Computational Science and Engineering
(International Master's Program)

Technische Universität München

Master's Thesis

Uncertainty Quantification and Global Sensitivity Analysis of a Lagrangian Acoustical-meteorological Coupled Simulation

Hessel Juliust

Computational Science and Engineering
(International Master's Program)

Technische Universität München

Master's Thesis

Uncertainty Quantification and Global Sensitivity Analysis of a Lagrangian Acoustical-meteorological Coupled Simulation

Author:	Hessel Juliust
Supervisor:	Univ.-Prof. Dr. Hans-Joachim Bungartz
Advisors (DLR):	Dr. Katharina Elsen
	Dr. Arthur Schady
Advisor (TUM):	Dr. Tobias Neckel
Submission Date:	October 29th, 2022

I hereby declare that this thesis is entirely the result of my own work except where otherwise indicated. I have only used the resources given in the list of references.

October 29th, 2022 Hessel Juliust

Acknowledgments

This thesis is part of my studies at the Technical University of Munich (TUM). The research reported in this thesis was carried out at the Institute of Atmospheric Physics at the German Aerospace Center (IPA-DLR) Oberpfaffenhofen.

In this opportunity, I would like to to thank Dr. Katharina Elsen and Dr. Arthur Schady as my advisors from DLR who have given me this opportunity to do research on this particular topic and for their willingness to provide assistance whenever I encounter difficulties on every scale. I am also thankful to Dr. Tobias Neckel as my advisor from TUM for the supports and discussions especially regarding to the uncertainty quantification.

Additionally, I would like to express my gratitude to Claussen-Simon-Stiftung for the supports for the first 4 semesters by Master Plus Scholarship. Last but not least, I am very grateful to my family and friends especially my mom for their supports throughout my studies.

"I think that when we know that we actually do live in uncertainty, then we ought to admit it; it is of great value to realize that we do not know the answers to different questions. This attitude of mind – this attitude of uncertainty – is vital to the scientist, and it is this attitude of mind which the student must first acquire"

-Richard P. Feynman

Abstract

The sound propagation model supports acoustic noise pollution studies. Since most of the noise sources are located outdoors, an acoustical-meteorological coupled simulation is required. However, meteorological parameters are always uncertain and can affect model output. In order to ensure that proper conclusions will be obtained from the simulation, an uncertainty analysis should be carried out. However, such an analysis is still rarely applied in the research field. Therefore, this thesis addresses the question of which approach is suitable for analyzing the uncertainty and sensitivity of a particular acoustic-meteorological coupled model. In this study, the uncertainty of the Lagrangian-based acoustical-meteorological coupled model is analyzed.

This study focuses on the methods of Monte Carlo and the generalized polynomial chaos expansion, both of which make it possible to obtain the first statistical moment and the Sobol' sensitivity indices for uncertainty and sensitivity analysis, respectively. These methods are applied to acoustical-meteorological simulations with several hill topographies and also with several variations of meteorological parameters. The results are compared and it is observed that both methods can be applied. The generalized polynomial chaos expansion method produces similar results to the Monte Carlo method but with lower computational costs. The results also show the same pattern as other studies on the meteorological effects on sound propagation for specific test cases.

Contents

1. Introduction

The noise caused by wind turbine is a major concern since the number of onshore wind power plants increases and small turbines are substituted by higher and more powerful ones, therefore reliable noise predictions are needed [15]. Sound waves are generated by the interaction between the wind and the rotating blades (aerodynamic noise) or by other sources at the turbine and propagate through the atmospheric boundary layer. The outdoor propagation of sound is strongly determined by characteristics of the atmosphere and the underlying ground [7].

Various models have been developed to simulate the sound propagation which also included the effects of the meteorological condition (or usually refer as acoustical-meteorological coupled simulation), for example AKUMET (sound ray particle based) [17], AKU3D (sound wave PDE, FDTD solver based) [2], or other commercial industrial software. Meteorological conditions are rarely constant, thus these models are subjects to uncertainties even though the results of these models are merely values of sound pressure level (SPL), without information on how uncertain these values are. These uncertainties that propagate through the model might affect the output and could affect the reliability of the simulations thus might also cause a misleading conclusions.

How meteorological conditions affects the sound propagation have previously been studied many times, such as by Heath [14] which explained the effect of wind and temperature and by Heimann [15] which studied the case of wind turbine on the top of a hill using 9 test cases which included the variation of geostrophic wind speed, horizontal large-scale pressure gradient, background temperature gradient, dew point difference and roughness length. These mentioned scenario are usually specific examples and can only be transferred to limited extent according to the local meteorological conditions. However, the uncertainties of sound pressure level that is caused by multiple meteorological condition samples with a certain distribution of the uncertain inputs have never been studied before, particularly the one that is simulated by the Lagrangian sound particle model. If the uncertainty can be quantified, then an additional action can also be done in order to reduce the sound pollution.

More to that, uncertainties that are propagated through the model may originate from many different parameters. Therefore, one parameter affects uncertainty to the model differently to other parameter and this property can described by it's sensitivity metric. Therefore, other than the analysis of the uncertainty, the sensitivity of the parameters should also be analyzed.

In this research, the model to be analyzed is AKUMET, since it has been used for many researches and the meteorological fields on the whole domain can be generated by certain

parameters only. Inspired by this fact, there are questions that will be discussed in this thesis, namely:

1. How to quantify the uncertainty of the sound pressure level, which is the output of AKUMET that is affected by the uncertain meteorological or topographical parameters?
2. How to quantify the sensitivity of uncertain meteorological or topographical parameters?
3. Which uncertain meteorological or topographical parameters that cause a lot of uncertainty in the output sound pressure level?
4. How do the uncertain input parameters affect the behaviour of the sound pressure level output?

The first and second research questions are answered by comparing different methods of uncertainty quantification and sensitivity analysis. The methods under consideration are Monte Carlo method and generalized polynomial chaos expansion (gPCE) method with the pseudo-spectral approach. After the first two questions are answered then the third question will be answered by doing the sensitivity analysis with the 2-dimensional setup and with the most extreme variability of parameters that can exist in the real world. The goal is, since the 2-dimensional and 3-dimensional use the same model, the sensitivity should be similar. For the fourth question, various cases of acoustical-meteorological simulation are studied and comparisons with previous research about effects of meteorological variability on sound propagation are done. Since most human activities are carried out at the ground level, the research for this thesis mostly focuses on the analysis of the ground level, especially for the 3-dimensional setup rather than the whole domain of sound propagation, even though the source of the sound is located above the ground.

The thesis is structured as follows. After introduction of the thesis from this first section, section 2 gives a theoretical overview about numerical quadrature and orthogonal polynomial for the basis of the generalized polynomial chaos expansion method with pseudo-spectral approach and ray acoustic in general for the Lagrangian based sound propagation model. The discussion in section 3, focuses on the methods of uncertainty quantification and sensitivity analysis. In section 4, an explanation of AKUMET as the sound propagation model is given, namely the governing equation and it's parameters that will be varied for the uncertainty and sensitivity analysis. Also on the fourth section, the uncertainty quantification programming libraries and the parallel computing method to assist the research are described. In the end of the fourth section, all scenario for uncertainty quantification and sensitivity analysis for this research are described. Finally, the results of this research and their relevance are discussed along with possible outlook for further research.

2. Theoretical Foundation

2.1. Numerical Quadrature and Sparse Grids

In this section quadrature rules that are used for approximating a definite integral of a function $f(x)$ will be discussed. These quadrature rules would be used for computing the coefficient of a chaos polynomial that is used to do the uncertainty quantification.

2.1.1. Gauss–Legendre Quadrature

A Gaussian quadrature rule with an n-point quadrature nodes is a quadrature rule that evaluates an integral as the sum of a finite number of terms [37]. The most common domain of integration for such problem is $[-1, 1]$, and the quadrature rule is written as follows:

$$\int_{-1}^{1} f(x)\,dx \approx \sum_{i=1}^{n} w_i f(x_i). \tag{2.1}$$

The integral is approximated by the weighted sum w_i of function values at specially selected nodes. The usual chosen quadrature nodes x_i are the roots of an orthogonal polynomial. Here, roots of Legendre Polynomials will be used (hence the name Gauss-Legendre Quadrature) since it is able to approximate an integral of polynomials of degree $2N+1$ degree relatively accurate by $N+1$ quadrature nodes. A compact expression for the Legendre polynomials is given by Rodrigues' formula as follows:

$$P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}\left(x^2-1\right)^n. \tag{2.2}$$

The weights w_i are obtained by solving the linear equation with the Legendre Polynomials, its roots and its integrals by 2.1 or by using [1] :

$$w_i = \frac{2}{\left(1-x_i^2\right)\left[P_n'\left(x_i\right)\right]^2}. \tag{2.3}$$

In order to approximate integral over $[a, b]$ instead $[-1, 1]$, change of interval can be done by this following way:

$$\int_a^b f(x)dx = \int_{-1}^{1} f\left(\frac{b-a}{2}\xi + \frac{a+b}{2}\right), \tag{2.4}$$

with $\frac{dx}{d\xi} = \frac{b-a}{2}$.

2.1.2. Clenshaw–Curtis and Fejér Quadrature

Clenshaw–Curtis quadrature is quadrature rules to integrate a function $f(x)$ by changing the variable $x = cos(\theta)$ [27]. Same as previous part, it is usually written for integration over the interval [-1,1] (which re-scaling can also be performed). This quadrature rule can be written as follows:

$$\int_{-1}^{1} f(x)dx = \int_{0}^{\pi} f(\cos\theta)\sin(\theta)d\theta. \tag{2.5}$$

Since cosine series for $f(cos\theta)$ is known therefore:

$$f(\cos\theta) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos(k\theta), \tag{2.6}$$

thus the integral approximation becomes:

$$\int_{0}^{\pi} f(\cos\theta)\sin(\theta)d\theta = a_0 + \sum_{k=1}^{\infty} \frac{2a_{2k}}{1-(2k)^2}, \tag{2.7}$$

with the coefficients as follows:

$$a_{2k} \approx \frac{2}{N}\left[\frac{f(1)+f(-1)}{2} + f(0)(-1)^k + \sum_{n=1}^{N/2-1}\{f(\cos[n\pi/N]) + f(-\cos[n\pi/N])\}\cos\left(\frac{nk\pi}{N/2}\right)\right]. \tag{2.8}$$

This is actually connected with the Chebyshev polynomials $T_k(x)$, since :

$$T_k(cos(\theta)) = cos(k\theta). \tag{2.9}$$

Hence, the cosine series above is really an approximation of $f(x)$ by Chebyshev polynomials as follows:

$$f(x) = \frac{a_0}{2}T_0(x) + \sum_{k=1}^{\infty} a_k T_k(x). \tag{2.10}$$

Therefore, what is done here is integrating approximate expansion of $f(x)$ in terms of Chebyshev polynomials. The points that are evaluated ($x_n = cos(\frac{n\pi}{N})$) are the extremas of the Chebyshev polynomial. The reason that Clenshaw–Curtis quadrature is also needed in the context of uncertainty quantification is that when dealing with quadrature approximation in order to get the coefficient for polynomial chaos expansion, the curse of dimensionality is often found particulary when increasing the dimension of the variables. By Clenshaw-Curtis, there will be a nested evaluation points which means in case of multidimensional quadrature, evaluation points coincide with other evaluation points and this reduces a lot of function evaluation when multidimensional quadrature is performed.

Fejér quadrature rules is very similar to Clenshaw–Curtis quadrature (also was proposed earlier which was in 1933). Fejér's quadrature rule is nearly identical to Clenshaw–Curtis with the only difference is that the endpoint $f(-1)$ and $f(1))$ are set to zero [38] and it means that Fejér quadrature used only the interior extrema of the Chebyshev polynomials.

2.2. Orthogonal Polynomials

In this section orthogonal polynomials that are used for approximating a random process of a probability distribution function will be discussed. These polynomials which are written here will be used for the uncertainty quantification method.

2.2.1. Legendre Polynomials

Legendre polynomials are a system of complete and orthogonal polynomials. Legendre polynomials are defined as an orthogonal system with respect to it's weight function $w(x) = 1$ over the interval $[-1, 1]$ [3]. $\mathbf{P}_n(x)$ is the Legendre polynomial with order n with the definition as follows:

$$\int_{-1}^{1} \mathbf{P}_m(x)\mathbf{P}_n(x)dx = 0, \quad \text{if } n \neq m. \tag{2.11}$$

Figure 2.1 shows the plot of Legendre polynomials until it's fifth order.

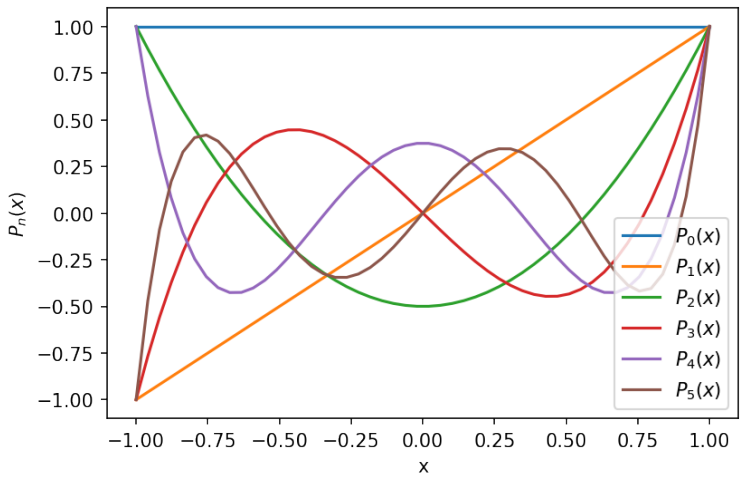

Figure 2.1.: Plot of Legendre Polynomials until fifth order.

2.2.2. Hermite Polynomials

Hermite Polynomials (The probabilist' one) are a system of also an orthogonal polynomials. $\mathbf{H}_n(x)$ is the Hermite polynomial with order n with the definition as follows [39]:

$$\mathbf{H}_n(x) = (-1)^n e^{\frac{x^2}{2}} \frac{d^n}{dx^n} e^{-\frac{x^2}{2}}. \tag{2.12}$$

Figure 2.2 shows the plot of Hermite polynomials until it's fifth order.

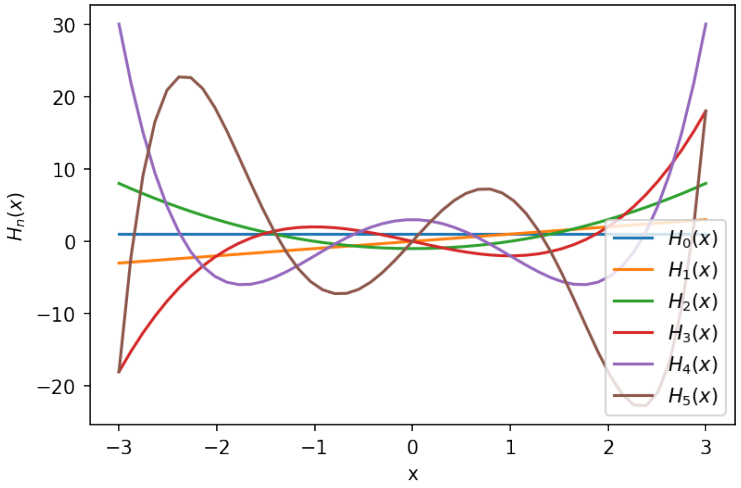

Figure 2.2.: Plot of Hermite Polynomials until fifth order.

2.3. Coupled Meteorological and Acoustical Simulations

The meteorological model that is coupled with the acoustical simulation is a mesoscale meteorological model, since it is a meteorological model which cover a limited area of 1 to 1000 km horizontal extension, thus topographic features such as mountains, valleys, coasts and varying land are the caused of distinct patterns of meteorological variables [17]. Meteorological mesoscale models are based on a set of physical equations which describe the conservation of momentum, mass, and internal energy. Since these equations are non-linear, a numerical solutions are usually required if a most accurate simulation is needed [28]. Since the effect of the topographic features are included, the sound acoustical simulation is then also affected by the topography.

2.3.1. Lagrangian Based Sound Propagation Model (Ray Acoustics)

There are many methods to simulate sound propagation which mostly are PDE-based model which solve the wave or Hemholtz equation [29]. However not all available models are suited to realistic meteorological conditions over complex topography, the problem is vary from models cannot consider in-homogeneous atmospheric condition until difficulty to simulate high frequency [17].

Ray Acoustics

Sound propagation can also be modeled as ray tracing instead of wave model. The important part of ray acoustic is the concept of wave front and ray path. A wavefront is a surface that move along at which a wave feature (usually peak) is being simultaneously received.

An illustration at fig 2.3 shows an example that if the time history of wave pressure has a single pronounced peak at x at time $\tau(x)$, the set of all points satisfying $t = \tau(x)$ describes the corresponding wavefront at time t. Wavefronts propagate with speed c if it is observed

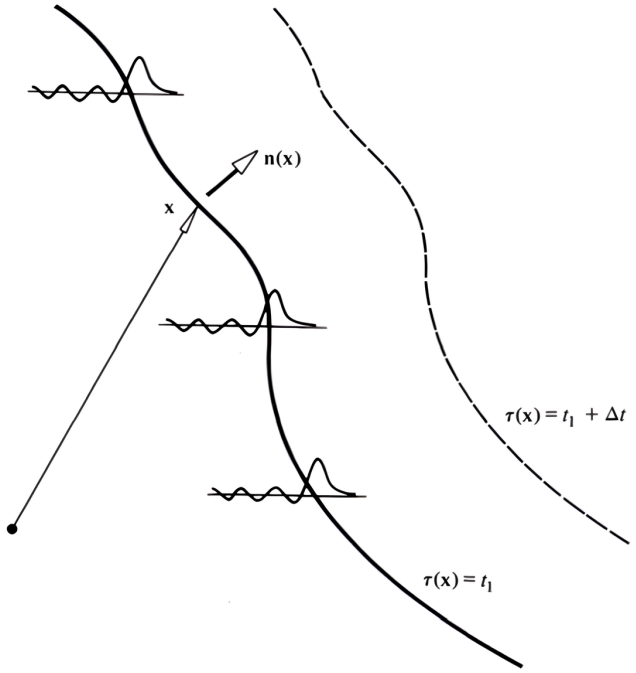

Figure 2.3.: Illustration of the concept of a wavefront in order to show points which the wavefront simultaneously passes receive the same waveform feature at the same time [29].

in a coordinate system in which the propagation medium appears at rest. [29]

If a medium is moving with velocity **n**, the wave velocity c**n** seen by observer moving with the moving medium becomes $v + c$**n** in a coordinate system at rest. Here **n** is the unit vector normal to the wavefront which coincides with the direction of propagation if the coordinate system is moving with the local ambient moving medium velocity **v**. The direction of propagation observed by a stationary observer is not the same as that of **n**. A ray path $\mathbf{x}_p$ can be defined as a point at a wave front and the relation of it's derivative with the vector normal to the wavefront **n**, the velocity of the moving medium **v** and the velocity of the propagated wave c is written as follows:

$$\frac{d\boldsymbol{x}_p(t)}{dt} = \boldsymbol{v} + c\boldsymbol{n}. \tag{2.13}$$

An illustration at fig 2.4 shows an example that if the point $\boldsymbol{X}_P(t)$ moves with velocity $c\boldsymbol{n} + \boldsymbol{v}$ such that it is always on wavefront $\tau(x) = t$. Here, the properties of n can also be

written as follows:

$$\frac{d\boldsymbol{n}}{dt} = -\vec{\nabla}c - \boldsymbol{n}\boldsymbol{v}. \tag{2.14}$$

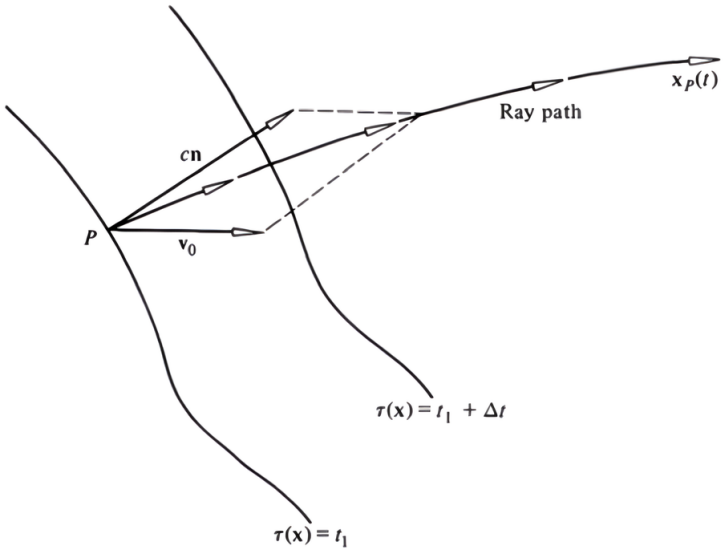

Figure 2.4.: Illustration concept of a ray path [29].

