(a) zeroth order (b) Fourth order (c) Eight order

Figure 3-25: Surrogate model for the Hornz Rev wind farm with a constant wind speed of 8 m/s with the new sectorwise PCE with a sectorwidth of 30° and 10 collocation points per wind direction sector.

In order to measure this ‘fitness’, a fitness function was developed. For the one dimensional case it is defined as

$$\left(1-\frac{\int_{0^\circ}^{360^\circ}\left|\hat{P}\left(\theta\right)-P\left(\theta\right)\right|\mathrm{d}\theta}{\int_{0^\circ}^{360^\circ}P\left(\theta\right)\mathrm{d}\theta}\right)100\,\%\,, \tag{3-21}$$

where $\hat{P}(\theta)$ represents the surrogate model and $P(\theta)$ the computationally expensive model, both in function of wind direction. The fitness in function of the order of the PCE surrogate model is shown in Figure 3-26. It can be seen that the fitness becomes better when the order of the surrogate model increases.

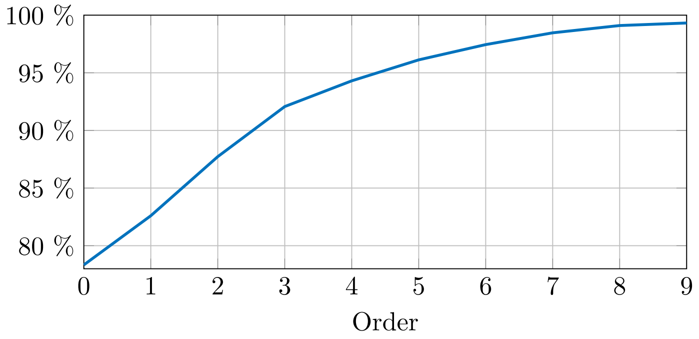
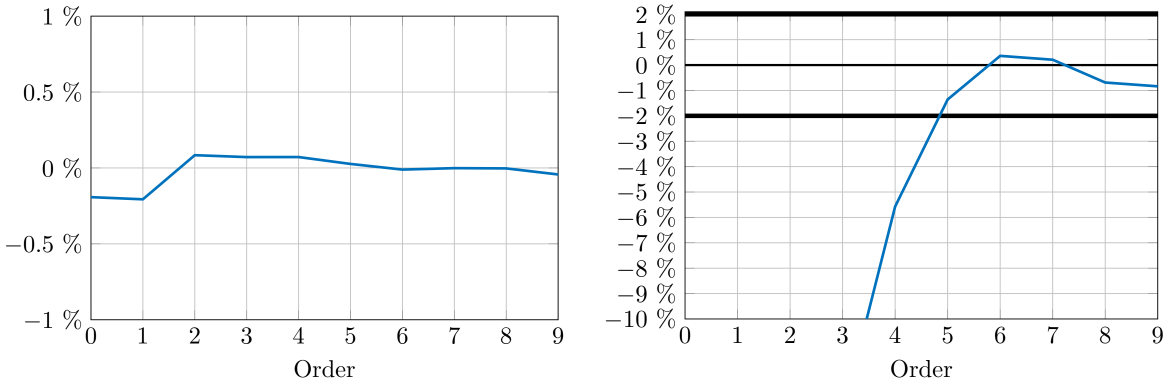


Figure 3-26: Fitness function (%) of the sectorwise PCE.

The estimate of the mean and standard deviation are not as straightforward as when considering one PCE for the whole wind direction domain. The results could be obtained by computing the weighted average of each wind direction bin (with respect to the frequency of each bin in the full data set, see Figure 2-5). However, the choice was made to evaluate the resulting polynomial surrogate model at each point of the time series data set and to use the mean and standard deviation of this result. Since the results from the time series are obtained in the same way, but with the use of the computationally expensive model, this proves to be an excellent way to compare the results of the sectorwise PCE with the exact results. Both results are shown in Figure 3-27. It can be seen that the estimation for the AEP (Figure 3-27(a)) is almost perfect. The accuracy of the standard deviation estimate increases with increasing order but still isn't perfect.



(a) Mean.

(b) Standard deviation.

Figure 3-27: Difference (%) of mean and standard deviation for the one dimensional sectorwise PCE surrogate with respect to exact solution.

In order to find the best combination of parameters, some comparisons can be made. First the sector width will be varied with a constant number of points per sector. Afterwards the number of points will be changed for a constant sector width.

Varying the sector width

A fixed number of 10 collocation points is chosen per wind sector. it can be seen in Figure 3-28 that the smaller the bin width, the better the fitness of the surrogate model. This is as expected, even more so since the number of collocation points per sector is constant. The surrogate model with the smallest wind sector has thus the largest number of points over the whole domain.

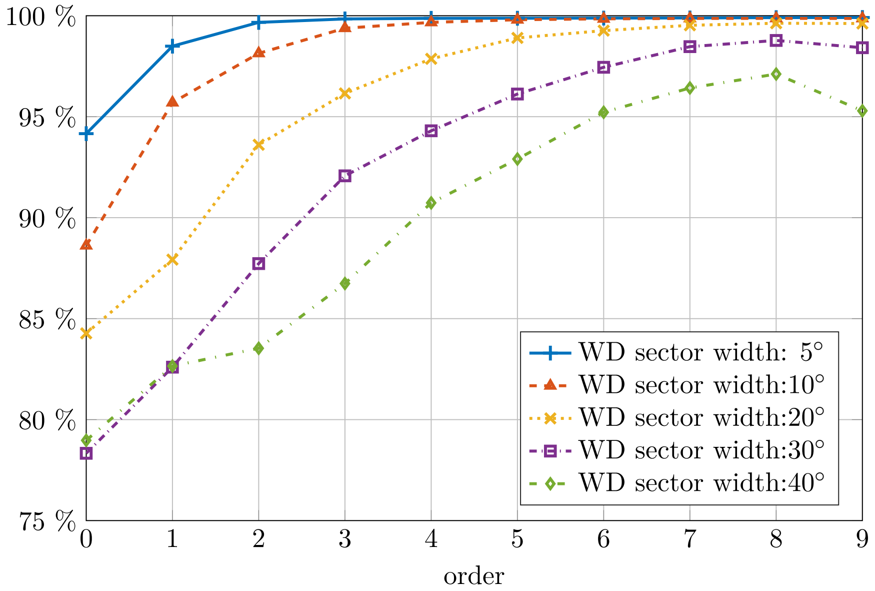
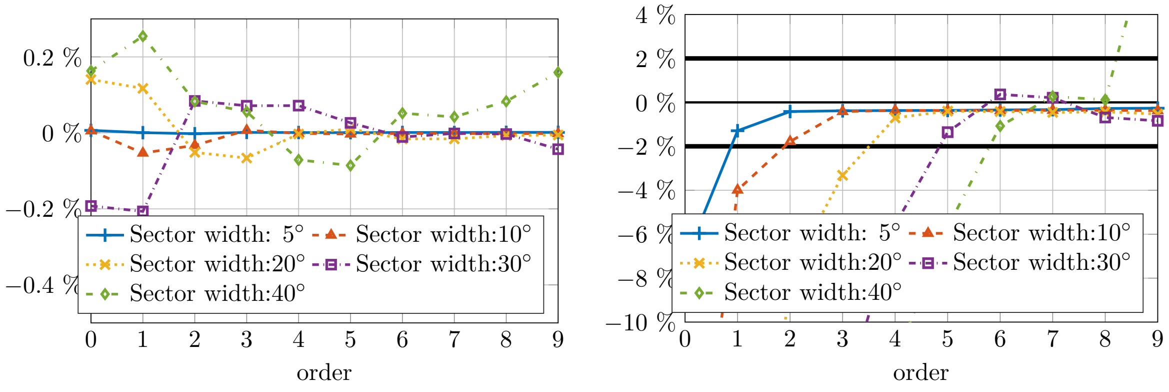


Figure 3-28: Surrogate fitness (%) for a sectorwise PCE with varying wind sector width and fixed number of points per wind sector.

In Figure 3-29(a), the estimate for the mean is shown in function of the order of the surrogate model. It can clearly be seen that the approximation is very good for all sector widths and perfect for a bin width of 5°. In Figure 3-29(b), the difference between the standard deviation from the surrogate model and the time series is shown. For a small sector width, the results are good, but the larger the bin width, the less accurate is the estimate. This result could also have been predicted from the fitness of the surrogate models (Figure 3-28).



(a) Mean

(b) Standard deviation

Figure 3-29: Differences (%) between the estimates of mean and standard deviation with respect to the time series results for a sectorwise PCE with varying wind sector width and fixed number of points per wind sector.

Varying the number of points per sector

Based on the results in Figure 3-29, a wind sector width of 20° was chosen for the next analysis. It was chosen because of its behaviour in the standard deviation analysis. The bin width is not too narrow, but it still gives good results and didn't overshoot the 0 % line like the larger sector widths. In Figure 3-30, the fitness values for the constructed surrogate models can be seen. Notice how the line of 5 points per sector takes a deep dive on the fifth and sixth order surrogate model. It can thus be seen that at least $n + 1$ collocation points for a nth order PCE surrogate model are required.

