

RESEARCH ARTICLE  

Given-data probabilistic fatigue assessment for offshore wind turbines using Bayesian quadrature

Elias Fekhari^{1,2} , Vincent Chabridon¹ , Joseph Muré¹ and Bertrand Iooss^{1,2,3}

¹EDF R&D, 6 Quai Watier, Chatou 78401, France

²Université Côte d'Azur, 28 Avenue de Valrose, Nice 06103, France

³GdR MASCOT-NUM - Méthodes d'Analyse Stochastique des Codes et Traitements Numériques, Paris, France

Corresponding author: Elias Fekhari; Email: elias.fekhari@gmail.com

Received: 21 March 2023; **Revised:** 17 October 2023; **Accepted:** 14 November 2023

Keywords: design of experiments; fatigue loads; kernel herding; uncertainty propagation; wind turbine

Abstract

Offshore wind turbines intend to take a rapidly growing share in the electric mix. The design, installation, and exploitation of these industrial assets are regulated by international standards, providing generic guidelines. Constantly, new projects reach unexploited wind resources, pushing back installation limits. Therefore, turbines are increasingly subject to uncertain environmental conditions, making long-term investment decisions riskier (at the design or end-of-life stage). Fortunately, numerical models of wind turbines enable to perform accurate multi-physics simulations of such systems when interacting with their environment. The challenge is then to propagate the input environmental uncertainties through these models and to analyze the distribution of output variables of interest. Since each call of such a numerical model can be costly, the estimation of statistical output quantities of interest (e.g., the mean value, the variance) has to be done with a restricted number of simulations. To do so, the present paper uses the kernel herding method as a sampling technique to perform Bayesian quadrature and estimate the fatigue damage. It is known from the literature that this method guarantees fast and accurate convergence together with providing relevant properties regarding subsampling and parallelization. Here, one numerically strengthens this fact by applying it to a real use case of an offshore wind turbine operating in Teesside, UK. Numerical comparison with crude and quasi-Monte Carlo sampling demonstrates the benefits one can expect from such a method. Finally, a new Python package has been developed and documented to provide quick open access to this uncertainty propagation method.

Impact Statement

Wind energy companies constantly deploy cutting-edge wind turbines to reach unexploited wind resources. In these highly uncertain environments, numerical models simulating such systems can be used to assess the performance and reliability of wind turbines. This numerical procedure is often computationally expensive. However, promising new kernel-based techniques aim at simplifying such a computational burden. This paper demonstrates the use of a fast sampling technique on the numerical model simulating a real offshore wind turbine operating in Teesside, UK. This sampling method conveniently extracts from a measured dataset a small number of relevant input points (i.e., environmental conditions) in order to use them as simulation inputs. Finally, this method is compatible with the intensive use of high-performance computer facilities.

  This research article was awarded Open Data and Open Materials badges for transparent practices. See the Data Availability Statement for details.

© The Author(s), 2024. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.



1. Introduction

As a sustainable and renewable energy source, offshore wind turbines (OWT) are likely to take a growing share of the global electric mix. However, to be more cost-effective, wind farm projects tend to move further from the coast, exploiting stronger and steadier wind resources. Going further offshore, wind turbines are subject to more severe and uncertain environmental conditions (i.e., wind and waves). In such conditions, their structural integrity should be certified. To do so, numerical simulation and probabilistic tools have to be used. In fact, according to Graf et al. (2016), for new environmental conditions or new turbine models, international standards such as IEC (2019) from the International Electrotechnical Commission and DNV-GL (2016b) from Det Norske Veritas recommend performing over 2×10^5 simulations distributed over a grid. However, numerical simulations are computed by a costly hydro-servo-aero-elastic wind turbine model, making the design process time-consuming. In the following, the simulated output cyclic loads studied are aggregated over the simulation period to assess the mechanical fatigue damage at hot spots of the structure. To compute the risks associated with wind turbines throughout their lifespan, one can follow the steps of the universal framework for the treatment of uncertainties (de Rocquigny et al., 2008) presented in Figure 1. After specifying the problem (Step A), one can quantify the uncertainties related to site-specific environmental conditions represented by the random vector $\mathbf{X} \in \mathcal{D}_{\mathbf{X}} \subset \mathbb{R}^p$, $p \in \mathbb{N}^*$ (Step B). Then, one can propagate them through the OWT simulation model (Step C) denoted by $g: \mathcal{D}_{\mathbf{X}} \rightarrow \mathbb{R}$, $\mathbf{X} \mapsto Y = g(\mathbf{X})$, and estimate a relevant quantity of interest $\psi(Y) = \psi(g(\mathbf{X}))$ (e.g., a mean, a quantile, a failure probability). An accurate estimation of the quantity of interest $\psi(Y)$ relies on both a relevant quantification of the input uncertainty and an efficient sampling method.

Regarding Step B, when dealing with uncertain environmental conditions, a specific difficulty often arises from the complex dependence structure such variables may exhibit. Here, two cases may occur: either measured data are directly available (i.e., the “given-data” context) or a theoretical parametric form for the joint input probability distribution can be postulated. Such existing parametric joint distributions often rely on prior data fitting combined with expert knowledge. For example, several parametric approaches have been proposed in the literature to derive such formulations, ranging from fitting conditional distributions (Vanem et al., 2023) to using vine copulas (Li and Zhang, 2020). However, when a considerable amount of environmental data is available, nonparametric approaches may be useful, even if fitting a joint probability distribution with a complex dependence structure may be a challenging task. In this case, the idea is to directly use the data as an empirical representation of input uncertainties in order to avoid a parametric fit that might be too restrictive in terms of dependence structure.

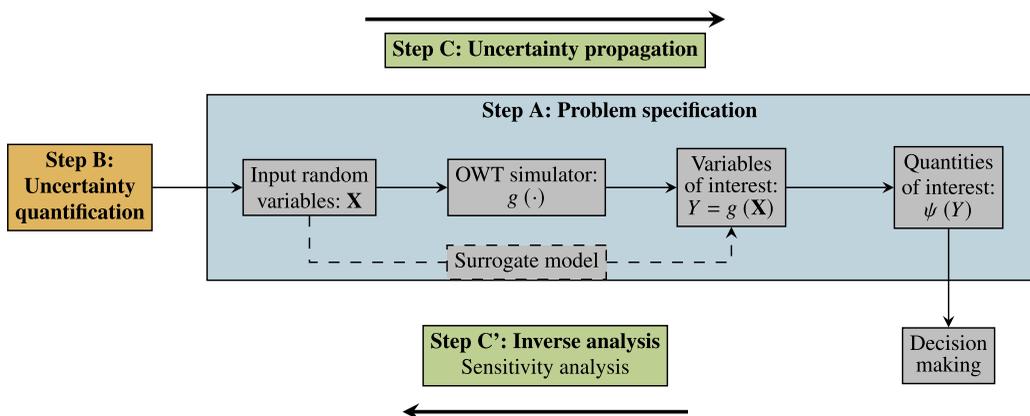


Figure 1. General framework for uncertainty quantification (scheme adapted from the one proposed by Ajenjo, 2023, originally introduced in de Rocquigny et al., 2008).

Step C usually focuses on propagating the input uncertainties in order to estimate the quantity of interest. Depending on the nature of $\psi(Y)$, one often distinguishes between two types of uncertainty propagation: a central tendency estimation (e.g., focusing on the output mean value or the variance) and a tail estimation (e.g., focusing on a high-order quantile or a failure probability). When uncertainty propagation aims at central tendency estimation, the usual methods can be split into two groups. First, those relying on sampling, that is, mainly Monte Carlo sampling (Graf et al., 2016), quasi-Monte Carlo sampling (Müller and Cheng, 2018), geometrical subsampling (Kanner et al., 2018), or deterministic quadrature rules (Van den Bos, 2020). All these methods estimate the quantity directly on the numerical simulator's outputs. Second, those that rely on the use of surrogate models (or metamodels, see Figure 1) to emulate the costly numerical model by a statistical model. Among a large panel of surrogates, one can mention, regarding wind energy applications, the use of polynomial chaos expansions (Dimitrov et al., 2018; Murcia et al., 2018), Gaussian process (GP) regression (Huchet, 2018; Teixeira et al., 2019a; Slot et al., 2020; Wilkie and Galasso, 2021), or artificial neural networks (Bai et al., 2023). When uncertainty propagation aims at studying the tail of the output distribution such as in risk or reliability assessment, one usually desires to estimate a quantile or a failure probability. In the wind energy literature, failure probability estimation has been largely studied, for example, in time-independent reliability assessment (Zwick and Muskulus, 2015; Slot et al., 2020; Wilkie and Galasso, 2021) or regarding time-dependent problems (Abdallah et al., 2019; Lataniotis, 2019).

During the overall process described in Figure 1, modelers and analysts often need to determine whether inputs are influential or not in order to prioritize their effort (in terms of experimental data collecting, simulation budget, or expert elicitation). Sometimes, they want to get a better understanding of the OWT numerical models' behavior or to enhance the input uncertainty modeling. All these questions are intimately related to the topic of sensitivity analysis (Saltelli et al., 2008; Da Veiga et al., 2021) and can be seen as an "inverse analysis" denoted by Step C' in Figure 1. In the wind energy literature, one can mention, among others, some works related to Spearman's rank correlation analysis and the use of the Morris method in Velarde et al. (2019) and Petrovska (2022). Going to variance-based analysis, the direct calculation of Sobol' indices after fitting a polynomial chaos surrogate model has been proposed in many works (e.g., in Murcia et al., 2018) while the use of distributional indices (e.g., based on the Kullback–Leibler divergence) has been investigated by Teixeira et al. (2019b).

The present paper focuses on the problem of uncertainty propagation, and more specifically, on the mean fatigue damage estimation (i.e., $\psi(Y) = \mathbb{E}[g(\mathbf{X})]$). Such a problem is usually encountered, by engineers, during the design phase. Most of the time, current standards as well as common engineering practices make them use regular grids (Huchet, 2018). Altogether, one can describe three alternative strategies: (i) direct sampling on the numerical model (e.g., using Monte Carlo), (ii) sampling on a static surrogate model (e.g., using GP regression), or (iii) using an "active learning" strategy (i.e., progressively adding evaluations of the numerical model to enhance the surrogate model fitting process). In practice, fitting a surrogate model in the context of OWT fatigue damage can be challenging due to the nonlinearity of the code. Moreover, the surrogate model validation procedure complexifies the process. Finally, active learning strategies restrict the potential number of parallel simulations, which limits the use of HPC facilities. Thus, the main contribution of this paper is to explore different ways to propagate uncertainties by directly evaluating the numerical model (i.e., without any surrogate model) with a relevant tradeoff between computational cost and accuracy. In the specific context of wind turbine fatigue damage, this work shows how to propagate uncertainties arising from a complex input distribution through a costly wind turbine simulator. The proposed work consists of evaluating the advantages and limits of kernel herding (KH) as a tool for given-data, fast, and fully distributable uncertainty propagation in OWT simulators. Additionally, this sampling method is highly flexible, allowing one to complete an existing design of experiments. Such a property can be crucial in practice when the analyst is asked to include some specific points to the design (e.g., characteristic points describing the system's behavior required by experts or by standards, see Huchet, 2018).

The paper is organized as follows. Section 2 presents the industrial use case related to a wind farm operating in Teesside, UK. Then, Section 3 introduces the various kernel-based methods for central

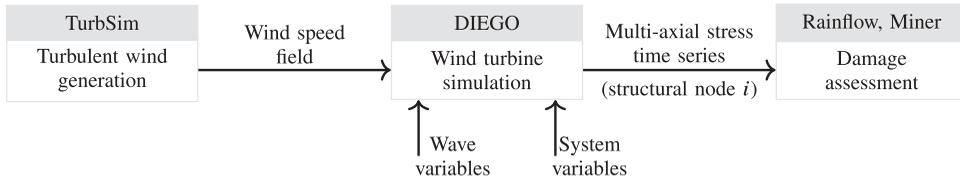


Figure 2. Diagram of the chained OWT simulation model.

tendency estimation. Section 4 analyzes the results of numerical experiments obtained on both analytical and industrial cases. Finally, the last section presents some discussions and draws some conclusions.

2. Treatment of uncertainties on the Teesside wind farm

An OWT is a complex system interacting with its environment. To simulate the response of this system against a set of environmental solicitations, multi-physics numerical models are developed. In the present paper, the considered use case consists of a chain of three numerical codes executed sequentially. As illustrated in Figure 2, a simulation over a time period is the sequence of, first, a turbulent wind speed field generation, then a wind turbine simulation (computing various outputs including mechanical stress), and finally, a post-processing phase to assess the fatigue damage of the structure.

2.1. Numerical simulation model

This subsection describes more precisely the modeling hypotheses considered in the industrial use case. The first block of the chain consists of a turbulent wind field simulator called “TurbSim” (developed by Jonkman (2009) from the National Renewable Energy Laboratory, USA) that uses, as a turbulence model, a Kaimal spectrum (Kaimal et al., 1972) (as recommended by the IEC, 2019). Moreover, to extrapolate the wind speed vertically, the shear is modeled by a power law. Since the wind field generation shows inherent stochasticity, each 10-min long simulation is repeated with different pseudorandom seeds and one averages the estimated damage over these repetitions. This question has been widely studied by some authors, (e.g., Slot et al., 2020), who concluded that the six repetitions recommended by the IEC (2019) may be insufficient to properly average this stochasticity. Thus, in the following, the simulations are repeated 11 times (picking an odd number also directly provides the empirical median value over the repetitions). This number of repetitions was chosen to suit the maximum number of simulations and the storage capacity of the generated simulations.

As a second block, one finds the “DIEGO” software (for “Dynamique Intégrée des Éoliennes et Génératrices Offshore¹”) which is developed by EDF R&D (Kim et al., 2022) to simulate the aero-hydro-servo-elastic behavior of OWTs. It takes the turbulent wind speed field generated by TurbSim as input and computes the dynamical behavior of the system (including the multiaxial mechanical stress at different nodes of the structure). For the application of interest here, the control system is modeled by the open-source DTU controller (Hansen and Henriksen, 2013), and no misalignment between the wind and the OWT is assumed. As for the waves, they are modeled in DIEGO using a JONSWAP spectrum (named after the 1975 Joint North Sea Wave Project). The considered use case here consists of a DIEGO model of a Siemens SWT 2.3 MW bottom-fixed turbine on a monopile foundation (see the datasheet in Table 1), currently operating in Teesside, UK (see the wind farm layout and wind turbine diagram in Figure 3). Although wind farms are subject to the wake effect, affecting the behavior and performance of some turbines in the farm, this phenomenon is not considered in this paper. To avoid numerical perturbations and reach the stability of the dynamical system, our simulation period is extended to 1000 s and the first 400 s are cropped in the post-processing step. This chained OWT numerical simulation model has been

¹ In English, “Integrated Dynamics of Wind Turbines and Offshore Generators.”