3. Uncertainty Quantification and Sensitivity Analysis

3.1. Uncertainty Quantification

Uncertainty quantification (UQ) is the quantitative characterization of uncertainties in both computational and real world applications. The goal is to determine how certain outcomes are if some aspects or inputs of the system are not exactly known. This research is focused on forward propagation of uncertainties which means to quantify uncertainties in system output(s) that is propagated from input(s) which has uncertainty. Analysis of the influence on the outputs from the parametric variability listed in the sources of uncertainty is done.

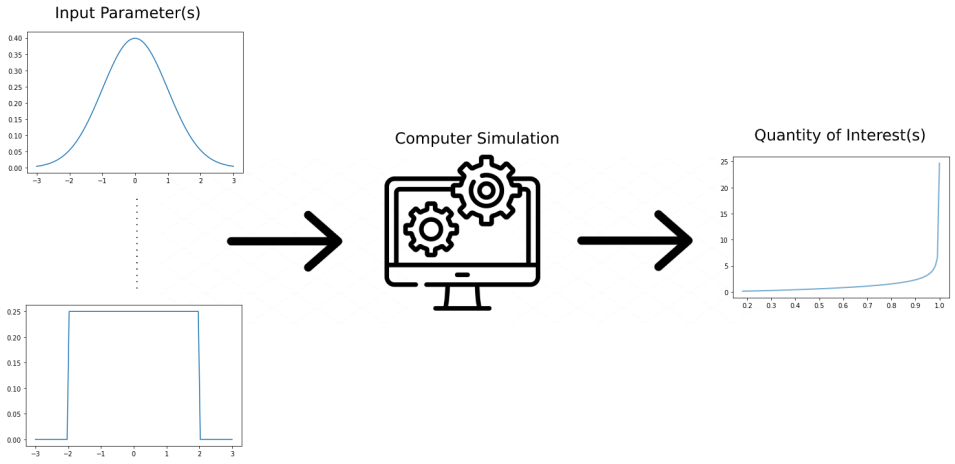

Figure 3.1.: Illustration of forward propagation uncertainty quantification scheme.

As an equation, problem at figure 3.1 can be expressed as:

$$Y = f(\mathbf{X}). \tag{3.1}$$

with Y is our quantity of interest (QoI) and $\mathbf{X}$ is the parameter(s) with uncertainty. The statistical moments of Y are then analyzed [33]. This approach requires the uncertainty

of input parameter **X** to be known, because **X** is treated as a random variable with a certain distribution. The probability density function of **X** is obtained by experiment or by Bayesian Inversion approach by means to get the input distribution [33].

One famous example is the calculation of the uncertainty propagation for the inverse tangent function using partial derivatives. Let f be:

$$f(x) = \arctan(x),$$

where Δ_x is the absolute uncertainty on our measurement of x. We have the derivative of $f(x)$ with respect to x is:

$$\frac{df}{dx} = \frac{1}{1+x^2}.$$

Therefore, we have the absolute propagated uncertainty Δ_f as:

$$\Delta_f \approx \frac{\Delta_x}{1+x^2}.$$

However, the real world problems are not as simple as inverse tangent function since the multiple inputs and more complex functions. Therefore, the quantification of the uncertainty cannot be done easily by calculating its mathematical derivative.

If the probability density function (PDF) of the input is known, then the input can be propagated with the uncertainty through the model. There are two approaches of propagating the uncertainty in the forward model namely intrusive and non-intrusive approaches. Intrusive approach requires access to the model in order to determine the QoI, whereas non-intrusive approach consider the function f as only as a black box (can be done without any knowledge about the mechanism of the function f) and only the input is needed.

The categorisation of techniques of UQ through a model differs among research communities, according to [33] those approaches are categorized as follows:

- **Sampling based methods**: Sampling methods evaluate a model multiple times for different sets of input values. The results are collected and described statistically. One example of sampling based method is Monte Carlo methods.
- **Spectral representations**: These methods are based on a series of polynomials (spectral representation) which are used to represent the random process and often referred to as (generalized) polynomial chaos expansions. The spectral representations are used to calculate the uncertainties or statistics for the model response.
- **Direct evaluation**: This statistical properties are calculated directly from the stochastic equations of the model. Sometimes dealing with the closure problem is needed because deriving statistical moments usually necessitates knowledge about properties of higher moments which means for linearly parameterized model, the uncertainty output is computed explicitly.

- **Perturbation methods**: Taylor expansion is done to the the function and the uncertainty is derived from that truncated Taylor expansions and characterized by mean, variance, or covariance. The problem with this method is that information about the shape of the probability density function of the output cannot be obtained.

In the case of this research, direct evaluation and perturbation methods are not applicable because neither can the writer access nor alter the code of the non-linear model, nor expanding it with the Taylor expansion. With that limitations, the sampling based and spectral representations methods will be used.

3.1.1. Monte Carlo Methods

Monte Carlo methods are a class of computational algorithms that are using a repeated random sampling to obtain numerical results. For example when it's used for numerical integration we can approximate the true integration value I as $\hat{I}_f$ as follows:

$$I := \int_0^1 f(x)\,dx \approx \hat{I}_f = \frac{1}{N}\sum_{i=1}^{N} f(x_i). \tag{3.2}$$

We can also calculate its root mean squared error (RMSE) as follows:

$$RMSE := \sqrt{\mathbb{E}[(I - \hat{I}_f)^2]} = \frac{\sigma_f}{\sqrt{N}} \approx \frac{\hat{\sigma_f}}{\sqrt{N}}, \tag{3.3}$$

with:

$$\hat{\sigma_f}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(f(x_i) - \hat{I}_f)^2. \tag{3.4}$$

Monte Carlo Methods for UQ

The algorithm of Monte Carlo methods for uncertainty quantification is similar to integration value. The difference is that instead of function evaluation to approximate integration value, function or evaluation is done in order to get the statistical properties. The algorithm for Monte Carlo methods for uncertainty quantification is written at algorithm 1.

Quasi Monte Carlo

Quasi-Monte Carlo method (QMC) is a modified Monte Carlo methods which using low-discrepancy sequences (also called quasi-random sequences or sub-random sequences) instead of sequences of pseudo-random numbers.

Algorithm 1 Algorithm of Uncertainty Quantification with Monte Carlo

Require: N number of sample, f probability distribution function
Generate a sample of the input random variable $Z \in \mathbb{R}^d$,distributed according to f,
for $k = 0$ to $N - 1$ **do**
Run the model with these parameters $Z^{(i)}$
Get the model output as $u\left(Z^{(i)}\right)$
end for
Compute the statistical properties of the Qols as the results of all the model run of the samples, e.g, $\mathbb{E}[u(Z)] \approx \frac{1}{N}\sum_{i=1}^{N} u\left(Z^{(i)}\right)$.

Koksma–Hlawka inequality and Low-Discrepancy Sequences

The basic idea of QMC comes from the Koksma–Hlawka inequality when we assume that $V(f)$ is constant and we minimize the error by reducing D_N i.e. by lowering the discrepancy of $\{x_i\}_{i=1}^{N}$ that is done by making it "better" spaced. [23]

Theorem 3.1 (Koksma–Hlawka inequality)**.** *Let $V(f)$ the variation of f,*

$$D_N = \sup_{A\subset[0,1]} \left|\frac{\text{card}(A)}{N} - \text{vol}(A)\right| \Rightarrow D_N \textit{ is the discrepancy of } \{x_i\}_{i=1}^{N}.$$

Hence, we have the relation of sampling quality as follows:

$$\left|I - \hat{I}_f\right| \leq V(f)D_N. \tag{3.5}$$

Sobol' sequence

The intuition of the Sobol' sequence is to subdivide each dimension in 2^N points. i.e. we have:

$$x, y, z, \ldots \in \left\{\frac{k}{2^N}, k \in \left[0.2^N - 1\right]\right\}.$$

The sequence is then associated an index $n \in \left[0 \ldots 2^N - 1\right]$ with a single tuple $\mathbf{p} = (x_n, y_n, z_n, \ldots)$. In order to select each element of the tuple to ensure a good distribution, generative matrices $C_x, C_y, C_z, \ldots$ that convert the binary representation of the index n to the binary representation of the position on each dimension are used.

$$x_n = \sum_{k=0}^{N-1} t_k 2^{-k-1}, \quad \text{with } \mathbf{t} = \mathbf{n} \times \mathbf{C}_x, \tag{3.6}$$

with t_k the k-th element of $\mathbf{t} = [t_0, \ldots t_{N-1}]$. Note that the matrix multiplication above is done in modulo 2. [24]

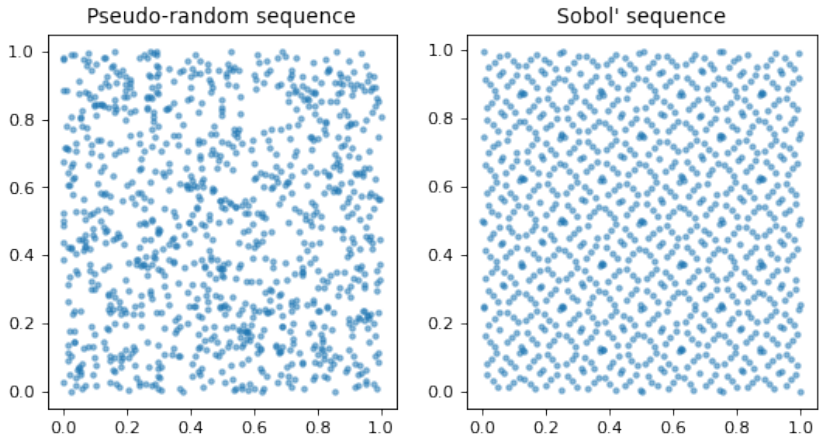

Figure 3.2.: Comparison of the pseudo-random sampling and sampling with Sobol' Sequence.

3.1.2. Generalized Polynomial Chaos Expansion

Since in order to perform UQ using MC, a repeated high-number of sampling is needed, problem occurs when facing a program which takes (a lot of) time to be run even though can be parallelized. The idea is to construct another system that the uncertainty can be easily quantified either by having faster running time or its statistical properties can be easily obtained. The idea is then a further mathematical tools are needed to replace this sampling process. Wiener used Hermite polynomials to represent Gaussian stochastic process and later, this is generally known as generalized polynomial chaos expansion (gPCE) and there exist many more polynomials to represent other stochastic process. [40]

To describe the method, the previous problem 3.1 is modified to multiple variables with variable(s) which are deterministic and independent are denoted as t and variable(s) which are uncertain are denoted as ω. The stochastic model is now written as:

$$Y = f(t, \omega) \tag{3.7}$$

The spectral expansion method in general or in this gPCE works similarly to the Fourier series. However, different with Fourier series which is a sum that represents a function with a sinusoidal function, Y is represented with a polynomial instead, as follows:

$$f(t, \omega) = \sum_{n=0}^{\infty} \hat{f}_n(t)\phi_n(\omega) \approx \sum_{n=0}^{N-1} \hat{f}_n(t)\phi_n(\omega) \tag{3.8}$$

$\phi_n(\omega)$ are polynomials of degree n and $\hat{f}_n(t)$ are coefficients that only depend on t (which at this point are still unknown) and the sum itself is always truncated after N terms in order for it to be able to be computed (which raises a truncation error). The polynomials

Distribution	Type of Polynomials
Gaussian	Hermite
Gamma	Laguerre
Beta	Jacobi
Uniform	Legendre

Table 3.1.: Commonly appear distribution and the polynomials that can be used to represent it [40].

$\phi_n(\omega)$ are chosen according to the PDF of ω [20] and list of polynomials that can be used to represent often used distribution are listed in 3.1.

Pseudo-spectral Approach

After choosing the right basis polynomial for the spectral expansion, the coefficients $\hat{f}_n(t)$ still need to be computed. It is no coincidence that all the polynomials which have been mentioned above are orthogonal since the orthonormality of the basis can be exploited to calculate the coefficients $\hat{f}_n(t)$. In function space the inner products of two function $p(\omega)$ and $q(\omega)$ with a weight function $\rho(\omega)$ is:

$$< p(\omega), q(\omega) >_\rho = \int_{\text{supp}(\rho)} p(\omega)q(\omega)\rho(\omega)d\omega, \tag{3.9}$$

with

$$< p(\omega), q(\omega) >_\rho = 0, \tag{3.10}$$

if $p(\omega)$ and $q(\omega)$ is orthogonal to each other. Therefore, with the fact that the basis polynomial $\phi_n(\omega)$ are orthogonal to each other, we have:

$$< \phi_i(\omega), \phi_j(\omega) >_\rho = \int \phi_i(\omega)\phi_j(\omega)\rho(\omega)d\omega = \gamma_i\delta_{ij} \qquad \forall\phi_0, \phi_1, \ldots, \phi_{N-1}, \tag{3.11}$$

$$\gamma_i = < \phi_i(\omega), \phi_i(\omega) >_\rho, \tag{3.12}$$

with γ_i the normalization constant which is 1 if the polynomial is also normalized. Since the polynomials that represent commonly appear distribution are also normalized, we assume that every $\phi_i(\omega)$ is normalized and we have:

$$< \phi_i(\omega), \phi_j(\omega) >_\rho = \delta_{ij}. \tag{3.13}$$

This property then is used to compute the coefficients $\hat{f}_n(t)$. Inner product is now done to the both side of the equation 3.8 and we have:

$$< \sum_{n=0}^{N-1} \hat{f}_n(t)\phi_n(\omega), \phi_m(\omega) >_\rho = < f(t,\omega), \phi_m(\omega) >_\rho . \tag{3.14}$$

Since $\hat{f}_n(t)$ is constant we can move the inner product inside the sum and obtain as follows:

$$\sum_{n=0}^{N-1} \hat{f}_n(t) \underbrace{< \phi_n(\omega), \phi_m(\omega) >_\rho}_{\delta_{nm}} =< f(t,\omega), \phi_m(\omega) >_\rho, \tag{3.15}$$

$$\Rightarrow \hat{f}_n(t) =< f(t,\omega), \phi_n(\omega) >_\rho, \tag{3.16}$$

$$\Rightarrow \hat{f}_n(t) = \int_\Omega f(t,\omega)\phi_n(\omega)\rho(\omega)d\omega. \tag{3.17}$$

It can be seen now that problem arises since $f(t,\omega)$ is only available as a black box since we want a non-intrusive approach, therefore the solution is using numerical quadrature such as Gaussian quadrature in 2.1.1 or Clenshaw-Curtis/Fejér Quadrature in 2.1.2.

Statistical Properties Computation

For our polynomials we have:

$$\phi_0(\omega) \equiv 1, \tag{3.18}$$

and also :

$$\mathbb{E}\left[\phi_0(\omega)\phi_n(\omega)\right] =< \phi_0(\omega), \phi_n(\omega) >_\rho . \tag{3.19}$$

Since the expectation value of our model is also the expectation value of our spectral expansion we can write it as:

$$\mathbb{E}[f(t,\omega)] \approx \mathbb{E}\left[\sum_{n=0}^{N-1} \hat{f}_n(t)\phi_n(\omega)\right] = \sum_{n=0}^{N-1} \hat{f}_n(t)\mathbb{E}\left[1 \cdot \phi_n(\omega)\right]. \tag{3.20}$$

By 3.18 we can also write equation above as:

$$\mathbb{E}[f(t,\omega)] \approx \sum_{n=0}^{N-1} \hat{f}_n(t) \underbrace{\mathbb{E}\left[\phi_0(\omega)\phi_n(\omega)\right]}_{=\delta_0} = \hat{f}_0(t). \tag{3.21}$$

Thus we obtained that the mean of the model is just the 0^{th} order $\hat{f}$ coefficient. To get the variance, we also write the variance formula in terms of the spectral expansion also with the knowledge that the expectation value of the model is $\hat{f}_0(t)$ and we have the equation as follows:

$$\mathrm{Var}[f(t,\omega)] = \mathbb{E}\left[(f(t,\omega) - \mathbb{E}[f(t,\omega)])^2\right] \approx \mathbb{E}\left[\left(\sum_{n=0}^{N-1} \hat{f}_n(t)\phi_n(\omega) - \hat{f}_0(x)\right)^2\right]. \tag{3.22}$$

Since the mixed terms is zero due to orthogonality we have:

$$\mathrm{Var}[f(t,\omega)] = \sum_{n=1}^{N-1} \hat{f}_n^2(t) \underbrace{\mathbb{E}\left[\phi_n(\omega)^2\right]}_{=1} = \sum_{n=1}^{N-1} \hat{f}_n^2(t), \tag{3.23}$$

which shows that the variance is just summation of the square coefficient of the spectral expansion without the 0^{th} term. This shows that for gPCE method the statistical properties of a model can be computed easily. The full length algorithm to computer the coefficient $\hat{f}_n(t)$ can now be written as in algorithm 2 with N is the maximum order of polynomials and K is the number of quadrature points. There any already many cases of UQ using gPCE for non-linear model for example gPCE for hydrology model [4] and crowd simulation [30], thus given the idea that similar setup can also be performed for this research.

