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

Elias Fekhari, Vincent Chabridon, Joseph Muré, Bertrand Iooss

Fast given-data uncertainty propagation in offshore wind turbine simulator using Bayesian quadrature

Elias Fekhari[1,2], Vincent Chabridon[1], Joseph Muré[1], and Bertrand Iooss[1,2]

[1]EDF R&D, 6 Quai Watier, Chatou, 78401, France
[2]Université Nice Côte d'Azur, 28 Avenue de Valrose, Nice, 06000, France

March 30, 2023

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 more and more subject to uncertain environmental conditions, making long-term investment decisions riskier. Fortunately, numerical models of wind turbines can perform accurate multi-physics simulations of such systems interacting with their environment. The challenge is then to propagate the input environmental uncertainties through such a model and analyze the distribution of model outputs of interest. Since each simulation of this numerical model is costly, the estimation of output quantities of interest (e.g., mean, variance) must be done with a restricted number of simulations. To do so, this paper combines kernel herding sampling with Bayesian quadrature to estimate the mean fatigue damage,with an application to an offshore wind turbine operating in Teesside, UK. It is demonstrated theoretically and numerically that this method guarantees fast and accurate convergence. Moreover, this approach is highly flexible by allowing to subsample directly from a given dataset and being fully distributable on high-performance computing facilities. Finally, a new Python package was developed and documented to provide quick open access to this uncertainty propagation method.

Keywords Wind turbine; Uncertainty propagation; Fatigue loads; Bayesian quadrature; Design of experiments; Active learning

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 more regular 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 200,000 simulations distributed over a grid. 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 Fig. 1. After defining the problem (Step A), one can quantify the uncertainties related to site-specific environmental conditions denoted by the random variable $\mathbf{X}$ (Step B). Then, one can propagate them through an OWT simulation model $g(\cdot)$ (Step C), and estimate a relevant quantity of interest $\psi(Y) = \psi(g(\mathbf{X}))$ (e.g., mean, quantile, failure probability). A proper estimation of the quantity relies on a good uncertainty model and an efficient sampling method to estimate the quantity of interest.

The uncertainties related to the OWT environment follow a joint distribution with a complex dependence structure. This challenging distribution has been fitted with different parametric approaches in the literature (step B in Fig. 1), mainly using conditional distributions (Vanem et al., 2023), but also vine copulas (Li and Zhang, 2020). When one has access to data, another way is to directly use the data as empirical representation of input uncertainties.

These uncertainties have then been propagated to fatigue damage (the output), making it random, and to the associated quantities of interest (step C in Fig. 1). When uncertainty propagation aims at central tendency studies,

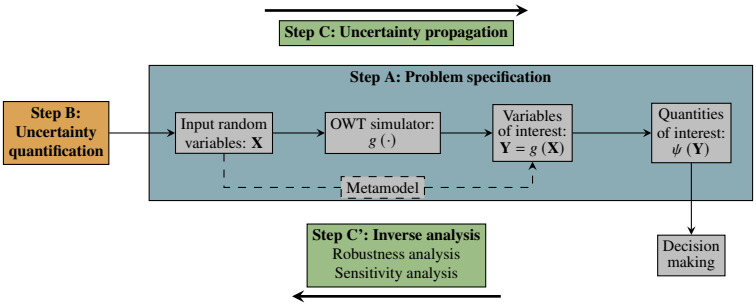

Figure 1: General uncertainty quantification and propagation framework (adapted from Ajenjo (2023))

the methods employed are split into two groups. Methods of the first group rely on numerical integration by Monte Carlo sampling (Graf et al., 2016), quasi-Monte Carlo sampling (Müller and Cheng, 2018), or deterministic quadrature rules (Van den Bos, 2020). All these methods estimate the quantity directly on the numerical simulator's outputs. Methods of the second group use metamodels (or surrogate models) to emulate the costly numerical model by a statistical model such as polynomial chaos expansion (Dimitrov et al., 2018; Murcia et al., 2018), Gaussian processes (Huchet, 2019; 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 (e.g., reliability analysis), one can estimate a quantile or a failure probability. Failure probabilities were studied, in static reliability analysis (Zwick and Muskulus, 2015; Slot et al., 2020; Wilkie and Galasso, 2021) or time-dependent reliability analysis (Abdallah et al., 2019; Lataniotis, 2019). To get a better understanding of the OWT numerical models behavior, authors have used sensitivity analysis methods (Da Veiga et al., 2021), which determine the most influential inputs on the damage (step C' in Fig. 1). Among others, one can cite the application of Spearman's rank correlation coefficients and Morris method's by Velarde et al. (2019); Petrovska (2022), the direct calculation of Sobol' indices after fitting a polynomial chaos model by Murcia et al. (2018) and the use of Kullback–Leibler divergence by Teixeira et al. (2017). Each of those methods brings something different to the analysis.

This paper will focus on central tendency estimation (i.e., $\psi(\mathbf{X}) = \mathbb{E}(\mathbf{X})$) by: (1) direct sampling on the numerical model, (2) sampling on a static regression model, or (3) sampling on an active regression model (i.e., observations of the numerical model are progressively added to enhance a goal-oriented metamodel). In the specific context of wind turbines, this paper explores how to study the central tendency study of the fatigue damage output, by carrying out the uncertainty propagation of a complex input joint distribution through a costly wind turbine simulator. This work proposes a new approach for given data, fast, and fully-distributable uncertainty propagation for OWT models. Overall, this paper reviews the methods of Bayesian quadrature and presents its application on the industrial wind turbines case. In this paper, Section 2 will detail the industrial use-case related to a wind farm operating in Teesside, UK. Then, Section 3 will introduce different methods for central tendency estimation. Section 4 will analyze the results of numerical experiments on analytical and industrial cases. Then, the last section will present discussions and 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 our case, it is a chain of three numerical codes executed sequentially. As illustrated in Fig. 2, a simulation over a time period is the sequence of (1) turbulent wind speed field generation, (2) wind turbine simulation (computing various outputs including mechanical stress), and (3) post-processing to assess the fatigue damage of the structure.

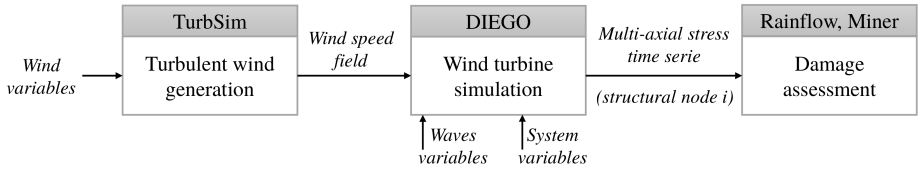

Figure 2: Diagram of the chained OWT simulation model

2.1 Numerical simulation model

This section describes more precisely the modeling hypotheses considered in this case. Starting from the turbulent wind field simulator TurbSim (developed by Jonkman (2009) from the National Renewable Energy Laboratory,

Table 1: Teesside Offshore Wind turbine 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

USA) that uses a Kaimal spectrum (Kaimal et al., 1972). To extrapolate the wind speed vertically, the shear is modeled by a power law. Since the wind field generation shows inherent stochasticity, each 10-minute long simulation is repeated with different pseudo-random seeds and the average damage over these repetitions is studied. This question was widely studied by some authors, (e.g., Slot et al. (2020)), who concluded that the six repetitions recommended by the IEC (2019) are insufficient to properly average this stochasticity. In the following, the simulations are repeated eleven times (allowing direct access to the median value). This number of repetitions was chosen as a compromise between the general number of simulations and the storage capacity of the generated simulations.