Table 1. Teesside OWT datasheet

Siemens SWT-2.3–93	
Rated power	2.3 MW
Rotor diameter	93 m
Hub height	83 m
Cut-in, cut-out wind speed	4 m/s, 25 m/s

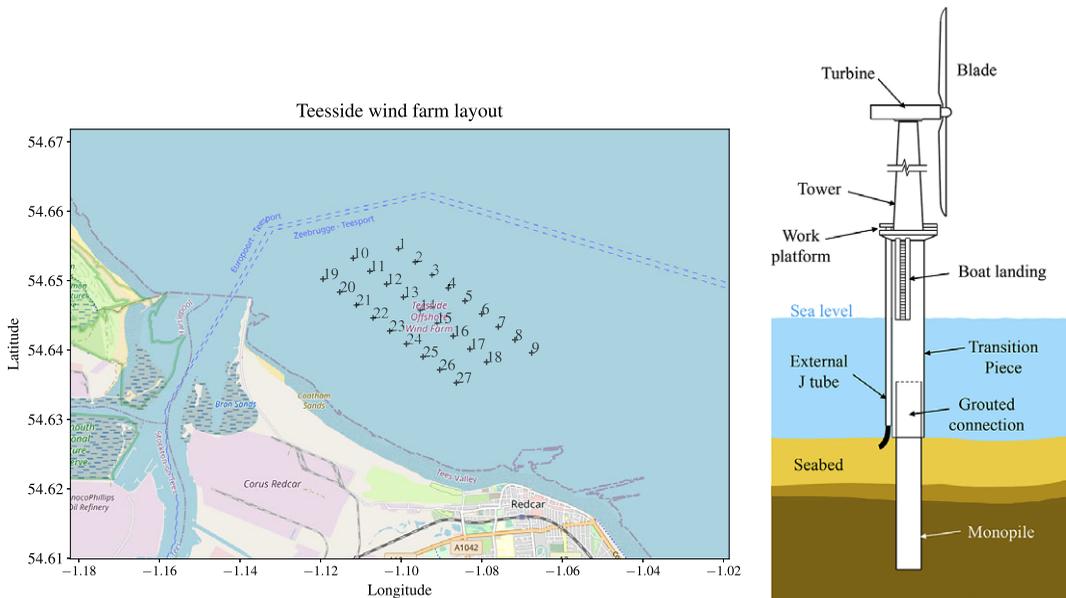


Figure 3. Teesside wind farm layout (left) and monopile offshore wind turbines (OWT) diagram from Chen et al. (2018) (right).

deployed on an EDF R&D HPC facility to benefit from parallel computing speed up (a single simulation on one CPU takes around 20 min).

2.2. Measured environmental data

During the lifespan of a wind farm project, environmental data are collected at different phases. In order to decide on the construction of a wind farm, meteorological masts, and wave buoys are usually installed on a potential site for a few years. After its construction, each wind turbine is equipped with monitoring instruments (e.g., cup anemometers). In total, 5 years of wind data have been collected on the turbines which are not affected by the wake on this site. Their acquisition system (usually called “Supervisory Control And Data Acquisition”) has a sampling period of 10 min. The wave data arise from a buoy placed in the middle of the farm. These data describe the physical features listed in Table 2. A limitation of the present study is that it controller-induced uncertainty (like wind misalignment) is not considered.

The Teesside farm is located close to the coast, making the environmental conditions very different depending on the direction (see the wind farm layout in Figure 3). Since measures are also subject to uncertainties, a few checks were made to ensure that the data were physically consistent. Truncation bounds were applied since this study is focused on central tendency estimation (i.e., mean behavior) rather than extreme values. In practice, this truncation only removes extreme data points (associated with storm events). In addition, a simple trigonometric transform is applied to each directional feature to take into

Table 2. Description of the environmental data

Variable	Notation	Unit	Description
Mean wind speed	U	ms^{-1}	10-min. average horizontal wind speed
Wind turbulence	σ_U	ms^{-1}	10-min. wind speed standard deviation
Wind direction ²	θ_{wind}	$^\circ$	10-min. average wind direction
Significant wave height	H_s	m	Significant wave height
Peak wave period	T_p	s	Peak 1-h spectral wave period
Wave direction	θ_{wave}	$^\circ$	10-min. average wave direction

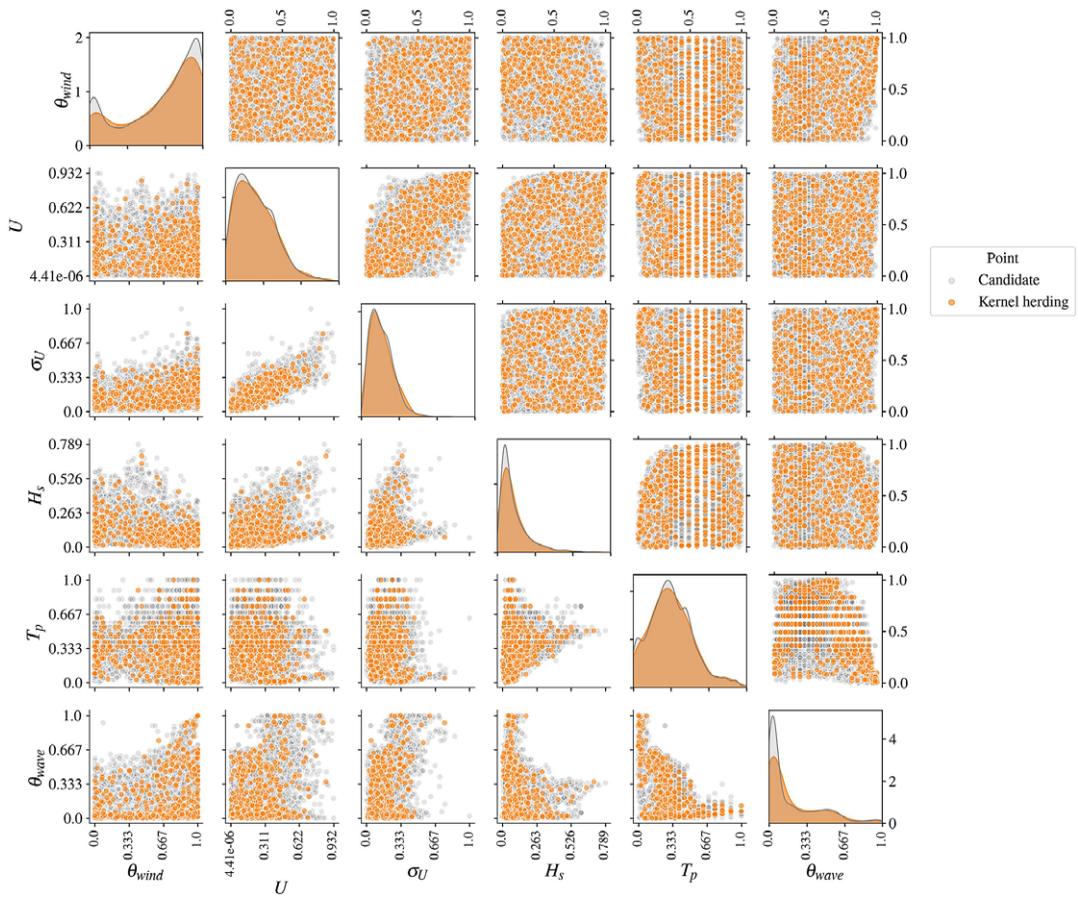


Figure 4. Copulogram of the Teesside measured data ($N = 10^4$ in gray) and kernel herding subsample ($n = 500$ in orange). Marginals are represented by univariate kernel density estimation plots (diagonal) and the dependence structure with scatterplots in the rank space (upper triangle). Scatterplots on the bottom triangle are set in the physical space.

account their cyclic structure. Finally, the remaining features are rescaled (i.e., using a min-max normalization).

The matrix plot of the transformed data presented in Figure 4 is (to the best of the authors’ knowledge) a new kind of plot named *copulogram*. A copulogram is a graphical tool that enables the analysis of the

²Note that the two directional variables could be replaced by a wind-wave misalignment variable for a bottom-fixed technology; however, our framework can be directly transposed to floating models.

probabilistic data structure into two parts: the marginal distributions (in the diagonal, in a similar fashion as a usual matrix plot) and the dependence structure between features. To do so, it represents the marginals with univariate kernel density estimation plots (in the diagonal), and the dependence structure with scatterplots in the rank space (in the upper triangle). Looking at data in the rank space instead of the initial space allows one to observe the ordinal associations between variables. The common practice is to normalize the ranks of each marginal between zero and one. Two independent variables will present a uniformly distributed scatterplot in the rank space. In the lower triangular matrix, the scatterplots are set in the physical space, merging the effects of the marginals and the dependencies (as in the usual visualization offered by the matrix plot). Since the dependence structure is theoretically modeled by an underlying copula, this plot is called *copulogram*, generalizing the well-known “correlogram” to nonlinear dependencies. It gives a synthetic and empirical decomposition of the dataset that was implemented in a new open-source Python package named *copulogram*³.

In Figure 4, a large sample $\mathcal{S} \subset \mathcal{D}_{\mathbf{X}}$ (with size $N = 10^4$) is randomly drawn from the entire Teesside data (with size $N_{\text{Teesside}} = 2 \times 10^5$) and plotted in gray. In the same figure, the orange matrix plot is a subsample of the sample \mathcal{S} , selected by KH, a method minimizing some discrepancy measure with the sample \mathcal{S} that will be presented in Section 3). For this example, generating the KH subsample takes less than 1 min, which is negligible compared with the simulation time of OWT models. Visually, this orange subsample seems to be representative of the original sample both in terms of marginal distributions and dependence structure. In the following study, the large samples \mathcal{S} will be considered as an empirical representation of the distribution of the random vector $\mathbf{X} \in \mathcal{D}_{\mathbf{X}}$, with probability density function $f_{\mathbf{X}}$, and called *candidate set*. KH allows direct subsampling from this large and representative dataset, instead of fitting a joint distribution and generating samples from it. Indeed, fitting a joint distribution would introduce an additional source of error in the uncertainty propagation process. Note that a proper parametric model fit would be challenging for complex dependence structures such as the one plotted in Figure 4. As examples of works that followed this path, one can mention the work of Li and Zhang (2020), who built a parametric model of a similar multivariate distribution using vine copulas.

For a similar purpose and to avoid some limits imposed by the parametric framework, a nonparametric approach coupling empirical Bernstein copula (EBC) fitting with kernel density estimation of the marginals is proposed in Section 2.3.

2.3. Nonparametric fit with EBC

Instead of directly subsampling from a dataset such as the one from Figure 4, one could first infer a multivariate distribution and generate a sample from it. However, accurately fitting such complex multivariate distributions is challenging. The amount of available data is large enough to make nonparametric inference a viable option.

The Sklar theorem (Joe, 1997) states that the multivariate distribution of any random vector $\mathbf{X} \in \mathbb{R}^p$, $p \in \mathbb{N}^*$ can be broken down into two objects:

1. A set of univariate marginal distributions to describe the behavior of the individual variables.
2. A function describing the dependence structure between all variables, called a *copula*.

This theorem states that considering a random vector $\mathbf{X} \in \mathbb{R}^p$, with its cumulative distribution function F and its marginals $\{F_i\}_{i=1}^p$, there exists a copula $C : [0, 1]^p \rightarrow [0, 1]$, such that:

$$F(x_1, \dots, x_p) = C(F_1(x_1), \dots, F_p(x_p)). \quad (1)$$

It allows us to divide the problem of fitting a joint distribution into two independent problems: fitting the marginals and fitting the copula. Note that, when the joint distribution is continuous, this copula is unique. Copulas are continuous and bounded functions defined on a compact set (the unit hypercube). Bernstein polynomials allow uniform approximation as closely as desired any continuous and real-valued function

³ <https://github.com/efekhari27/copulogram>.

defined on a compact set (Weierstrass approximation theorem). Therefore, they are good candidates to approximate unknown copulas. This concept was introduced as EBC by Sancetta and Satchell (2004) for applications in economics and risk management. Later on, Segers et al. (2017) offered further asymptotic studies. Formally, the multivariate Bernstein polynomial for a function $C : [0, 1]^p \rightarrow \mathbb{R}$ on a grid over the unit hypercube $G := \left\{ \frac{0}{h_1}, \dots, \frac{h_1}{h_1} \right\} \times \dots \times \left\{ \frac{0}{h_p}, \dots, \frac{h_p}{h_p} \right\}$, $\mathbf{h} = (h_1, \dots, h_p) \in \mathbb{N}^p$, is written as follows:

$$B_{\mathbf{h}}(C)(\mathbf{u}) := \sum_{t_1=0}^{h_1} \dots \sum_{t_p=0}^{h_p} C\left(\frac{t_1}{h_1}, \dots, \frac{t_p}{h_p}\right) \prod_{j=1}^p b_{h_j, t_j}(u_j), \tag{2}$$

with $\mathbf{u} = (u_1, \dots, u_p) \in [0, 1]^p$, and the Bernstein polynomial $b_{h,t}(u) := \frac{t!}{h!(h-t)!} u^t (1-u)^{h-t}$. When C is a copula, then $B_{\mathbf{h}}(C)$ is called ‘‘Bernstein copula.’’ Therefore, the EBC is an application of the Bernstein polynomial in Eq. (2) to the so-called ‘‘empirical copula.’’ In practice, considering a sample $\mathbf{X}_n = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\} \in \mathbb{R}^{np}$ and the associated ranked sample $\mathbf{R}_n = \{\mathbf{r}^{(1)}, \dots, \mathbf{r}^{(n)}\}$, the corresponding empirical copula is written:

$$C_n(\mathbf{u}) := \frac{1}{n} \sum_{i=0}^n \prod_{j=1}^p \left\{ \frac{r_j^{(i)}}{n} \leq u_j \right\}, \mathbf{u} = (u_1, \dots, u_p) \in [0, 1]^p. \tag{3}$$

Provided a large enough learning set \mathbf{X}_n (over 5 years in the present case), the EBC combined with kernel density estimation for the marginals enable to fit well the environmental joint distribution related to the dataset in Figure 4. Moreover, the densities of the EBC are available in an explicit form, making Monte Carlo or quasi-Monte Carlo generation easy. Nevertheless, this method is sensitive to the chosen polynomial orders $\{h_j\}_{j=1}^p$ and the learning set size. For a thorough presentation of this method, practical recommendations and theoretical results regarding EBC tuning, see the manuscript of Lasserre (2022). Further discussions and numerical experiments on the estimation of nonparametric copula models are presented in Nagler et al. (2017).