Algorithm 2 Algorithm of Uncertainty Quantification with gPCE

Require: $N-1, K-1$
  generate polynomials ϕ_i
  for $k = 0$ to $K-1$ **do**
    generate x_k, w_k ▷ x_k and w_k are quadrature points and weights
    evaluate $f(t, x_k)$
  end for
  for $n = 0$ to $N-1$ **do**
    compute $\hat{f}_n(t)$ ▷ Computing the coefficients
    if $n \geq 1$ **then**
      compute $\hat{f}_n^2(t)$ ▷ Computing the square for the variance
    end if
  end for
  for $n = 0$ to $N-1$ **do**
    compute $\hat{f}_n^2(t)$ ▷ Computing the coefficients
  end for
  compute $\sum_{n=1}^{N-1} \hat{f}_n^2(t)$ ▷ Computing the variance

In case of multivariate models, expanding the model using gPCE is also possible. The difference is that the random variable ω is changed to a random vector $\boldsymbol{\omega}$, where for example if there are d random variables, we would we have:

$$\omega_1, \omega_2, \ldots, \omega_d \in \boldsymbol{\omega},$$

which we also have the multi index $\boldsymbol{n}$ as:

$$\boldsymbol{n} = (n_1, \ldots, n_d) \in \mathbb{N}_0^d.$$

The polynomials are also multivariate which are products of the univariate polynomials, i.e:

$$\phi_{\boldsymbol{n}}(\boldsymbol{\omega}) = \phi_{n_1}(\omega_1) \cdots \phi_{n_d}(\omega_d),$$

which is still orthogonal (also normalized in case of polynomials that would be used in this research) polynomials and therefore can also be written as:

$$< \phi_{\boldsymbol{n}}(\boldsymbol{\omega}), \phi_{\boldsymbol{m}}(\boldsymbol{\omega}) >_w = \delta_{\boldsymbol{nm}}, \quad \delta_{\boldsymbol{nm}} = \delta_{n_1 m_1} \cdots \delta_{n_d m_d}.$$

Hence, the multivariate polynomial chaos expansion is:

$$f(t, \omega) \approx \sum_{\boldsymbol{n}} \hat{f}_{\boldsymbol{n}}(t) \phi_{\boldsymbol{n}}(\boldsymbol{\omega}), \tag{3.24}$$

with the approximation to get the coefficient $\hat{f}_{\boldsymbol{n}}$ by the pseudo-spectral approach as follows:

$$\hat{f}_{\boldsymbol{n}}(t) \approx \sum_{\boldsymbol{k}} f\left(t, \boldsymbol{x_k}\right) \phi_{\boldsymbol{n}}\left(\boldsymbol{x_k}\right) w_{\boldsymbol{k}}. \tag{3.25}$$

3.2. Sensitivity Analysis

According to [5], sensitivity analysis is defined as study to evaluate the impact of varying some input parameters on some output response . Most of the sensitivity analysis met in the literature are based on derivative which is computationally expensive though effective [32]. The practices developed for sensitivity analysis are often categorized as local or global.

Global Sensitivity Analysis

Local sensitivity analysis methods vary the uncertain on one parameter, hence the information about the sensitivity of the model is valid only for a restricted range of the chosen parameter. Global sensitivity analysis methods instead take into account the uncertainty of multiple parameters and therefore allowing the sensitivity analysis on the whole interval of the uncertain parameters. Other than that, by doing global sensitivity analysis, the information about the interaction among the parameters on the output are also acquired. [32]

By these facts, global sensitivity analysis serves better for the this research because the model has many input parameters that might interact. There are many methods of global sensitivity analysis such as regression-based and variance-based methods [36]. Linear regression methods however are not available when analyzing non-linear model [36]. The later is also chosen for this research since the variance of the model can be easily acquired as the post processing of the UQ which are mentioned at 3.1 (more explanations at subsection 3.2.1 and 3.2.1).

3.2.1. Variance Based Sensitivity Analysis

Variance-based methods analyze the effect of the variance in the input to a model on the variance of the output [32]. A metric of how sensitive a parameter is usually represented by

Sobol' Indices [35]. As the method of UQ has been discussed in previous section, with both the used methods (MC and gPCE), the Sobol' Indices can be obtained even though the way to obtain those are totally different. This gives a huge advantage for the computational cost because the sensitivity analysis then can be obtained only as a byproduct of the uncertainty quantification.

Sobol' Representation

Sobol' Representation which also sometimes called High Dimensional Model Representation (HDMR) is a way to construct an expansion of the function f which has d uncertain input parameters $\boldsymbol{\omega}$ ($\omega_1 \dots \omega_d \in \boldsymbol{\omega}$). Similar with previous section about UQ, here f from equation 3.1 is expanded by its uncertain parameter as follows [33]:

$$\begin{aligned} f(t,\boldsymbol{\omega}) =& f_0(t) + \sum_{i=1}^{d} f_i\left(t,\omega_i\right) + \sum_{1\le i<j\le d} f_{ij}\left(t,\omega_i,\omega_j\right) + \\ & \sum_{1\le i_1<\ldots<i_s\le d} f_{i_1,\ldots,i_s}\left(t,\omega_{i_1},\ldots,\omega_{i_s}\right) + \ldots + f_{12\ldots d}(t,\omega), \end{aligned} \tag{3.26}$$

where $f_0(t)$ is the mean, $f_i(t,\omega_i)$ are functions with 1 uncertainty input and $f_{ij}(t,\omega_i,\omega_j)$ are functions with 2 uncertainty input and so on. The functions $f_i(t,\omega_i)$ represent independent contribution due to individual parameters, while functions with more uncertain variables (Such as f_{ij}) represent the interactions of multi variables on the response f. Based on the assumption that higher-order interaction terms have negligible effect, then equation 3.27 while emitting t is approximated as:

$$f(\boldsymbol{\omega}) \approx f_0 + \sum_{i=1}^{d} f_i(\omega_i) + \sum_{1\le i<j\le d} f_{ij}(\omega_i,\omega_j). \tag{3.27}$$

The variance of the equation is estimated as follows:

$$\mathrm{Var}[f(\boldsymbol{\omega})] \approx \sum_{i=1}^{d} \mathrm{Var}[f_i(\omega_i)] + \sum_{1\le i<j\le d} \mathrm{Var}[f_{ij}(\omega_i,\omega_j)], \tag{3.28}$$

dividing all by the total variance $\mathrm{Var}[f(\boldsymbol{\omega})]$ yields:

$$1 \approx \sum_{i=1}^{d} \frac{\mathrm{Var}[f_i]}{\mathrm{Var}[f]} + \sum_{1\le i<j\le d} \frac{\mathrm{Var}[f_{ij}]}{\mathrm{Var}[f]} \tag{3.29}$$

$$\Rightarrow 1 \approx \sum_{i=1}^{d} S_i + \sum_{1\le i<j\le d} S_{ij}, \tag{3.30}$$

or for simplicity just:

$$1 = \sum_{i=1}^{d} S_i + \sum_{1 \le i < j \le d} S_{ij}, \tag{3.31}$$

where:

$$S_i = \frac{\mathrm{Var}[f_i]}{\mathrm{Var}[f]}, \quad S_{ij} = \frac{\mathrm{Var}[f_{ij}]}{\mathrm{Var}[f]}.$$

Equation 3.31 defines the relation of ratio of partial variances with the total variance. Now we have S_i as the first order Sobol' indices which measure individual contributions by measuring the variance of the model response when a single input parameter ω_i varies. Next, the Total sensitivity indices S_{T_i} which characterize individual contributions and also its interaction between other input parameters can also be calculated by adding the higher-order interaction terms. By these definitions we can also write the expression of S_i and S_{T_i} as follows:

$$S_i = \frac{\mathrm{Var}[\mathbb{E}[Y \mid \omega_i]]}{\mathrm{Var}[Y]}, \tag{3.32}$$

$$S_{T_i} = 1 - \frac{\mathrm{Var}[\mathbb{E}[Y \mid \omega_{\sim i}]]}{\mathrm{Var}[Y]} = \frac{\mathbb{E}[\mathrm{Var}[Y \mid \omega_{\sim i}]]}{\mathrm{Var}[Y]}. \tag{3.33}$$

Here it has been shown that in order to get the Sobol' indices, only variance and partial variance are needed and there are many ways to approximate those values [31].

Monte Carlo Variance Based Sensitivity Analysis

Saltelli et al. proposed a method to calculate the partial variances that are needed for sensitivity analysis which lead to relatively more accurate results [31]. It is proposed that to calculate the first Sobol' Indices, the approximation as follows is used:

$$\mathrm{Var}[\mathbb{E}[Y \mid \omega_i]] = \frac{1}{N} \sum_{j=1}^{N} f(\mathbf{B})_j \left(f\left(\mathbf{A}_{\mathbf{B}}^{(i)}\right)_j - f(\mathbf{A})_j \right). \tag{3.34}$$

In order to approximate the partial variance to calculate the total sensitivity indices the partial variance approximation that is proposed to be used is the one that is proposed by Jansen [22] which is:

$$\mathrm{Var}[\mathbb{E}[Y \mid \omega_{\sim i}]] = \frac{1}{2N} \sum_{j=1}^{N} \left(f(\mathbf{A})_j - f\left(\mathbf{A}_{\mathbf{B}}^{(i)}\right)_j \right)^2. \tag{3.35}$$

$\mathbf{A}$, $\mathbf{B}$ are $N \times k$ dimensional sample matrices of input parameters (N is the sample size and k is the number of input parameters). Elements of matrices $\mathbf{A}$ and $\mathbf{B}$ are generated randomly by a distribution or can also be generated by using low discrepancy sequences

[34]. $\mathbf{A}_{\mathbf{B}}^{(i)}$ is a matrix which its column i are elements of matrix B and all other k 1 columns are elements of matrix A. It is more intuitive to show these as matrix form as follows:

$$\boldsymbol{A} = \begin{bmatrix} \omega_{11}^{A} & \cdots & \omega_{1k}^{A} \\ \cdots & \cdots & \cdots \\ \omega_{N1}^{A} & \cdots & \omega_{Nk}^{A} \end{bmatrix},$$

$$\boldsymbol{B} = \begin{bmatrix} \omega_{11}^{B} & \cdots & \omega_{1k}^{B} \\ \cdots & \cdots & \cdots \\ \omega_{N1}^{B} & \cdots & \omega_{Nk}^{B} \end{bmatrix},$$

$$\mathbf{A}_{\mathbf{B}}^{(i)} = \begin{bmatrix} \omega_{11}^{A} & \cdots & \omega_{1i}^{B} & \cdots & \omega_{1k}^{A} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \omega_{N1}^{A} & \cdots & \omega_{Ni}^{B} & \cdots & \omega_{Nk}^{A} \end{bmatrix}.$$

This also shows how many times the model is evaluated which is $(2 + d)N$ times.

gPCE Based Sensitivity Analysis

Sudret proposed a method to calculate the Sobol' Indices by the byproduct (coefficients) of gPCE method [36]. Since the Sobol' representation of the model f from equation 3.27 is analogous to gPCE representation from equation 3.24, it is proposed to use the coefficients of the gPCE representation to calculate the Sobol' indices. The Sobol' indices S_i is then computed as follows:

$$S_i(t) = \frac{D_i(t)}{\mathrm{Var}[f(t, \boldsymbol{\omega})]}, \quad D_i(t) = \sum_{\boldsymbol{n} \in A_i} \hat{f}_{\boldsymbol{n}}^2(t), \tag{3.36}$$

where A_i depends on the order of the contribution, which for the first order:

$$A_i = \left\{ \boldsymbol{n} \in \mathbb{N}_0^d : \forall j \neq i, \boldsymbol{n}_j = 0, \boldsymbol{n}_i \neq 0 \right\}, \tag{3.37}$$

and for the higher contribution:

$$A_i = \left\{ \boldsymbol{n} \in \mathbb{N}_0^d : \boldsymbol{n}_i > 0 \right\}. \tag{3.38}$$

Since it is known now that the computational cost of gPCE is for estimating the gPCE coefficients, therefore it is also the computational cost of the gPCE based sensitivity analysis and a little addition of post processing (no further model run required).

4. Implementation

4.1. Sound Propagation Model

In this subsection the sound propagation model based on a publication by Heimann [17] will be explained. The sound propagation model that will be analyzed here is an acoustical-meteorological coupled model. As it has been mentioned, the atmospheric model that will be used is a mesoscale model, thus the effect of topography can be seen.

4.1.1. AKUMET

AKUMET is a Lagrangian based sound propagation model which based on ray acoustic. AKUMET simulates the propagation of sound over hilly terrain on an inhomogeneous atmosphere. It calculates the sound pressure amplitude p at designated receiver points (or also all grid points).

Initialization

For AKUMET, x_1 and x_3 are the horizontal and vertical coordinates, respectively and $H(x_1)$ is the height of the terrain. The sound is emitted from an acoustical source at $\vec{x}_s = (x_{s1}; x_{s3})$ and is located at a height $h_s = x_{s3} - H(x_{s1})$ above the ground. The sound energy dispersion is simulated by releasing a large number N of sound particles simultaneously at time $t = 0$ at the position of the source. The emitted sound can also be composed of different frequencies $f_i (i = 1, \ldots, n)$. The frequency-dependent fraction of sound pressure of each particle $j (j = 1, \ldots, N)$ is defined as follows:

$$p_j(f) = \frac{1}{N}\sqrt{2\rho_s c_s J_o(f)}. \tag{4.1}$$

with ρ_s, c_s and J_o are the air density, the sound speed at the source, and is the sound intensity at the distance s_o from the source, respectively.

Geometrical spreading

Particles which pass through a spherical sampling area of radius d around a receiver point $\vec{x}_r$ decreases it's sound pressure as it's travel through the distance between the source and the receiver. The sound pressure amplitude at the receiver is given by the equation as

follows:

$$p\left(f,\vec{x}_r\right) = \frac{\Delta\psi s_o}{d}\sum_{j}^{N_r} W_j \cdot p_j(f), \tag{4.2}$$

with $W_j = a_2\left(1 - d_j/d\right)$ is a weighting factor with $a_2 = 1$ for a point source, $d_j \leq d$ is the minimum distance between a passing particle and the receiver, r_j is the path length between source and receiver and $\Delta\psi$ is the limiting launching angles.

Refraction

As it has been mentioned in ray acoustic section, if there exist gradients in the wind field or temperature field the particles move along refracted rays. Ray vector $\vec{x}_j(t)$ is the path of the j^{th} particle and its unit vector normal to the wavefront is $\vec{n}_j(t)$. From the restriction to two dimensions follows that $x_{j2} = n_{j2} = 0$. We have again the differential equation for the rays and normal vector as follows:

$$\frac{d\vec{x}_j}{dt} = \vec{v} + c\vec{n}_j, \tag{4.3}$$

$$\frac{d\vec{n}_j}{dt} = -\vec{\nabla}c - \sum_{i=1}^{3} n_{ji}\vec{\nabla}v_i, \tag{4.4}$$

with c the speed of sound depends on variable as follows:

$$c = \sqrt{\kappa R_L T}. \tag{4.5}$$

κ, R_L and T are the ratio of specific heat capacities, the gas constant of dry air and the temperature, respectively. These differential equations are numerically integrated for all particles and are repeated until the particle has left the model domain. The j^{th} particle travels the distance r_j every time step which is computed as follows:

$$r_j(t + \Delta t) - r_j(t) = \left|\vec{x}_j(t + \Delta t) - \vec{x}_j(t)\right| = \left|\vec{v} + c\vec{n}_j(t)\right|\Delta t, \tag{4.6}$$

The phase angle φ_j that is needed to simulate interference is also changing every time step and is computed as follows:

$$\varphi_j(f, t + \Delta t) - \varphi_j(f, t) = \frac{2\pi f}{c}\left|\vec{v} + c\vec{n}_j(t)\right|\Delta t. \tag{4.7}$$

Sound Pressure Level

The model calculates the sound pressure p at a grid point as a result of ray spreading or other physical effect such as reflection, interference, or absorption. The frequency-dependent sound pressure $p\left(f,\vec{x}_r\right)$ at any receiver point $\vec{x}_r$ is determined by the superposition of the contributions of all sound particles j which pass the receiver point within

a distance d and is computed as follows:

$$\begin{aligned} p\left(f,\vec{x}_r\right) =& \frac{\Delta\psi s_o}{d}\sum_{j}^{N}\Big(W_j p_j \exp\left[-A_j^{(a)}(f,t)\right]\times \\ & A_j^{(d)}(f,t)\cdot A_j^{(r)}(f,t)\exp\left[i\left(\varphi_j(f,t)+\delta\varphi_j^{(r)}(f,t)\right)\right]\Big). \end{aligned} \tag{4.8}$$

Parameters which account for reflection $A_j^{(r)}(f,t)$ and $\delta\varphi_j^{(r)}(f,t)$ also absorption $A_j^{(a)}(f,t)$ are defined in the next subsection but $A_j^{(d)}(f,t)$ which account for diffraction will not be discussed since the diffraction setting is not used in this research.

Boundary condition of the ground

There are two boundary conditions which are analyzed for this research namely total ground absorption and complex impedance ground. Total absorption of particles to the ground is used in order to model a ground that is ideal and reduces noise, whereas a complex impedance ground is used since it makes the particles loses sound pressure level and also changes the phase, yet still reflects those upwards according to the incident angle, which is close to reality. This complex impedance ground governs by the widely used complex reflection coefficient for spherical waves that is computed every time a particle hit the ground according to its angle of incident ϕ_j. The complex reflection coefficient is computed as follows:

$$Q\left(f,\phi_j\right) = R_p\left(\phi_j\right) + \left[1 - R_p\left(\phi_j\right)\right]F(w). \tag{4.9}$$

R_p is called the plane-wave reflection coefficient for locally reacting ground, which is calculated as follows:

$$R_p\left(\phi_j\right) = \frac{\sin\phi_j - \beta}{\sin\phi_j + \beta}. \tag{4.10}$$

β is the (complex) normalized admittance of the ground. $F(w)$ is the boundary loss function which is calculated as follows:

$$F(w) = 1 + i\sqrt{\pi}w\exp\left(-w^2\right)\operatorname{erfc}(-iw), \tag{4.11}$$

with the numerical distance w is given by:

$$w = \sqrt{0.5ikr_j(t)(\sin(\phi)+\beta)^2}, \tag{4.12}$$

where $k = 2\pi f/c$ is the local wave number. The parameters $A_j^{(r)}(f,t)$ and $\delta\varphi_j^{(r)}(f,t)$ (amplitude and phase) initially are set to :

$$A_j^{(r)}(f,t=0) = 1,$$

and

$$\delta\varphi_j^{(r)}(f, t = 0) = 0.$$

These will remain unchanged until there is a particle that hits the ground and is reflected. Afterwards, these will be changed according to:

$$A_j^{(r)}(f, t + \Delta t) = A_j^{(r)}(f, t) \cdot |Q(f)|, \tag{4.13}$$

$$\delta\varphi_j^{(r)}(f, t + \Delta t) = \delta\varphi_j^{(r)}(f, t) + \arg[Q(f)]. \tag{4.14}$$

Absorption in the air

Sound waves are continuously losing energy due to absorption in the air. The value of absorption coefficient α depends on temperature T, relative humidity U, and frequency f. The attenuation due to absorption in the air is computed as follows:

$$A_j^{(a)}(f, t + \Delta t) = A_j^{(a)}(f, t) + \alpha(f, T, U)\left(r_j(t + \Delta t) - r_j(t)\right). \tag{4.15}$$

4.2. Uncertainty Quantification and Sensitivity Analysis Library

Gödel listed several several open-source frameworks for uncertainty quantification [13]. Since the goal of the research is to do both uncertainty quantification and sensitivity analysis, we need frameworks which can accommodate both. Other than that, in connection to our problem there are requirements that have to be passed. For Monte Carlo case, other than generating random sampling, it is required to be able to generate low discrepancy sampling (for quasi-Monte Carlo) as well as generate sampling matrix for Monte Carlo based sensitivity analysis. For generalized polynomial chaos expansion case, other than creating orthogonal polynomial and calculating the coefficients, framework that offers also sparse grids creation is also preferred.