DIEGO (for "Dynamique Intégrée des Éoliennes et Génératrices Offshore[1]") is a code 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 our application, 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). Our study uses a DIEGO model of a Siemens SWT 2.3MW 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 Fig. 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 the following. To avoid numerical perturbations and reach the stability of the dynamical system, our simulation period is extended to 1000 seconds and the first 400 seconds are cropped in the post-processing step. This chained OWT numerical simulation model has been deployed on an EDF R&D HPC facility to benefit from parallel computing speed up (a single simulation on one CPU takes around 20 minutes).

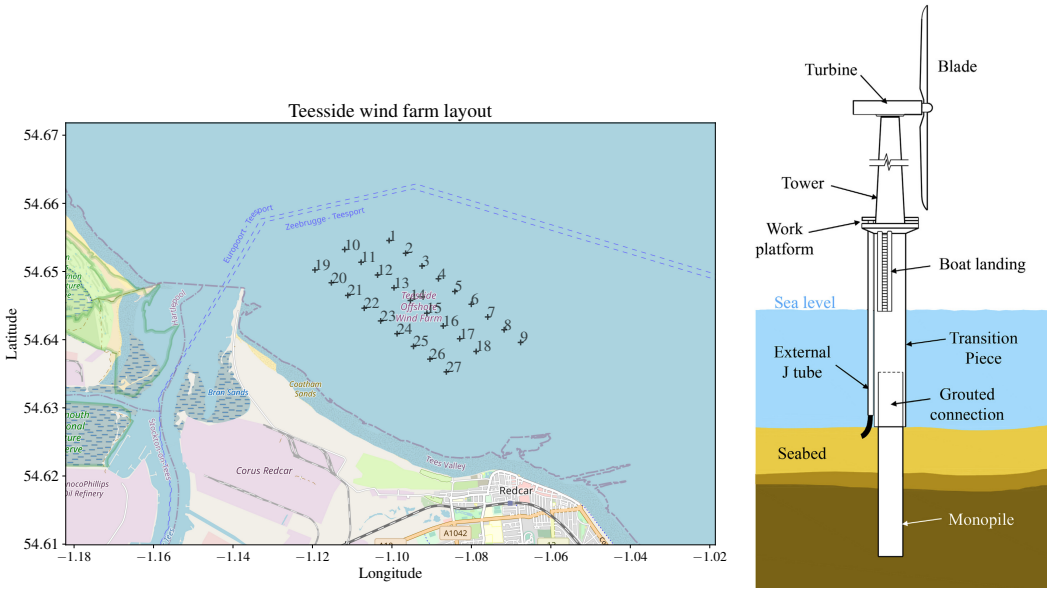

Figure 3: Teesside wind farm layout (left). Monopile OWT diagram (Chen et al., 2018) (right)

2.2 Measured environmental data

During the lifespan of a wind farm project, environmental data is 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, five years of wind data have been collected on the turbines which are not affected

[1] In english, "Integrated Dynamics of Wind Turbines and Offshore Generators".

by the wake on this site. Their acquisition system (usually called SCADA, for "Supervisory Control And Data Acquisition") have a sampling period of ten minutes. The wave data arise from a buoy placed in the middle of the farm. This data describes the physical features listed in Table 2.

The farm of Teesside is located close to the coast, making the environmental conditions very different depending on the direction (see the wind farm layout in Fig. 3). Since measures are also subject to uncertainties, a few checks were realized to ensure that the data were physically consistent. The truncation bounds defined in Table 2 were applied since this study is not interested in extreme values but central tendency estimation (i.e., mean behavior). In addition, a simple trigonometric transform is applied to each directional feature to take into account their cyclic structure. Finally, the remaining features are rescaled (i.e., using a min-max normalization). The matrix plot of the transformed data in Fig. 4 is an innovative plot named *copulogram*. A copulogram is an innovative plot as it decomposes the data between the effects of the marginals and those of the dependence between features. To do so, it represents the marginals with univariate kernel density estimation plots (diagonal), and the dependence structure with scatter plots in the ranked space (upper triangle). On the bottom triangle the scatter plots are set in the physical space, gathering the effects of the marginals and the dependencies. 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.

On Fig. 4, a large sample $\mathcal{S}$ (with a size $N = 10^4$) is randomly drawn from the entire Teesside data (with size $N_{\text{Teesside}} = 2 \times 10^5$), and plotted in grey. In the same figure, the orange matrix plot is a subsample of the sample $\mathcal{S}$, selected by kernel herding, a method that will be presented in Section 3. Visually, this orange subsample seems to match 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 multivariate environmental distribution $\mathbf{X} \in \mathcal{D}_{\mathbf{X}} \subset \mathbb{R}^p$, of density $f_{\mathbf{X}}$, and called *candidate set*. Contrarily to parametric approaches which can be used to describe the joint environmental uncertainty, this method intends to directly subsample from this large and representative dataset. This technique samples a joint distribution without modeling it. Indeed, a proper parametric model fit would be challenging for complex dependence structures such as the one plotted on Fig. 4. Li and Zhang (2020) built a parametric model of a similar multivariate distribution using vine copulas. Alternatively, a nonparametric approach coupling empirical Bernstein copula fitting with kernel density estimation of the marginals is introduced in Section 2.3.

Table 2: Description of the environmental data.

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

2.3 Non parametric fit with empirical Bernstein copula

The Sklar theorem (Joe, 1997) states that the multivariate distribution of any random vector $\mathbf{X} \in \mathbb{R}^p$ 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 distribution F and its marginals $\{F_i\}_{i=1}^p$, there exists a copula $C : [0,1]^p \to [0,1]$, such that:

$$F(x_1, \ldots, x_p) = C\left(F_1(x_1), \ldots, F_p(x_p)\right). \tag{1}$$

It allows us to divide the problem of fitting a joint distribution into two independent problems: fitting the marginals and fitting the copula. Additionally, 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 to uniformly approximate as closely as desired any continuous and real-valued function defined on a compact set (Weierstrass approximation theorem). Therefore, they are good candidates to approximate unknown copulas. This concept was introduced as *empirical Bernstein copula* (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 $C : [0,1]^p \to \mathbb{R}$ on a grid over the unit hypercube $G := \left\{\frac{0}{h_1}, \ldots, \frac{h_1}{h_1}\right\} \times \cdots \times \left\{\frac{0}{h_p}, \ldots, \frac{h_p}{h_p}\right\}$, $\mathbf{h} = (h_1, \ldots, h_p) \in \mathbb{N}^p$, is written as follows:

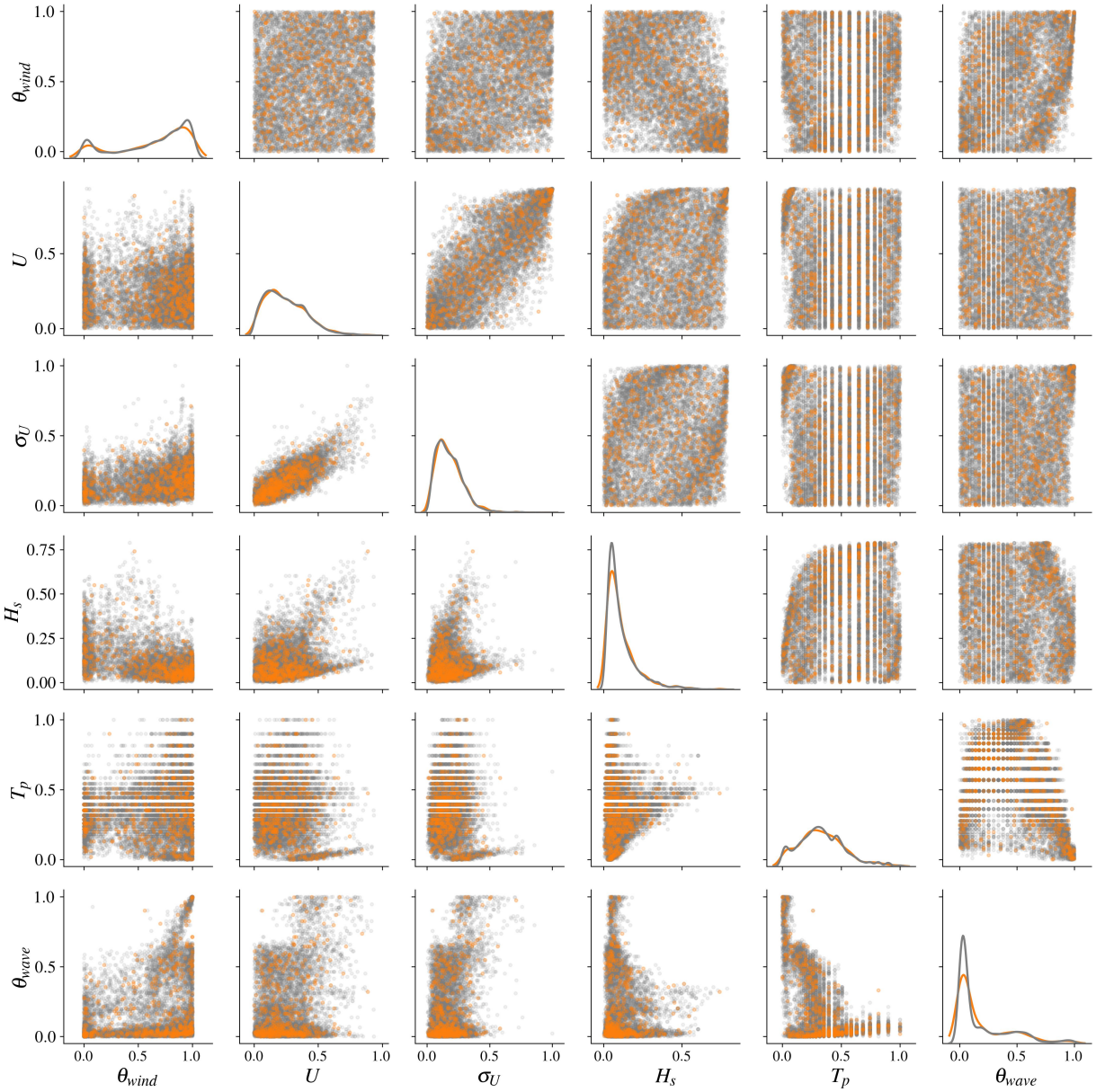

Figure 4: Copulogram of the Teesside measured data ($N = 10^4$ in grey), kernel herding subsample ($n = 500$ in orange). Marginals are represented by univariate kernel density estimation plots (diagonal), the dependence structure with scatter plots in the ranked space (upper triangle). Scatter plots on the bottom triangle are set in the physical space

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

with $\mathbf{u} = (u_1, \ldots, u_p) \in [0,1]^p$, and the Bernstein polynomial $P_{h,t}(u) := \frac{t!}{h!(t-h)!} u^h (1-u)^{t-h}$. When C is a copula, then $B_{\mathbf{m}}(C)$ is called "Bernstein copula". Therefore, the empirical Bernstein copula is an application of the Bernstein polynomial in Eq. (2) to the so-called "empirical copula". In practice, considering a sample $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\right\} \in \mathbb{R}^{np}$ and the associated ranked sample $\mathbf{R}_n = \left\{\mathbf{r}^{(1)}, \ldots, \mathbf{r}^{(n)}\right\}$, the corresponding empirical copula is written:

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

Provided a large enough learning set $\mathbf{X}_n$, the EBC combined with kernel density estimation for the marginals

fit well the environmental joint distribution related to the dataset in Fig. 4. Moreover, the densities of the EBC are available in an explicit form, making Monte Carlo or quasi-Monte Carlo generation easy. For a thorough presentation of this method and theoretical results regarding the 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 Fig. 2, a typical DIEGO simulation returns a 10-minute multiaxial stress time series at each node $i \in \mathbb{N}$ of the 1D meshed structure. Since fatigue laws are established for uniaxial stresses, the first step is to compute one equivalent Von Mises stress time series at each structural node.

However, the foundation and the tower of an OWT are a succession of tubes with various sections connected by bolted or welded joints. Our work studies the welded joints at the mudline level, identified as a critical area for the structure. To compute fatigue in this 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 Fig. 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 Fig. 5). According to 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 counting algorithm from the Python package `rainflow`[2]). For each identified stress cycle of amplitude s, the so-called "Stress vs. Number of cycles" curve (also called the "Wöhler curve") allows one to estimate the number N_c of similar stress cycles necessary to reach fatigue ruin:

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

Finally, a usual approach to compute the damage is to consider the fatigue contribution of each stress cycle identified using Miner's rule. Damage occurring during a 10-minute operating time is simulated and then scaled up to the OWT lifetime. More details regarding damage assessment are available in Appendix 5. For a realization of environmental conditions $\mathbf{x} \in \mathcal{D}_{\mathbf{X}}$, at a structural node i, an azimuth angle θ; k 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 $g_{i,\theta}(\mathbf{x}) : \mathcal{D}_{\mathbf{X}} \to \mathbb{R}_+$ by:

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

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

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

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 Fig. 6 (with a log scale). The log-damage presents a little asymmetry,so it is unlikely to be normally distributed.