2.4. Fatigue assessment

As described in Figure 2, a typical DIEGO simulation returns a 10-min multiaxial stress time series at each node $i \in \mathbb{N}$ of the one-dimensional (1D) meshed structure. Since classical fatigue laws are established for uniaxial stresses, the first step is to compute one equivalent Von Mises stress time series at each structural node.

The foundation and the tower of an OWT are a succession of tubes with various sections connected by bolted or welded joints. Our work focuses on the welded joints at the mudline level, identified as a critical area for the structure. This hypothesis is confirmed in the literature by different contributions, see, for example, the results of Müller and Cheng (2018) in Figure 12, or Katsikogiannis et al. (2021). At the top of the turbine, the fatigue is commonly studied at the blade root, which was not studied here since the blades in composite material have different properties (see, e.g., Dimitrov, 2013). Note that the OWT simulations provide outputs allowing us to similarly study any node along the structure (without any additional computational effort).

To compute fatigue in a welded joint, the external circle of the welding ring is discretized for a few azimuth angles $\theta \in \mathbb{R}_+$ (see the red points in the monopile cross section on the right in Figure 5). The equivalent Von Mises stress time series is then reported on the external welding ring for an azimuth θ . According to Li and Zhang (2020) and our own experience, the most critical azimuth angles are roughly aligned with the main wind and wave directions (whose distributions are illustrated in Figure 5). Looking at these illustrations, the wind and wave conditions have a very dominant orientation, which is explained by the closeness of the wind farm to the shore. Then, it is assumed that azimuth angles in these directions will be more solicited, leading to higher fatigue damage. To assess fatigue damage, rainflow counting (Dowling, 1972) first identifies the stress cycles and their respective amplitudes (using the implementation of the ASTM E1049-85 rainflow cycle

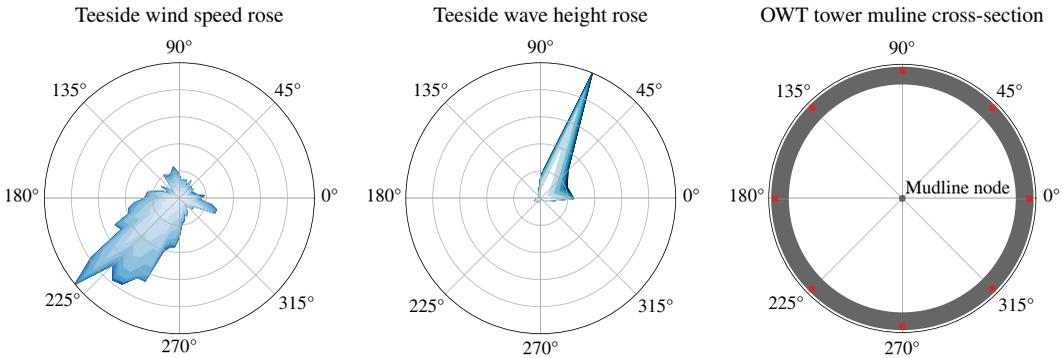


Figure 5. Angular distribution of the wind and waves with a horizontal cross section of the offshore wind turbines (OWT) structure and the mudline. Red crosses represent the discretized azimuths for which the fatigue is computed.

counting algorithm from the Python package *rainflow*⁴). For each identified stress cycle of amplitude $s \in \mathbb{R}_+$, the so-called “Stress versus Number of cycles” curve (also called the “SN curve” or “Wöhler curve”) allows one to estimate the number N_c of similar stress cycles necessary to reach fatigue ruin. The SN curve, denoted by $W(\cdot)$ is an affine function in the log–log scale with slope $-m$ and y-intercept a :

$$N_c := W(s) = as^{-m}, \quad a \in \mathbb{R}_+, \quad m \in \mathbb{R}_+. \tag{4}$$

Finally, a usual approach to compute the damage is to aggregate the fatigue contribution of each stress cycle identified using Miner’s rule. Damage occurring during a 10-min operating time is simulated and then scaled up to the OWT lifetime. More details regarding damage assessment and the Wöhler curve used are available in Section 2.4.6 from (DNV-GL, 2016a). For the realization $\mathbf{x} \in \mathcal{D}_X$ of environmental conditions, at a structural node i , an azimuth angle θ ; $k \in \mathbb{N}$ stress cycles of respective amplitude $\{s_{i,\theta}^{(j)}(\mathbf{x})\}_{j=1}^k$ are identified. Then, Miner’s rule (Fatemi and Yang, 1998) defines the damage function $g_{i,\theta}(\mathbf{x}) : \mathcal{D}_X \rightarrow \mathbb{R}_+$ by:

$$g_{i,\theta}(\mathbf{x}) = \sum_{j=1}^k \frac{1}{N_c^{(j)}} = \sum_{j=1}^k \frac{1}{W(s_{i,\theta}^{(j)}(\mathbf{x}))}. \tag{5}$$

As defined by the DNV standards for OWT fatigue design (DNV-GL, 2016a), the quantity of interest in the present paper is the “mean damage” $d_{i,\theta}$ (also called “expected damage”), computed at a node i , for an azimuth angle θ :

$$d_{i,\theta} = \mathbb{E}[g_{i,\theta}(\mathbf{X})] = \int_{\mathcal{D}_X} g_{i,\theta}(\mathbf{x})f_{\mathbf{x}}(\mathbf{x})d\mathbf{x}. \tag{6}$$

To get a preview of the distribution of this output random variable $g_{i,\theta}(\mathbf{X})$, a histogram of a large Monte Carlo simulation ($N_{\text{ref}} = 2000$) is represented in Figure 6 (with a log scale). In this case, the log-damage histogram presents a little asymmetry but it is frequently modeled by a normal distribution (see, e.g., Teixeira et al., 2019b).

⁴ <https://github.com/iamlikeme/rainflow>.

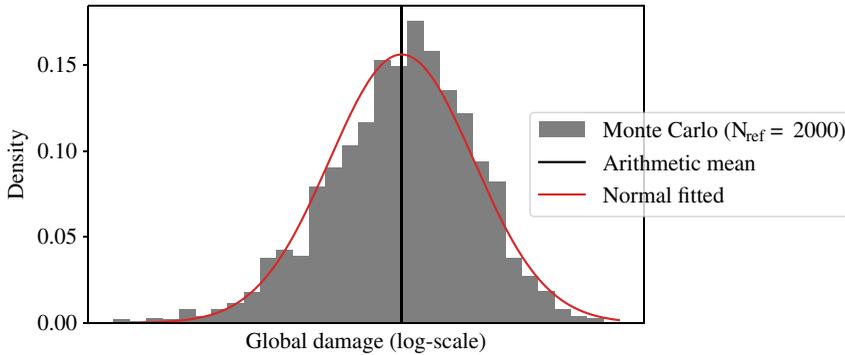


Figure 6. Histogram of the log-damage, at mudline, azimuth 45° (Monte Carlo reference sample).

3. Numerical integration procedures for mean damage estimation

The present section explores different methods aiming at approximating the integral of a function against a probability measure. In the case of OWT mean damage estimation, these methods can be used for defining efficient design load cases. This problem is equivalent to the central tendency estimation of $\mathbf{Y} = g(\mathbf{X})$, the image of the environmental random variable \mathbf{X} by the damage function $g(\cdot) : \mathcal{D}_X \rightarrow \mathbb{R}$ (see, e.g., Eq. (6)). Considering a measurable space $\mathcal{D}_X \subset \mathbb{R}^p$, $p \in \mathbb{N}^*$, associated with a known probability measure μ , this section studies the approximation of integrals of the form $\int_{\mathcal{D}_X} g(\mathbf{x}) d\mu(\mathbf{x})$.

3.1. Quadrature rules and quasi-Monte Carlo methods

Numerical integration authors may call this generic problem *probabilistic integration* (Briol et al., 2019). In practice, this quantity of interest is estimated on an n -sized set of damage realizations $\mathbf{y}_n = \{g(\mathbf{x}^{(1)}), \dots, g(\mathbf{x}^{(n)})\}$ of an input sample $\mathbf{X}_n = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$. A weighted arithmetic mean of the realizations $\{g(\mathbf{x}^{(1)}), \dots, g(\mathbf{x}^{(n)})\}$ is called a *quadrature rule* with a set of unconstrained weights $\mathbf{w}_n = \{w_1, \dots, w_n\} \in \mathbb{R}^n$:

$$I_\mu(g) := \int_{\mathcal{D}_X} g(\mathbf{x}) d\mu(\mathbf{x}) \approx \sum_{i=1}^n w_i g(\mathbf{x}^{(i)}) \tag{7}$$

Our numerical experiment framework often implies that the function g is costly to evaluate, making the realization number limited. For a given sample size n , our goal is to find a set of tuples $\{\mathbf{x}^{(i)}, w_i\}_{i=1}^n$ (i.e., quadrature rule), giving the best approximation of our quantity. In the literature, a large panel of numerical integration methods has been proposed to tackle this problem. For example, Van den Bos (2020) recently developed a quadrature rule based on arbitrary sample sets which was applied to a similar industrial OWT use case.

Alternatively, sampling methods rely on generating a set of points \mathbf{X}_n drawn from the input distribution to compute the arithmetic mean of their realizations (i.e., uniform weights $\{w_i = \frac{1}{n}\}_{i=1}^n$). Among them, low-discrepancy sequences, also called “quasi-Monte Carlo” sampling (e.g., Sobol’, Halton, Faure sequences) are known to improve the standard Monte Carlo convergence rate and will be used as a deterministic reference method in the following numerical experiments (Leobacher and Pillichshammer, 2014).

3.2. Kernel discrepancy

Quasi-Monte Carlo sampling methods widely rely on a uniformity metric, called *discrepancy*. This section first presents the link between discrepancy and numerical integration. Then, it introduces a kernel-based discrepancy, generalizing the concept to nonuniform measures. This tool is eventually used to build a sequential quadrature rule by subsampling from a finite dataset.

3.2.1. Quantization of probability measures and quadrature

When dealing with probabilistic integration such as Eq. (7), a quadrature rule is a finite representation of a continuous measure μ by a discrete measure $\zeta_n = \sum_{i=1}^n w_i \delta(\mathbf{x}^{(i)})$ (weighted sum of Dirac distributions at the design points \mathbf{X}_n). In the literature, this procedure is also called *quantization* of a continuous measure μ . Overall, numerical integration is a particular case of probabilistic integration against a uniform input measure. For uniform measures, the Koksma–Hlawka inequality (Morokoff and Caflisch, 1995) provides a useful upper bound on the absolute error of a quadrature rule:

$$\left| \int_{[0,1]^p} g(\mathbf{x}) d\mathbf{x} - \frac{1}{n} \sum_{i=1}^n g(\mathbf{x}^{(i)}) \right| \leq V(g) D_n^*(\mathbf{X}_n). \tag{8}$$

As presented in Oates (2021), $V(g) = \sum_{u \subseteq \{1, \dots, p\}} \int_{[0,1]^u} \left| \frac{\partial^u g}{\partial \mathbf{x}_u}(\mathbf{x}_u, 1) \right| d\mathbf{x}$ quantifies the complexity of the integrand, while $D_n^*(\mathbf{X}_n)$ evaluates the discrepancy to uniformity of the design \mathbf{X}_n . Therefore, the Koksma–Hlawka inequality shows that the quadrature rule’s accuracy relies on the good quantization of μ by \mathbf{X}_n . For a uniform measure μ , the star discrepancy $D_n^*(\mathbf{X}_n)$ is a metric assessing how far from uniformity a sample \mathbf{X}_n is. When generalizing to a nonuniform measure, a good quantization of μ should also lead to a good approximation of the quantity.

3.2.2. Reproducing kernel Hilbert space and kernel mean embedding

To generalize the Koksma–Hlawka inequality to any probability measure, let us assume that the integrand g lives in a specific function space $\mathcal{H}(k)$. $\mathcal{H}(k)$ is a *reproducing kernel Hilbert space* (RKHS), which is an inner product space of functions $g: \mathcal{D}_X \rightarrow \mathbb{R}$. Considering a symmetric and positive definite function $k: \mathcal{D}_X \times \mathcal{D}_X \rightarrow \mathbb{R}$, later called a “reproducing kernel” or simply a “kernel,” an RKHS verifies the following axioms:

- The “feature map” $\phi: \mathcal{D}_X \rightarrow \mathcal{H}(k); \phi(\mathbf{x}) = k(\cdot, \mathbf{x})$ belongs to the RKHS: $k(\cdot, \mathbf{x}) \in \mathcal{H}(k), \forall \mathbf{x} \in \mathcal{D}_X$.
- The “reproducing property”: $\langle g, k(\cdot, \mathbf{x}) \rangle_{\mathcal{H}(k)} = g(\mathbf{x}), \forall \mathbf{x} \in \mathcal{D}_X, \forall g \in \mathcal{H}(k)$.

Note that it can be shown that every positive semi-definite kernel defines a unique RKHS (and vice versa) with a feature map ϕ , such that $k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}(k)}$. This framework allows us to embed a continuous or discrete probability measure in an RKHS, as illustrated in Figure 7. For any measure μ , let us define its *kernel mean embedding* (Sejdinovic et al., 2013), also called “potential” $P_\mu(\mathbf{x})$ in Pronzato and Zhigljavsky (2020), associated with the kernel k as:

$$P_\mu(\mathbf{x}) := \int_{\mathcal{D}_X} k(\mathbf{x}, \mathbf{x}') d\mu(\mathbf{x}'). \tag{9}$$

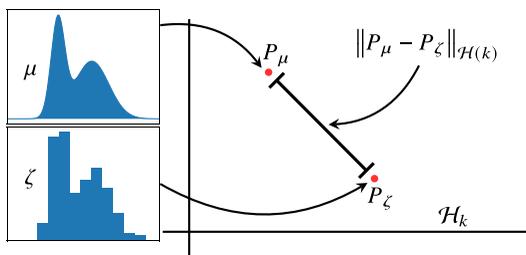


Figure 7. Kernel mean embedding of a continuous and discrete probability distribution.