It is also preferable if the framework is available for Python language since even though AKUMET is written in Fortran, the script to call it many times and the script to modify the input parameters are written in Python. By these mentioned requirements, SALib (Sensitivity Analysis Library) [19, 21] and Chaospy [9, 8] are chosen.

SALib

SALib is a sensitivity analysis library framework for Python framework that is developed by University of California and University of Oxford. SALib does not directly analyze the mathematical or computational model. The way SALib works is by generating the input parameters for the model using one of the sample functions (randomized or low-discrepancy sequences) and computing the sensitivity indices from the model outputs.

SALib also provides several sensitivity analysis methods, such as Sobol, Morris, and FAST. [19, 21]

The sensitivity analysis that is used for this research is Sobol' sensitivity analysis and the sampling that is used for this research is also used for uncertainty quantification. Hence, SALib satisfies all requirements that are needed for Monte Carlo based uncertainty quantification and sensitivity analysis of this research.

Chaospy

Chaospy is a library framework for Python language that is used also for performing uncertainty quantification and Sobol' sensitivity analysis using polynomial chaos expansion. It includes a full suite of tools for doing polynomial sampling creation, quadrature creation, polynomial chaos expansions with dependent variables, and also even sparse grids creation. [9, 8] Hence, Chaospy satisfies all requirements that are needed for gPCE based uncertainty quantification and sensitivity analysis of this research.

Verification of the Used Uncertainty Quantification and Sensitivity Analysis Library

To verify the used frameworks, an uncertainty quantification and Sobol' sensitivity analysis of a mathematical function will be done. As it is often used by the UQ community, the mathematical function that will be analyzed is Ishigami function as follows:

$$f(x) = \sin(x_1) + a\sin^2(x_2) + bx_3^4 \sin(x_1) \tag{4.16}$$

The Ishigami function is often used to since it has a strong non-linearity and non-monotonicity properties. In order to verify these frameworks, comparison of uncertainty quantification and sensitivity analysis of equation 4.16 will be done in manually written Monte Carlo method, Sobol' sequences sampling quasi-Monte Carlo method using SALib and gPCE pseudospectral method using Chaospy. The input values x_1, x_2 and x_3 are following the distribution as follows:

$$x_1 \sim \mathcal{U}(-\pi, \pi),$$
$$x_2 \sim \mathcal{U}(-\pi, \pi),$$
$$x_3 \sim \mathcal{U}(-\pi, \pi).$$

The standard Monte Carlo method used 100.000 samples, while the QMC used 32.768 samples and the gPCE uses 9^{th} order of the Gaussian quadrature which means 1.000 samples. Note that this number of samples means number of a models which are needed to be run. For this mathematical case, it means how many times a function evaluation is done. The result of the comparison of these uncertainty quantification results are written in table 4.1. It is shown that the results are similar and in similar order (even though it can be further improved if using more samples or higher polynomial order for gPCE). This leads to a conclusion that this uncertainty quantification library frameworks can be used for the context of our research.

	MC	**QMC**	**gPCE**
Mean	3.500370649	3.500071058	3.499999
Variance	8.912312986	8.915974509	8.79899942

Table 4.1.: Comparison of mean and variance values as the result of uncertainty quantification using manually written code Monte Carlo method (MC), Sobol' sequences sampling quasi-Monte Carlo method using SALib (QMC) and gPCE pseudospectral method using Chaospy (gPCE).

		MC (Manually)	**QMC** (SALib)	**gPCE** (ChaosPy)
S_i	x_1	0.21928696	0.21461107	0.221447607
	x_2	0.68691114	0.68693592	0.682769706
	x_3	0.00082069	-0.00589628	5.54×10^{-30}
S_T	x_1	0.31330746	0.31330746	0.31723029
	x_2	0.6869165	0.6869165	0.68276971
	x_3	0.0946538	0.0946538	0.09578269

Table 4.2.: Comparison of first (S_i) and total (S_T) Sobol' sensitivity indices values as the result of uncertainty sensitivity analysis using manually written code Monte Carlo method (MC), Sobol' sequences sampling quasi-Monte Carlo method using SALib (QMC) and gPCE pseudospectral method using Chaospy (gPCE).

The sensitivity analysis by post-processing of previous uncertainty quantification process is also done. Comparison of Sobol' indices calculated by these 3 methods are shown in table 4.2. Same case with the uncertainty quantification, it is shown that the results are similar and in similar order (except for x_3 which is super close to 0). This leads to a conclusion that this sensitivity analysis library frameworks can be used for the context of our research.

4.3. Parallel Computing

Since both uncertainty quantification and sensitivity analysis algorithm are able to be parallelized, efforts are also made to parallelized the sampling processes. Python language are used since as it is mentioned before that even though AKUMET is written in Fortran, Python is used to write the input files and also to automatically run AKUMET multiple times in the Linux server.

4.3.1. Multiprocessing in Python

Multiprocessing is a feature of a system to support more than one processor run at the same time. In a multiprocessing system, applications are broken to smaller functions or

routines that run independently in a way that the operating system allocates these process to the processors [12]. The Python *multiprocessing* library is part of the standard library of the Python. It implements the shared memory programming paradigm which means that the programming of a system that consists of one or more processors that have access to a common memory [41].

Parallelizing UQ Process

As it can be seen for uncertainty quantification for both gPCE (algorithm 2) and Monte Carlo algorithm (algorithm 1), there exist a loop for which the function evaluations (or model run in this context) are done. This is the part that will be parallelized since as it has been mentioned before that the model run is the part which takes a lot of computational efforts. The method of multiprocessing that is used is *map* method. This *map* method returns a list of the results obtained by executing the function in parallel against each of the items in the iterable parameter [41]. The usage of this method is written as follows:

```
pool.map(func, iterable, chunksize=None)
```

The func parameter is the subroutine to be executed and the iterable is the parameters to the function. This multiprocessing method divides the iterable into a number of chunks. Afterwards, those are submitted to the process pool as separate tasks. The size of these chunks can also be specified by setting chunksize to a positive integer. [41]

In the case for this research, the func parameter that is passed, is the Python subroutine to run AKUMET on the Linux server and the iterable are the values which are generated either by Chaospy or SALib. The model is then run in the Linux server and the values of sound pressure level in each grid are the output which values are used in the post processing for uncertainty quantification and sensitivity analysis.

4.4. Uncertainty Quantification and Sensitivity Analysis of AKUMET

This study is divided into 2-dimensional and 3-dimensional study. Both are using the same Lagrangian sound propagation model, thus our hypothesis suggests that the sensitivity of the parameter would be similar. In addition, the main differences of 2 and 3-dimensional cases are that for 3-dimensional study, more cases will be studied since the real world applications are 3-dimensional. Since the real world applications are mostly characterizing the sound pressure level on ground level, therefore the particles that are propagated from the source, are only propagated downwards half circle for 2-dimensional and half sphere for 3-dimensional.

4.4.1. 2-Dimensional Cases

Since the running time of one simulation for 2-dimensional case is only a couple of seconds (mostly 10 until 20 seconds), Monte Carlo simulation can be applicable, thus also an uncertainty quantification and sensitivity analysis for high uncertain dimension can be done. Another thing only for 2-dimensional case is that additional uncertain dimension which is the variability of the topography complexity (that is not a parameter that already exist in AKUMET) can be introduced. The measure of the complexity of the topography is using geometrical tortuosity τ which is the arc-chord ratio or the ratio of the length of the curve L to the distance between its ends L_0 (as in figure 4.1) [11]. This relation can be easily written as follows:

$$\tau = \frac{L}{L_0} \tag{4.17}$$

This measure is normally used to characterized how complex path through a porous medium.

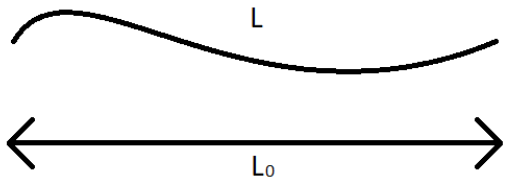

Figure 4.1.: Scheme of the measurement of geometric tortuosity.

Here, geometric tortuosity is also used since it can describe how winding a 2-dimensional topography is and therefore become our metric of topography complexity. In order to get various 2-dimensional topography which also could represent the real case, slicing of 3-dimensional topography of Vale do Cobrão in Portugal that was also used for wind energy, atmospheric and noise research [10, 6] will be acquired. The example results of the sampling are shown in figure 4.2, which look realistic even though having different complexity level. All y directions of this topography are sliced thus resulting of 201 different level of complexity with the maximum value 1.4427545 and the minimum value 1.4227340. These values become the bound of the uniform distribution that are used later.

Since there are no real world 2-dimensional case, the uncertainty in the input parameters are using the most extreme realistic possible parameter as a bounds of uniform distribution. Full descriptions of the uncertainty in the input parameters are shown in table 4.3. Other than that, since the topography is main factor that causes a huge meteorological variability [6], another setup without varying the topography complexity is also done since we already expect the topography is the most sensitive parameter and the sensitivity of other parameters then can be analyzed.

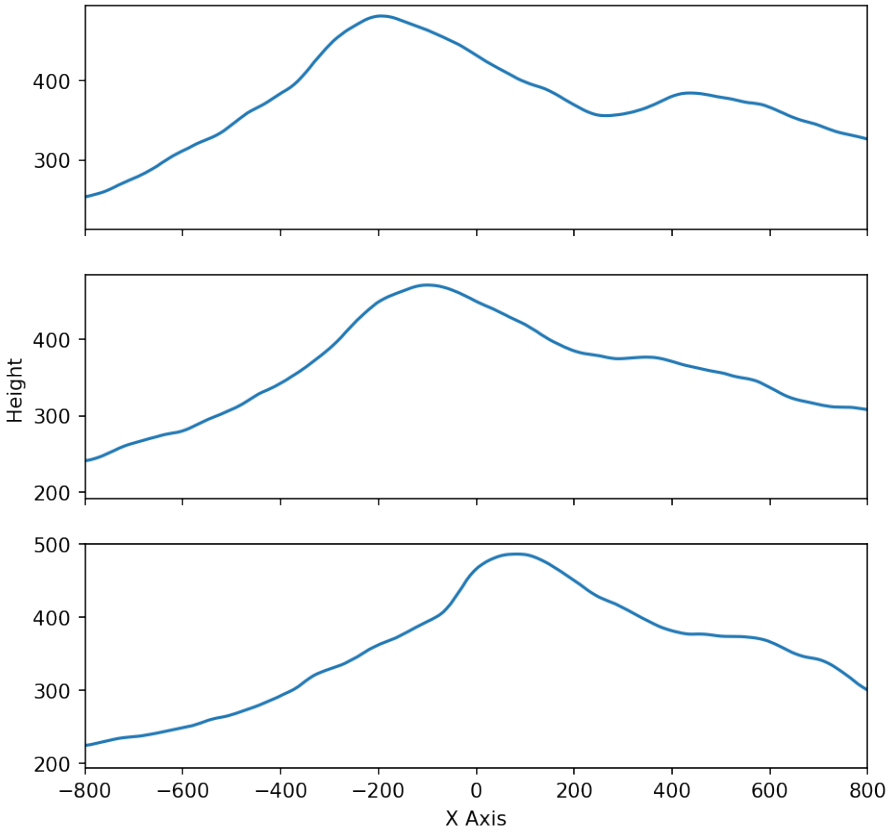

Figure 4.2.: Examples of three 2-dimensional topography with differences in complexity level (τ), which are 1.43868 (top), 1.42555 (middle) and 1.4388 (bottom).

The poi of interest (PoI) of this analysis are sound pressure level of all used grids which are available. In addition to that, sound pressure level on the ground will be the focused area since most human activities are done on the ground level. Other than that, a variation of boundary condition at the ground will also be tested to study the uncertainty of both cases. All these analysis that mentioned above will use quasi-Monte Carlo method since the model run for 2-dimension is fast. Since the 2-dimensional and 3-dimensional cases are based on the same model, the sensitivity analysis results of the 2-dimensional model will be used to reduce the uncertainty dimension for 3-dimensional case so that the number of samples that are needed, are reduced. gPCE based uncertainty quantification with 1 variable will also be performed in order to compare it with Monte Carlo based results to show that even for this case (uncertainty quantification of Lagrangian based sound propagation model), these 2 uncertainty quantification methods show similar results.

Parameter	Symbol	Distribution	Unit
Speed of sound	CEXP	$\mathcal{U}(330, 360)$	$\mathrm{m/s}$
Vertical gradient of the speed of sound	CEXPZ	$\mathcal{U}0, 5)$	$1/\mathrm{s}$
Temperature	TEXP	$\mathcal{U}(15, 25)$	$^{\circ}\mathrm{C}$
Vertical gradient of the temperature	TEXPZ	$\mathcal{U}(-20, 20)$	$\mathrm{K/km}$
Relative humidity	RFEXP	$\mathcal{U}(65, 75)$	$\%$
Wind speed at 10 meters height	UEXP	$\mathcal{U}(0, 10)$	$\mathrm{m/s}$
Wind direction	VEXP	$\mathcal{U}(268, 270)$	$^{\circ}$
Surface roughness	Z0EXP	$\mathcal{U}(0.1, 0.6)$	m
Air resistance on the ground	SIGEXP	$\mathcal{U}(290, 300)$	$\mathrm{KPas/m^{-2}}$
Topography complexity	τ	$\mathcal{U}(1.4227340, 1.4427545)$	$-$

Table 4.3.: The uncertainty of the input parameters for 2-dimensional case with also variation of topography.

4.4.2. 3-Dimensional Cases

There are many setup cases for 3-dimensional. Since the real world is three dimensional, the goal of uncertainty quantification in 3-dimensional is to do it in the real topography. The uncertainty quantification with 3-dimensional topography of Vale do Cobrão will be the final goal of 3-dimensional case. The research at Vale do Cobrão [10] or we refer as the project name which is Perdigão, also measured the variability data of the meteorological condition namely the wind speed, temperature gradient, and wind direction. Hence, this data is enabling this research to select the certain distribution to model the uncertainty of the uncertain input parameter.

3 Levels of Topography

The first cases that would be analyzed in the 3-dimension case is the 3 levels case of topography. The first level would be a simple hill that is artificially created by Gaussian distribution (figure 4.3), the second level is the projection of the topography that is used on 2-dimension case but equal on z direction which makes it a wavy topography (figure 4.4) and the third would be the actual topography of Vale do Cobrão (figure 4.5). In this analysis, only 1 variable that is the most sensitive after the topography would be used (results from 2-dimensional case). All sources of sound would be at the center of the topography and the winds are always flowing from left to right. Each cases will be divided again into 3 different simulation which are with distribution from Perdigão (1 specific frequency only and the whole third octave band) and with mean value from Perdigão but with a low variance value in order to just analyze an uncertainty that is caused by a measurement error and not the variability of the meteorological condition itself. All setup of 3-dimensional cases are recapped at table 4.4. Therefore, in total there are 9 cases of 3-dimensional setup.

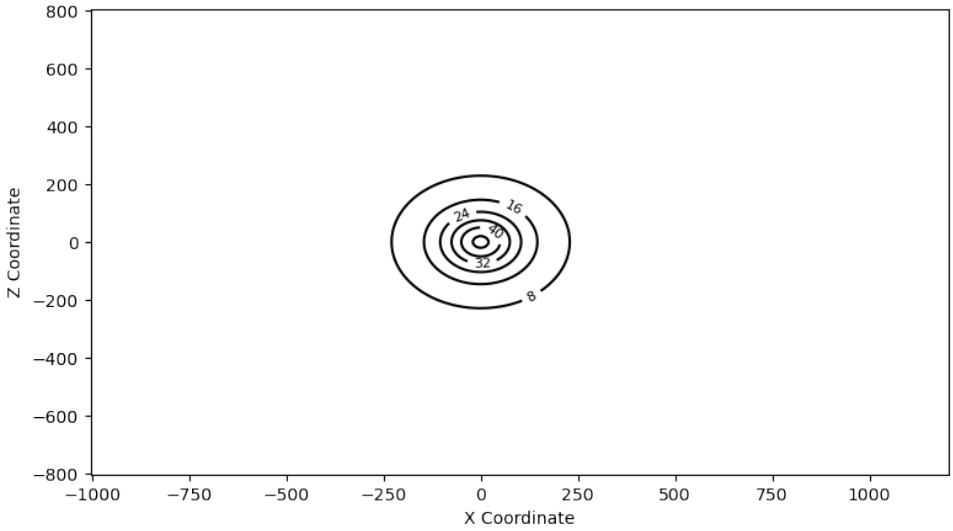

Figure 4.3.: Simple topography of a hill that is artificially created by Gaussian distribution in order to show that the source at the middle of the the topography will be located at the highest point of the topography.