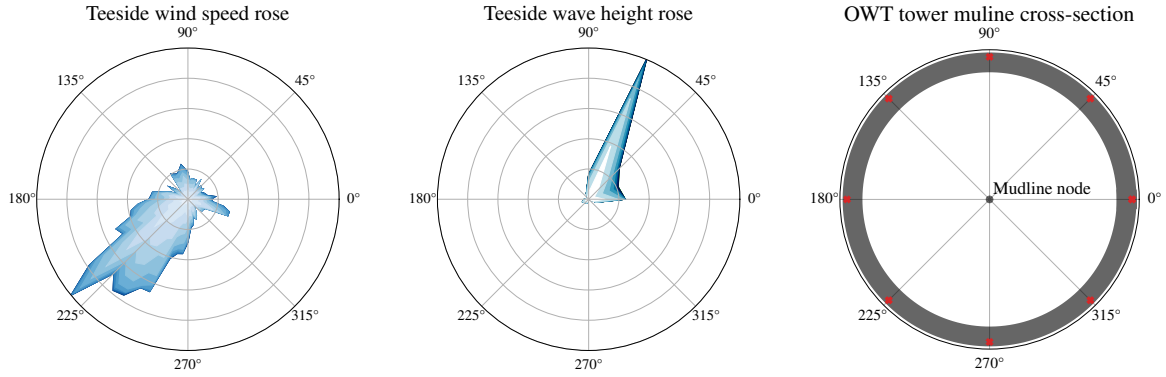

Figure 5: Angular distribution of the wind and waves with a horizontal cross-section of the OWT structure and the mudline

[2] https://github.com/iamlikeme/rainflow

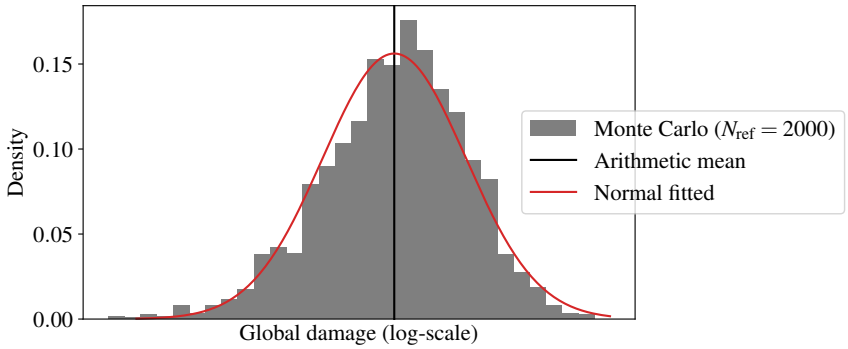

Figure 6: Histogram of the log-damage, at mudline, azimuth 45 deg. (Monte Carlo reference sample)

3 Numerical integration procedures for mean damage estimation

3.1 Quadrature rules and quasi-Monte Carlo methods

The present section explores different methods aiming at estimating the expected value of a function against a probability measure. Considering a measurable space $\mathcal{D}_{\mathbf{X}} \subset \mathbb{R}^p, p \in \mathbb{N}_+$, associated with a known Lebesgue measure μ, let us study the approximation of integrals of the form $\int_{\mathcal{D}_{\mathbf{X}}} g(\mathbf{x})\mathrm{d}\mu(\mathbf{x})$, with g the map $g(\mathbf{x}) : \mathcal{D}_{\mathbf{X}} \to \mathbb{R}$. 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 (see Eq. (6)). Some authors also named 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 $\mathbf{y}_n = \left\{g(\mathbf{x}^{(1)}), \ldots, g(\mathbf{x}^{(n)})\right\}$ of an input sample $\mathbf{X}_n = \left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\right\}$. Our numerical experiment framework often implies that the function g is costly to evaluate, making the realization number limited. A weighted arithmetic mean of the realizations $\left\{g(\mathbf{x}^{(1)}), \ldots, g(\mathbf{x}^{(n)})\right\}$ is called a *quadrature rule* with a set of unconstrained weights $\mathbf{w}_n = \{w_1, \ldots, w_n\} \in \mathbb{R}^n$:

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

For a given sample size n, our goal is to find a set of tuples $\left\{\mathbf{x}^{(n)}, w_i\right\}_{i=1}^{n}$ (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. In a recent work, Van den Bos (2020) applies a first family of numerical integration methods based on tensor products of quadrature rules to a similar industrial OWT use case. Unfortunately, the tensor formulation fails when inputs present a strong dependency structure and will not be studied in this paper. 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 $\left\{w_i = \frac{1}{n}\right\}_{i=1}^{n}$). 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 (Morokoff and Caflisch, 1995).

3.2 Kernel discrepancy

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:

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

As explained in Oates (2021), $V(g) = \sum_{\mathbf{u} \subseteq \{1,\ldots,p\}} \int_{[0,1]^{\mathbf{u}}} \left| \frac{\partial^{\mathbf{u}} g}{\partial \mathbf{x}_{\mathbf{u}}}(\mathbf{x}_{\mathbf{u}}, 1) \right| \mathrm{d}\mathbf{x}$, 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

target measure μ, the star discrepancy is a metric assessing how far from uniformity a sample $\mathbf{X}_n$ is. When generalizing to a non-uniform measure, a good quantization of μ should also lead to a good approximation of the quantity.

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 $\mathcal{H}(k)$ of functions $g : \mathcal{D}_{\mathbf{X}} \to \mathbb{R}$. Considering a symmetric and positive definite function $k : \mathcal{D}_{\mathbf{X}} \times \mathcal{D}_{\mathbf{X}} \to \mathbb{R}$, later called a "reproducing kernel" or simply a "kernel", an RKHS verifies the following axioms:

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

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 Fig. 7. For any measure μ, 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 is defined as:

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

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:

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

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 design $\mathbf{X}_n$. If $P_{\zeta_n}(\mathbf{x})$ is close to $P_\mu(\mathbf{x})$, it can be interpreted to mean that ζ_n is an adequate quantization or representation of μ by the discrete distribution ζ_n (and therefore lead 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.

Definition 1. The *energy* of a measure μ is defined as the integral of the potential P_μ against the measure, which leads to the following scalar quantity:

$$\varepsilon_\mu := \int_{\mathcal{D}_{\mathbf{X}}} P_\mu(\mathbf{x}) \mathrm{d}\mu(\mathbf{x}) = \iint_{\mathcal{D}_{\mathbf{X}}^2} k(\mathbf{x}, \mathbf{x}') \, \mathrm{d}\mu(\mathbf{x}) \mathrm{d}\mu(\mathbf{x}'). \tag{11}$$

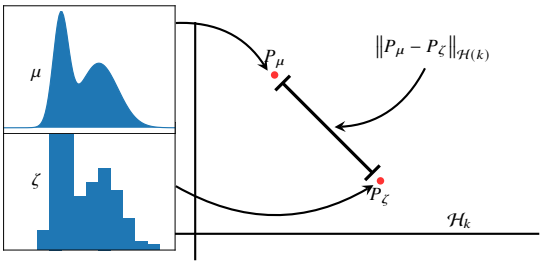

Figure 7: Kernel mean embedding of a continuous and discrete probability distribution

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)):

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

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

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

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:

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

According to the inequality in Eq. (12c), $\mathrm{MMD}_k(\mu,\zeta) = \left\|P_\mu - P_\zeta\right\|_{\mathcal{H}(k)}$, meaning that the MMD fully relies on the difference of potentials. Moreover, Sriperumbudur et al. (2010) defines a kernel as "characteristic kernel" when the following equivalence is true: $\mathrm{MMD}_k(\mu,\zeta) = 0 \Leftrightarrow \mu = \zeta$. This property makes the MMD a metric on $\mathcal{D}_\mathbf{X}$. The squared MMD has been used for other purposes than numerical integration: e.g., statistical testing (Gretton et al., 2006), and global sensitivity analysis (Da Veiga, 2015). It can be written as follows:

$$\begin{aligned}
\mathrm{MMD}_k(\mu,\zeta)^2 &= \left\|P_\mu(\mathbf{x}) - P_\zeta(\mathbf{x})\right\|^2_{\mathcal{H}(k)} && \text{(14a)}\\
&= \left\langle \left(P_\mu(\mathbf{x}) - P_\zeta(\mathbf{x})\right), \left(P_\mu(\mathbf{x}) - P_\zeta(\mathbf{x})\right)\right\rangle_{\mathcal{H}(k)} && \text{(14b)}\\
&= \left\langle P_\mu(\mathbf{x}), P_\mu(\mathbf{x})\right\rangle_{\mathcal{H}(k)} - 2\left\langle P_\mu(\mathbf{x}), P_\zeta(\mathbf{x})\right\rangle_{\mathcal{H}(k)} + \left\langle P_\zeta(\mathbf{x}), P_\zeta(\mathbf{x})\right\rangle_{\mathcal{H}(k)} && \text{(14c)}\\
&= \iint_{\mathcal{D}^2_\mathbf{X}} k(\mathbf{x},\mathbf{x}')\, \mathrm{d}\mu(\mathbf{x})\mathrm{d}\mu(\mathbf{x}') - 2\iint_{\mathcal{D}^2_\mathbf{X}} k(\mathbf{x},\mathbf{x}')\, \mathrm{d}\mu(\mathbf{x})\mathrm{d}\zeta(\mathbf{x}') + \iint_{\mathcal{D}^2_\mathbf{X}} k(\mathbf{x},\mathbf{x}')\, \mathrm{d}\zeta(\mathbf{x})\mathrm{d}\zeta(\mathbf{x}'). && \text{(14d)}
\end{aligned}$$

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:

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

3.3 Kernel herding sampling

Herein, the MMD is used to quantize the known target measure μ by a design sample $\mathbf{X}_n$. For practical reasons, design construction is done sequentially. Sequential strategies can also be 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, we fix the quadrature weights as uniform during the construction of the design $\mathbf{X}_n$.

Kernel herding (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 can be written as an optimization problem involving the following criterion over the point $\mathbf{x}^{(n+1)} \in \mathcal{D}_\mathbf{X}$:

$$\mathbf{x}^{(n+1)} \in \underset{\mathbf{x}\in\mathcal{D}_\mathbf{X}}{\arg\min} \left\{ \mathrm{MMD}_k\left(\mu, \frac{1}{n+1}\left(\delta(\mathbf{x}) + \sum_{i=1}^n \delta(\mathbf{x}^{(i)})\right)\right)^2 \right\}. \tag{16}$$

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. To see this, let us develop the MMD from Eq. (15):

$$\begin{aligned}
\mathrm{MMD}_k\left(\mu, \frac{1}{n+1}\left(\delta(\mathbf{x}) + \sum_{i=1}^n \delta(\mathbf{x}^{(i)})\right)\right)^2 &= \varepsilon_\mu - \frac{2}{n+1}\sum_{i=1}^{n+1} P_\mu\left(\mathbf{x}^{(i)}\right) + \frac{1}{(n+1)^2}\sum_{i,j=1}^{n+1} k\left(\mathbf{x}^{(i)},\mathbf{x}^{(j)}\right) && \text{(17a)}\\
&= \varepsilon_\mu - \frac{2}{n+1}\left(P_\mu(\mathbf{x}) + \sum_{i=1}^n P_\mu\left(\mathbf{x}^{(i)}\right)\right) && \text{(17b)}\\
&\quad + \frac{1}{(n+1)^2}\left(\sum_{i,j=1}^n k\left(\mathbf{x}^{(i)},\mathbf{x}^{(j)}\right) + 2\sum_{i=1}^n k\left(\mathbf{x}^{(i)},\mathbf{x}\right) - k(\mathbf{x},\mathbf{x})\right). && \text{(17c)}
\end{aligned}$$

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:

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

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 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:

$$\mathbf{x}^{(n+1)} \in \underset{\mathbf{x}\in\mathcal{D}_{\mathbf{X}}}{\arg\min} \left\{ P_{\zeta_n}(\mathbf{x}) - P_{\mu}(\mathbf{x}) \right\}. \tag{19}$$

Remark 2. In practice, the optimization problem is solved by a brute-force approach on a fairly dense finite subset $\mathcal{S} \subseteq \mathcal{D}_{\mathbf{X}}$ 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 $P_{\mu}(\mathbf{x}) \approx \frac{1}{N}\sum_{i=1}^{N} k\left(\mathbf{x}^{(i)}, \mathbf{x}\right)$.

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 kernel choice. A kernel is defined, both by 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 kernel herding 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 Gaussian process 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), $\theta_i = n^{-1/p}, i \in \{1, \ldots, p\}$.

Table 3: Kernels considered in the following numerical experiments.

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

Fig. 8 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.

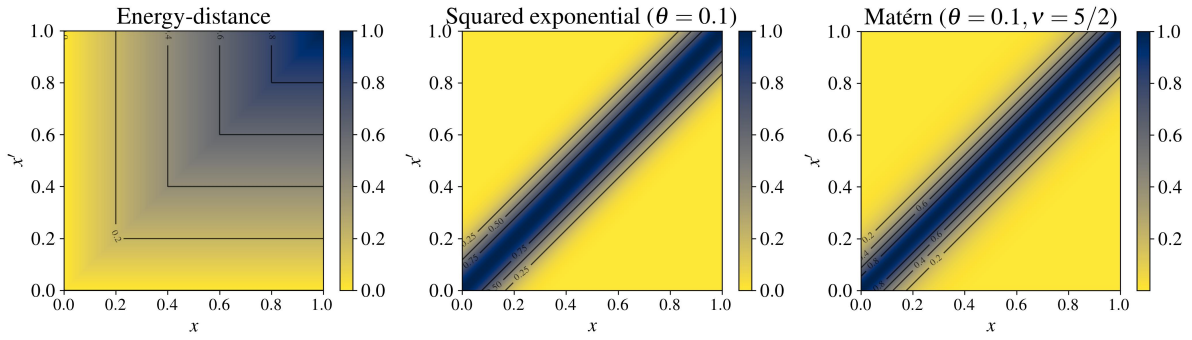

Figure 8: Kernel illustrations (left to right: energy-distance, squared exponential, and Matérn 5/2)

To illustrate the sequential sampling of a complex distribution, Fig. 9 shows three nested kernel herding 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 seems to build a parsimonious design between each mode of the distribution. The candidate set (in light grey) 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 kernel herding is the estimation of the potentials, which becomes costly in high dimension.