Respectively, the potential $P_{\zeta_n}(\mathbf{x})$ of a discrete distribution $\zeta_n = \sum_{i=1}^n w_i \delta(\mathbf{x}^{(i)})$, $w_i \in \mathbb{R}_+$ associated with the kernel k can be written as:

$$P_{\zeta_n}(\mathbf{x}) = \int_{\mathcal{D}_X} k(\mathbf{x}, \mathbf{x}') d\zeta_n(\mathbf{x}') = \sum_{i=1}^n w_i k(\mathbf{x}, \mathbf{x}^{(i)}) \tag{10}$$

The potential $P_\mu(\mathbf{x})$ of the targeted measure μ will be referred to as ‘‘target potential’’ and the potential $P_{\zeta_n}(\mathbf{x})$ associated with the discrete distribution ζ_n called ‘‘current potential’’ when its support is the current design \mathbf{X}_n . When $P_{\zeta_n}(\mathbf{x})$ is close to $P_\mu(\mathbf{x})$, it can be interpreted as ζ_n being an adequate quantization or representation of μ (which leads to a good estimation of a quantity such as $I_\mu(g)$ from Eq. (7)). Potentials can be computed in closed forms for specific pairs of distribution and associated kernel. Summary tables of some of these cases are detailed in Briol (2019) (Section 3.4), Pronzato and Zhigljavsky (2020) (Section 4), and extended in Fekhari et al. (2023). However, in most cases, the target potentials must be estimated on a large and representative sample, typically a large quasi-Monte Carlo sample of μ .

The energy of a measure μ is defined as the integral of the potential P_μ against the measure μ , which leads to the following scalar quantity:

$$e_\mu := \int_{\mathcal{D}_X} P_\mu(\mathbf{x}) d\mu(\mathbf{x}) = \iint_{\mathcal{D}_X^2} k(\mathbf{x}, \mathbf{x}') d\mu(\mathbf{x}) d\mu(\mathbf{x}') \tag{11}$$

Finally, using the reproducing property and writing the Cauchy–Schwarz inequality on the absolute quadrature error leads to the following inequality, similar to the Koksma–Hlawka inequality Eq. (8) (see Briol et al., 2019):

$$\left| \sum_{i=1}^n w_i g(\mathbf{x}^{(i)}) - \int_{\mathcal{D}_X} g(\mathbf{x}) d\mu(\mathbf{x}) \right| = \left| \langle g, P_{\zeta_n}(\mathbf{x}) \rangle_{\mathcal{H}(k)} - \langle g, P_\mu(\mathbf{x}) \rangle_{\mathcal{H}(k)} \right| \tag{12a}$$

$$= \left| \langle g, (P_{\zeta_n}(\mathbf{x}) - P_\mu(\mathbf{x})) \rangle_{\mathcal{H}(k)} \right| \tag{12b}$$

$$\leq \|g\|_{\mathcal{H}(k)} \|P_\mu(\mathbf{x}) - P_{\zeta_n}(\mathbf{x})\|_{\mathcal{H}(k)} \tag{12c}$$

3.2.3. Maximum mean discrepancy

A metric of discrepancy and quadrature error is offered by the *maximum mean discrepancy* (MMD). This distance between two probability distributions μ and ζ is given by the worst-case error for any function within a unit ball of the Hilbert space $\mathcal{H}(k)$, associated with the kernel k :

$$\text{MMD}_k(\mu, \zeta) := \sup_{\|g\|_{\mathcal{H}(k)} \leq 1} \left| \int_{\mathcal{D}_X} g(\mathbf{x}) d\mu(\mathbf{x}) - \int_{\mathcal{D}_X} g(\mathbf{x}) d\zeta(\mathbf{x}) \right| \tag{13}$$

According to the inequality in Eq. (12c), $\text{MMD}_k(\mu, \zeta) = \|P_\mu - P_\zeta\|_{\mathcal{H}(k)}$, meaning that the MMD fully relies on the difference of potentials. Moreover, Sriperumbudur et al. (2010) define a kernel as ‘‘characteristic kernel’’ when the following equivalence is true: $\text{MMD}_k(\mu, \zeta) = 0 \Leftrightarrow \mu = \zeta$. This property makes the MMD a metric on \mathcal{D}_X . The squared MMD has been used for other purposes than numerical integration: for example, statistical testing (Gretton et al., 2006), global sensitivity analysis (Da Veiga, 2015), or clustering distributions (Lovera et al., 2023). It can be developed as follows:

$$\text{MMD}_k(\mu, \zeta)^2 = \|P_\mu(\mathbf{x}) - P_\zeta(\mathbf{x})\|_{\mathcal{H}(k)}^2 \tag{14a}$$

$$= \langle (P_\mu(\mathbf{x}) - P_\zeta(\mathbf{x})), (P_\mu(\mathbf{x}) - P_\zeta(\mathbf{x})) \rangle_{\mathcal{H}(k)} \tag{14b}$$

$$= \langle P_\mu(\mathbf{x}), P_\mu(\mathbf{x}) \rangle_{\mathcal{H}^{(k)}} - 2 \langle P_\mu(\mathbf{x}), P_\zeta(\mathbf{x}) \rangle_{\mathcal{H}^{(k)}} + \langle P_\zeta(\mathbf{x}), P_\zeta(\mathbf{x}) \rangle_{\mathcal{H}^{(k)}} \tag{14c}$$

$$= \iint_{\mathcal{D}_X^2} k(\mathbf{x}, \mathbf{x}') d\mu(\mathbf{x}) d\mu(\mathbf{x}') - 2 \iint_{\mathcal{D}_X^2} k(\mathbf{x}, \mathbf{x}') d\mu(\mathbf{x}) d\zeta(\mathbf{x}') + \iint_{\mathcal{D}_X^2} k(\mathbf{x}, \mathbf{x}') d\zeta(\mathbf{x}) d\zeta(\mathbf{x}'). \tag{14d}$$

Taking a discrete distribution with uniform weights $\zeta_n = \frac{1}{n} \sum_{i=1}^n \delta(\mathbf{x}^{(i)})$, the squared MMD reduces to:

$$\text{MMD}_k(\mu, \zeta_n)^2 = \varepsilon_\mu - \frac{2}{n} \sum_{i=1}^n P_\mu(\mathbf{x}^{(i)}) + \frac{1}{n^2} \sum_{i,j=1}^n k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}). \tag{15}$$

3.3. KH sampling

Herein, the MMD is used to quantize the known target measure μ by a design sample \mathbf{X}_n . For practical reasons, the design construction is done sequentially. Sequential strategies have also been used to learn and validate regression models for statistical learning (see Fekhari et al., 2023). Moreover, since each realization is supposed to be obtained at the same unitary cost, the quadrature weights are first fixed as uniform during the construction of the design \mathbf{X}_n .

KH, proposed by Chen et al. (2010), is a sampling method that offers a quantization of the measure μ by minimizing a squared MMD when adding points iteratively. With a current design \mathbf{X}_n and its corresponding discrete distribution with uniform weights $\zeta_n = \frac{1}{n} \sum_{i=1}^n \delta(\mathbf{x}^{(i)})$, a KH iteration is as an optimization of the following criterion, selecting the point $\mathbf{x}^{(n+1)} \in \mathcal{D}_X$:

$$\mathbf{x}^{(n+1)} \in \arg \min_{\mathbf{x} \in \mathcal{D}_X} \left\{ \text{MMD}_k \left(\mu, \frac{1}{n+1} \left(\delta(\mathbf{x}) + \sum_{i=1}^n \delta(\mathbf{x}^{(i)}) \right) \right)^2 \right\}. \tag{16}$$

In the literature, two formulations of this optimization problem can be found. The first one uses the Frank–Wolfe algorithm (or “conditional gradient algorithm”) to compute a linearization of the problem under the convexity hypothesis (see Lacoste-Julien et al. (2015) and Briol et al. (2015) for more details). The second one is a straightforward greedy optimization. Due to the combinatorial complexity, the greedy formulation is tractable for sequential construction only. Let us develop the MMD criterion from Eq. (15):

$$\text{MMD}_k \left(\mu, \frac{1}{n+1} \left(\delta(\mathbf{x}) + \sum_{i=1}^n \delta(\mathbf{x}^{(i)}) \right) \right)^2 = \varepsilon_\mu - \frac{2}{n+1} \sum_{i=1}^{n+1} P_\mu(\mathbf{x}^{(i)}) + \frac{1}{(n+1)^2} \sum_{i,j=1}^{n+1} k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \tag{17a}$$

$$= \varepsilon_\mu - \frac{2}{n+1} \left(P_\mu(\mathbf{x}) + \sum_{i=1}^n P_\mu(\mathbf{x}^{(i)}) \right) \tag{17b}$$

$$+ \frac{1}{(n+1)^2} \left(\sum_{i,j=1}^n k(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) + 2 \sum_{i=1}^n k(\mathbf{x}^{(i)}, \mathbf{x}) - k(\mathbf{x}, \mathbf{x}) \right). \tag{17c}$$

In the previously developed expression, only a few terms actually depend on the next optimal point $\mathbf{x}^{(n+1)}$ since the target energy, denoted by ε_μ , and $k(\mathbf{x}, \mathbf{x}) = \sigma^2$ are constant (by taking a stationary kernel). Therefore, the greedy minimization of the MMD can be equivalently written as:

$$\mathbf{x}^{(n+1)} \in \arg \min_{\mathbf{x} \in \mathcal{D}_X} \left\{ \frac{1}{n+1} \sum_{i=1}^n k(\mathbf{x}^{(i)}, \mathbf{x}) - P_\mu(\mathbf{x}) \right\} = \arg \min_{\mathbf{x} \in \mathcal{D}_X} \left\{ \frac{n}{n+1} P_{\zeta_n}(\mathbf{x}) - P_\mu(\mathbf{x}) \right\}. \tag{18}$$

Remark 1. For the sequential and uniformly weighted case, the formulation in Eq. (18) is almost similar to the Frank–Wolfe formulation. Our numerical experiments showed that these two versions generate very

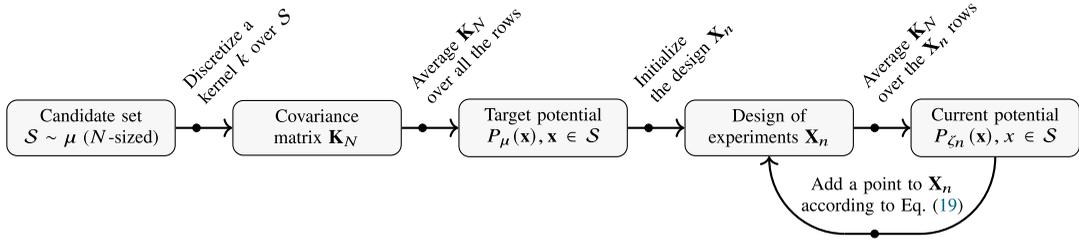


Figure 8. Greedy kernel herding algorithm.

close designs, especially as n becomes large. Pronzato and Rendas (2021) express the Frank–Wolfe formulation in the sequential and uniformly weighted case as follows:

$$\mathbf{x}^{(n+1)} \in \arg \min_{\mathbf{x} \in \mathcal{D}_X} \{P_{\zeta_n}(\mathbf{x}) - P_{\mu}(\mathbf{x})\}. \tag{19}$$

Remark 2. In practice, the optimization problem is solved by a brute-force approach on a fairly dense finite subset $\mathcal{S} \subseteq \mathcal{D}_X$ of candidate points with size $N \gg n$ that emulates the target distribution, also called the “candidate set.” This sample is also used to estimate the target potential $P_{\mu}(\mathbf{x}) \approx \frac{1}{N} \sum_{i=1}^N k(\mathbf{x}^{(i)}, \mathbf{x})$.

The diagram illustrated in Figure 8 summarizes the main steps of a KH sampling algorithm. One can notice that the initialization can either be done using a median point (maximizing the target potential) or from any existing design of experiments. This second configuration showed to be practical when the analyst must include some characteristic points in the design (e.g., points with a physical interpretation).

As explained previously, choosing the kernel defines the function space on which the worst-case function is found (see Eq. (13)). Therefore, this sampling method is sensitive to the kernel’s choice. A kernel is defined by both the choice of its parametric family (e.g., Matérn, squared exponential) and the choice of its tuning. The so-called “support points” method developed by Mak and Joseph (2018) is a special case of KH that uses the characteristic and parameter-free “energy-distance” kernel (introduced by Székely and Rizzo, 2013). In the following numerical experiments, the energy-distance kernel will be compared with an isotropic tensor product of a Matérn kernel (with regularity parameter $\nu = 5/2$ and correlation lengths θ_i), or a squared exponential kernel (with correlation lengths θ_i) defined in Table 3. Since the Matérn and squared exponential kernels are widely used for GP regression (Rasmussen and Williams, 2006), they were naturally picked to challenge the energy-distance kernel. The correlation lengths for the squared exponential and Matérn kernels are set using the heuristic given in Fekhari et al. (2023), $\theta_i = n^{-1/p}$, $i \in \{1, \dots, p\}$.

Figure 9 represents the covariance structure of the three kernels. One can notice that the squared exponential and Matérn $\nu = 5/2$ kernels are closer to one another than they are to the energy distance. In fact, as ν tends to infinity, the Matérn kernel tends toward the squared exponential kernel (which has infinitely differentiable sample paths, see Rasmussen and Williams, 2006). For these two stationary kernels, the correlation length controls how fast the correlation between two points decreases as their distance from one another increases.

Meanwhile, the energy distance is not stationary (but still positive and semi-definite). Therefore, its value does not only depend on the distance between two points but also on the norm of each of the points. Interestingly, the energy-distance kernel is almost similar to the kernel used by Hickernell (1998) to define a widely used space-filling metric called the centered L^2 -discrepancy. A presentation of these kernel-based discrepancies from the design of experiment point of view is also provided in Chapter 2 from Fang et al. (2018).