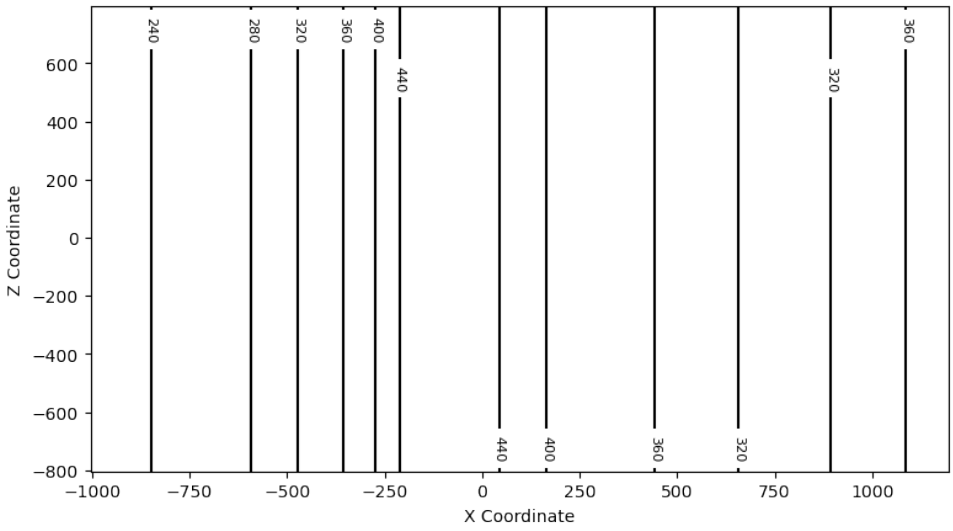

Figure 4.4.: Topography of a wavy hill that is artificially created by projecting topography in 2d case in order to show that the source at the middle of the the topography will be located at the highest point of the topography but on one direction still have the same height.

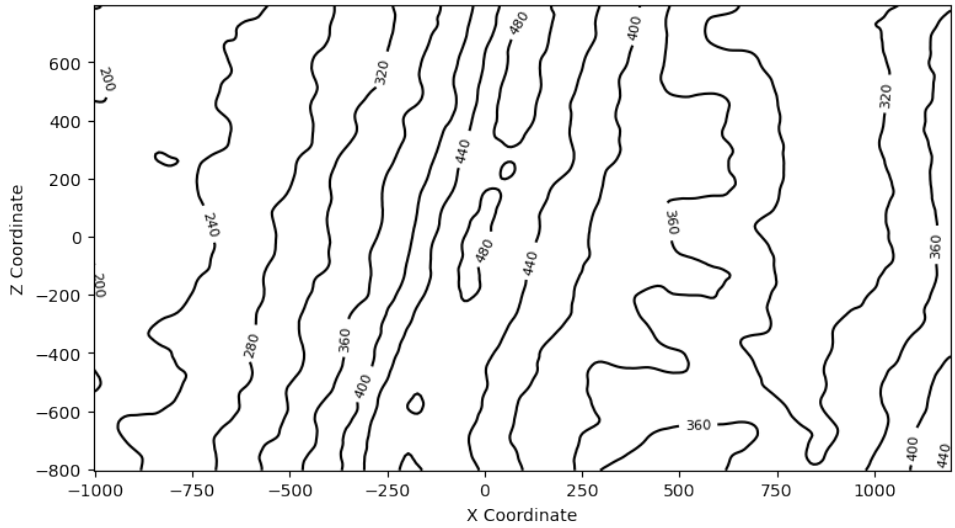

Figure 4.5.: Real topography of Vale do Cobrão with the source at the middle of the the topography which is the highest point of the topography.

Topography	Frequency Range	Uncertainty Dimension	Uncertainty Distribution Source
Level 1	100 Hz	1	Distribution by Perdigao Data
Level 1	100 Hz	1	Mean by Perdigao data but low variance
Level 1	3rd Octave Band	1	Distribution by Perdigao Data
Level 2	100 Hz	1	Distribution by Perdigao Data
Level 2	100 Hz	1	Mean by Perdigao data but low variance
Level 2	3rd Octave Band	1	Distribution by Perdigao Data
Level 3	100 Hz	1	Distribution by Perdigao Data
Level 3	100 Hz	1	Mean by Perdigao data but low variance
Level 3	3rd Octave Band	1	Distribution by Perdigao Data

Table 4.4.: List of setup that are used for 3-dimensional uncertainty quantification case.

4.5. Inverse Rosenblatt Transformation

If the covariance of the chosen distributions causes big value of matrix condition number, expansion generation will also likely be prone to error and the precision will drop quickly as well with the truncation of polynomial order. However, for gPCE (particularly when using ChaosPy library), there exist a workaround for this phenomenon namely Inverse

Rosenblatt transformation.

The trick is to create a proxy distribution that behaves nicely (for example Gaussian distributions). The polynomials, nodes and weights are then generated and those are used instead of the one of interest. Finally the nodes are transferred back to the domain of interest using Inverse Rosenblatt Transformation. Code example of this trick is displayed at source code 4.1.

```
import chaospy

define solves(someinput):
            #some function

dist1 = chaospy.Normal(3,2303)
#First distribution with big variance compare to the mean value

dist2 = chaospy.Normal(5,4)
#Second distribution with variance value comparable to the mean value

dist3 = chaospy.Normal(2,0.001)
#Third distribution with low variance compare to the mean value

distJ = chaospy.J(dist1, dist2, dist3)
#Joint distribution of the distributions

dist_proxy = chaospy.J\
        (chaospy.Normal(0, 1), chaospy.Normal(0, 1), chaospy.Normal(0, 1))
#Creation of distributions which are behave nicely as a proxy

poly = chaospy.orth_ttr(order=3, dist=dist_proxy)
#Generating polynomials from the proxy distribution

nodes_proxy, weights = chaospy.generate_quadrature\
        (order=5, dist=dist_proxy, rule='gaussian')
#Creating nodes for the proxy distribution

nodes = distJ.inv(dist_proxy.fwd(nodes_proxy))
#Getting the quadrature nodes from the proxy distribution

approx = chaospy.fit_quadrature(poly, nodes_proxy, weights,solves())
expected_pce = chaospy.E(approx, dist_proxy)
```

Source Code 4.1.: Inverse Rosenblatt transformation that can be applied in ChaosPy Python library in order to reduce error that is caused by big value of matrix condition number.

5. Results and Discussions

5.1. Parallelization

The analysis of parallelization is only done in 2-dimensional case. The reason is that 3-dimensional case model run takes a lot of times (4 to 12 hours depends on the number of particle which is chosen), while 2-dimensional model could be run in less that 1 minutes (since it needs fewer particle) but still able to represent the parallelization result of the model since it is the same simulation (only differs in dimensional setting) and the sampling process of both methods (Monte Carlo and gPCE) should be *embarrassingly parallel* [26]. Table 5.1 shows the speedup of each number of processes (cores) that are used for the sampling process. The speedups that are achieved, are close but not exactly linear. This happened since the parallelization is done only to the sampling part of the code (which for 2-dimensional simulation only takes 2-3 seconds) and not the part of generating random samples, calculating the statistical properties or transmitting the randomly generated parameters to the Linux server in order for it to run the model, hence according to the Amdahl's law [18], the speedup cannot be fully linear. Other than that, error in the server connection which resulting to reiterate a model run also happened multiple times which the running time also accumulated to this calculation.

Number of Process	Time (s)	Speedup
1	41275	–
2	22543	1.83096516
4	11938	3.457484306
6	6991	5.904083486
8	5643	7.314451116
10	4795	8.608018279
12	3740	11.03621595

Table 5.1.: Table of speedup of Monte Carlo process with 1792 samples by different number of processes (cores) compare to single process.

5.2. 2-Dimensional Cases with All Possible Meteorology and Topography Uncertainty

It has been mention in subsection 4.4.1 that there will be no uncertainty quantification for this scenario, since the goal of this analysis is to prove the hypothesis that the topography is the most sensitive parameter. All the uncertain input parameters is written in table 4.3.

5.2.1. Sensitivity Analysis

Figure 5.2 shows the Sobol' indices of each uncertain parameters. It is shown that when the topography is also varied, it becomes the most sensitive parameter. The first and total order Sobol' indices also show similar pattern, thus give understanding that the interaction terms do not influence strongly to the variance. This result proves the hypothesis that mentioned above and confirms that topography is the most important aspect that affects outdoor sound propagation, even though there exist other varying meteorological condition. However this should not be a problem in a real world case simulation since the topography is usually fixed and certain if the study is done at land.

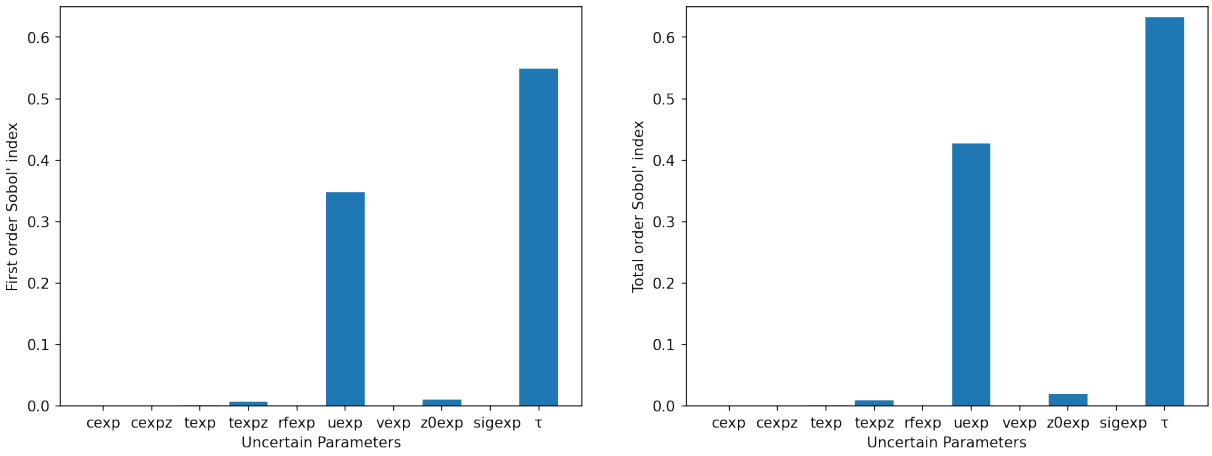

Figure 5.1.: First order Sobol' sensitivity indices (left) and total order Sobol' sensitivity indices (right) of all used meteorological input parameters of AKUMET including topography that are obtained using Monte Carlo method and Jansen/Saltelli approximation [22, 31].

5.3. 2-Dimensional Cases with All Possible Meteorology Parameters Uncertainty with Fixed Topography

Here, the uncertain parameters will be the same as previous study which are those which are written in table 4.3, with the exception for the topography complexity τ which is set to be constant to have the 2-dimensional topography as figure 5.2. Both uncertainty quantification and sensitivity analysis will be done in this study since the effects of variation of meteorology in a 2-dimensional fixed topography is also studied [16] as a simplified of phenomenon that happen in the real world. Other than that, sensitivity of parameters outside the complexity of topography are also needed in order to assist further research in 3-dimensional case, so that the uncertainty dimension is reduced (thus also the computational cost).

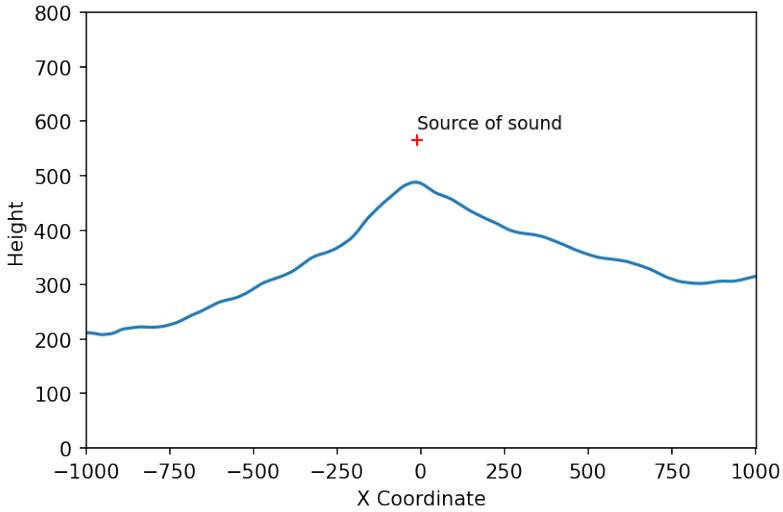

Figure 5.2.: Topography and sound source of uncertainty quantification for 2d-simulation scenario.

5.3.1. Uncertainty Quantification

There are 2 uncertainty quantification cases which are a ground boundary condition which absorbs particle when particle that hits it (total absorption) and the one that using complex impedance boundary condition (from subsection 4.1.1). The analysis of total absorption is done since it is a simplified case and shows a smooth sound propagation level profile, but it is also unknown how uncertain the sound pressure level if this boundary condition is applied when there exist uncertainties on the input parameters. Complex impedance ground boundary condition is studied since it's close to reality [17] and the study of a

model that is close to reality should also be able to assist the research in the real physical world since presumably it shows similar result or pattern with a real life measurement.

It is shown both in figure 5.3 and figure 5.4 (note that the plot of standard deviation is set to maximum of 4 dB to make the the lower standard deviation more visible, since the high standard deviation (larger than 8 dB) are located at the boundary grid on the top of the source that are not taken into account) that SPL in both ground boundary conditions have larges uncertainty in the up-wind (left side on the figure, since the wind goes from the left side to the right side with details on wind speed profile on figure B.1) and long field region. This result shows consistency with [15] which mentioned that up-wind and long field region are more affected by variations of meteorological condition.

For total absorption ground boundary condition case, there are simple deviations which are shaped like a ray patterns which looks more visible the farther it is from the sound source. This pattern exist because the uncertainty in input parameters cause the variation of how fast the particles hit the ground since as it is written before in 4.1.1 that wind speed vectors have influence. Since no particles are reflected after hitting the ground with this ground boundary condition, more particles hit the ground faster means also less particles that propagate in the domain, thus also less sound pressure level.

For total complex impedance ground boundary condition case, there are a lot of deviation that happen above the ground as well. This pattern exist with the same reason as before, which is because the uncertainty in input parameters cause the variation of how fast the particles hit the ground since as it is written before in 4.1.1 that wind speed vectors have influence. However, since particles are reflected after hitting the ground with this ground boundary condition, more particles are causing interference effect and according to it's phase, can cause higher or lower sound pressure level. The particles are reflected depending on their angle of incident ϕ_j thus showing more weird interference pattern.

5.3.2. Sensitivity Analysis

The sensitivity analysis is done as a post processing of uncertainty quantification data, thus having the same case as before but only done for complex impedance ground boundary condition since it's the more realistic model. Figure 5.5 shows the Sobol' indices of each uncertain parameters. It is obtained now that the wind speed *uexp* does have the most contribution on the total variance or the most sensitive model because of its unmatched Sobol' index. This results motivates to do another uncertainty quantification since the dimensionality of the uncertainty now can be reduced into only 1 dimension which means only *uexp* as a variable with uncertainty.

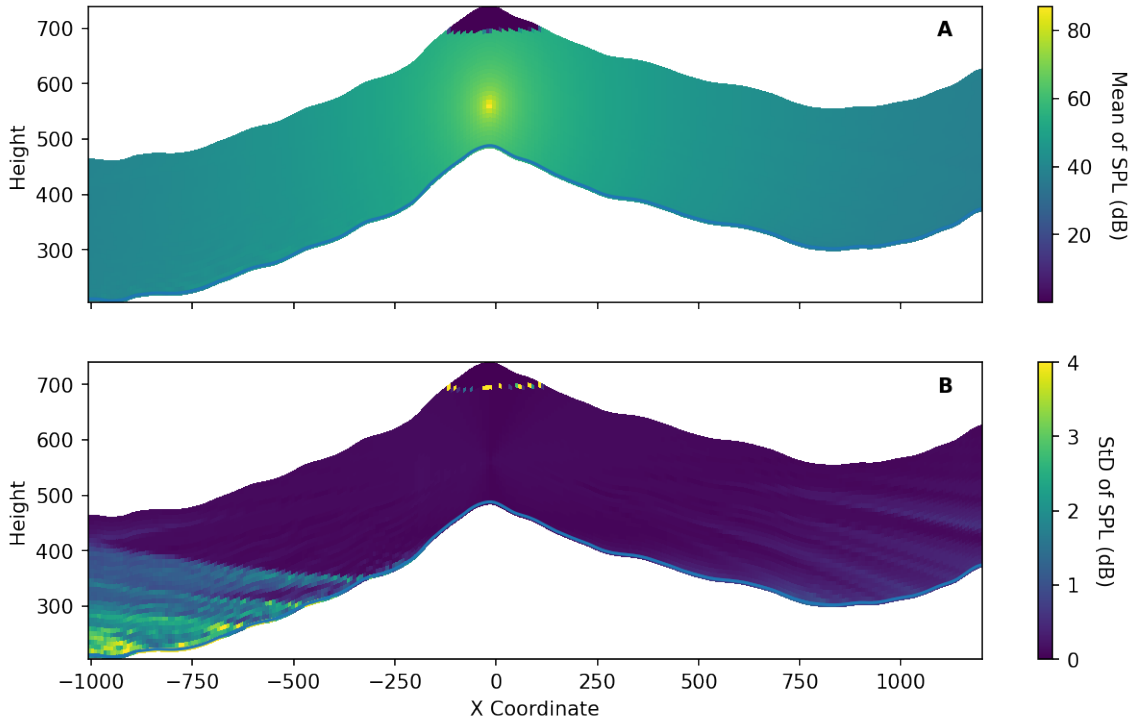

Figure 5.3.: Plot of mean values of SPL on every coordinates (A) and standard deviation of SPL on every coordinates (B) as a results of Monte Carlo uncertainty quantification using 45056 samples with total absorption ground boundary condition.

5.4. 2-Dimensional Cases with Wind Speed only as Uncertain Paramater with Fixed Topography

Reducing the dimensionality is important since doing uncertainty quantification for 3-dimensional is also planned and therefore can reduced the number of samples that are needed. Here, comparison with gPCE will also be shown. Since the Sobol' index is high, a smaller range of uncertainty will be used in order to simulate error because of measurement and not error because of the atmospheric wind variability. The variation of the speed of wind therefore will be as follows:

$$x_1 \sim \mathcal{N}(5.248, 0.5),$$

with the mean values of wind speed obtained from real data from Vale do Cobrão [10]. The result of uncertainty quantification for the whole domain is shown in figure 5.6. Both pattern of mean and standard deviation look similar to 5.4, therefore analysis of the uncertainty of the model will be focused only in the uncertain *uexp*.