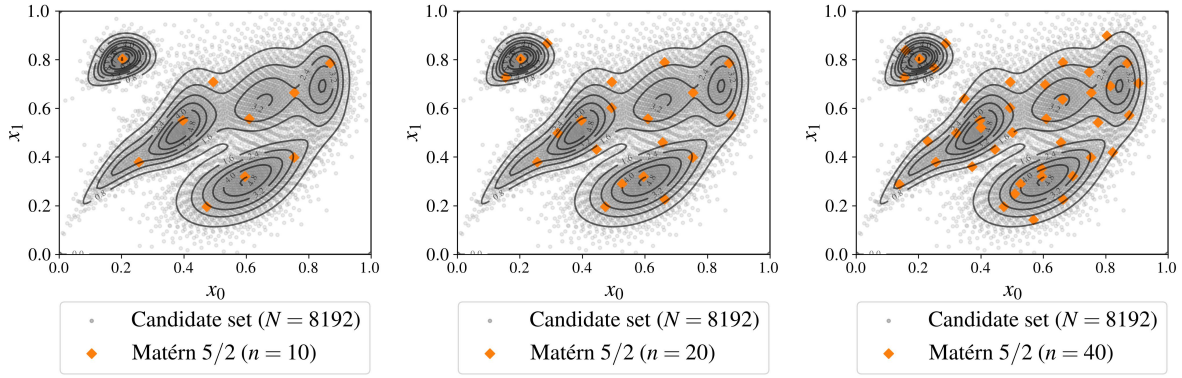

Figure 9: Sequential kernel herding for increasing design sizes ($n \in \{10, 20, 40\}$) built on a candidate set of $N = 8196$ points drawn from a complex Gaussian mixture μ

Other approaches take advantage of the progressive knowledge acquired sequentially on g 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 Gaussian process (or Kriging) metamodel. To solve a probabilistic integration problem, the concept of Bayesian quadrature is introduced in the following.

3.4 Bayesian quadrature

Gaussian processes for Bayesian quadrature Kernel methods and Gaussian processes present a lot of connections and equivalences, thoroughly reviewed by Kanagawa et al. (2018). In numerical integration, Gaussian processes have been used to build quadrature rules in the seminal paper of O'Hagan (1991), introducing the concept of *Bayesian quadrature* (BQ). Let us recall the probabilistic integration problem $I_\mu(g) = \int_{\mathcal{D}_\mathbf{X}} g(\mathbf{x})\mathrm{d}\mu(\mathbf{x})$ (introduced 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 the distribution of the output.

Let us assume, adopting a Bayesian point of view, that ξ is a stochastic process describing the uncertainty affecting the knowledge about the true function g. Let ξ be a Gaussian process (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 $\xi_n := (\xi|\mathbf{y}_n) \sim \mathcal{GP}(\eta_n, s_n^2)$ has been conditioned on the function observations $\mathbf{y}_n = \left[g\left(\mathbf{x}^{(1)}\right), \ldots, g\left(\mathbf{x}^{(n)}\right)\right]^\top$ 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)):

$$\left\{\begin{array}{ll} \eta_n(\mathbf{x}) & := \mathbf{k}_n^\top(\mathbf{x})\mathbf{K}_n^{-1}\mathbf{y}_n \\ s_n^2(\mathbf{x}) & := k_n(\mathbf{x},\mathbf{x}) - \mathbf{k}_n^\top(\mathbf{x})\mathbf{K}_n^{-1}\mathbf{k}_n(\mathbf{x}) \end{array}\right. \tag{20}$$

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

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

$$I_\mu(\xi_n) = \int_{\mathcal{D}_\mathbf{X}} (\xi(\mathbf{x})|\mathbf{y}_n)\mathrm{d}\mu(\mathbf{x}) = \int_{\mathcal{D}_\mathbf{X}} \xi_n(\mathbf{x})\mathrm{d}\mu(\mathbf{x})\,. \tag{21}$$

Fig. 10 provides a one-dimensional illustration of the Bayesian quadrature of an unknown function (dashed black curve) against a given input measure μ (with corresponding grey distribution at the bottom). For an arbitrary design, one can fit a Gaussian process model, interpolating the function observations (black crosses). Then, multiple trajectories of this conditioned Gaussian process ξ_n are drawn (orange curves) whilst its mean function, also called "predictor", is represented by the red curve. Therefore, the input measure μ is propagated through the conditioned Gaussian process to obtain the random variable $I_\mu(\xi_n)$, 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).

Optimal weights computed by Bayesian quadrature Taking the random process ξ_n as Gaussian conveniently implies that its posterior distribution $a_\mu(\xi_n)$ is also Gaussian. This comes from the linearity of the infinite sum of realizations of a Gaussian process. The posterior distribution is described in a closed form through its mean

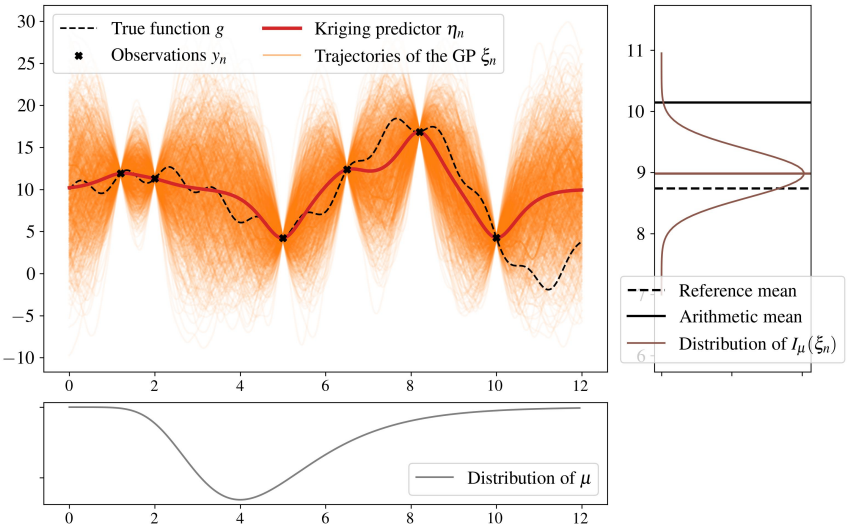

Figure 10: Bayesian quadrature on a one-dimensional case

and variance by applying Fubini's theorem (see the supplementary materials from Briol et al. (2019) for the proof regarding the variance):

$$\bar{y}_n^{\mathrm{BQ}} := \mathbb{E}\left[I_\mu(\xi_n)|\mathbf{y}_n\right] = \int_{\mathcal{D}_\mathbf{X}} \eta_n(\mathbf{x})\mathrm{d}\mu(\mathbf{x}) = \left[\int_{\mathcal{D}_\mathbf{X}} \mathbf{k}_n^\top(\mathbf{x})\mathrm{d}\mu(\mathbf{x})\right]\mathbf{K}_n^{-1}\mathbf{y}_n = P_\mu(\mathbf{X}_n)\mathbf{K}_n^{-1}\mathbf{y}_n, \tag{22}$$

$$\left(\sigma_n^{\mathrm{BQ}}\right)^2 := \mathrm{Var}\left(I_\mu(\xi_n)\right) = \iint_{\mathcal{D}_{\mathbf{X}^2}} k_n(\mathbf{x},\mathbf{x}')\mathrm{d}\mu(\mathbf{x})\mathrm{d}\mu(\mathbf{x}') = \varepsilon_\mu - P_\mu(\mathbf{X}_n)\mathbf{K}_n^{-1}P_\mu(\mathbf{X}_n)^\top. \tag{23}$$