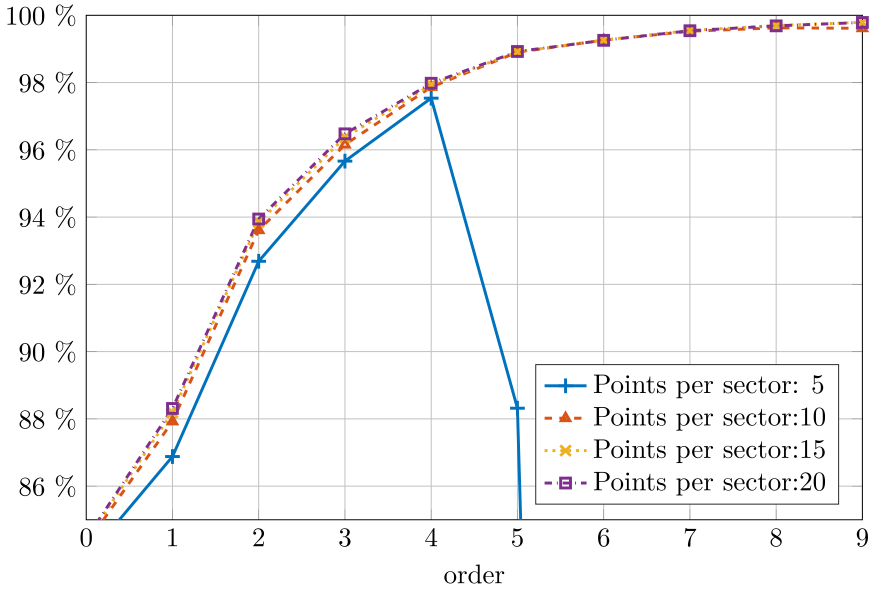
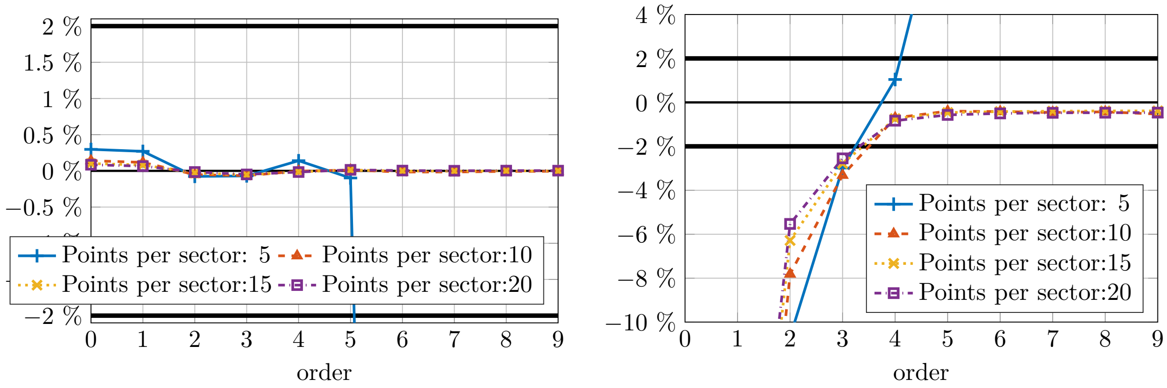


Figure 3-30: Surrogate fitness (%) for a sectorwise PCE with fixed wind sector width and varying number of points per wind sector.

In Figure 3-31 it can be seen that the influence of the number of points per sector is limited, as long as it is at least one more than the order of the PCE.

Different conclusions can be drawn depending on the application of the surrogate model:

- Estimation of the mean: In this case the sectorwise PCE is flexible. A sector width of 40° and a fourth order surrogate model give good results and doesn't require too many collocation points.
- Estimation of the standard deviation: In this case a smaller bin width or a higher order expansion (and thus higher number of collocation points) is advisable.
- Construction of a good fitting surrogate model: It is a good idea to go with a sector width of 5° or 10° and an appropriate order of the surrogate model.



(a) Mean

(b) Standard deviation

Figure 3-31: Differences (%) between the estimates of mean and standard deviation with respect to the time series results for a sectorwise PCE with fixed wind sector width and varying number of points per wind sector.

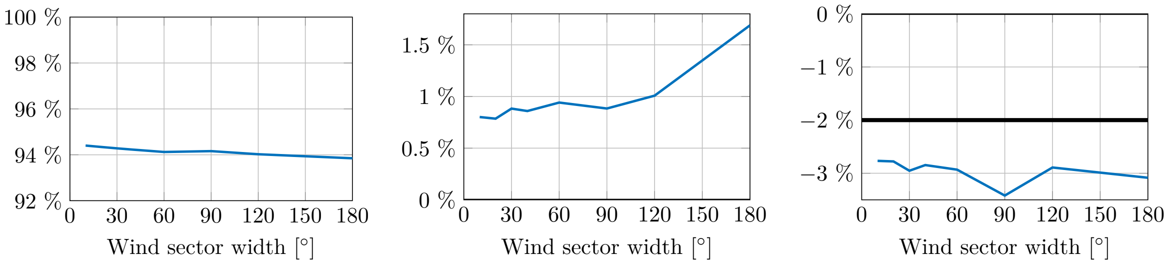
3-5-2 Two-dimensional sectorwise PCE

The previous analysis was done in one dimension. Now it is time to extend this to incorporate the wind speed distribution and to obtain a two-dimensional sectorwise PCE. The results can not be shown like in Figure 3-25 since this is a two-dimensional problem. The 1D fitness function is extended into

$$\left(1-\frac{\int_{0^\circ}^{360^\circ}\int_{v_{\text{cut-in}}}^{v_{\text{cut-out}}}\left|\hat{P}\left(v,\theta\right)-P\left(v,\theta\right)\right|\mathrm{d}v\,\mathrm{d}\theta}{\int_{0^\circ}^{360^\circ}\int_{v_{\text{cut-in}}}^{v_{\text{cut-out}}}P\left(v,\theta\right)\mathrm{d}v\,\mathrm{d}\theta}\right)100\,\%\,, \tag{3-22}$$

and now uses a surrogate model and computationally expensive model in function of wind speed and wind direction (v, θ). The results of this fitness function, together with the difference of the mean and standard deviation with respect to the time series result, are the only way to evaluate the surrogate model. Since there are now multiple parameters, it is decided to fix the order of the surrogate model at 5 (which has proven to be an appropriate value). Now, the number of wind sectors, the number of collocation points in these wind sectors and the number of wind speed collocation points can be changed. From the one-dimensional analysis it could be seen that the minimum amount of collocation points is one more than the order of the surrogate. It thus follows that the amount of wind speed or wind direction collocation points should be at least 6. As a basis the following parameters are chosen: 10 wind speed collocation points, 10 wind direction collocation points and a sector width of 20°.

Unfortunately, not a lot of conclusions can be drawn from the analysis. In Figure 3-32, the wind sector width was varied. As expected, the fitness of the surrogate and accuracy of the mean and standard deviation decrease with increasing sector width. Figure 3-33 shows a variation in collocation points per wind direction sector for a constant sector width of 20°. From the graphs it seems like the number of wind direction collocation points does not matter. The same analysis was done for a variation of the number of wind speed collocation points. The results are shown in Figure 3-34. A small increase in the fitness can be noticed when increasing the number of collocation points while the error in the estimation for the mean and standard deviation grows.

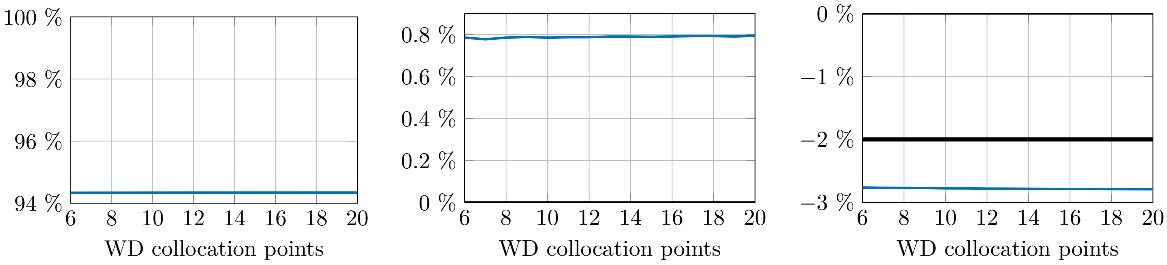


(a) Fitness function.

(b) Difference: mean

(c) Difference: standard deviation

Figure 3-32: Sectorwise PCE surrogate with varying sector width.