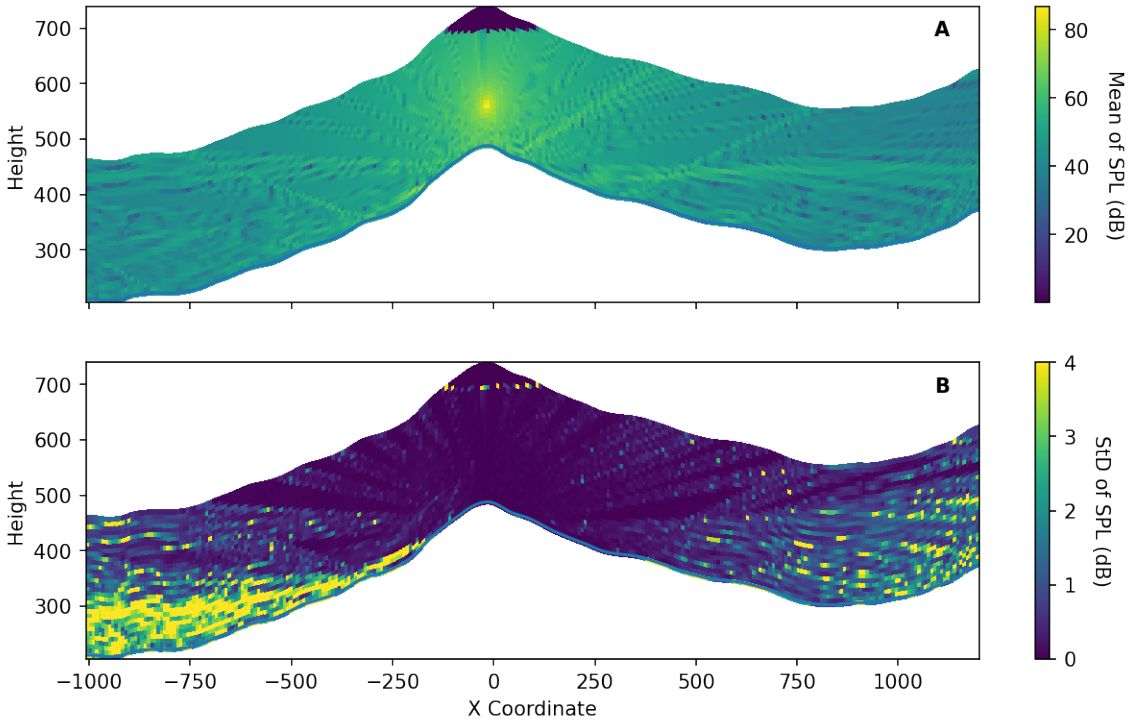

Figure 5.4.: Plot of mean values of SPL on every coordinates (A) and standard deviation of SPL on every coordinates (B) as a results of Monte Carlo uncertainty quantification using 45056 samples with complex impedance ground boundary condition 4.4.1.

5.4.1. Uncertainty Quantification at the Ground Level

Since the region that is most needed to be studied is the ground level, this region will be focused more in this section. Figure 5.7 shows the result of SPL at the ground level with several example of wind speed at it's mean using Monte Carlo.

It is shown that at up-wind region, a wider variation of SPL are more observed than in down-wind region, especially the farther it is from the location of the source. It is observed that at down-wind region near the source, the variation of wind speed does not really affect the value of SPL.

Comparison of Monte Carlo and gPCE

In this subsection, the 2 mentioned uncertainty quantification methods will be compared. The uncertainty quantification of Monte Carlo data (8032 samples) are still used and will be compared to gPCE with order 10 quadrature, which means 20 quadrature point, thus

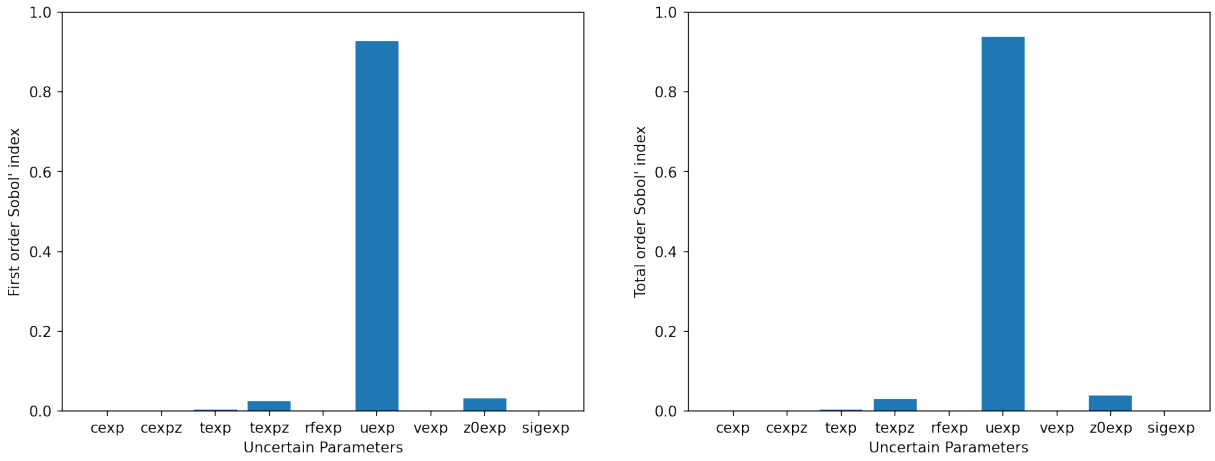

Figure 5.5.: First order Sobol' sensitivity indices (left) and total order Sobol' sensitivity indices (right) of all used meteorological input parameters of AKUMET with fixed topography that are obtained using Monte Carlo method and Jansen/Saltelli approximation [22, 31].

20 times model run (20 samples). The comparison results of mean and variance as the ground level are shown in figure 5.8 and figure 5.9. The results show for mean values, the result of gPCE and Monte Carlo almost identical even though gPCE needs only $\frac{1}{400}$ samples that Monte Carlo needs. For Standard deviation, we observe that both methods shows the same pattern but sometimes differs. However, the pattern of standard deviation is similar. Therefore, it can be concluded that uncertainty quantification using gPCE are useful for the research and can be applied also for 3-dimensional case.

5.5. 3-Dimensional Cases

The results of uncertainty quantification of 3 levels of topography are shown at figure 5.10 for the mean values and figure 5.11 for the standard deviation values. The results for 3-dimensional are unexpected, it is found that the uncertainties are not as big as in the 2-dimensional cases. Both figure 5.10 and 5.11 show the results of the uncertainty which are using the distribution that are achieved from Perdigão project data and the others are put at the appendix (B.3, B.4, B.5, B.6, B.7 and B.8). In addition to the fact that the result of sensitivity analysis of the 2-dimensional cases shows the speed of wind at 10 meters height is the most sensitive parameter, it is also chosen since our knowledge that wind speeds are always varying (data is shown at B.2). The mean figures that are shown in figure 5.10 show an intuitive results since it looks similar to the result when the uncertain inputs are changed to the deterministic with the mean values as it's deterministic values. In order to check the results, another comparison is done. Since the 2-dimensional simulation, uses

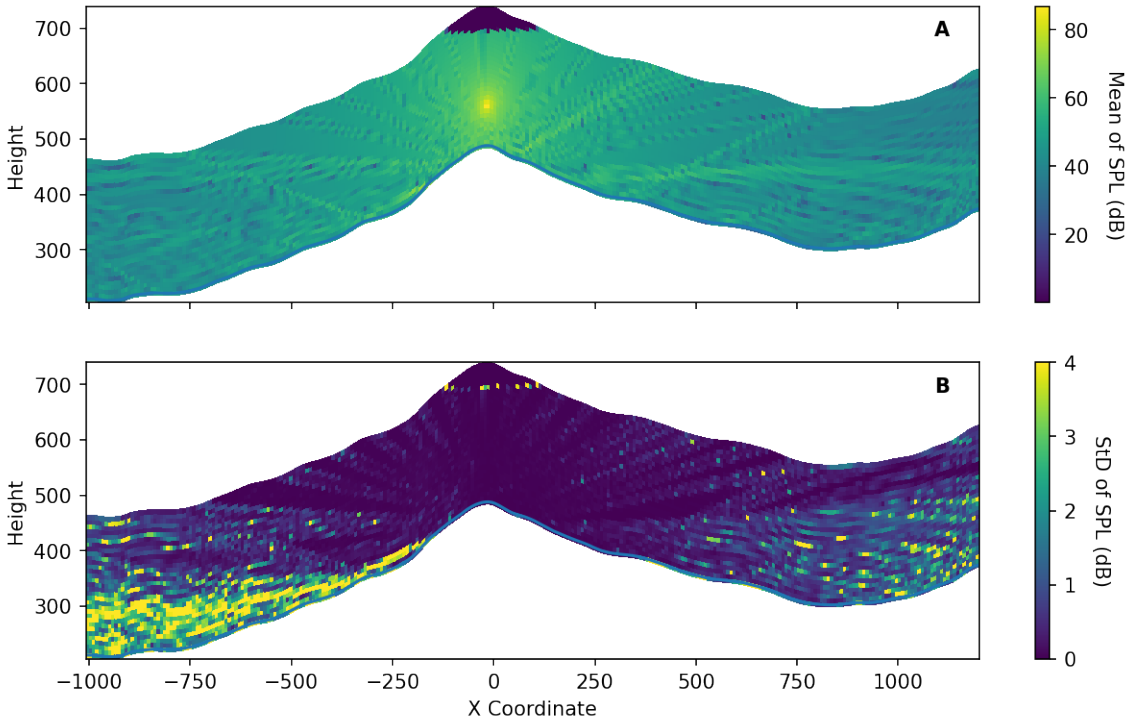

Figure 5.6.: Plot of mean values of SPL on every coordinates (A) and standard deviation of SPL on every coordinates (B) as a results of Monte Carlo uncertainty quantification using 8032 samples with complex impedance ground boundary condition but with only little uncertainty in wind speed.

the slice of Vale do Cobrão's topography (which slice is shown in figure 5.12) and uses the same parameters as the 3-dimensional one, we can slice the 3-dimensional results at the same point and compare the data. Figures 5.13 shows this comparison and it is shown that the mean data has similar pattern (even more at the down-wind case) and gives an understanding that the model is running correctly. However it is also shown in figure 5.13 that the standard deviation values for the 3-dimension differ a lot even in different order.

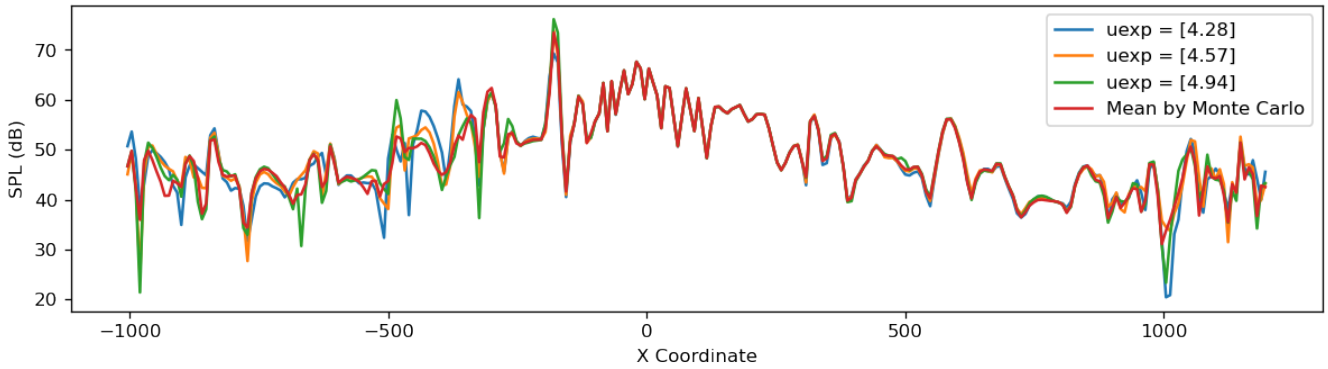

Figure 5.7.: Plot of mean values of SPL at ground level compare to SPL at the ground level with variation of *uexp*, a more focused version of figure 5.6.

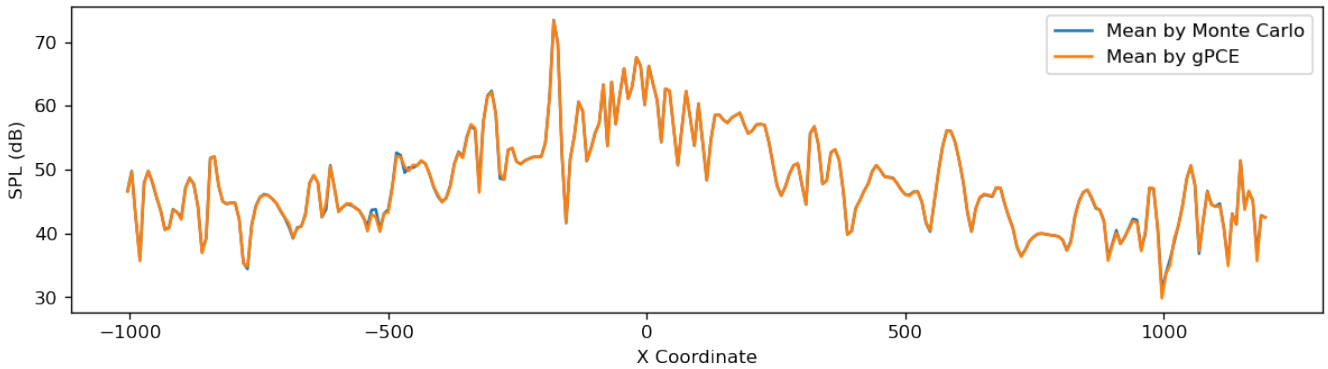

Figure 5.8.: Comparison of mean value of SPL at ground level by using Monte Carlo method (8032 samples) and gPCE method (20 samples).

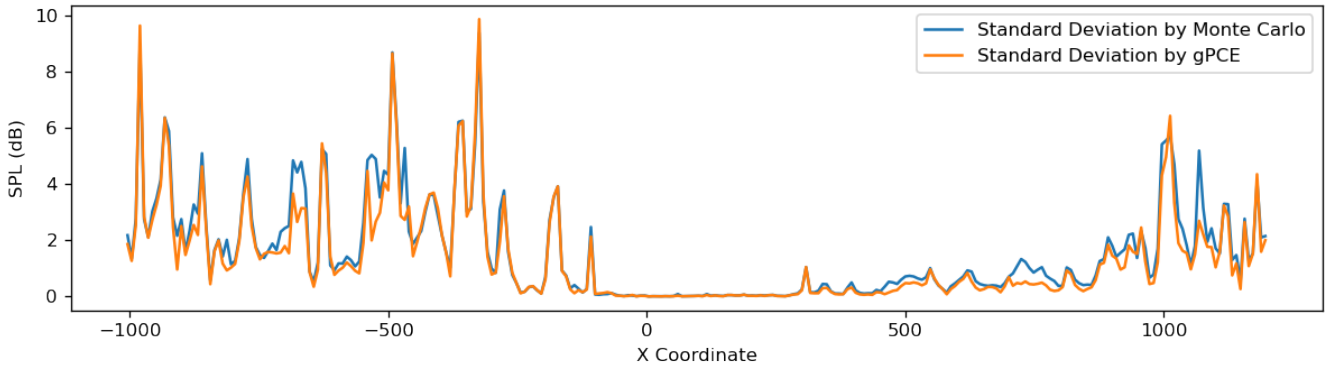

Figure 5.9.: Comparison of standard deviation value of SPL at ground level by using Monte Carlo method (8032 samples) and gPCE method (20 samples).

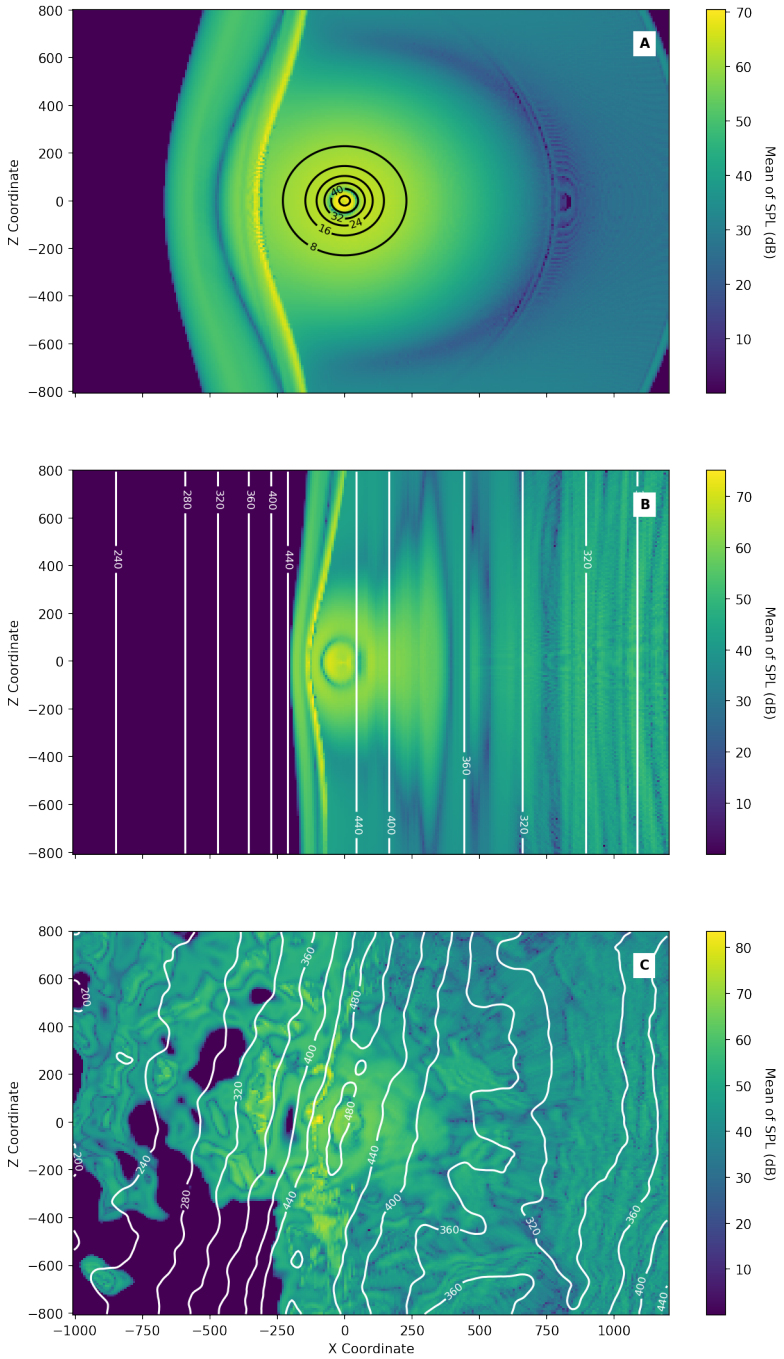

Figure 5.10.: Plot of mean values of SPL on every coordinates at the lowest ground with different topography which are (A) simple topography, (B) wavy topography and (C) Vale do Cobrão topography when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 0.5)$) using gPCE method with 20 samples.