Where $P_\mu(\mathbf{X}_n)$ is the row vector of potentials $\left[\int k_n(\mathbf{x},\mathbf{x}^{(1)})\mathrm{d}\mu(\mathbf{x}),\ldots,\int k_n(\mathbf{x},\mathbf{x}^{(n)})\mathrm{d}\mu(\mathbf{x})\right]$, and ε_μ is given in Eq. (1). As in the one-dimensional example presented in Fig. 10, the expected value of $I_\mu(\xi_n)$ expressed in Eq. (22) is a direct estimator of the quantity of interest Eq. (7). The so-called "Bayesian quadrature estimator" appears to be a simple linear combination of the observations by taking the row vector of "optimal weights" as:

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

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, either 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 Gaussian processes 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 β_n is used, which leads to:

$$\mathbb{E}\left[I_\mu(\xi_n)\right] = \beta_n + P_\mu(\mathbf{X}_n)\mathbf{K}_n^{-1}\left(\mathbf{y}_n - \beta_n\mathbf{1}_n\right). \tag{25}$$

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

$$\bar{y}_n^{\mathrm{BQ}} \in \left[\bar{y}_n^{\mathrm{BQ}} - 2\sigma_n^{\mathrm{BQ}}, \bar{y}_n^{\mathrm{BQ}} + 2\sigma_n^{\mathrm{BQ}}\right]. \tag{26}$$

Variance-based Bayesian quadrature 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 Bayesian quadrature $\mathrm{Var}\left(I_\mu(\xi_n)\right)$ is the MMD between the target distribution μ and $\zeta_n = \sum_{i=1}^n \mathbf{w}_{\mathrm{BQ}}^{(i)}\delta(\mathbf{x}^{(i)})$. The proof is reproduced below (as well as in Proposition 6.1 from Kanagawa et al. (2018)):

$$\begin{aligned}
\mathrm{Var}\left(I_\mu(\xi_n)\right) &= \mathbb{E}\left[\left(I_\mu(\xi_n) - I_{\zeta_n}(\xi_n)\right)^2\right] && \text{(27a)}\\
&= \mathbb{E}\left[\left(\langle \xi_n, P_\mu\rangle_{\mathcal{H}(k)} - \langle \xi_n, P_{\zeta_n}\rangle_{\mathcal{H}(k)}\right)^2\right] && \text{(27b)}\\
&= \mathbb{E}\left[\langle \xi_n, P_\mu - P_{\zeta_n}\rangle_{\mathcal{H}(k)}^2\right] && \text{(27c)}\\
&= \|P_\mu - P_{\zeta_n}\|_{\mathcal{H}(k)}^2 && \text{(27d)}\\
&= \mathrm{MMD}_k(\mu,\zeta_n)^2. && \text{(27e)}
\end{aligned}$$

Note that the transition from equation (27c) to (27d) relies on the property stating that if ξ is a standard Gaussian process then $\forall g \in \mathcal{H}(k) : \langle \xi, g \rangle_{\mathcal{H}(k)} \sim \mathcal{N}(0, \|g\|^2_{\mathcal{H}(k)})$. The method that sequentially builds a quadrature rule by minimizing this variance is called by the authors "Sequential Bayesian Quadrature" (SBQ). According to the previous proof, this criterion can be seen as an optimally-weighted version of the kernel herding criterion, as stated in the title of the paper from Huszár and Duvenaud (2012). Later, Briol et al. (2015) proved the weak convergence of $I_\mu(\xi_n)$ towards the target integral. Closer to wind turbines applications, Huchet (2019) 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 both scale hardly in high dimension.

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) presents 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. 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 quasi-Monte Carlo Sobol' sequences (abbreviation by QMC in the next figures), and kernel herdings with the three kernels defined in Table 3. Notice that the quasi-Monte Carlo designs are first generated on the unit cube, then transformed using the generalized Nataf transformation to follow the target distribution (Lebrun and Dutfoy, 2009). Additionally, the performances of kernel herding for both uniform and optimally-weighted Eq. (25) estimators are compared.

The kernel-based sampling and BQ methods were implemented in a new open-source Python package named `otkerneldesign`[3]. This development mostly relies on OpenTURNS[4], an "Open source initiative for the Treatment of Uncertainties, Risks'N Statistics", see Baudin et al. (2017). The following numerical experiments are available in the Git repository named `ctbenchmark`[5].

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 (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 Fig. 11 (toy-case #1) and Fig. 12 (toy-case #2). Kernel herding samples with kernels defined in Table 3 (squared exponential in green, Matérn in orange, and energy-distance in red), are compared with a quasi-Monte Carlo sample (Sobol' sequences in grey). 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.

Table 4: Analytical toy-cases

Toy-case #1	$dim = 2$	$g_1(\mathbf{x}) = x_1 + x_2$	Gaussian mixture from Fig. 9
Toy-case #2	$dim = 10$	$g_2(\mathbf{x}) = \prod_{i=1}^{10} \frac{\lvert 4x_i - 2 \rvert + a_i}{1 + a_i}, \{a_i = 2\}_{i=1}^{10}$	Gaussian $\mathcal{N}(\mathbf{0.5}, \mathbf{I}_{10})$

[3] `https://efekhari27.github.io/otkerneldesign/master/index.html`
[4] `https://openturns.github.io/www/`
[5] `https://github.com/efekhari27/ctbenchmark`

Remark 4. Different kernels are used in these numerical experiments. First, the generation kernel, used by the kernel herding 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).

About toy-case #1. 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 to convergence rate.

About toy-case #2. 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-weighted KH does slightly better. Moreover, the results confirm that $\mathrm{MMD}_{\mathrm{BQ}} < \mathrm{MMD}_{\mathrm{unif}}$ for all our experiments. The application of optimal-weights to the quasi-Monte Carlo sample slightly improves the estimation on 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, recalled in Eq. (8).

4.2 Application to the Teesside wind turbine fatigue estimation

Before analyzing the performance of the KH on the industrial application, let us notice that the copulogram Fig. 14 seems to be in line with the global sensitivity analysis presented in Murcia et al. (2018) and Li and Zhang (2020). In particular, the fact that the scatter plot of mean wind speed vs. 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 high effective dimension, however, the input dependence structure is challenging and the damage assessment induces strong nonlinearities (see Eq. (4)).