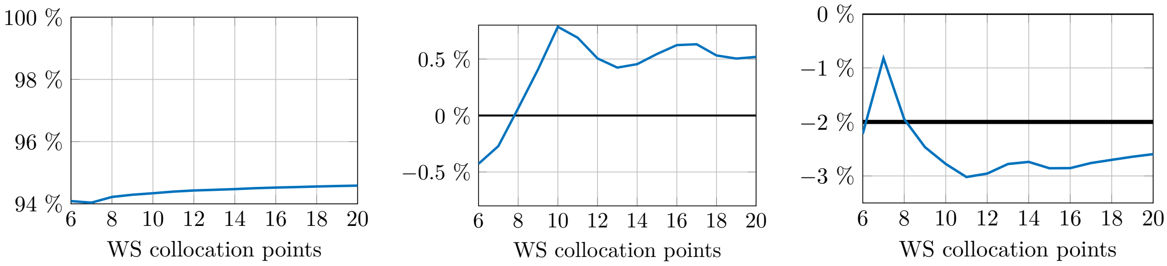


(a) Fitness function.

(b) Difference: mean

(c) Difference: standard deviation

Figure 3-33: Sectorwise PCE surrogate with varying number of collocation points per wind sector.



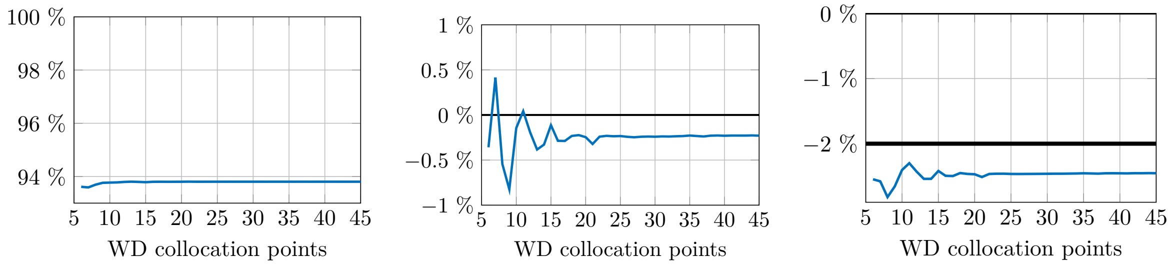
(a) Fitness function.

(b) Difference: mean

(c) Difference: standard deviation

Figure 3-34: Sectorwise PCE surrogate with varying number of wind speed collocation points.

As a test, the analysis was repeated with a sector width of 120° and a varying number of wind direction collocation points per wind sector. The results are shown in Figure 3-35. Note that the same behaviour in the estimation for the mean is seen as in the normal PCE surrogate model (see Figure 3-15 on page 35). This indicates that the sector width needs to be chosen sufficiently small so that a good surrogate model is obtained and that 120° is too big. It is a surprise that the fitness of this surrogate model is still really high. This may indicate that a better fitness function should be found to evaluate the accuracy of the surrogate model. Compared with the results from the standard approach for two-dimensional PCE, it can be concluded that the accuracy for the estimation of the mean is increased, but that no significant improvement can be seen for the estimation of the standard deviation.



(a) Fitness function.

(b) Difference: mean

(c) Difference: standard deviation

Figure 3-35: Surrogate model based on sectorwise PCE for the Hornz Rev wind farm with varying number of collocation points per wind sector.

3-6 Conclusion

In this chapter, polynomial chaos expansion was introduced. The principles and construction choices were explained and evaluated. It can be concluded that PCE works in a wind energy context, mainly to estimate the AEP of a wind farm.

It is recommended to use data-driven moments since these result in polynomials that are more representative of the data. Moments from statistical distributions can also be used, but only when this distribution is a good fit to the data set.

The Gram-Schmidt process was chosen as the construction method for the orthonormal polynomials. It uses an iterative process and as a result it means that numerical inaccuracies in higher order moments do not have an influence on the accuracy of the first polynomials of a lower order. It also provides a good extension into a second dimension.

The wind direction does, even with the data-driven moments, not result in a good surrogate model. It is still possible to compute the AEP but the standard deviation is not accurate enough.

For the extension into two dimensions, the modified Gram-Schmidt process worked as expected. The difference with the other method (combination of one-dimensional polynomial bases) is small and this is most likely the result of the fact that the wind direction does not result in a good fitting surrogate model.

A sectorwise PCE approach was attempted. This worked well in one dimension (only wind direction) to give accurate estimates of the mean and standard deviation. In two dimensions, a slight increase in accuracy for the estimation of the mean can be seen, but no (big) improvement is seen for the standard deviation as compared with the standard PCE approach in two dimensions.

It is recommended to use a polynomial basis of the 5th order and to use 6 linearly spaced collocation points for the wind speed and 30 linearly spaced collocation points for the wind direction.

Chapter 4

Wind farm layout optimisation

4-1 Introduction

In this chapter the surrogate models constructed with the polynomial chaos expansion technique will be used to perform a wind farm layout optimisation. First, an optimisation algorithm will be selected and introduced, after which the performance of the conventional annual energy production (AEP) calculation and polynomial chaos expansion (PCE) will be compared.

4-2 Objective functions

Various optimisation objectives exist and show a different view on the focus of the optimisation problem. The simpler methods can provide a fast evaluation of the model but may not be as accurate as the more involved, computationally expensive models [39]. Some optimisation objectives include:

- **Maximum power production:** a common choice of optimisation objective. This approach produces a layout with the highest energy output but does not take the price of this energy into account.
- **Maximum profit:** For this approach, more information is needed on tax incentives or subsidies since it tries to maximise the profit for the turbine and/or site owner.
- **Minimization of cost of energy:** This approach is the most beneficial from the standpoint of the user of the produced electricity. It provides a balance between the maximum power production and the associated costs.

Often the levelised cost of energy (LCOE) is used as the objective function of the optimisation process. This falls in the last category of the above list. An example of such function is

$$\text{LCOE} = \frac{C_{\text{inv}}}{a_f \cdot AEP} + \frac{C_{\text{OM}}}{AEP} + \frac{C_{\text{decom}}\,(1 + r_i)^{-T_e}}{a_f \cdot AEP}\,, \tag{4-1}$$

where a_f is the annuity factor, C_{inv} is the upfront investment cost, C_{OM} is the annual operational and maintenance cost, C_{decom} is the decommissioning cost at the end of life, r_i is the interest rate and T_e is the length of the economic life of the wind farm [40].

Since this research project is however more focussed on the implementation of PCE, it can be more practical to choose for a simpler objective function that does not require as much 'outside' information. Mosetti et al [4] proposed the following objective function to minimize

$$w_1 \frac{1}{P_{\mathrm{tot}}} + w_2 \frac{C_{\mathrm{tot}}}{P_{\mathrm{tot}}}\,, \tag{4-2}$$

where p_{tot} is the AEP and w_1 and w_2 are the weights of the respective parts of the equation [4]. C_{tot} represents the costs of the wind farm which is computed as

$$C_{\mathrm{tot}} = N_t \left(\frac{2}{3} + \frac{1}{3} e^{\left(-0.00174 N_t^2\right)} \right)\,, \tag{4-3}$$

where N_t is the number of turbines installed in the wind farm. The factors in the equation come from the assumption that the non-dimensional cost of the turbines can be reduced by a maximum of $^1/_3$ when many of them are installed. According to a study by Herbert et al., more than 80 % of the papers they investigated used this optimisation objective [5]. They however also state that it is very unlikely that wind turbine manufacturers used this simple cost function since for values of N_t higher than 35 it converges to $^2/_3 N_t$. However, since this cost function is so widely used, it provides a good way to compare results between numerical methods [5]. Since the parameters w_1 and w_2 can be chosen at will (depending on the focus of the objective), the objective function is often reduced to [41]

$$\frac{C_{\mathrm{tot}}}{P_{\mathrm{tot}}}\,. \tag{4-4}$$

It was decided to use wind farms with a fixed number of wind turbines so there is no need for an objective function which takes a varying number of wind turbines into account. Therefore, the following objective function was used in all following analyses:

$$-P_{\mathrm{tot}}\,, \tag{4-5}$$

which maximises the power production. The negative of the generated power of the wind farm is chosen since all optimisation algorithms in MATLAB are implemented as minimisation problems. By using the negative of the power, the intermediate results in the MATLAB console can easily be interpreted. This would not have been the case if an objective function like $^1/_{P_{\mathrm{tot}}}$ was used.

4-3 Optimisation algorithms

In the same way that multiple objective functions exist, also multiple optimisation algorithms are used by various researchers. The methods can be put on a scale ranging from deterministic (fast but lower quality) to random (slow but higher quality). In this context 'quality' is a measure for the ability to find the most optimal layout. The mostly used methods are gradient search algorithm (GSA), greedy heuristic algorithm (GHA) and genetic algorithm (GA).