To illustrate the KH sampling of a complex distribution, Figure 10 shows three nested samples (orange crosses for different sizes $n \in \{10, 20, 40\}$) of a mixture of Gaussian distributions with complex nonlinear dependencies (with density represented by the black isoprobability contours). In this example, the method

Table 3. *Kernels considered in the following numerical experiments*

Energy distance	$k_E(\mathbf{x}, \mathbf{x}') = \frac{1}{2} (\ \mathbf{x}\ + \ \mathbf{x}'\ - \ \mathbf{x} - \mathbf{x}'\)$	
Squared exponential	$k_G(\mathbf{x}, \mathbf{x}') = \prod_{i=1}^p k_{\theta_i}(x_i - x'_i)$	$k_{\theta}(x - x') = \exp\left(-\frac{(x-x')^2}{2\theta^2}\right)$
Matérn ($\nu = 5/2$)	$k_M(\mathbf{x}, \mathbf{x}') = \prod_{i=1}^p k_{5/2, \theta_i}(x_i - x'_i)$	$k_{5/2, \theta}(x - x') = \left(1 + \frac{\sqrt{5}}{\theta} x - x' + \frac{5}{3\theta^2}(x - x')^2\right) \exp\left(-\frac{\sqrt{5}}{\theta} x - x' \right)$

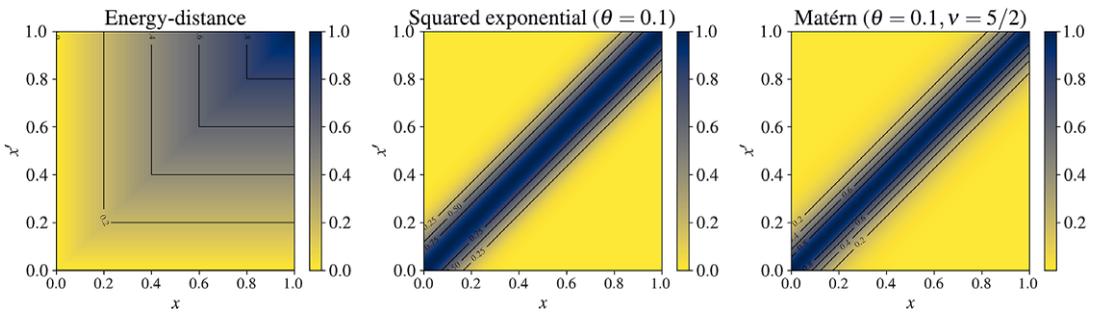


Figure 9. *Kernel illustrations (left to right: energy-distance, squared exponential, and Matérn 5/2).*

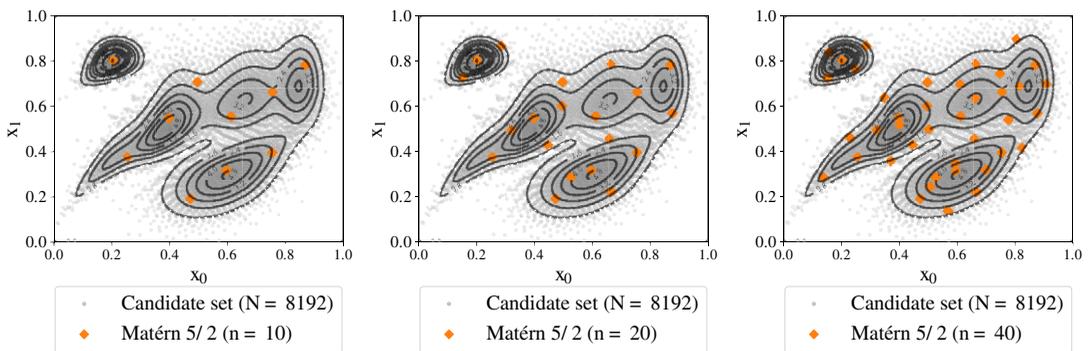


Figure 10. *Sequential kernel herding for increasing design sizes ($n \in \{10, 20, 40\}$) built on a candidate set of $N = 8196$ points drawn from a complex Gaussian mixture μ .*

seems to build a parsimonious design between each mode of the distribution (by subsampling directly without any transformation). The candidate set (in light gray) was generated by a large quasi-Monte sample of the underlying Gaussian mixture. In this two-dimensional case, this candidate set is sufficient to estimate the target potential P_{μ} . However, the main bottleneck of KH is the estimation of the potentials, which becomes costly in high dimensions.

Other approaches take advantage of the progressive knowledge acquired sequentially from the outputs to select the following points in the design. These methods are sometimes called “active learning” or “adaptive strategies” (Fuhg et al., 2021). Many of them rely on a sequentially updated GP (or Kriging) metamodel. To solve a probabilistic integration problem, the concept of Bayesian quadrature (BQ) is introduced in the following.

3.4. Bayesian quadrature

3.4.1. GPs for BQ

Kernel methods and GPs present a lot of connections and equivalences, thoroughly reviewed by Kanagawa et al. (2018). In numerical integration, GPs have been used to build quadrature rules in the seminal paper of O’Hagan (1991), introducing the concept of BQ. Let us recall the probabilistic integration problem $I_\mu(g) = \int_{\mathcal{D}_X} g(\mathbf{x}) d\mu(\mathbf{x})$ (stated in Eq. (7)). From a general point of view, this quantity could be generalized by composing g with another function ψ (e.g., other moments, quantiles, exceedance probabilities). The quantity of interest then becomes, $I_\mu(\psi(g))$, for example, when ψ is a monomial, it gives a moment of the output distribution.

Let us assume, adopting a Bayesian point of view, that ζ is a stochastic process describing the uncertainty affecting the knowledge about the true function g . Let ζ be a GP prior with a zero trend (denoted by $\mathbf{0}$) to ease the calculation, and a stationary covariance kernel (denoted by $k(\cdot, \cdot)$). The conditional posterior $\zeta_n := (\zeta | \mathbf{y}_n) \sim \mathcal{GP}(\eta_n, s_n^2)$ has been conditioned on the function observations $\mathbf{y}_n = [g(\mathbf{x}^{(1)}), \dots, g(\mathbf{x}^{(n)})]^T$ computed from the input design \mathbf{X}_n and is fully defined by the well-known ‘‘Kriging equations’’ (see, e.g., Rasmussen and Williams, 2006):

$$\begin{cases} \eta_n(\mathbf{x}) & := \mathbf{k}_n^T(\mathbf{x})\mathbf{K}_n^{-1}\mathbf{y}_n \\ s_n^2(\mathbf{x}) & := k_n(\mathbf{x}, \mathbf{x}) - \mathbf{k}_n^T(\mathbf{x})\mathbf{K}_n^{-1}\mathbf{k}_n(\mathbf{x}) \end{cases} \quad (20)$$

where $\mathbf{k}_n(\mathbf{x})$ is the column vector of the covariance kernel evaluations $[k_n(\mathbf{x}, \mathbf{x}^{(1)}), \dots, k_n(\mathbf{x}, \mathbf{x}^{(n)})]$ and \mathbf{K}_n is the $(n \times n)$ variance–covariance matrix such that the (i, j) -element is $\{\mathbf{K}_n\}_{ij} = k_n(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$.

In BQ, the main object is the random variable $I_\mu(\zeta_n)$. According to Briol et al. (2019), its distribution on \mathbb{R} is the pushforward of ζ_n through the integration operator $I_\mu(\cdot)$, sometimes called *posterior distribution*:

$$I_\mu(\zeta_n) = \int_{\mathcal{D}_X} (\zeta(\mathbf{x}) | \mathbf{y}_n) d\mu(\mathbf{x}) = \int_{\mathcal{D}_X} \zeta_n(\mathbf{x}) d\mu(\mathbf{x}). \quad (21)$$

Figure 11 provides a 1D illustration of the BQ of an unknown function (dashed black curve) against a given input measure μ (with corresponding gray distribution at the bottom). For an arbitrary design, one can fit a GP model, interpolating the function observations (black crosses). Then, multiple trajectories of this conditioned GP ζ_n are drawn (orange curves) while its mean function, also called ‘‘predictor,’’ is

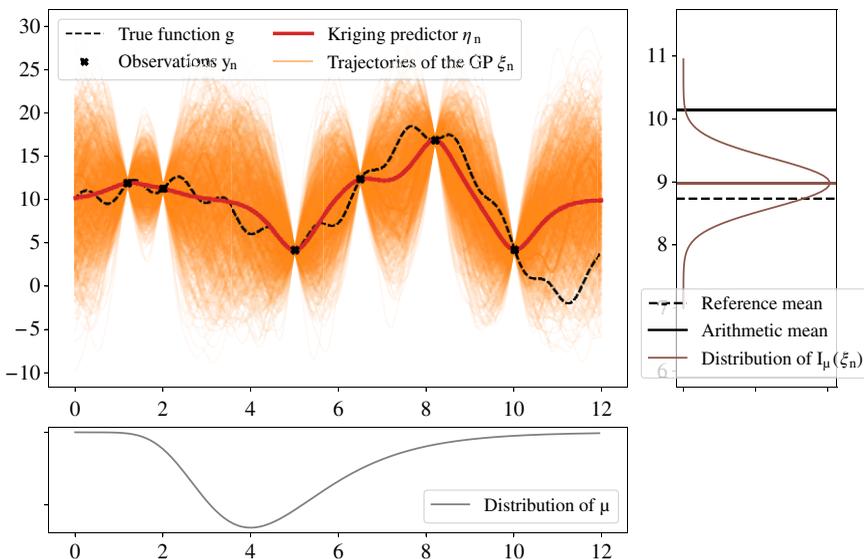


Figure 11. Bayesian quadrature on a one-dimensional case.

represented by the red curve. Therefore, the input measure μ is propagated through the conditioned GP to obtain the random variable $I_\mu(\zeta_n)$, with distribution represented on the right plot (brown curve). Again on the right plot, remark how the mean of this posterior distribution (brown line) is closer to the reference output expected value (dashed black line) than the arithmetic mean of the observations (black line). This plot was inspired by the paper of Huszár and Duvenaud (2012).

3.4.2. Optimal weights computed by BQ

Taking the random process ζ_n as Gaussian conveniently implies that its posterior distribution $a_\mu(\zeta_n)$ is also Gaussian. This comes from the linearity of the infinite sum of realizations of a GP. The posterior distribution is described in a closed form through its mean and variance by applying Fubini’s theorem (see the supplementary materials from Briol et al. (2019) for the proof regarding the variance):

$$\bar{y}_n^{\text{BQ}} := \mathbb{E}[I_\mu(\zeta_n)|\mathbf{y}_n] = \int_{\mathcal{D}_X} \eta_n(\mathbf{x}) d\mu(\mathbf{x}) = \left[\int_{\mathcal{D}_X} \mathbf{k}_n^T(\mathbf{x}) d\mu(\mathbf{x}) \right] \mathbf{K}_n^{-1} \mathbf{y}_n = P_\mu(\mathbf{X}_n) \mathbf{K}_n^{-1} \mathbf{y}_n \quad (22)$$

$$(\sigma_n^{\text{BQ}})^2 := \text{Var}(I_\mu(\zeta_n)) = \iint_{\mathcal{D}_X^2} k_n(\mathbf{x}, \mathbf{x}') d\mu(\mathbf{x}) d\mu(\mathbf{x}') = \varepsilon_\mu - P_\mu(\mathbf{X}_n) \mathbf{K}_n^{-1} P_\mu(\mathbf{X}_n)^T. \quad (23)$$

where $P_\mu(\mathbf{X}_n)$ is the row vector of potentials $[\int k_n(\mathbf{x}, \mathbf{x}^{(1)}) d\mu(\mathbf{x}), \dots, \int k_n(\mathbf{x}, \mathbf{x}^{(n)}) d\mu(\mathbf{x})]$, and ε_μ is given in Eq. (11). As in the 1D example presented in Figure 11, the expected value of $I_\mu(\zeta_n)$ expressed in Eq. (22) is a direct estimator of the quantity of interest Eq. (7). The so-called “BQ estimator” appears to be a simple linear combination of the observations by taking the row vector of “optimal weights” as:

$$\mathbf{w}_{\text{BQ}} := P_\mu(\mathbf{X}_n) \mathbf{K}_n^{-1}. \quad (24)$$

For any given sample, an optimal set of weights can be computed, leading to the mean of the posterior distribution. Remark here that this enhancement depends on the evaluation of the inverse variance–covariance matrix \mathbf{K}_n^{-1} , which can present numerical difficulties, typically when design points are too close, making the conditioning bad. Moreover, a prediction interval on the BQ estimator can be computed since the posterior distribution is Gaussian, with a variance expressed in closed-form in Eq. (23). The expressions in Eq. (22) and Eq. (23) were extended to GPs in the case of constant and linear trends in Pronzato and Zhigljavsky (2020). In the following numerical experiments, the expression with a hypothesis of constant trend $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)^T$ is used, which leads to:

$$\bar{y}_n^{\text{BQ}} = \boldsymbol{\beta} + P_\mu(\mathbf{X}_n) \mathbf{K}_n^{-1} (\mathbf{y}_n - \boldsymbol{\beta} \mathbf{1}_n). \quad (25)$$

Then, an a posteriori 95% prediction interval around the mean Bayesian estimator is directly given by:

$$[\bar{y}_n^{\text{BQ}} - 2\sigma_n^{\text{BQ}}, \bar{y}_n^{\text{BQ}} + 2\sigma_n^{\text{BQ}}]. \quad (26)$$

3.4.3. Variance-based BQ rule

The link between the posterior variance and the squared MMD has been first made by Huszár and Duvenaud (2012) in their Proposition 1: the expected variance in the BQ $\text{Var}(I_\mu(\zeta_n))$ is the MMD between the target distribution μ and $\zeta_n = \sum_{i=1}^n \mathbf{w}_{\text{BQ}}^{(i)} \delta(\mathbf{x}^{(i)})$. The proof is reproduced below (as well as in Proposition 6.1 from Kanagawa et al., 2018):

$$\text{Var}(I_\mu(\zeta_n)) = \mathbb{E} \left[(I_\mu(\zeta_n) - I_{\zeta_n}(\zeta_n))^2 \right] \quad (27a)$$

$$= \mathbb{E} \left[\left(\langle \zeta_n, P_\mu \rangle_{\mathcal{H}(k)} - \langle \zeta_n, P_{\zeta_n} \rangle_{\mathcal{H}(k)} \right)^2 \right] \quad (27b)$$

$$= \mathbb{E} \left[\langle \zeta_n, P_\mu - P_{\zeta_n} \rangle_{\mathcal{H}(k)}^2 \right] \quad (27c)$$

$$= \|P_\mu - P_{\zeta_n}\|_{\mathcal{H}(k)}^2 \quad (27d)$$

$$= \text{MMD}_k(\mu, \zeta_n)^2. \quad (27e)$$

Note that the transition from equation (27c) to (27d) relies on the property stating that if ζ is a GP with null trend and covariance kernel k , then $\forall g \in \mathcal{H}(k) : \langle \zeta, g \rangle_{\mathcal{H}(k)} \sim \mathcal{N}\left(0, \|g\|_{\mathcal{H}(k)}^2\right)$. The method that sequentially builds a quadrature rule by minimizing this variance is called by the authors “sequential BQ” (SBQ). According to the previous proof, this criterion can be seen as an optimally weighted version of the KH criterion, as stated in the title of the paper from Huszár and Duvenaud (2012). Later, Briol et al. (2015) proved the weak convergence of $I_\mu(\zeta_n)$ toward the target integral.