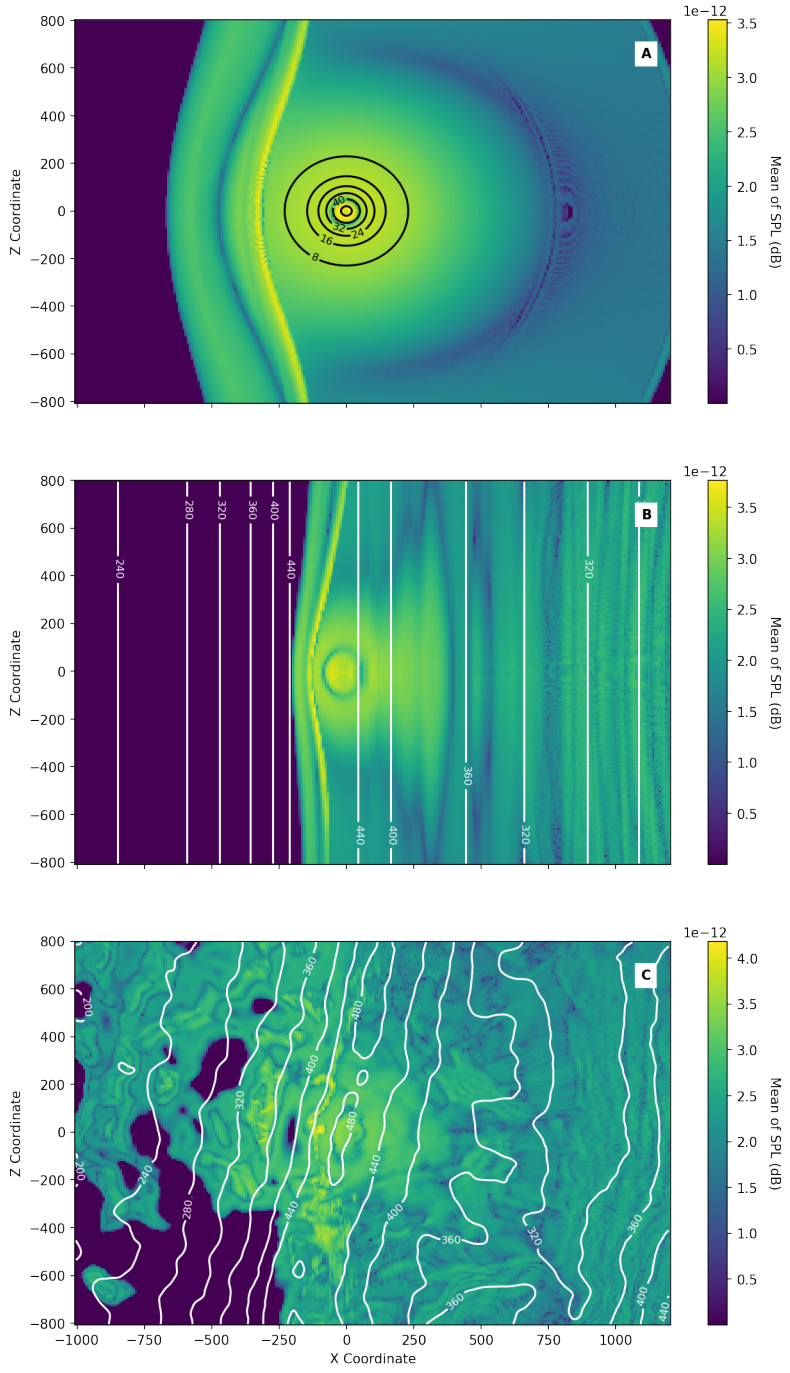

Figure 5.11.: Plot of standard deviation values of SPL on every coordinates at the lowest ground with different topography which are (A) simple topography, (B) wavy topography and (C) Vale do Cobrão topography when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 0.5)$) using gPCE method with 20 samples.

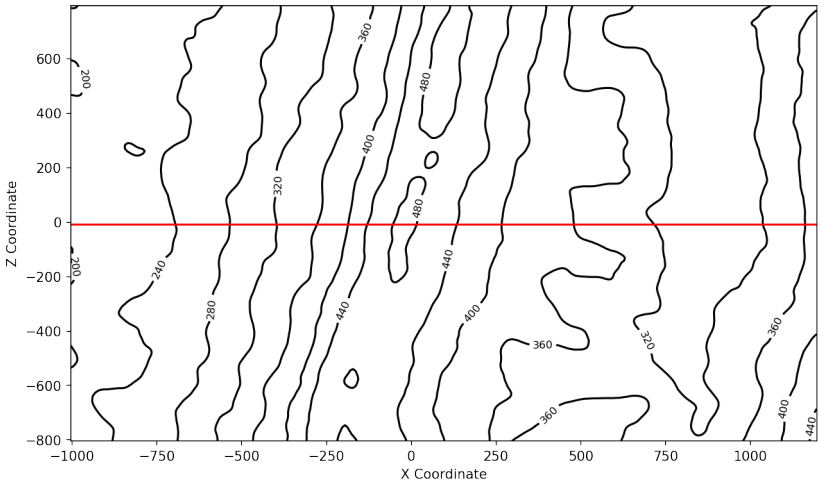

Figure 5.12.: Marked by the red line is the slice of the 3-dimensional topography that is used as input topography for the 2-dimensional simulation.

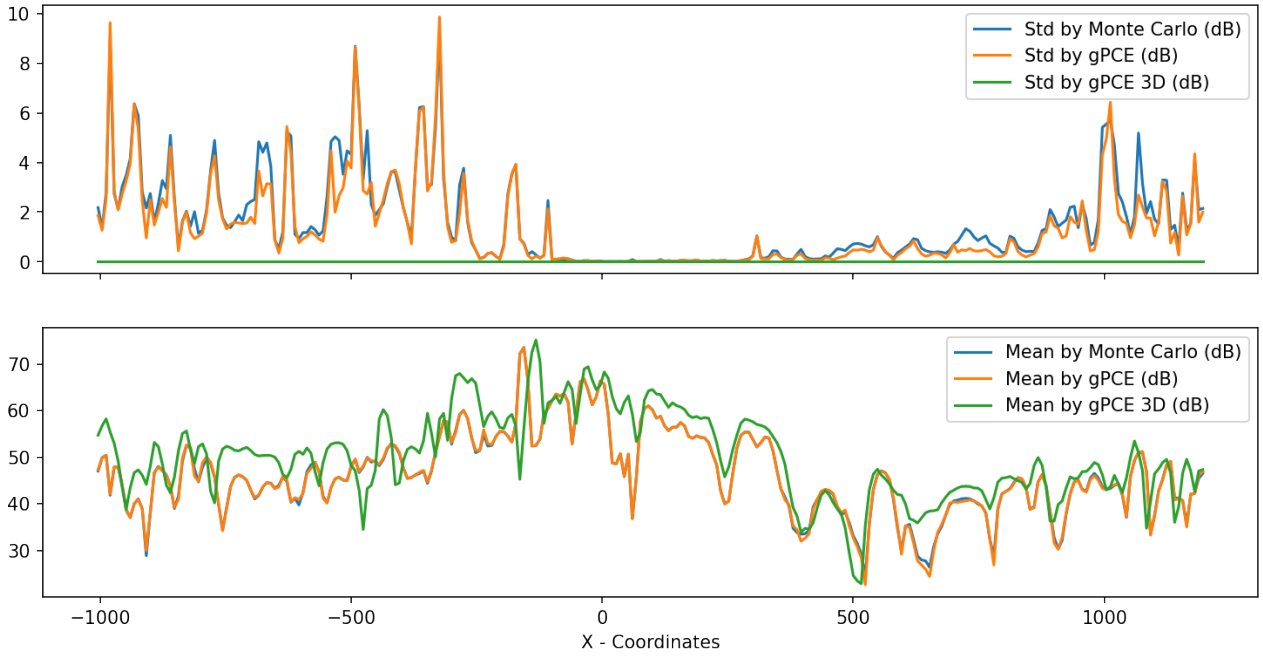

Figure 5.13.: Comparison of standard deviation and mean values at one level of the 2-dimension simulation with the slice of Vale do Cobrão's topography uses as input topography, compares to the 2-dimensional slicing of 3-dimensional simulation results with the Vale do Cobrão's topography full 3-dimensional topography as input topography (which slice is shown in figure 5.12).

6. Conclusions and Outlook

The uncertainty quantification and sensitivity analysis method chosen for this study proved to be functional. Both methods, Monte Carlo and gPCE show similar results and both can be used to analyze the uncertainty and sensitivity of Lagrangian-based sound propagation models which takes meteorological conditions into account. Another important finding is the fact that because of many of the uncertain parameters cannot be reduced and uncertainty quantification using gPCE method does not require a lot of samples, quantifying the uncertainty using gPCE method in every research that uses acoustic-meteorology coupled model can be done as a complement to a normal simulation where usually only the value of the sound pressure level is obtained without any information of it's uncertainty (for example the standard deviation of the sound pressure level), even though it is well known that meteorological values are uncertain.

6.1. Conclusion of The Specific Results

Specifically from the 2-dimensional case, it is evident that the topography is the most important parameter to be looked on to reduce the uncertainty of the model. After the topography is made to be fixed/constant, the Sobol' sensitivity index of the speed of wind at 10 meters from the ground is shown to be the highest and not comparable to other parameters. Reducing the uncertainty of the speed of wind is not easy since in every outdoor case, the speed of wind is always changing. In 2-dimensional case, it is confirmed that the pattern of uncertainty of AKUMET follows the uncertainty of the real world sound propagation model ([15] as reference). Even in the case of low input uncertainty, the up-wind and long-field regions experience great uncertainty only with variations in wind speed.

However, in the case of 3-dimensional for all different kind of topography (even when the topography of Vale do Cobrão is used), the values of the standard deviation are unusually small (order of 10^{-12}), even when the uncertain input parameter which is chosen has the highest Sobol' sensitivity index. Many scenario have been tested by changing the input parameters that is object to uncertainty and also the frequency of the sound wave, but those results still show small standard deviations while the mean plots look intuitively correct. This usually indicates the robustness of the model, however this does not seem to fit with the physical reality and further investigation of this result is needed. This results serve as preliminary results which was using Sobol' Indices from 2-dimensional case as reference and this is proven to be not correct thus, there is a possibility that the uncertainty of 2-dimensional and 3-dimensional model behave differently. Therefore, the uncertainty

and sensitivity analysis results that can be said as reliable results are the cases from 2-dimension one.

6.2. Outlook

The first outlook for the future work is obviously about the 3-dimensional model. Since the analyzed results are always on the lowest ground, there probably could be an unknown effect of the model that has not yet known. The other thing is the creation of the meteorological field that AKUMET performs before the sound propagation simulation is done, could also be analyzed deeply.

For 2-dimensional case, a variation of a topography that includes a barrier can also be done in order to understand how a topographical barrier properties (height, how close the receiver/source to the barrier and reflection/absorption properties) might affect the sound pressure level. Also for all cases in this research, a non-moving receiver and source are always used and a moving one might also be interesting to study since many studies about noises also consider a moving receiver or source (for example the noise that came from traffics [25]).

In this study, a forward model has always been used since the input distribution of uncertain parameters are known. There actually exist other uncertainty quantification methods that can be tested such as Bayesian inversion which is a backward model or other gPCE methods such as point collocation or stochastic Galerkin approach.

Appendix

A. Detailed Descriptions

Parameter	Symbol	S_i	S_T
Speed of sound	cexp	$6.82\text{E}-09$	$4.14\text{E}-12$
Vertical gradient of the speed of sound	cexpz	$0.00\text{E}+00$	$0.00\text{E}+00$
Temperature	texp	$1.26\text{E}-03$	$1.73\text{E}-03$
Vertical gradient of the temperature	texpz	$6.85\text{E}-03$	$9.10\text{E}-03$
Relative humidity	rfexp	$2.57\text{E}-05$	$1.75\text{E}-05$
Wind speed at $10\ \text{m}$ height	uexp	$3.48\text{E}-01$	$4.28\text{E}-01$
Wind direction	vexp	$-1.27\text{E}-05$	$1.89\text{E}-06$
Ground roughness	z0exp	$1.05\text{E}-02$	$1.91\text{E}-02$
Air resistance on the ground	sigexp	$0.00\text{E}+00$	$0.00\text{E}+00$
Topography complexity	τ	$5.48\text{E}-01$	$6.33\text{E}-01$

Table A.1.: Table of first order and total order Sobol' sensitivity indices of all used meteorological input parameters of AKUMET including topography that are obtained using Monte Carlo method and Jansen/Saltelli approximation [22, 31].

Parameter	Symbol	S_i	S_T
Speed of sound	cexp	$2.14\text{E}-08$	$7.99\text{E}-12$
Vertical gradient of the speed of sound	cexpz	$0.00\text{E}+00$	$0.00\text{E}+00$
Temperature	texp	$3.89\text{E}-03$	$4.47\text{E}-03$
Vertical gradient of the temperature	texpz	$2.56\text{E}-02$	$3.03\text{E}-02$
Relative humidity	rfexp	$7.77\text{E}-05$	$7.32\text{E}-05$
Wind speed at $10\ \text{m}$ height	uexp	$9.28\text{E}-01$	$9.37\text{E}-01$
Wind direction	vexp	$-3.91\text{E}-05$	$1.06\text{E}-05$
Ground roughness	z0exp	$3.25\text{E}-02$	$3.90\text{E}-02$
Air resistance on the ground	sigexp	$0.00\text{E}+00$	$0.00\text{E}+00$

Table A.2.: Table of first order and total order Sobol' sensitivity indices (right) of all used meteorological input parameters of AKUMET that are obtained using Monte Carlo method and Jansen/Saltelli approximation [22, 31].

Parameter	Symbol	Value	Unit
Speed of sound	cexp	340	$\mathrm{m/s}$
Vertical gradient of the speed of sound	cexpz	5	$1/\mathrm{s}$
Temperature	texp	18	$^{\circ}\mathrm{C}$
Vertical gradient of the temperature	texpz	2	$\mathrm{K/km}$
Relative humidity	rfexp	60	$\%$
Wind speed at $10\ \mathrm{m}$ height	uexp	RV	$\mathrm{m/s}$
Ground roughness	z0exp	0.3	M
Air resistance on the ground	sigexp	300	$\mathrm{KPaS}M^{-2}$

Table A.3.: Chosen meteorological parameters when uncertainty quantification with only the speed of wind as the parameter with uncertainty is done.

B. Complementary Results

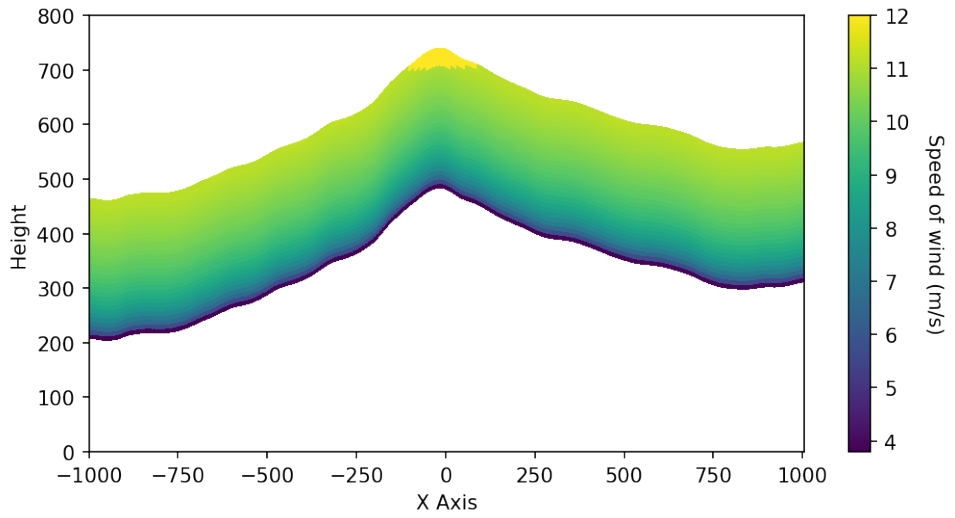

Figure B.1.: Plot of the profile of the speed of wind at every point of the domain when the sample generator set the speed of wind at 10 meters (*uexp*) to 4.21 m/s and at other height is calculated logarithmically.

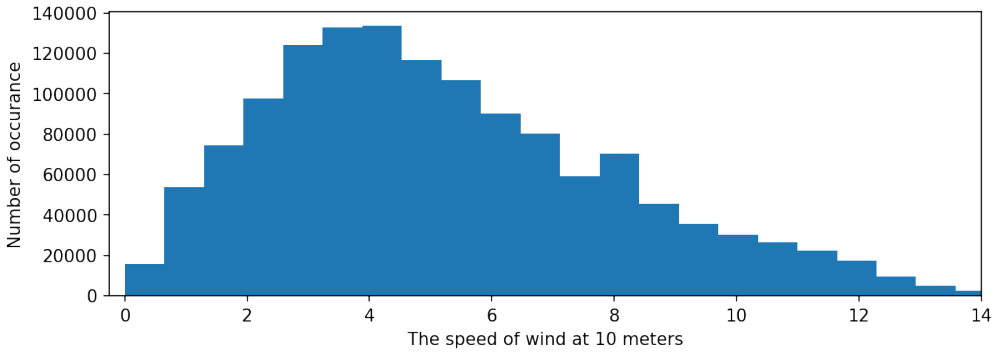

Figure B.2.: Data of wind speed at Vale do Cobrão [10].

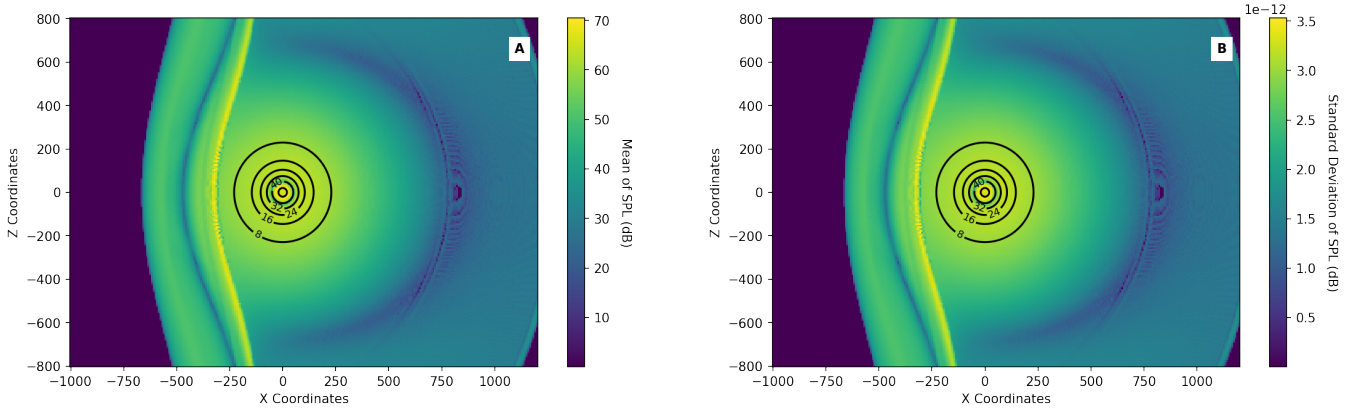

Figure B.3.: Plot of mean values of SPL (100 Hz frequency) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 1 topography.