4-3-1 Gradient search algorithm

Gradient based algorithms use information about the gradient of the objective function. They search in the decreasing direction (of a minimisation objective function) and in this way limit the search in the design space. Since optimisation problems are often non-linear, the gradients need to be recalculated at every new state of the objective function's inputs. When using an objective function that is not analytical, a finite-difference approximation of the gradient can be used for this algorithm. This approximation can however have a misleading effect on the search [42]. The disadvantage of using a GSA is that the method only looks for minima starting from its initial seed. This means that, depending on the configuration where the method was started, the algorithm can get stuck in a local minimum instead of the global minimum. This makes the algorithm however perfect for a two step strategy where the result of a coarser algorithm (that looks for a global minimum) can be refined by using GSA [43].

4-3-2 Greedy heuristic algorithm

In order to overcome the behaviour of gradient search algorithm, researchers developed methods which introduce a form or randomness in their search, like the greedy heuristic algorithm (GHA). The GHA initialises with a guess. The algorithm then performs the following actions on the initial seed in order to produce potential solutions. The actions by the algorithm can be constrained by e.g. the number of turbines or layout constraints.

- **Add:** A specified number of new turbines is added to the initial seed. They are added one by one and for each new layout, the objective function is evaluated.
- **Remove:** A number of turbines is removed. The new layout is evaluated for each affected turbine.
- **Move:** A certain percentage of the turbines is moved in a certain direction. Per step in direction and per turbine, the objective function is evaluated.

At the end of each iteration (which consists of the above actions), all potential layouts are compared and the best of these potential solutions is selected. It becomes the seed for the next iteration. The name greedy originates from the fact that only the best solution is selected to 'live on' [39]. When no improvement is seen for a predetermined amount of iterations, the algorithm is stopped.

4-3-3 Genetic algorithm

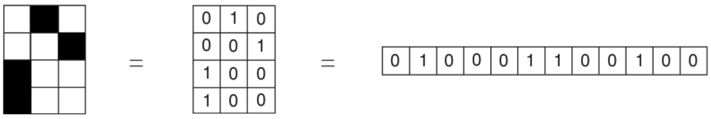
The most used class of optimisation algorithm for the wind farm layout optimisation problem (WFLOP) is the genetic algorithm (GA). They are based on Darwin's law of natural selection. The algorithm emulates the process of genetic reproduction where information of the parents' chromosomal string is combined into a new offspring who shares genetic information. Just like in biology, the fittest individuals survive and reproduce [4, 17].

The GA performs certain actions to combine the genetic information from the parents in order to produce new offspring [4, 17, 44–46].

- **Selection:** Individuals from the population are selected to become parents. They are chosen based on a fitness function. It can be an advantage to use a fitness function that is not the same as the objective since doing so can lead to premature convergence.
- **Crossover:** A number of crossover points is selected. In case of one crossover point, two parents are selected and the first parent copies its genetic information up to the crossover point. The second parent finishes the genetic string till the end. As such, two children are created (child one as described above, while child two first contains information of the second parent and then of the first one).
- **Mutation:** In this step a random change is done on one of the elements of the genetic string. This is an important step in the algorithm since without it evolution would not really occur and no new information would enter the population.
- **Reinsertion:** A combination of the parents and children are selected to become the new population. This selection is most often based on some sort of fitness function in order to prevent unfit children of replacing fit parents.

Binary genetic algorithm

In the binary genetic algorithm (BGA), the design space is coded into a gene. The location of the wind turbines in the wind farm is coded in ones and zeros in the genetic string where each position in the string indicates a position in the design space (see figure 4-1) [39]. This means that a design space of 10 x 10 squares where turbines can be placed, results in 2^{100} possible layouts. This is of course reduced by a constraint on the amount of turbines that can be present in the wind farm, but it shows that the scale of this type of optimisation can grow very rapidly.



=

0	1	0
0	0	1
1	0	0
1	0	0

=

0	1	0	0	0	1	1	0	0	1	0	0

Figure 4-1: Example of a wind farm and the genetic representation of that wind farm. Black squares and 1's indicate cells occupied by turbines. White squares and 0's indicate empty cells [39].

As indicated by Samorani, the BGA approach has a disadvantage in the sense that its solution space is discrete [1]. Since the layout space is divided in cells, not all wind turbine positions can be evaluated and hence the most optimal solution may just not be a part of the solution subset that was selected at the start of the algorithm. This can be overcome by choosing a finer grid, but this comes at a huge computational cost.

General genetic algorithm

Instead of the BGA, it is also possible to use coordinates for each wind turbine and to encode them in a large string. This string is then subjected to the same actions as explained for the BGA above. This approach ensures that the wind turbines can be placed at any possible position in the wind farm, and are not limited to the squares that were selected for the binary GA. There is however a slight complication with this reasoning: there is a minimum distance between two turbines which is currently neglected in the optimisation algorithm. Two distance criteria exist:

- **Physical:** Wind turbines can not be placed within one rotor diameter of each other, otherwise the blades would touch their neighbours' blades.
- **Computational:** The used wind farm model of Bo Hu uses a wake model which is only valid starting from a distance that is at least two rotor diameters from the turbine. This is the strongest constraint and will thus be used in this optimisation algorithm.

In order to make sure that the optimisation algorithm will return valid layouts so that valid results from the wake model can be obtained, a non-linear constraint will be used. It states that no wind turbine (WT) can be placed in a square with sides of length $4 \cdot \mathrm{WT}_D$ where the turbine in question is in the middle of the square.

$$c_n : \max\left(\left|\mathrm{WT}_{x,n} - \mathrm{WT}_{x,i}\right|, \left|\mathrm{WT}_{y,n} - \mathrm{WT}_{y,i}\right|\right) > 2 \cdot \mathrm{WT}_D \tag{4-6}$$

c_n should be true for all n for each wind turbine i.

4-3-4 Choice of optimisation algorithm

As the GA is the most used algorithm in wind farm layout optimisation and Matlab has it built-in in the optimisation toolbox, it was decided to use genetic algorithm for all following analyses. A choice was made for the version of GA that uses coordinates over the binary version. Both are a valid choice, but the non-binary variant of the algorithm has a much bigger solution space and can thus possibly also result in more optimal layouts. In the next sections the coordinate version of GA will simply be indicated by GA.

4-4 Implementation of the algorithm in Matlab

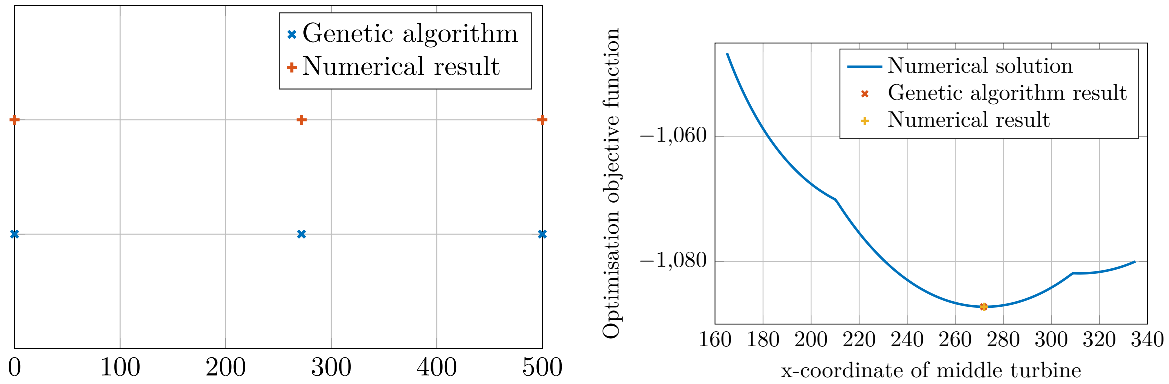
The implementation of the optimisation process in Matlab was done in various phases. The first steps were done with the direct use of the original Bo Hu model (so without the use of the PCE surrogate model), since it has already been proven to work when used in an optimisation method. The following steps were taken:

- **One-dimensional:** The first step was to perform an optimisation for the one-dimensional case (a line) with a constant wind speed and wind direction. The results were checked and verified.
- **Two-dimensions:** The code was extended to a second dimension and for a fixed wind speed and wind direction, the expected power production for a 3 by 3 (9 wind turbines) wind farm layout was used to optimise the layout.
- **Varying wind conditions:** Once the algorithm was coded for two dimensions, a varying wind speed and wind direction were used in order to compute the AEP of the 3 by 3 wind farm layouts. This number was then used in the optimisation process.

- **Testing of the optimisation parameters:** A lot of options are available in the optimisation toolbox in Matlab. An analysis to find the best combination of options was performed.
- **Including PCE:** All previous results and the PCE method were combined in the final step: an optimisation algorithm that uses polynomial chaos expansion.

4-4-1 One-dimensional optimisation

The algorithm was first implemented using a wind farm consisting of three turbines on a horizontal line with constant wind speed of 8 m/s. The wind direction was fixed at 270° (see windrose in Figure 2-4) since otherwise the wind conditions from both left and right would be the same and the optimal layout would be symmetric. For this first implementation, the wind farm model was evaluated directly since only one wind speed and wind direction were used. The possible position for each of the three turbines was limited between 0 m and 500 m. Additionally, the constraint from Equation (4-6) was used to make sure that the distance between the turbines is at least two rotor diameters ($= 2 \cdot 80\,\text{m} = 160\,\text{m}$). An initial layout vector $[0, 200, 400]$ was provided to the algorithm. The results can be seen below in Figure 4-2(a) (blue markers). It can be seen that the most optimal layout results in a turbine placed on $x = 0$, another one on $x = 500$ and the last in the middle at $x = 272$.