Closer to wind turbine applications, Huchet (2018) and Huchet et al. (2019) introduced the “Adaptive Kriging Damage Assessment” method: a Kriging-based method for mean damage estimation that is very close to SBQ. However, this type of method inherits the limits from both KH and BQ since it searches for optimal design points among a candidate set and computes an inverse variance–covariance matrix. These numerical operations cannot easily scale up to high dimensions.

Remark 3. Every quadrature method introduced in this section has been built without any observation of the possibly costly function g . Therefore, they cannot be categorized as active learning approaches. Contrarily, Kanagawa and Hennig (2019) present a set of methods for BQ with transformations (i.e., adding a positivity constraint on the function g), which are truly active learning methods.

4. Numerical experiments

This section presents numerical results computed on two different analytical toy cases, respectively, in dimension 2 (toy case #1) and dimension 10 (toy case #2), with easy-to-evaluate functions $g(\cdot)$ and associated input distributions μ (see Table 4). Therefore, reference values can easily be computed with great precision. For each toy case, a large reference Monte Carlo sample ($N_{\text{ref}} = 10^8$) is taken. This first benchmark compares the mean estimation of toy cases given by a quasi-Monte Carlo technique (abbreviated by QMC in the next figures) which consists herein using a Sobol’ sequence, and KH with the three kernels defined in Table 3. Notice that the quasi-Monte Carlo designs are first generated on the unit hypercube and then, transformed using the generalized Nataf transformation to follow the target distribution (Lebrun and Dutfoy, 2009). Additionally, the performances of KH for both uniform and optimally weighted Eq. (25) estimators are compared.

All the following results and methods (i.e., the kernel-based sampling and BQ methods) have been implemented in a new open-source Python package named `otkerneldesign`⁵. This development mostly relies on the open source software `OpenTURNS`⁶ (“Open source initiative for the Treatment of Uncertainties, Risks’N Statistics”) devoted to uncertainty quantification and statistical learning (Baudin et al., 2017). Finally, note that the numerical experiments for the toy cases are available in the Git repository named `ctbenchmark`⁷ for reproducibility purposes.

4.1. Illustration through analytical toy cases

The toy cases were chosen to cover a large panel of complex probabilistic integration problems, completing the ones from Fekhari et al. (2021). To assess the complexity of numerical integration problems, Owen (2003) introduced the concept of the “effective dimension” of an integrand function

⁵ <https://efekhari27.github.io/otkerneldesign/master/index.html>.

⁶ <https://openturns.github.io/www/>.

⁷ <https://github.com/efekhari27/ctbenchmark>.

Table 4. Analytical toy cases

toy case #1	$dim = 2$	$g_1(\mathbf{x}) = x_1 + x_2$	Gaussian mixture from Figure 10
toy case #2	$dim = 10$	$g_2(\mathbf{x}) = \prod_{i=1}^{10} \frac{ 4x_i - 2 + a_i}{1 + a_i}, \{a_i = 2\}_{i=1}^{10}$	Gaussian $\mathcal{N}(\mathbf{0.5}, \mathbf{I}_{10})$

(number of the variables that actually impact the integral). The author showed that functions built on sums yield a low effective dimension (unlike functions built on products). In the same vein, Kucherenko et al. (2011) build three classes of integrand sorted by difficulty depending on their effective dimension:

- *class A*: problem with a few dominant variables.
- *class B*: problem without unimportant variables, and important low-order interaction terms.
- *class C*: problems without unimportant variables, and important high-order interaction terms.

The 10-dimensional “GSobol function” (toy case #2) with a set of coefficient $\{a_i = 2\}_{i=1}^{10}$ has an effective dimension equal to 10 and belongs to the hardest class C from Kucherenko et al. (2011). In the case of the two-dimensional Gaussian mixture problem, the complexity is carried by the mixture of Gaussian distributions with highly nonlinear dependencies. Probabilistic integration results are presented in Figure 12 (toy case #1) and Figure 13 (toy case #2). KH samples using the energy-distance kernel are in red, while quasi-Monte Carlo samples built from Sobol’ sequences are in gray. To ease the reading, the results from the other kernels defined in Table 3 are moved to Appendix A, in Figures 17 and 18. Convergences of the arithmetic means are plotted on the left and MMDs on the right. The respective BQ estimators of the means are plotted in dashed lines.

Remark 4. Different kernels are used in these numerical experiments. First, the generation kernel, used by the KH algorithm to generate designs (with the heuristic tuning defined in Section 3.3). Second, the BQ kernel allows computation of the optimal weights (arbitrarily set up as a Matérn 5/2 with the heuristic tuning). Third, the evaluation kernel, which must be common to allow a fair comparison of the computed MMD results (same as the BQ kernel).

Results analysis for toy case #1. Convergence plots are provided in Figure 12. KH consistently converges faster than quasi-Monte Carlo in this case, especially for small sizes in terms of MMD. BQ weights tend to reduce the fluctuations in the mean convergence, which ensures better performance for any size. Overall, applying the weights enhances the convergence rate.

Results analysis for toy case #2. Convergence plots are provided in Figure 13. Although quasi-Monte Carlo is known to suffer the “curse of dimensionality,” KH does not outperform it drastically in this example. In fact, KH with uniform weights performs worse than quasi-Monte Carlo while optimally

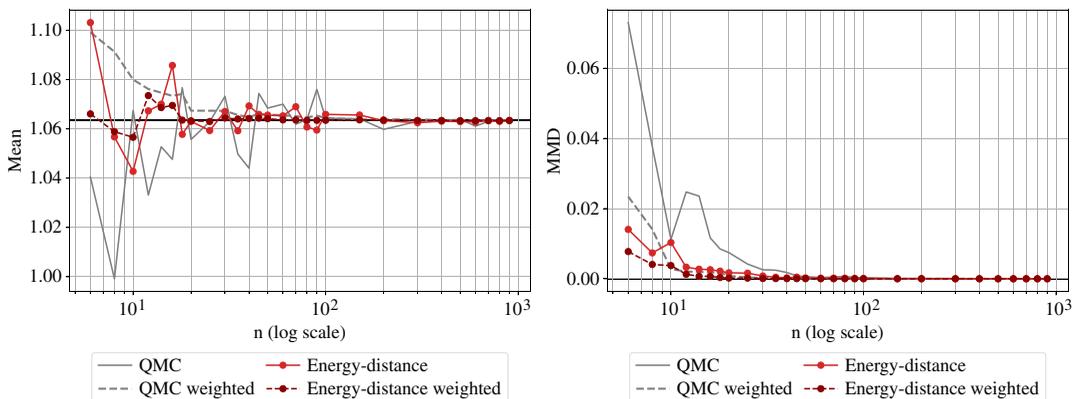


Figure 12. Analytical benchmark results on the toy case #1.

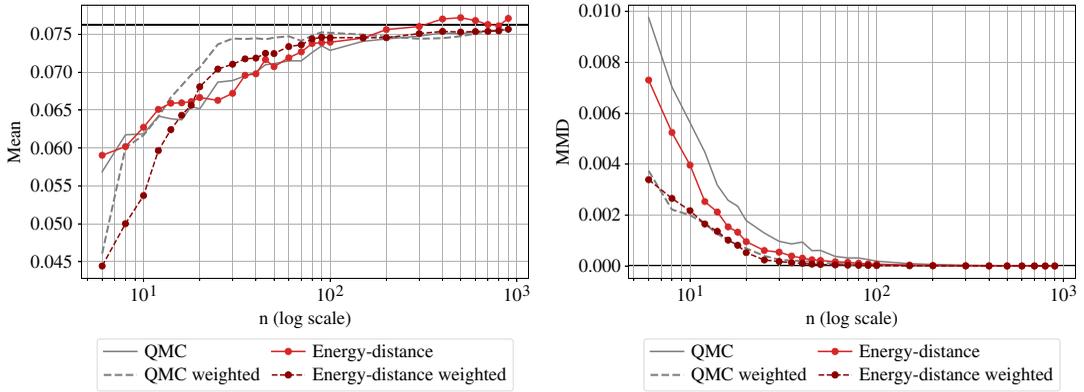


Figure 13. Analytical benchmark results on the toy case #2.

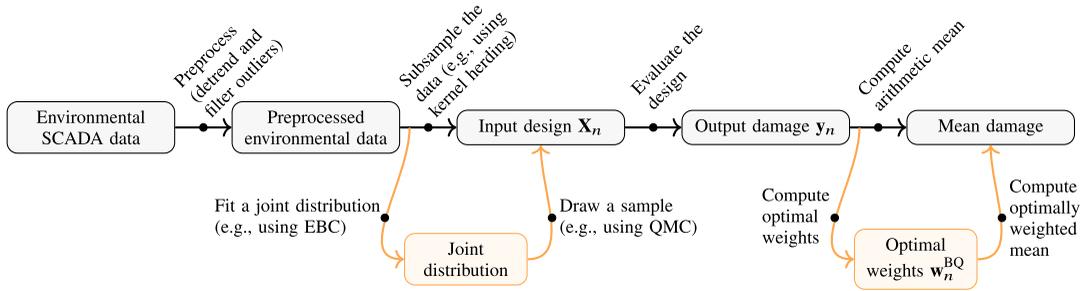


Figure 14. Mean damage estimation workflows for the industrial use case. The orange parts represent optional alterations to the workflow: the first one is an alternative to input data subsampling where the underlying distribution is sampled from, and the second one improves mean damage calculation by using optimal weights over the output data.

weighted KH does slightly better. Moreover, the results confirm that $MMD_{BQ} < MMD_{unif}$ for all our experiments. The application of optimal weights to the quasi-Monte Carlo sample slightly improves the estimation in this case. Note that the prediction interval around the BQ estimator is not plotted for the sake of readability.

In these two toy cases, the MMD is shown to quantify numerical integration convergence well, which illustrates the validity of the inequality given in Eq. (12c), similar to the Koksma–Hlawka inequality (as recalled in Eq. (8)).

4.2. Application to the Teesside wind turbine fatigue estimation

Let us summarize the mean damage estimation strategies studied in this paper. The diagram represented in Figure 14 describes the different workflows computed. The simplest workflow is represented by the gray horizontal sequence. It directly subsamples a design of experiments from a large and representative dataset (previously referred to as candidate set). This workflow simply estimates the mean damage by computing an arithmetic average of the outputs.

Alternatively, one can, respectively, fit a joint distribution and sample from it. In our case, this distribution is only known empirically via the candidate set. Since its dependence structure is complex (see Figure 4), a parametric method might fit the distribution poorly (and therefore lead to a poor estimation of the quantity). Then, a nonparametric fit using the EBC (introduced in Section 2.3) coupled with a kernel density estimation on each marginal is applied to the candidate set (with the EBC parameter $m = 100 > m_{MISE}$ to avoid bias, see Lasserre (2022, p. 117). The sampling on this hybrid joint distribution

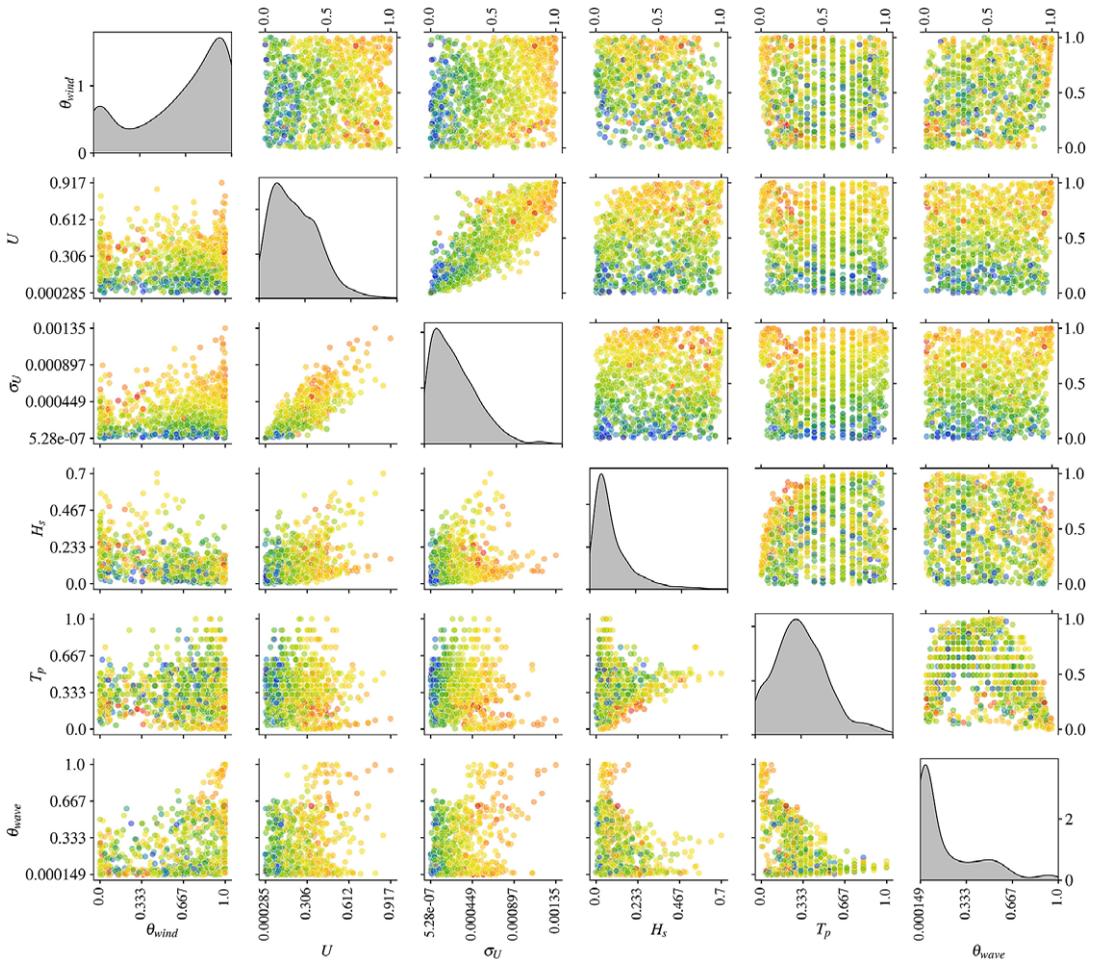


Figure 15. Copulogram of the kernel herding design of experiments with corresponding outputs in color (log-scale) on the Teesside case ($n = 10^3$). The color scale ranges from blue for the lowest values to red for the largest. Marginals are represented by histograms (diagonal), the dependence structure with scatterplots in the ranked space (upper triangle). Scatterplots on the bottom triangle are set in the physical space.

is realized with a quasi-Monte Carlo method. A Sobol’ low-discrepancy sequence generates a uniform sample in the unit hypercube, which can then be transformed according to a target distribution. Remember that quasi-Monte Carlo sampling is also sensitive to the choice of a low-discrepancy sequence, each presenting different properties (e.g., Sobol’, Halton, Faure, etc.). Finally, the estimation by an arithmetic mean can be replaced by an optimally weighted mean. To do so, optimal weight must be computed using the formulas introduced in Eq. (24).