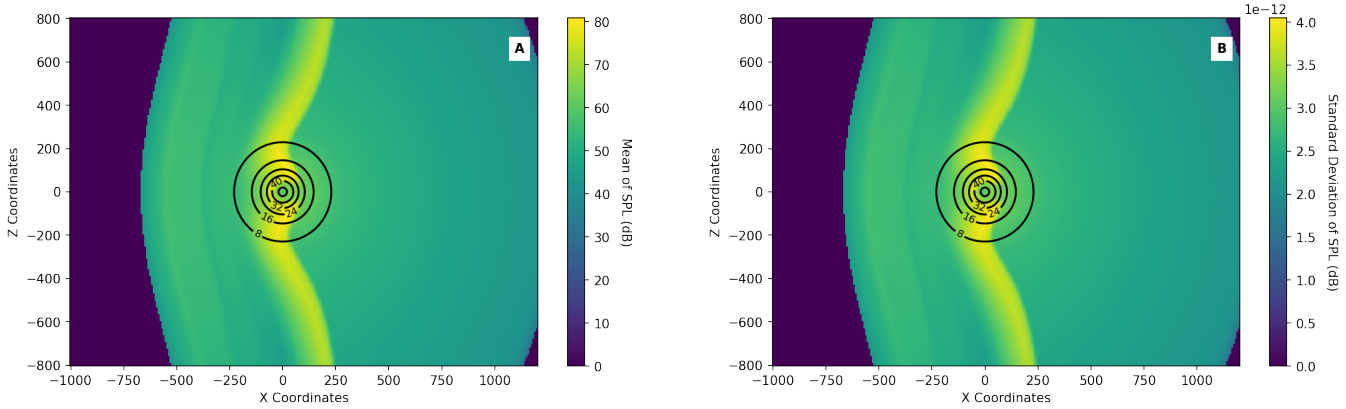

Figure B.4.: Plot of mean values of SPL (Whole third octave band) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 1 topography.

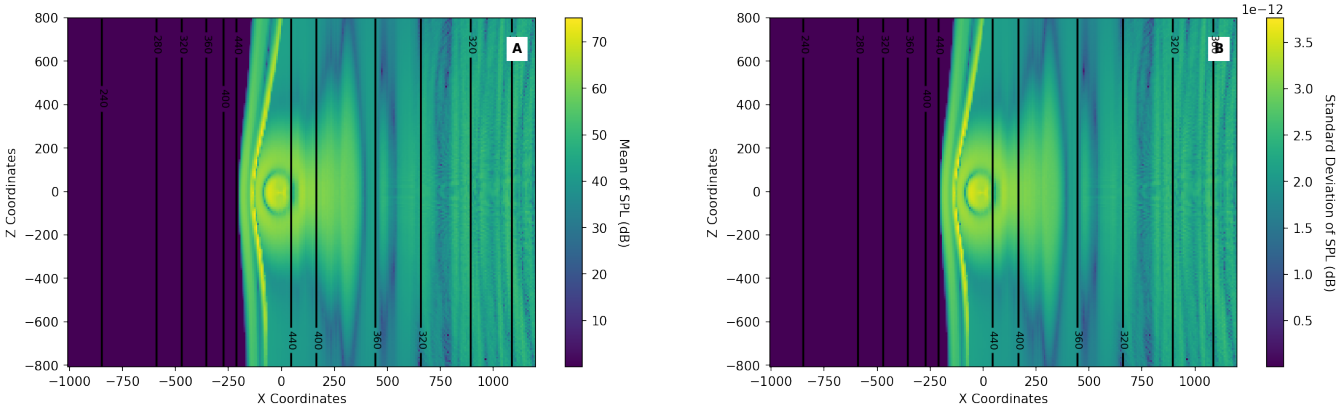

Figure B.5.: Plot of mean values of SPL (100 Hz frequency) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 2 topography.

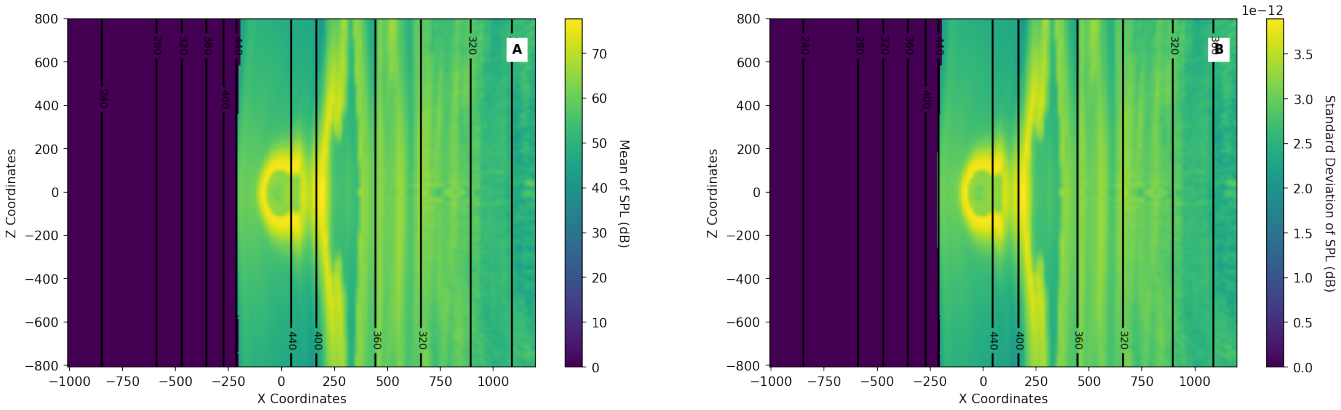

Figure B.6.: Plot of mean values of SPL (whole third octave band) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 2 topography.

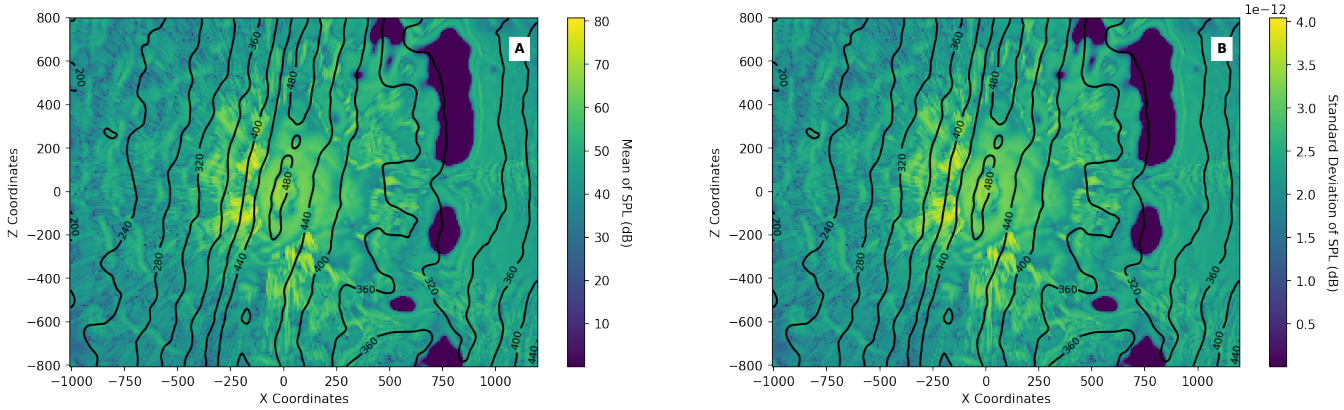

Figure B.7.: Plot of mean values of SPL (100 Hz frequency) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 3 (Vale do Cobrão) topography.

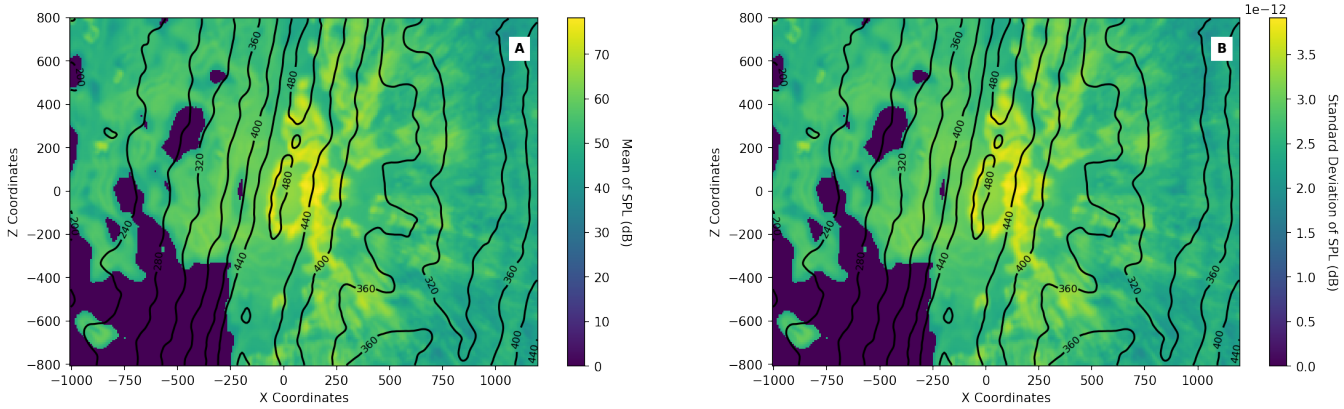

Figure B.8.: Plot of mean values of SPL (whole third octave band) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 3 (Vale do Cobrão) topography.

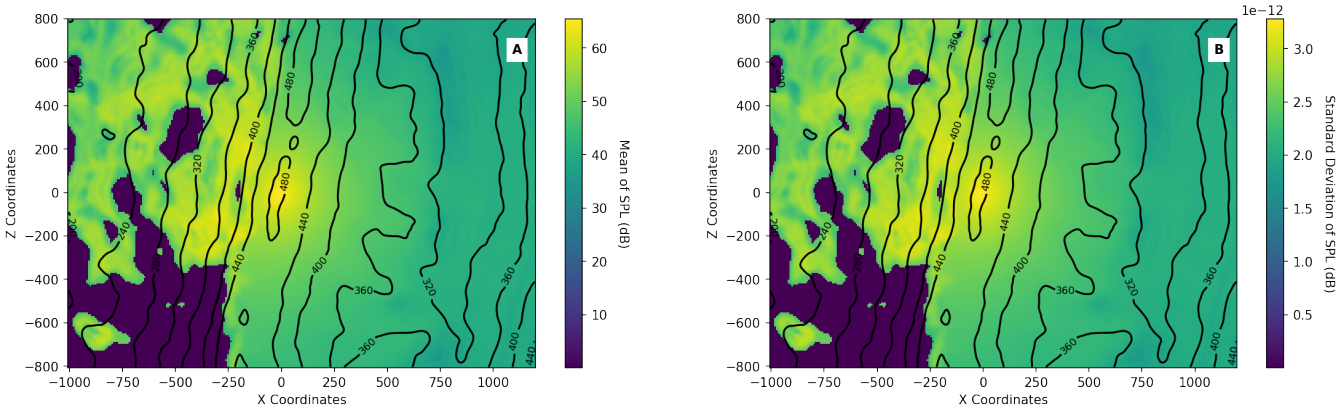

Figure B.9.: Plot of mean values of SPL (whole third octave band) on every coordinates (A) and standard deviation of SPL on every coordinates (B) when the speed of wind at 10 meters is uncertain ($\mathcal{N}(5.248, 2.8639)$) as a results of gPCE uncertainty quantification with 20 samples at level 3 (Vale do Cobrão) topography but with ground boundary condition set into total absorption.

Glossary

dew point temperature at which air must be cooled to become saturated with water vapor, with the assumption of constant air pressure and water content. 1

down-wind in the same direction to that in which the wind is blowing; with the wind. 40

geostrophic wind a wind which is resulted from an exact balance between the Coriolis force and the pressure gradient force. 1

roughness length a parameter of a vertical wind profile equations that model the horizontal mean wind speed nearby ground. 1

up-wind in the opposite direction to that in which the wind is blowing; against the wind. 38

Bibliography

[1] Milton Abramowitz and Irene A Stegun. Handbook of mathematical functions with formulas, graphs, and mathematical tables. In *Appl*, 1972.

[2] Reinhard Blumrich and Dietrich Heimann. A linearized eulerian sound propagation model for studies of complex meteorological effects. *The Journal of the Acoustical Society of America*, 112(2):446–455, August 2002.

[3] Mary L Boas. *Mathematical methods in the physical sciences*. John Wiley & Sons, 2006.

[4] Ivana Jovanovic Buha, Florian Künzner, Tobias Neckel, and Hans-Joachim Bungartz. Efficient uncertainty quantification and global time-varying sensitivity analysis of conceptual hydrological model. In *SIAM Conference on Computational Science and Engineering (CSE21)*, Fort Worth, Texas, U.S.A., Mar 2021. SIAM.

[5] Jef Caers. *Modeling uncertainty in the earth sciences*. John Wiley & Sons, 2011.

[6] Katharina Maria Elsen and Arthur Schady. Influence of meteorological conditions on sound propagation of a wind turbine in complex terrain. In *Proceedings of Meetings on Acoustics LRSP*, volume 41, page 032001. Acoustical Society of America, 2020.

[7] Tony FW Embleton. Tutorial on sound propagation outdoors. *The Journal of the Acoustical Society of America*, 100(1):31–48, 1996.

[8] Jonathan Feinberg, Vinzenz Gregor Eck, and Hans Petter Langtangen. Multivariate polynomial chaos expansions with dependent variables. *SIAM Journal on Scientific Computing*, 40(1):A199–A223, 2018.

[9] Jonathan Feinberg and Hans Petter Langtangen. Chaospy: An open source tool for designing methods of uncertainty quantification. *Journal of Computational Science*, 11:46–57, 2015.

[10] HJS Fernando, Jakob Mann, JMLM Palma, Julie K Lundquist, Rebecca Jane Barthelmie, M Belo-Pereira, WOJ Brown, FK Chow, Thomas Gerz, CM Hocut, et al. The perdigao: Peering into microscale details of mountain winds. *Bulletin of the American Meteorological Society*, 100(5):799–819, 2019.

[11] Behzad Ghanbarian, Allen G Hunt, Robert P Ewing, and Muhammad Sahimi. Tortuosity in porous media: a critical review. *Soil science society of America journal*, 77(5):1461–1477, 2013.

[12] Micha Gorelick and Ian Ozsvald. *High Performance Python: Practical Performant Programming for Humans*. O'Reilly Media, 2020.

[13] Marion Gödel. *Systematic parameter analysis to reduce uncertainty in crowd simulations*. Dissertation, Technische Universität München, München, 2022.

[14] Stephanie Heath and Gerry McAninch. Propagation effects of wind and temperture on acoustic ground contour levels. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, page 411, 2006.

[15] Dietrich Heimann. Modelling sound propagation from a wind turbine under various atmospheric conditions. *Meteorologische Zeitschrift*, 27(4):265–275, 2018.

[16] Dietrich Heimann, Antonia Englberger, and Arthur Schady. Sound propagation through the wake flow of a hilltop wind turbine—a numerical study. *Wind Energy*, 21(8):650–662, 2018.

[17] Dietrich Heimann and Günter Gross. Coupled simulation of meteorological parameters and sound level in a narrow valley. *Applied Acoustics*, 56(2):73–100, 1999.

[18] John L Hennessy and David A Patterson. *Computer architecture: a quantitative approach*. Elsevier, 2011.

[19] Jon Herman and Will Usher. SALib: An open-source python library for sensitivity analysis. *The Journal of Open Source Software*, 2(9), jan 2017.

[20] Serhat Hosder, Robert Walters, and Michael Balch. Efficient sampling for non-intrusive polynomial chaos applications with multiple uncertain input variables. In *48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, page 1939, 2007.

[21] Takuya Iwanaga, William Usher, and Jonathan Herman. Toward SALib 2.0: Advancing the accessibility and interpretability of global sensitivity analyses. *Socio-Environmental Systems Modelling*, 4:18155, May 2022.

[22] Michiel JW Jansen. Analysis of variance designs for model output. *Computer Physics Communications*, 117(1-2):35–43, 1999.

[23] JF Koksma. Een algemeene stelling uit de theorie der gelijkmatige verdeeling modulo 1. *Mathematica B (Zutphen)*, 11(7-11):43, 1942.

[24] Christian Lemieux. *Monte Carlo and Quasi-Monte Carlo Sampling*. Springer, New York, NY, USA, 2009.

[25] Manfred Liepert, Viktor Skowronek, Geske Eberlei, Martin Crljenkovic, Sarah Müller, Arthur Schady, and Katharina Elsen. *Aufbau einer Datenbank zur Berechnung exemplarischer Lärmsituationen unter Einbeziehung von Geräuschemissionsdaten des Verkehrsträgers Straße und meteorologischer Daten*. 2021.

[26] T. Neckel. *Lecture notes in Algorithms for Uncertainty Quantification*. TUM School of Computation, Information and Technology, Department of Computer Science, Technical University of Munich, 2022.

[27] Martina Novelinkova. Comparison of clenshaw-curtis and gauss quadrature. In *WDS*, volume 11, pages 67–71, 2011.

[28] Roger A Pielke Sr. *Mesoscale meteorological modeling*. Academic press, 2013.

[29] Allan Pierce. *Acoustics: An Introduction to Its Physical Principles and Applications*, volume 34. 06 1989.

[30] Simon Rahn. Uncertainty quantification in microscopic crowd simulation based on polynomial chaos expansions. Master's thesis, Hochschule für Angewandte Wissenschaften München, 2020.

[31] Andrea Saltelli, Paola Annoni, Ivano Azzini, Francesca Campolongo, Marco Ratto, and Stefano Tarantola. Variance based sensitivity analysis of model output. design and estimator for the total sensitivity index. *Computer physics communications*, 181(2):259–270, 2010.

[32] Andrea Saltelli, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, Michaela Saisana, and Stefano Tarantola. *Global sensitivity analysis: the primer*. John Wiley & Sons, 2008.

[33] Ralph C Smith. *Uncertainty quantification: theory, implementation, and applications*, volume 12. Siam, 2013.

[34] Ilya M Sobol and Sergei S Kucherenko. On global sensitivity analysis of quasi-monte carlo algorithms. 2005.

[35] IM Sobol and Yu L Levitan. On the use of variance reducing multipliers in monte carlo computations of a global sensitivity index. *Computer Physics Communications*, 117(1):52–61, 1999.

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

[37] William T Vetterling, William H Press, Saul A Teukolsky, and Brian P Flannery. *Numerical recipes example book (c++): The art of scientific computing*. Cambridge University Press, 2002.

[38] Jörg Waldvogel. Fast construction of the fejér and clenshaw–curtis quadrature rules. *BIT Numerical Mathematics*, 46(1):195–202, 2006.

[39] Eric W Weisstein. Hermite polynomial. *https://mathworld. wolfram. com/*, 2002.

[40] Dongbin Xiu. Numerical methods for stochastic computations. In *Numerical Methods for Stochastic Computations*. Princeton university press, 2010.

[41] Giancarlo Zaccone. *Python parallel programming cookbook*. Packt Publishing Ltd, 2015.