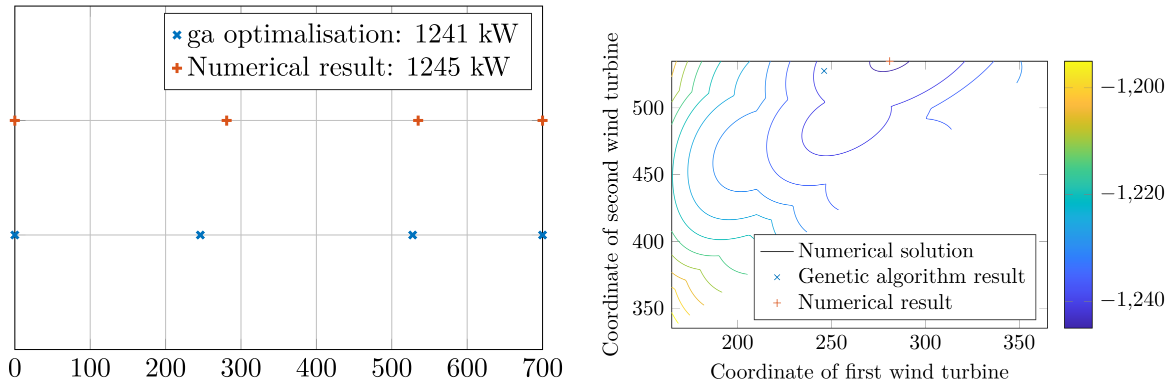
(a) The two resulting optimised layouts.

(b) Numerical solution for every position of the middle turbine.

Figure 4-2: Optimisation with genetic algorithm and numerical solution of a wind farm consisting of three wind turbines.

This result can easily be verified since only three wind turbines are present. It is clear that two turbines need to be placed at the edges of the position constraint ($x = 0$ and $x = 500$) while the other can have a position between those two. With a for loop, the objective function was evaluated at each layout $[0, 2 \cdot WT_D ... 500 - 2 \cdot WT_D, 500]$ in steps of 1 m. As can be seen in Figure 4-2(b), the same optimal layout is obtained as with the optimisation process.

The test was repeated with 4 wind turbines instead of 3. All turbines now need to have an x-coordinate between 0 m and 700 m. The initial layout vector is $[0, 200, 400, 600]$. In order to verify the results, all possible layouts with a turbine at $x = 0$ and $x = 700$ and two other, freely spaced turbines were evaluated. This resulted in Figure 4-3 where it can be seen that the optimisation algorithm could optimise the initial layout. The result is not totally the same as the numerical solution. This may be due to the characteristics of the genetic algorithm as will be discussed in the next sections. Another reason may be the fact that there is not such a big difference in objective function values. It is possible that this makes it more difficult for the algorithm to find the optimal results.



(a) The two resulting optimised layouts.

(b) Analytical solution for every position of the middle turbines.

Figure 4-3: Optimisation with genetic algorithm and analytical solution of a wind farm consisting of four wind turbines.

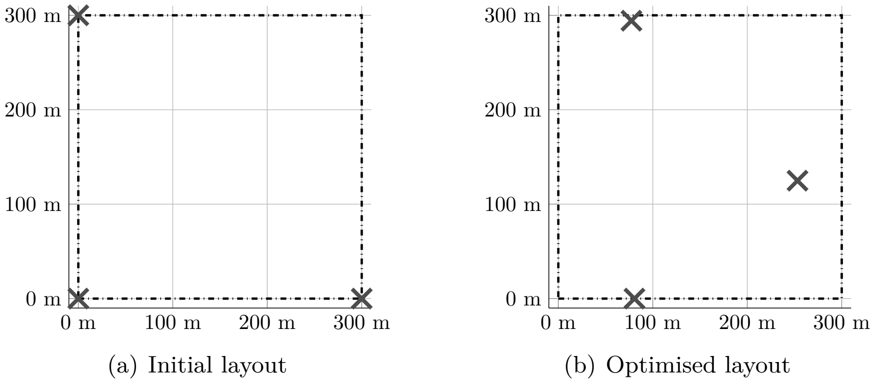
Another test that can be performed in the one-dimensional case is a test to see whether the boundary conditions get respected or not. In the previous examples, the maximum and minimum boundary conditions were respected, but these are (in the MATLAB GA toolbox) linear boundary conditions. The minimum distance constraint between two turbines is an example of a non-linear boundary condition. It could already be seen in the previous examples that this condition gets respected, but what happens when the design space gets severely limited? The following test was performed: optimisation of a wind farm of 3 wind turbines, placed on a line and a constant wind speed of 8 m/s. A (linear) boundary condition which states that turbines must be positioned between 0 m and 800 m and the non-linear boundary condition that the turbines must be at least two rotor diameters from each other. An additional constraint was added which states that the second wind turbine can not have an x-coordinate less than 600 m which hence creates a severely limited design space. The result of this optimisation was the following layout: $[0, 600, 800]$ which shows that all types of boundary conditions, linear or non-linear, indeed get respected.

4-4-2 Two-dimensional optimisation

Once it was proven that the algorithm performed well in one dimension, a second dimension was implemented. In order to do so, all (x, y) coordinates needed to be encoded in one string which was optimised. In the objective function, this string needs to be decoded again in order to produce a matrix with coordinates of the wind turbines. The analysis was done on a 3 by 3 wind farm (9 wind turbines) since the optimisation of the full Horns Rev wind farm was too computationally expensive to perform during the implementation phase. The extension into

2D also means that no numerical solutions can be computed. The previous case with 4 wind turbines in a one-dimensional wind farm resulted in two varying positions. In this case there are 9 wind turbines with each an x and y coordinate which results in 18 variables to optimise. It is not practically feasible to do this in a numerical way. The validity of the results can however still be checked by implementing some test cases and seeing how the optimisation algorithm responds to them.

The first step is to check whether a wind farm of three wind turbines in an L-shape (see Figure 4-4(a)) gets optimised so that the rightmost turbine will be 'removed' from the wake of the turbine in front, considering wind from the left with a wind speed of 8 m/s. The result of the optimisation can be seen in Figure 4-4(b). The rightmost turbine moved in between the wakes of the turbines in front and experiences (almost) no wake effects. The optimised production in this case is 2069.6 MW which is 99.98 % of three times the power produce by a 2 MW wind turbine at 8 m/s. There are positions for the rightmost turbine where 100 % could be obtained. The algorithm stopped at 99.98 % since the stopping criterion for the algorithm (change in difference of the objective function) did not change for a few generations. A lot of possible solutions where no wake effects are present exist in the solution space so it is no surprise that the front turbines also moved slightly and did not stay at $x = 0$.

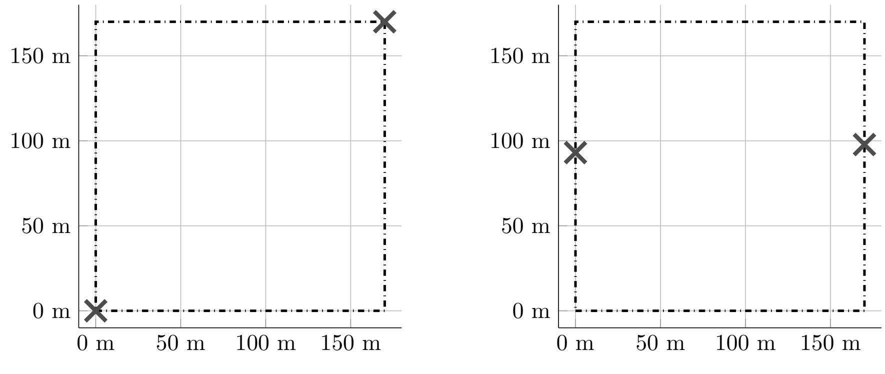


(a) Initial layout

(b) Optimised layout

Figure 4-4: First test of the 2D optimisation algorithm for an L-shaped three wind turbine wind farm with a wind of 8 m/s from the left (270°).