The copulogram in Figure 15 illustrates the intensity of the computed damages, proportionally to the color scale. Note that the numerical values of the damage scale are kept confidential since it models the state of an operating asset. Before analyzing the performance of the KH on this industrial application, let us notice that the copulogram Figure 15 seems to be in line with the global sensitivity analysis presented in Murcia et al. (2018) and Li and Zhang (2020). In particular, the scatterplot of mean wind speed versus turbulence wind speed is the main factor explaining the variance of the output $Y = g(\mathbf{X})$. Judging from these references, the numerical model does not seem to have a highly effective dimension; however, the input dependence structure is challenging and the damage assessment induces strong nonlinearities (see Eq. (4)).

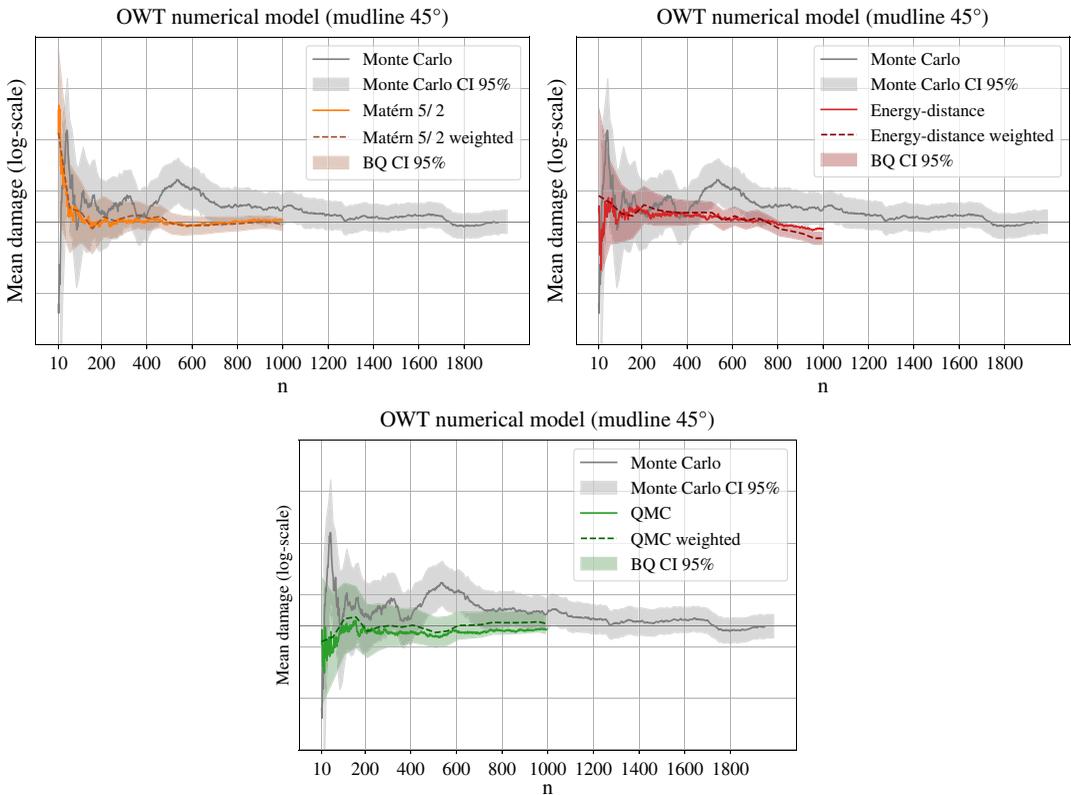


Figure 16. Mean estimation convergence (at the mudline, azimuth $\theta = 45\text{deg.}$) on the Teesside case. Monte Carlo confidence intervals are all computed by bootstrap.

The results presented are compared in the following to a large reference Monte Carlo sample (size 2000) with a confidence interval computed by bootstrap (see Figure 16). This reference is represented by a horizontal line intersecting with the most converged Monte Carlo estimation. Once again, the mean damage scale is hidden for confidentiality reasons, but all the plots are represented for the same vertical scale. The performance of the KH is good: it quickly converges toward the confidence interval of the Monte Carlo obtained with the reference sample. In addition, the BQ estimator also offers a posteriori prediction interval, which can reassure the user. The BQ prediction intervals are smaller than the ones obtained by bootstrap on the reference Monte Carlo sample.

To provide more representative results, note that a set of scale parameters is computed with a Kriging procedure to define the kernel used to compute BQ intervals. Since other methods do not generate independent samples, bootstrapping them is not legitimate. Contrarily to the other kernels, we observe that the energy-distance kernel presents a small bias with the MC reference for most of the azimuth angles computed in this experiment. Meanwhile, combining nonparametric fitting with quasi-Monte Carlo sampling also delivers good results as long as the fitting step does not introduce a bias. In our case, any potential bias due to poor fitting would be the result of a poorly tuned EBC. Fortunately, Nagler et al. (2017) formulated recommendations regarding how to tune EBCs. We follow these recommendations in the present work.

5. Conclusion

Wind energy assets are subject to highly uncertain environmental conditions. Uncertainty propagation through numerical models is performed to ensure their structural integrity (and energy production). For

this case, the method recommended by the standards (regular grid sampling) is intractable for even moderate-fidelity simulators. In practice, such an approach can lead to poor uncertainty propagation, especially when facing simulation budget constraints.

In the present paper, a real industrial wind turbine fatigue estimation use case is investigated by considering site-specific data. As a perspective, other sites with different environmental conditions could be studied. This use case induces two practical constraints: first, usual active learning methods are hard to set up on such a model (mainly due to the nonlinearity of the variable of interest), and they restrict the use of high-performance computing facilities; second, the input distribution of the environmental conditions presents a complex dependence structure which is hard to infer with common parametric approaches.

In this work, the association of KH sampling with BQ for central tendency estimation is explored theoretically and numerically. This method fits with the practical constraints induced by the industrial use case. To be more specific, the KH method easily subsamples the relevant points directly from a given dataset (here, from the measured environmental data). Moreover, the method is fully compatible with intensive high-performance computer use. Moreover, the present work outlined an upper bound based on the MMD on numerical integration absolute error. KH and BQ both aim at finding the quadrature rule minimizing the MMD, and therefore the absolute integration error. The numerical experiments confirm that the MMD is an appropriate criterion since it leads to results being better or equivalent to quasi-Monte Carlo sampling. Finally, the proposed numerical benchmark relies on a Python package, called *otkerneldesign*, which implements the methods and allows anyone to reproduce the results.

The limits of the proposed method are reached when the input dimension of the problem increases, requiring a larger candidate set and therefore a larger covariance matrix. Moreover, the numerical experiments show that the method can be sensitive to the choice of the kernel and its tuning (although good practices can be derived). From a methodological viewpoint, further interpretation of the impact of the different kernels could be explored. Besides, extensions of KH sampling for quantile estimation could be investigated, in a similar fashion as the work on randomized quasi-Monte Carlo for quantiles proposed by Kaplan et al. (2019). KH could also be used to quantize conditional distributions, using the so-called “conditional kernel mean embedding” concept reviewed by Klebanov et al. (2020). Finally, regarding the industrial use case, the next step should be to perform a reliability analysis by considering another group of random variables (related to the wind turbine) or to explore the possibilities offered by reliability-oriented sensitivity analysis in the context of kernel-based indices, as studied in Marrel and Chabridon (2021).

Data availability statement. Replication data and code of the analytical benchmark can be found in the GitHub repository: <https://github.com/efekhari27/ctbenchmark>.

Acknowledgments. The authors are grateful to the two anonymous reviewers for their comments and suggestions. The authors are also thankful for the technical assistance from various contributors to the HIPERWIND European project, A. Lovera, N. Dimitrov, and M. Capaldo. Moreover, the authors would also like to thank S. Da Veiga, L. Pronzato, M. Munoz Zuniga, and M. Baudin, for their fruitful discussions about the several topics found in this paper.

Author contribution. Conceptualization: E.F.; V.C.; J.M.; B.I. Methodology: E.F.; J.M.; V.C.; B.I. Data curation: E.F. Data visualization: E.F. Writing original draft: E.F. All authors approved the final submitted draft.

Funding statement. Part of this study is part of HIPERWIND project which has received funding from the European Union’s Horizon 2020 Research and Innovation Programme under Grant Agreement No. 101006689. Part of this work was also supported by project INDEX (INcremental Design of EXperiments) ANR-18-CE91-0007 of the French National Research Agency (ANR).

Competing interest. The authors declare no competing interests exist.

Ethical standard. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

References

- Abdallah I, Lataniotis C and Sudret B** (2019) Parametric hierarchical kriging for multi-fidelity aero-servo-elastic simulators – Application to extreme loads on wind turbines. *Probabilistic Engineering Mechanics* 55, 67–77.
- Ajenjo A** (2023) *Info-gap robustness assessment of reliability evaluations for the safety of critical industrialsystems*. PhD thesis, Université Bourgogne Franche-Comté.

- Bai H, Shi L, Aoues Y, Huang C and Lemosse D** (2023) Estimation of probability distribution of long-term fatigue damage on wind turbine tower using residual neural network. *Mechanical Systems and Signal Processing* 190, 110101.
- Baudin M, Duffoy A, Iooss B and Popelin A** (2017) Open TURNS: An industrial software for uncertainty quantification in simulation. In Ghanem R, Higdon D and Owhadi H (eds.), *Handbook on Uncertainty Quantification*. Cham: Springer, pp. 2001–2038.
- Briol F, Oates C, Girolami M, Osborne M and Sejdinovic D** (2019) Probabilistic integration: A role in statistical computation? *Statistical Science* 34, 1–22.
- Briol F-X** (2019) *Statistical computation with kernels*. PhD thesis, University of Warwick.
- Briol F-X, Oates C, Girolami M and Osborne M** (2015) Frank-wolfe Bayesian quadrature: Probabilistic integration with theoretical guarantees. In Jordan MI, LeCun Y and Solla SA (eds.), *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press.
- Chen T, Wang X, Yuan G and Liu J** (2018) Fatigue bending test on grouted connections for monopile offshore wind turbines. *Marine Structures* 60, 52–71.
- Chen Y, Welling M and Smola A** (2010) Super-samples from kernel herding. In Grunwald P and Spirites P (eds.), *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*. Arlington, VA: AUAI Press, pp. 109–116.
- Da Veiga S** (2015) Global sensitivity analysis with dependence measures. *Journal of Statistical Computation and Simulation* 85, 1283–1305.
- Da Veiga S, Gamboa F, Iooss B and Prieur C** (2021) *Basics and Trends in Sensitivity Analysis: Theory and Practice in R*. Society for Industrial and Applied Mathematics.
- de Rocquigny E, Devictor N and Tarantola S** (2008) *Uncertainty in Industrial Practice: A Guide to Quantitative Uncertainty Management*. John Wiley & Sons.
- Dimitrov N** (2013) *Structural reliability of wind turbine blades: Design methods and evaluation*. PhD thesis, Technical University of Denmark.
- Dimitrov N, Kelly M, Vignaroli A and Berg J** (2018) From wind to loads: Wind turbine site-specific load estimation with surrogate models trained on high-fidelity load databases. *Wind Energy Science* 3, 767–790.
- DNV-GL** (2016a) *DNVGL-RP-C203: Fatigue design of offshore steel structures*. Technical report, DNVGL.
- DNV-GL** (2016b) *DNVGL-ST-0437: Loads and site conditions for wind turbines*. Technical report, DNVGL.
- Dowling NE** (1972) Fatigue failure predictions for complicated stress-strain histories. *Journal of Materials, JMLSA* 7, 71–87.
- Fang K, Liu M-Q, Qin H and Zhou Y-D** (2018) *Theory and Application of Uniform Experimental Designs*, vol. 221. Singapore: Springer.
- Fatemi A and Yang L** (1998) Cumulative fatigue damage and life prediction theories: A survey of the state of the art for homogeneous materials. *International Journal of Fatigue* 20(1), 9–34.
- Fekhari E, Iooss B, Chabridon V and Muré J** (2021) Efficient techniques for fast uncertainty propagation in an offshore wind turbine multi-physics simulation tool. In *Proceedings of the 5th International Conference on Renewable Energies Offshore*. CRC Press, pp. 837–846.
- Fekhari E, Iooss B, Muré J, Pronzato L and Rendas M-J** (2023) Model predictivity assessment: Incremental test-set selection and accuracy evaluation. In Salvati N, Perna C, Marchetti S and Chambers R (eds.), *Studies in Theoretical and Applied Statistics, SIS 2021, Pisa, Italy, June 21–25*. Cham: Springer.
- Fuhg J, Fau A and Nackenhorst U** (2021) State-of-the-art and comparative review of adaptive sampling methods for kriging. *Archives of Computational Methods in Engineering* 28, 2689–2747.
- Graf P, Stewart G, Lackner M, Dykes K and Veers P** (2016) High-throughput computation and the applicability of Monte Carlo integration in fatigue load estimation of floating offshore wind turbines. *Wind Energy* 19(5), 861–872.
- Gretton A, Borgwardt KM, Rasch M, Schölkopf B and Smola A** (2006) A kernel method for the two-sample-problem. In Schölkopf B, Platt J and Hoffman T (eds.), *Advances in Neural Information Processing Systems*. MIT Press. pp. 513–520.
- Hansen M and Henriksen L** (2013) *Basic DTU Wind Energy Controller*. DTU Wind Energy.
- Hickernell F** (1998) A generalized discrepancy and quadrature error bound. *Mathematics of Computation* 67(221), 299–322.
- Huchet Q** (2018) *Kriging based methods for the structural damage assessment of offshore wind turbines*. PhD thesis, Université Blaise Pascal.
- Huchet Q, Mattrand C, Beaurepaire P, Relun N and Gayton N** (2019) AK-DA: An efficient method for the fatigue assessment of wind turbine structures. *Wind Energy* 22(5), 638–652.
- Huszár F and Duvenaud D** (2012) Optimally-weighted herding is Bayesian quadrature. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*. Arlington, Virginia, VA: AUAI Press. pp. 377–386.
- IEC** (2019) IEC 61400–1: Wind energy generation systems - part 1: Design requirements. Technical report, International Electrotechnical Commission (IEC).
- Joe H** (1997) *Multivariate Models and Multivariate Dependence Concepts*. New York: Chapman and Hall.
- Jonkman B** (2009) *Turbsim User's Guide: Version 1.50*. Technical report, NREL.
- Kaimal J, Wyngaard J, Izumi Y and Coté O** (1972) Spectral characteristics of surface-layer turbulence. *Quarterly Journal of the Royal Meteorological Society* 98(417), 563–589.
- Kanagawa M and Hennig P** (2019) Convergence guarantees for adaptive Bayesian quadrature methods. *Advances in Neural Information Processing Systems* 32, 6237–6248.

- Kanagawa M, Hennig P, Sejdinovic D and Sriperumbudur B** (2018) Gaussian processes and kernel methods: A review on connections and equivalences. Preprint, arXiv:1807.02582.
- Kanner S, Aubault A, Peiffer A and Yu B** (2018) Maximum dissimilarity-based algorithm for discretization of meteocean data into clusters of arbitrary size and dimension. *International Conference on Offshore Mechanics and Arctic Engineering 51319*, 9.
- Kaplan Z, Li Y, Nakayama M and Tuffin B** (2019) Randomized quasi-monte carlo for quantile estimation. In *2019 Winter Simulation Conference (WSC)*, pp. 428–439.
- Katsikogiannis G, Sørum S, Bachynski E and Amdahl J** (2021) Environmental lumping for efficient fatigue assessment of large-diameter monopile wind turbines. *Marine Structures 77*, 102939.
- Kim T, Natarajan A, Lovera A, Julian E, Peyrard E, Capaldo M, Huwart G, Bozonnet P and Guiton M** (2022) A comprehensive code-to-code comparison study with the modified IEA15MW-UMaine floating wind turbine for H2020 HIPER-WIND project. *Journal of Physics: Conference Series 2265*(4), 042006.
- Klebanov I, Schuster I and Sullivan T** (2020) A rigorous theory of conditional mean embeddings. *SIAM Journal on Mathematics of Data Science 2*(3), 583–606.
- Kucherenko S, Feil B, Shah N and Mauntz W** (2011) The identification of model effective dimensions using global sensitivity analysis. *Reliability Engineering & System Safety 96*, 440–449.
- Lacoste-Julien S, Lindsten F and Bach F** (2015) Sequential kernel herding: Frank-Wolfe optimization for particle filtering. *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics 38*, 544–552.
- Lasserre M** (2022) *Apprentissages dans les réseaux bayésiens à base de copules non-paramétriques*. PhD thesis, Sorbonne Université.
- Lataniotis C** (2019) *Data-driven uncertainty quantification for high-dimensional engineering problems*. PhD thesis, ETH Zürich.
- Lebrun R and Dutfoy A** (2009) A generalization of the Nataf transformation to distributions with elliptical copula. *Probabilistic Engineering Mechanics 24*(2), 172–178.
- Leobacher G and Pillichshammer F** (2014) *Introduction to Quasi-Monte Carlo Integration and Applications*. Cham: Springer.
- Li X and Zhang W** (2020) Long-term fatigue damage assessment for a floating offshore wind turbine under realistic environmental conditions. *Renewable Energy 159*, 570–584.
- Lovera A, Fekhari E, Jézéquel B, Dupoirion M, Guiton M and Ardillon E** (2023) Quantifying and clustering the wake-induced perturbations within a wind farm for load analysis. *Journal of Physics: Conference Series 2505*, 012011.
- Mak S and Joseph V** (2018) Support points. *Annals of Statistics 46*, 2562–2592.
- Marrel A and Chabridon V** (2021) Statistical developments for target and conditional sensitivity analysis: Application on safety studies for nuclear reactor. *Reliability Engineering & System Safety 214*, 107711.
- Morokoff WJ and Cafilisch RE** (1995) Quasi-Monte Carlo integration. *Journal of Computational Physics 122*(2), 218–230.
- Müller K and Cheng P** (2018) Application of a Monte Carlo procedure for probabilistic fatigue design of floating offshore wind turbines. *Wind Energy Science 3*, 149–162.
- Murcia J, Réthoré P, Dimitrov N, Natarajan A, Sørensen J, Graf P and Kim T** (2018) Uncertainty propagation through an aeroelastic wind turbine model using polynomial surrogates. *Renewable Energy 119*, 910–922.
- Nagler T, Schellhase C and Czado C** (2017) Nonparametric estimation of simplified vine copula models: Comparison of methods. *Dependence Modeling 5*, 99–120.
- O'Hagan A** (1991) Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference 29*, 245–260.
- Oates CJ** (2021) Minimum Discrepancy Methods in Uncertainty Quantification. Lecture Notes at École Thématique sur les Incertitudes en Calcul Scientifique (ETICS21). Available at <https://www.gdr-mascotnum.fr/etics.html>.
- Owen A** (2003) The dimension distribution and quadrature test functions. *Statistica Sinica 13*, 1–17.
- Petrovska E** (2022) *Fatigue life reassessment of monopile-supported offshore wind turbine structures*. PhD thesis, University of Edinburgh.
- Pronzato L and Rendas M-J** (2021) Validation of Machine Learning Prediction Models. *The New England Journal of Statistics in Data Science 1*, 394–414.
- Pronzato L and Zhigljavsky A** (2020) Bayesian quadrature and energy minimization for space-filling design. *SIAM/ASA Journal on Uncertainty Quantification 8*, 959–1011.
- Rasmussen C and Williams C** (2006) *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press.
- Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D, Saisana M and Tarantola S** (2008) *Global Sensitivity Analysis. The Primer*. Wiley.
- Sancetta A and Satchell S** (2004) The Bernstein copula and its applications to modeling and approximations of multivariate distributions. *Econometric Theory 20*(3), 535–562.
- Segers J, Sibuya M and Tsukahara H** (2017) The empirical beta copula. *Journal of Multivariate Analysis 155*, 35–51.
- Sejdinovic D, Sriperumbudur B, Gretton A and Fukumizu K** (2013) Equivalence of distance-based and RKHS-based statistics in hypothesis testing. *Annals of Statistics 41*, 2263–2291.
- Slot RM, Sørensen JD, Sudret B, Svenningsen L and Thøgersen ML** (2020) Surrogate model uncertainty in wind turbine reliability assessment. *Renewable Energy 151*, 1150–1162.
- Sriperumbudur B, Gretton A, Fukumizu K, Schölkopf B and Lanckriet G** (2010) Hilbert space embeddings and metrics on probability measures. *Journal of Machine Learning Research 11*, 1517–1561.
- Székely GJ and Rizzo ML** (2013) Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference 143*, 1249–1272.

Teixeira R, Nogal M, O'Connor A, Nichols J and Dumas A (2019a) Stress-cycle fatigue design with kriging applied to offshore wind turbines. *International Journal of Fatigue* 125, 454–467.

Teixeira R, O'Connor N and Nogal M (2019b) Probabilistic sensitivity analysis of offshore wind turbines using a transformed Kullback–Leibler divergence. *Structural Safety* 81, 101860.

Van den Bos L (2020) *Quadrature Methods for Wind Turbine Load Calculations*. PhD thesis, Delft University of Technology.

Vanem E, Fekhari E, Dimitrov N, Kelly M, Cousin A and Guiton M (2023) A joint probability distribution model for multivariate wind and wave conditions. In *International Conference on Offshore Mechanics and Arctic Engineering*, vol. 86847. American Society of Mechanical Engineers, V002T02A013.

Velarde J, Kramhøft C and Sørensen J (2019) Global sensitivity analysis of offshore wind turbine foundation fatigue loads. *Renewable Energy* 140, 177–189.

Wilkie D and Galasso C (2021) Gaussian process regression for fatigue reliability analysis of offshore wind turbines. *Structural Safety* 88, 102020.

Zwick D and Muskulus M (2015) The simulation error caused by input loading variability in offshore wind turbine structural analysis. *Wind Energy* 18, 1421–1432.

A. Complementary results to the analytical benchmark

This section presents complementary results to the two analytical problems using the Matérn and squared-exponential kernels.

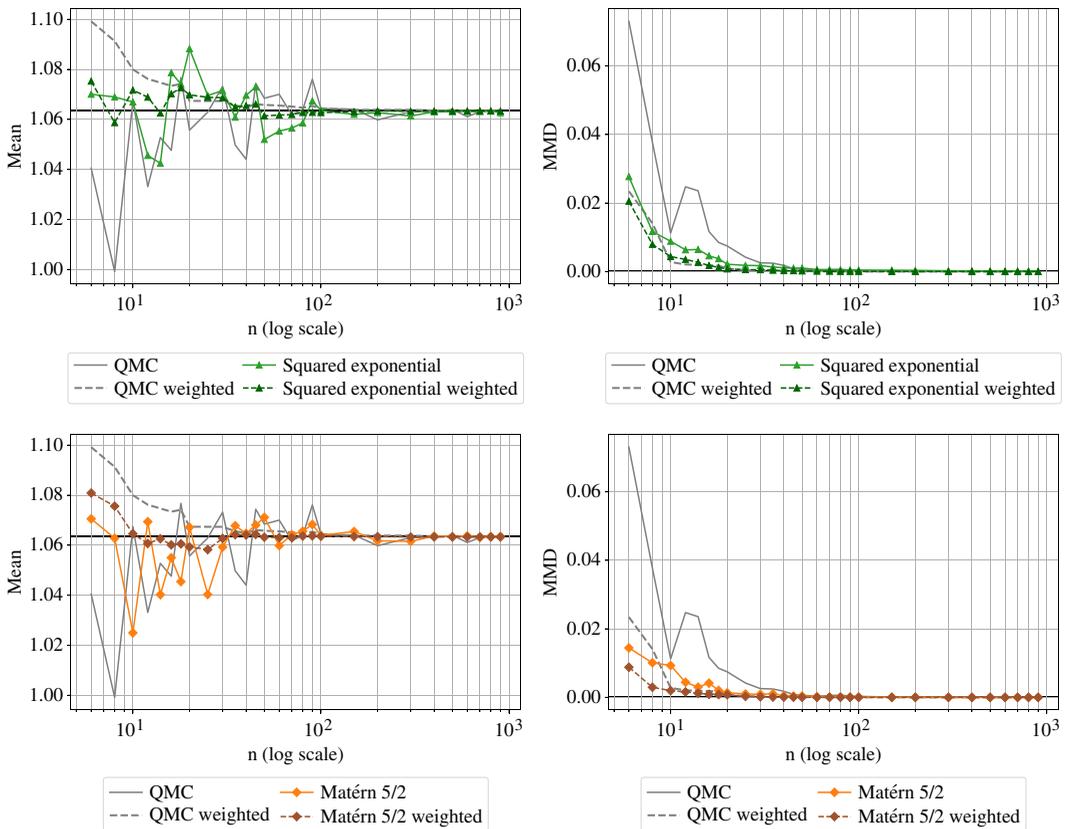


Figure 17. Analytical benchmark results on the toy case #1.

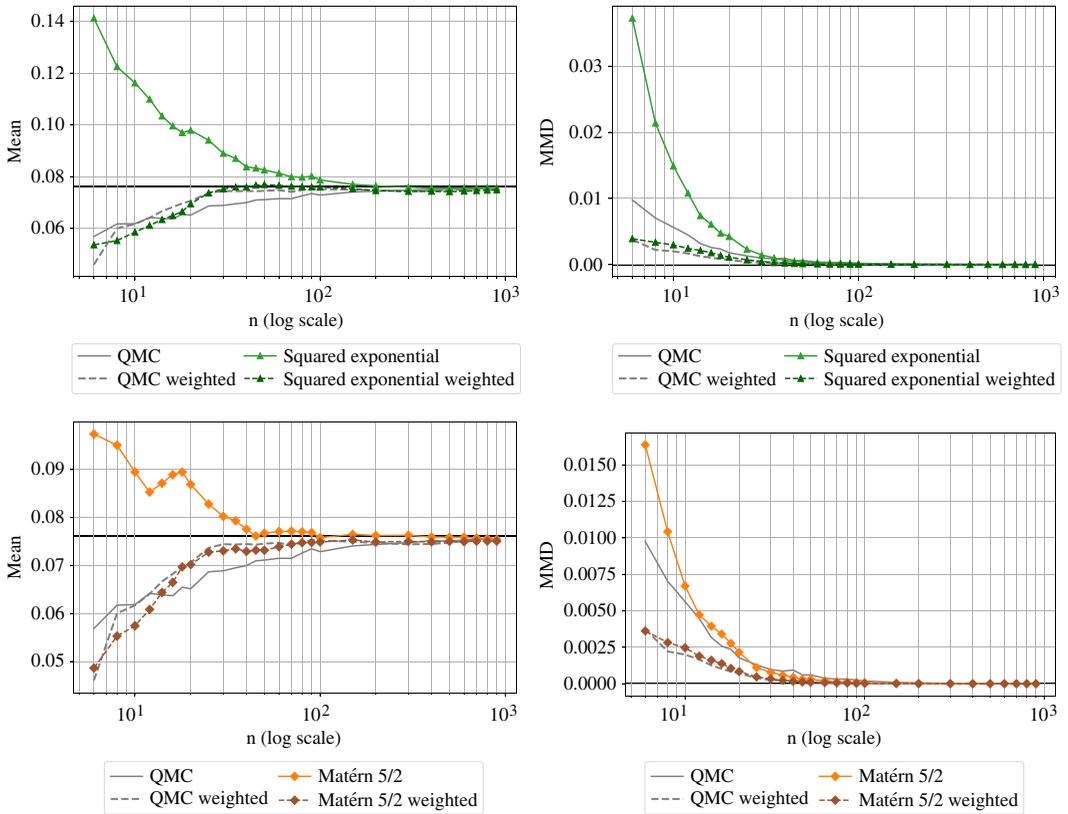


Figure 18. Analytical benchmark results on the toy case #2.