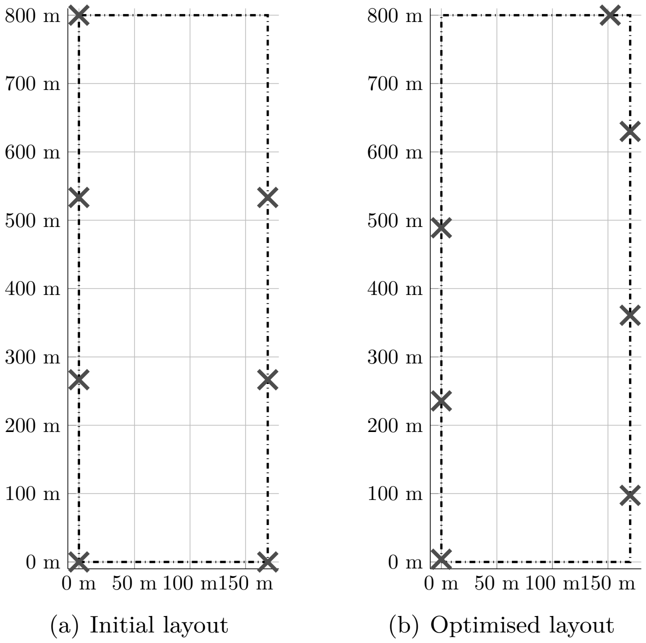
A next check that can be performed is the optimisation of a farm of two wind turbines which are placed behind each other in the direction of the wind speed. The same wind speed of 8 m/s is used, but now the wind direction is 225°. The constraints of the wind farm are set at two times the rotor diameter + 10 m. The initial layout can be seen in Figure 4-5(a) while the optimised layout can be seen in Figure 4-5(b). As can be seen in these figures, the optimisation made sure that the turbines are not in each others' wake any more. Again, a lot of layouts exist where this is the case.



(a) Initial layout (b) Optimised layout

Figure 4-5: Second test of the 2D optimisation algorithm for a two wind turbine wind farm placed in a diagonal line. Wind conditions are $8\,\mathrm{m/s}$ from the bottom left ($225°$).

Finally, a wind farm consisting of 7 wind turbines is tested. The original layout consists of two rows of turbines, a row of four and a row of thee wind turbines, as can be seen in Figure 4-6(a). The expected result would be the same as in the first case: the second row of turbines shifts up so that they are out of the wake of the first row. The result of the optimisation process can be seen in Figure 4-6(b). The obtained result is not really as expected, but the power production of the optimised layout is 4815 MW which is 99.7 % of seven times the power produced by one wind turbine at 8 m/s.



(a) Initial layout (b) Optimised layout

Figure 4-6: Third test of the 2D optimisation algorithm for a wind farm consisting of seven wind turbines. Wind conditions are $8\,\mathrm{m/s}$ from the left ($270°$).

4-4-3 Varying wind conditions

Up to now, the algorithm has been using the wind farm model directly for one wind speed and one wind direction. As stated before, the objective of the optimisation algorithm is to use the AEP in the objective function, so an additional layer between the optimisation algorithm and the wind farm model is implemented. This layer takes the current layout under investigation by the optimisation algorithm and the wind resource as inputs and executes a lot of wind farm model calculations. The result is the AEP which is reported back to the algorithm so it can decide on the next set of layouts.

4-4-4 Analysis of optimisation options

A lot of options exist in the optimisation toolbox of MATLAB. The MathWorks site states that the best combination of options is case dependent. No literature could be found with an overview of the best combination of these options, so a comparison of the most important options was performed.

- **Optimisation algorithm:** 'auglag' vs 'penalty'
- **Creation:** 'gacreationnonlinearfeasible' vs 'gacreationlinearfeasible'
- **Mutation:** 'mutationadaptfeasible' vs 'mutationuniform' vs 'mutationgaussian'
- **Crossover:** 'crossoverheuristic' vs 'crossoverintermediate' vs 'crossovertwopoint' vs 'crossoversinglepoint' vs 'crossoverscattered'
- **Selection:** 'selectionstochunif' vs 'selectionroulette' vs 'selectiontournament' vs 'selectionremainder'

During the analysis the sectorwise AEP method of the section above was used with a wind sector width of 15° (so 24 wind sectors) and a wind speed bin width of 3.125 m/s (so 8 wind speeds). The layout used is a three by three wind farm with boundary conditions such that no turbine can have an x- or y-coordinate larger than 600 m. The combination of optimised AEP and the time needed to obtain this value were used to find the most appropriate combination of optimisation options.

Optimisation algorithm In the first two runs the two options for the optimisation algorithm were compared. The decision was easily made to use the penalty in all of the following analyses. It gave much better results, but most importantly, it really uses the procedure of the GA whereas 'auglag' uses an alternative approach that uses the creation phase of a normal GA but then uses a special non-linear optimisation which does not involve the mutation, crossover or selection steps of the algorithm. It also does not show any output except the final results.

Creation The choice for the creation options was fairly straightforward since we are dealing with multiple non-linear boundary conditions (the minimum distance between the turbines). All other creation options resulted in either less performing optimised layouts, or did not respect the boundary conditions

Mutation Matlab proposes to use 'mutationadaptfeasible' since according to them this would perform the best with non-linear boundary conditions. This advice is contradicting the results that were found which indicate that 'mutationuniform' performs a lot better. It gives better optimised layouts in less time. 'mutationgaussian' was also tested, but this one did not respect any of the boundary conditions (linear or non-linear).

Crossover Most of the crossover options resulted in a suboptimal optimised layout. Some of the crossover options resulted in an algorithm which could not converge to a certain solution and kept jumping around. 'crossoverscattered' was chosen as the option for crossover.

Selection 'selectionremainder' gave the best results in the fastest time and was thus selected for this option.

The selection of these options did not happen as linearly as written down above. It was more an iterative process in order to come up with the best combination. The choice for these options was also validated using a smaller grid. The resulting combination of options (mutationuniform, gacreationnonlinearfeasible, crossoverscattered, selectionremainder, penalty) were used in all following analyses.

4-4-5 Implementing PCE

When the best options were known, it was time to implement the PCE algorithm in the optimisation algorithm. This was not such a difficult task thanks to the clear division in 'layers' as stated above. The only thing that needed to be changed was the link between the optimisation algorithm and the wind farm model which calculates the AEP. It was chosen to compute the polynomial basis and the collocation points before the optimisation algorithm starts so that no time is wasted during the optimisation process with needless (and repetitive) computations.

4-5 Optimisation with PCE

To analyse how efficient the optimisation algorithm works and to investigate how PCE performs during an optimisation process, a lot of optimisation of the three by three wind farm were performed. Each optimisation used another technique (sectorwise, Weibull per wind direction sector, PCE and sectorwise PCE) and different numbers of collocation points for wind speeds and wind directions. The computations were performed on the INSY cluster at the EWI faculty of the TU Delft [47, 48]. The optimisation algorithm was initialised by a small wind farm in a square layout with x and y coordinates $[0, 200, 400]\,m$. The boundary conditions were set to be $[0, 600]\,m$. All methods were programmed to perform the optimisations with the same set of number of collocation points.

$$
\begin{aligned}
N_{WS} &= [5, 6, 10, 15, 20] \\
N_{WD} &= [9, 12, 18, 24, 36, 72]
\end{aligned}
$$

All combinations of number of collocation points were made, which resulted in 30 different combinations of wind speed and wind direction collocation points for each method.

Figure 4-7 shows the number of model evaluations used to compute the AEP versus the time it took for the algorithm to complete the optimisation process. Note that the computations were performed on a computational cluster while using 16 cores. As the optimisation algorithm is making use of this parallel computing capabilities, a lot of data needs to be copied between the main script and the workers. The computational efficiency thus highly depends on how efficiently the different methods were programmed. Unfortunately, it was not so easy to implement the sectorwise PCE in an efficient way. This is visible in the results as it can be seen this method is a lot slower than the other three methods. PCE is the fastest method since the only data that is required to compute the AEP is the polynomial basis and a set of collocation points. The method which uses discrete sectors for wind speed and wind direction and the method which uses the Weibull distribution per wind direction sector performed on an equal level.

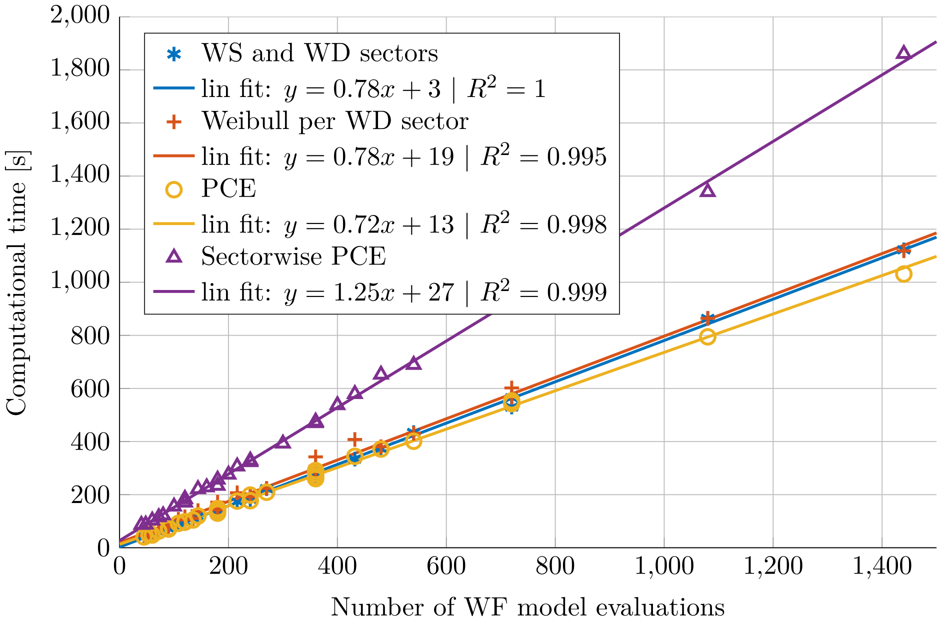


Figure 4-7: Number of model evaluations versus total computational time of the optimisation algorithm for three different calculation methods for the AEP.

Figure 4-8 shows the result of the optimisation process as evaluated by each (surrogate) model. Here it seems that sectorwise PCE is a very bad method to use in WFLOP. However, it is important to note that the optimised power values used in this plot come directly from the surrogate model itself. This means that values from different methods can not directly be compared with each other.

In Figure 4-9, the optimised layouts were reevaluated with one function. It is the same function as used in the previous chapter on PCE to compute the exact solution. A discrete sectorwise evaluation on a 1 m/s and 1° grid was performed for each optimised layout. In the resulting figure, it can be seen that the sectorwise PCE underperforms compared to the other methods. It is slower and produces less optimal results. The discrete sectors method and Weibull per wind direction sector perform on an equal level. Using PCE results in layouts which are slightly more optimised, but only when good parameters for the number of collocation points are chosen. The most optimal result was achieved by PCE by using respectively 10 and 24 wind speed and wind direction collocation points. The dots of PCE that are way below the others are all resulting from using 9 wind direction collocation points.

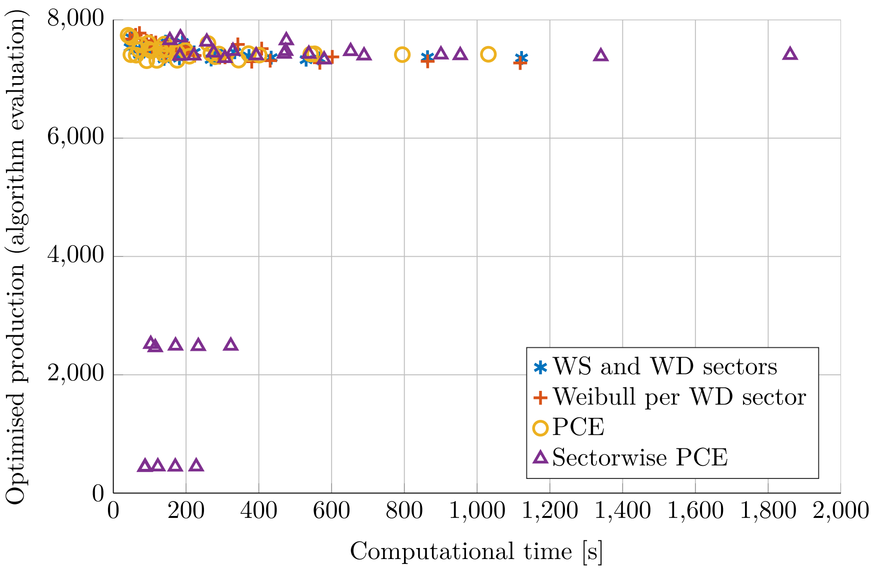


Figure 4-8: Optimised production of the three by three wind farm layout according to the respective surrogate model.

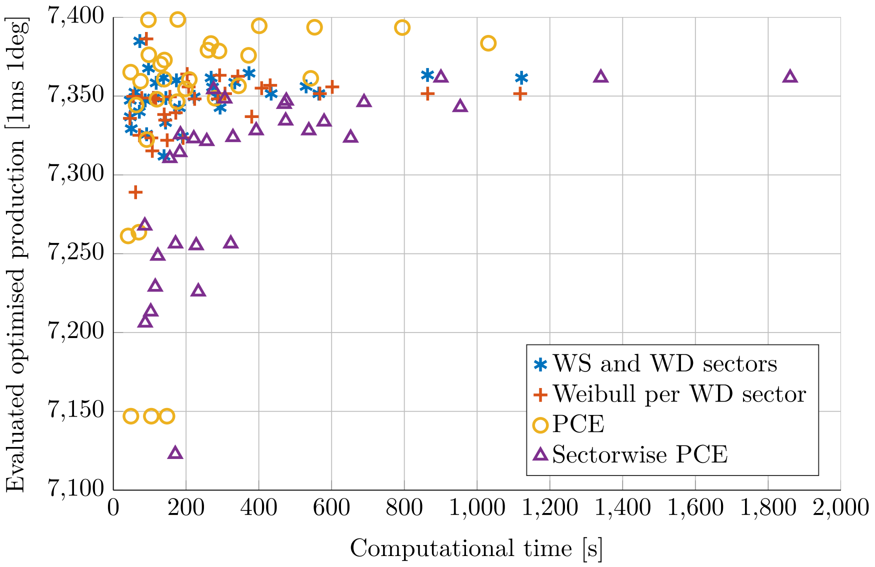


Figure 4-9: Optimised production of the three by three wind farm layout according to a sectorwise evaluation on a $1\,\mathrm{m/s}$ and 1° grid.

Overall it seems like PCE has no real added value. The method was promising to speed up the optimisation process by providing a faster way to compute the AEP of a wind farm. No real speed increase can be seen and when too few collocation points are used, the resulting layouts are subperforming.

There was also an attempt to optimise the whole Horns Rev wind farm. This however failed, even with the conventional discrete sectors approach, and it is unclear why this happened. It is possible that the settings that are required to run such big optimisations are different than for the small 3x3 wind farm. Even with the computational cluster, one run for an optimisation took more than a day. It was hence not possible to perform the same search for the most optimal options like in Section 4-4-4.

4-6 Conclusion

In this chapter, the genetic optimisation algorithm was introduced and its options were analysed. The following combination of options are determined to work optimally for the optimisation of a small 3x3 wind farm layout: mutationuniform, gacreationnonlinearfeasible, crossoverscattered, selectionremainder and penalty algorithm.

A larger layout was unable to use the optimisation algorithm to obtain an optimised wind farm layout. This may be due to the selected options, but other (unknown) factors may also apply.

The optimisation of a three by three wind farm was performed and it can be concluded that polynomial chaos expansion in wind farm layout optimisation has a limited functionality. While the optimised layouts are slightly better performing than the layouts from the conventional methods, there is no noticeable decrease in the computational time that is required to optimise the wind farm layouts.

The sectorwise PCE does not work as well in optimisation problems as it does for constructing accurate surrogate models.

Chapter 5

Conclusion and recommendations

5-1 Conclusion

The aim of this project was to investigate whether polynomial chaos expansion (PCE) can be used in the wind energy field or not.

In Chapter 3, the PCE technique was introduced. From the analysis it can be concluded that the method works best when use is made of data-driven moments. The polynomial basis construction methods perform equally well, but the Gram-Schmidt process gives more insight and offers an extension to two dimensions which is not that difficult to implement. Special attention is needed when working with a large polynomial basis since this means that a lot of raw moments are needed. These can have very high values which results in computational errors when computing the polynomials. There errors started being noticable when the order was higher than 10.

PCE performs well for the one-dimensional case of wind speed, but not so well for wind direction. It is difficult to fit a polynomial on the output of a wind farm (with wake effects) and the resulting surrogate models were not an accurate fit. Therefore it was decided to try a sectorwise PCE approach. This resulted in very good surrogate models for the wind direction.

Going from one dimension to two dimensions proved to be not easy. The PCE over the whole domain struggled with the wind direction. This was also the case for the sectorwise PCE when the wind direction sectors were too large. In general it proves to be difficult to obtain an accurate estimation of the standard deviation using the surrogate model.

It is recommended to use a polynomial basis of the 5th order and to use at least 6 linearly spaced collocation points for the wind speed and 30 linearly spaced collocation points for the wind direction when using the PCE method.

In Chapter 4, PCE was used to optimise a small wind farm. It can be concluded that PCE has limited functionality in wind farm layout optimisation. The optimised layouts perform slightly better, but no decrease in required computational time could be obtained.

The new approach using the sectorwise PCE performs worse than the conventional methods during the optimisation process.

5-2 Recommendations for future work

In this work a simple method to select the collocation points and to fit the polynomial basis to the expensive model was used. It is recommended to look in to more advanced and efficient methods like sparse grids and the stochastic collocation method.

It is recommended to also apply PCE as an uncertainty quantification method. It could for example be used to investigate the uncertainty in annual energy production (AEP) due to uncertainties in the wind resource.

Bibliography

[1] M. Samorani, "The wind farm layout optimization problem," in *Handbook of wind power systems*, pp. 21–38, Springer, 2013.

[2] S. Turner, D. Romero, P. Zhang, C. Amon, and T. Chan, "A new mathematical programming approach to optimize wind farm layouts," *Renewable Energy*, vol. 63, pp. 674–680, 2014.

[3] S. Chowdhury, J. Zhang, A. Messac, and L. Castillo, "Unrestricted wind farm layout optimization (uwflo): Investigating key factors influencing the maximum power generation," *Renewable Energy*, vol. 38, no. 1, pp. 16–30, 2012.

[4] G. Mosetti, C. Poloni, and B. Diviacco, "Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 51, no. 1, pp. 105–116, 1994.

[5] J. F. Herbert-Acero, O. Probst, P.-E. Réthoré, G. C. Larsen, and K. K. Castillo-Villar, "A review of methodological approaches for the design and optimization of wind farms," *Energies*, vol. 7, no. 11, pp. 6930–7016, 2014.

[6] P.-Y. Yin, T.-H. Wu, and P.-Y. Hsu, "Risk management of wind farm micro-siting using an enhanced genetic algorithm with simulation optimization," *Renewable Energy*, vol. 107, pp. 508–521, 2017.

[7] C. Audet, A. J. Booker, J. Dennis Jr, P. D. Frank, and D. W. Moore, "A surrogate-model-based method for constrained optimization," *AIAA paper*, vol. 4891, 2000.

[8] D. Xiu, *Numerical methods for stochastic computations: a spectral method approach*. Princeton University Press, 2010.

[9] M. N. Jimenez, J. Witteveen, and J. Blom, "Polynomial chaos expansion for general multivariate distributions with correlated variables," 2014.

[10] X. Qin, S. J. Zhang, and D. X. Yan, "A new circular distribution and its application to wind data," *Journal of Mathematics Research*, vol. 2, no. 3, p. 12, 2010.

[11] M. Koivisto, J. Ekström, I. Mellin, J. Millar, and M. Lehtonen, "Statistical wind direction modeling for the analysis of large scale wind power generation," *Wind Energy*, vol. 20, no. 4, pp. 677–694, 2017.

[12] Online. Accessed on 2017-11-15 `https://www.vestas.com/en/products/turbines/v90%203_0_mw#!power-curve`.

[13] N. L. Johnson, S. Kotz, and N. Balakrishnan, "Continuous univariate distributions," *Journal of the Royal Statistical Society-Series A Statistics in Society*, vol. 159, no. 2, p. 343, 1996.

[14] M. A. Baseer, J. P. Meyer, S. Rehman, and M. M. Alam, "Wind power characteristics of seven data collection sites in Jubail, Saudi Arabia using Weibull parameters," *Renewable Energy*, vol. 102, pp. 35–49, 2017.

[15] B. Hrafnkelsson, G. V. Oddsson, and R. Unnthorsson, "A Method for Estimating Annual Energy Production Using Monte Carlo Wind Speed Simulation," *Energies*, vol. 9, no. 4, p. 286, 2016.

[16] J. Feng and W. Z. Shen, "Modelling wind for wind farm layout optimization using joint distribution of wind speed and wind direction," *Energies*, vol. 8, no. 4, pp. 3075–3092, 2015.

[17] S. Grady, M. Hussaini, and M. M. Abdullah, "Placement of wind turbines using genetic algorithms," *Renewable energy*, vol. 30, no. 2, pp. 259–270, 2005.

[18] Online. Accessed on 2018-06-07 `http://www.noordzeewind.nl/`.

[19] H. Korterink and P. Eecen, "Meteorological measurements OWEZ," tech. rep., ECN, 2008.

[20] I. Wijnant, A. . Stepek, M. Savenije, and H. van den Brink, *User manual of the (KNMI North Sea Wind) KNW-atlas* , 2016.

[21] B. Hu, "Design of a simple wake model for the wind farm layout optimization considering the wake meandering," Master's thesis, TU Delft, 2016.

[22] M. Bastankhah and F. Porté-Agel, "A new analytical model for wind-turbine wakes," *Renewable Energy*, vol. 70, pp. 116–123, oct 2014.

[23] A. Niayifar and F. Porté-Agel, "A new analytical model for wind farm power prediction," *Journal of Physics: Conference Series*, vol. 625, p. 012039, jun 2015.

[24] P. A. Jiménez, J. Navarro, A. M. Palomares, and J. Dudhia, "Mesoscale modeling of offshore wind turbine wakes at the wind farm resolving scale: a composite-based analysis with the Weather Research and Forecasting model over Horns Rev," *Wind Energy*, vol. 18, no. 3, pp. 559–566, 2015.

[25] N. Wiener, "The homogeneous chaos," *American Journal of Mathematics*, vol. 60, no. 4, pp. 897–936, 1938.

[26] D. Xiu and G. E. Karniadakis, "Modeling uncertainty in flow simulations via generalized polynomial chaos," *Journal of computational physics*, vol. 187, no. 1, pp. 137–167, 2003.

[27] S. Oladyshkin and W. Nowak, "Data-driven uncertainty quantification using the arbitrary polynomial chaos expansion," *Reliability Engineering & System Safety*, vol. 106, pp. 179–190, 2012.

[28] J. A. Witteveen and H. Bijl, "Modeling arbitrary uncertainties using Gram-Schmidt polynomial chaos," in *44th AIAA aerospace sciences meeting and exhibit*, p. 896, 2006.

[29] A. Winkelbauer, "Moments and absolute moments of the normal distribution,"

[30] D. Wackerly, W. Mendenhall, and R. L. Scheaffer, *Mathematical Statistics with Applications*. DUXBURY, 2007.

[31] R. Ahlfeld, B. Belkouchi, and F. Montomoli, "Samba: sparse approximation of moment-based arbitrary polynomial chaos," *Journal of Computational Physics*, vol. 320, pp. 1–16, 2016.

[32] F. Wang, F. Xiong, H. Jiang, and J. Song, "An enhanced data-driven polynomial chaos method for uncertainty propagation," *Engineering Optimization*, pp. 1–20, 2017.

[33] D. C. Lay, *Linear Algebra and its applications, Third Edition*. Pearson Education, 2006.

[34] P. M. Congedo, G. Geraci, and G. Iaccarino, "On the use of high-order statistics in robust design optimization," in *6th European Conference on Computational Fluid Dynamics (ECFD VI)*, 2014.

[35] M. Eldred and J. Burkardt, "Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification," *AIAA paper*, vol. 976, no. 2009, pp. 1–20, 2009.

[36] B. Sudret, "Global sensitivity analysis using polynomial chaos expansions," *Reliability Engineering & System Safety*, vol. 93, no. 7, pp. 964–979, 2008.

[37] X. Huan, C. Safta, K. Sargsyan, Z. P. Vane, G. Lacaze, J. C. Oefelein, and H. N. Najm, "Compressive sensing with cross-validation and stop-sampling for sparse polynomial chaos expansions," *arXiv preprint arXiv:1707.09334*, 2017.

[38] G. Yang, F. Liu, L. Li, H. Wang, C. Zhao, and Z. Wang, "2d orthogonal polynomials for concurrent dual-band digital predistortion," in *Microwave Symposium Digest (IMS), 2013 IEEE MTT-S International*, pp. 1–3, IEEE, 2013.

[39] C. N. Elkinton, J. F. Manwell, and J. G. McGowan, "Algorithms for offshore wind farm layout optimization," *Wind Engineering*, vol. 32, no. 1, pp. 67–84, 2008.

[40] J. E. G. Martínez, "Layout optimisation of offshore wind farms with realistic constraints and options," Master's thesis, TU Delft, jun 2014.

[41] R. Shakoor, M. Y. Hassan, A. Raheem, and Y.-K. Wu, "Wake effect modeling: a review of wind farm layout optimization using Jensens model," *Renewable and Sustainable Energy Reviews*, vol. 58, pp. 1048–1059, 2016.

[42] J. Cagan, K. Shimada, and S. Yin, "A survey of computational approaches to three-dimensional layout problems," *Computer-Aided Design*, vol. 34, no. 8, pp. 597–611, 2002.

[43] A. Tesauro, P.-E. Réthoré, and G. C. Larsen, "State of the art of wind farm optimization," in *EWEA 2012-European Wind Energy Conference & Exhibition*, 2012.

[44] F. Wang, D. Liu, and L. Zeng, "Modeling and simulation of optimal wind turbine configurations in wind farms," in *World Non-Grid-Connected Wind Power and Energy Conference, 2009. WNWEC 2009*, pp. 1–5, IEEE, 2009.

[45] J. Wang, X. Li, and X. Zhang, "Genetic optimal micrositing of wind farms by equilateral-triangle mesh," in *Wind Turbines*, InTech, 2011.

[46] C. Szafron, "Offshore windfarm layout optimization," in *Environment and Electrical Engineering (EEEIC), 2010 9th International Conference on*, pp. 542–545, IEEE, 2010.

[47] Online. Accessed on 2018-06-08 `https://www.tudelft.nl/ewi/over-de-faculteit/afdelingen/intelligent-systems/`.

[48] Online. Accessed on 2018-06-08 `http://insy-login.hpc.tudelft.nl/`.