Crude Monte Carlo and kernel herding both subsample directly from a large dataset (previously referred to as candidate set). Unlike them, quasi-Monte Carlo generates a uniform sample in the unit hypercube, which can then be transformed according to a target distribution. In our case, this distribution is only known empirically via the candidate set. Since its dependence structure is complex (see Fig. 4), a parametric model might fit the distribution poorly (and therefore lead to a poor quasi-Monte Carlo estimation of the quantity). Then, a nonparametric fit using the empirical Bernstein copula (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). Subsequently, quasi-Monte Carlo sampling is applied to this nonparametric model. These two approaches are summarized in Fig. 13, showing a practical advantage to the subsampling methods.

The results presented are compared in the following to a reference Monte Carlo sample with a confidence interval computed by bootstrap (see Fig. 15). The performance of the KH is good: it quickly converges towards the confidence interval of the Monte Carlo obtained with the reference sample. In addition, the Bayesian quadrature estimator also offers a posteriori prediction interval, which can reassure the user. The BQ prediction intervals are smaller that 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 kernel, 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. Finally, combining nonparametric fitting with quasi-Monte Carlo sampling also delivers good results as long as the fitting step does not introduce a bias.

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. This can lead, in practice, to poor uncertainty propagation under the constraint of a simulation budget. This industrial use case induces two practical constraints. First, active learning methods are hard to set up on such a numerical model, and they restrict the use of high-performance computers. Second, the input distribution of the environmental conditions presents a complex dependence structure, hard to model with parametric approaches.

In this paper, the association of kernel herding sampling with Bayesian quadrature for central tendency has been both explored theoretically and numerically. This method fits with the practical constraints induced by the industrial use case. Kernel herding sampling 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. This work provides an MMD-based upper bound on numerical integration absolute error. Kernel herding and Bayesian quadrature both aim at finding the quadrature rule minimizing the MMD, and therefore the absolute integration error. The numerical experiments confirmed that the MMD is an appropriate criterion since it

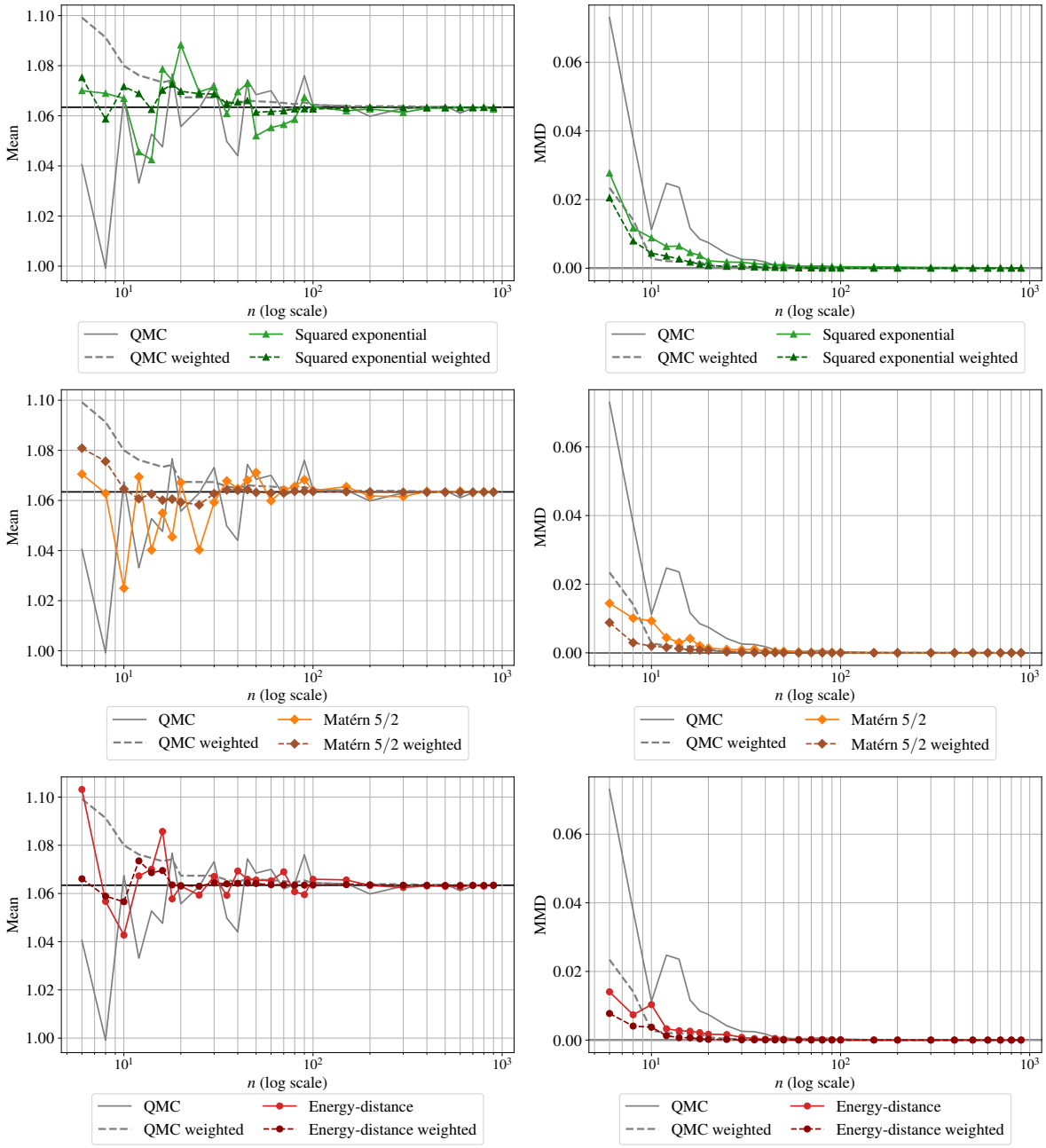

Figure 11: Analytical benchmark results on the toy-case #1

leads to results being better or equivalent to quasi-Monte Carlo sampling. This numerical benchmark relied on the Python package, called `otkerneldesign`, implementing the methods.

The limits of this method are reached when the problem dimension increases considerably. Moreover, it showed to be sensitive to the choice of the kernel and its tuning (although good practices were offered). From a methodological point of view, further interpretation of the impact of the different kernels should be explored. Then, the kernel herding sampling could be used to estimate quantiles, following the work on randomized quasi-Monte Carlo for quantiles of Kaplan et al. (2019). Kernel herding could also be used to quantize conditional distributions, using the conditional kernel mean embedding concept reviewed by Klebanov et al. (2020). Regarding the industrial use case, the next step is to realize a reliability analysis by considering another group of random variables (related to the wind turbine). Among other ideas, our upcoming work could explore a reliability-oriented sensitivity analysis by adapting recent kernel-based sensitivity indices (Marrel and Chabridon, 2021) to the sensitivity of a failure probability.

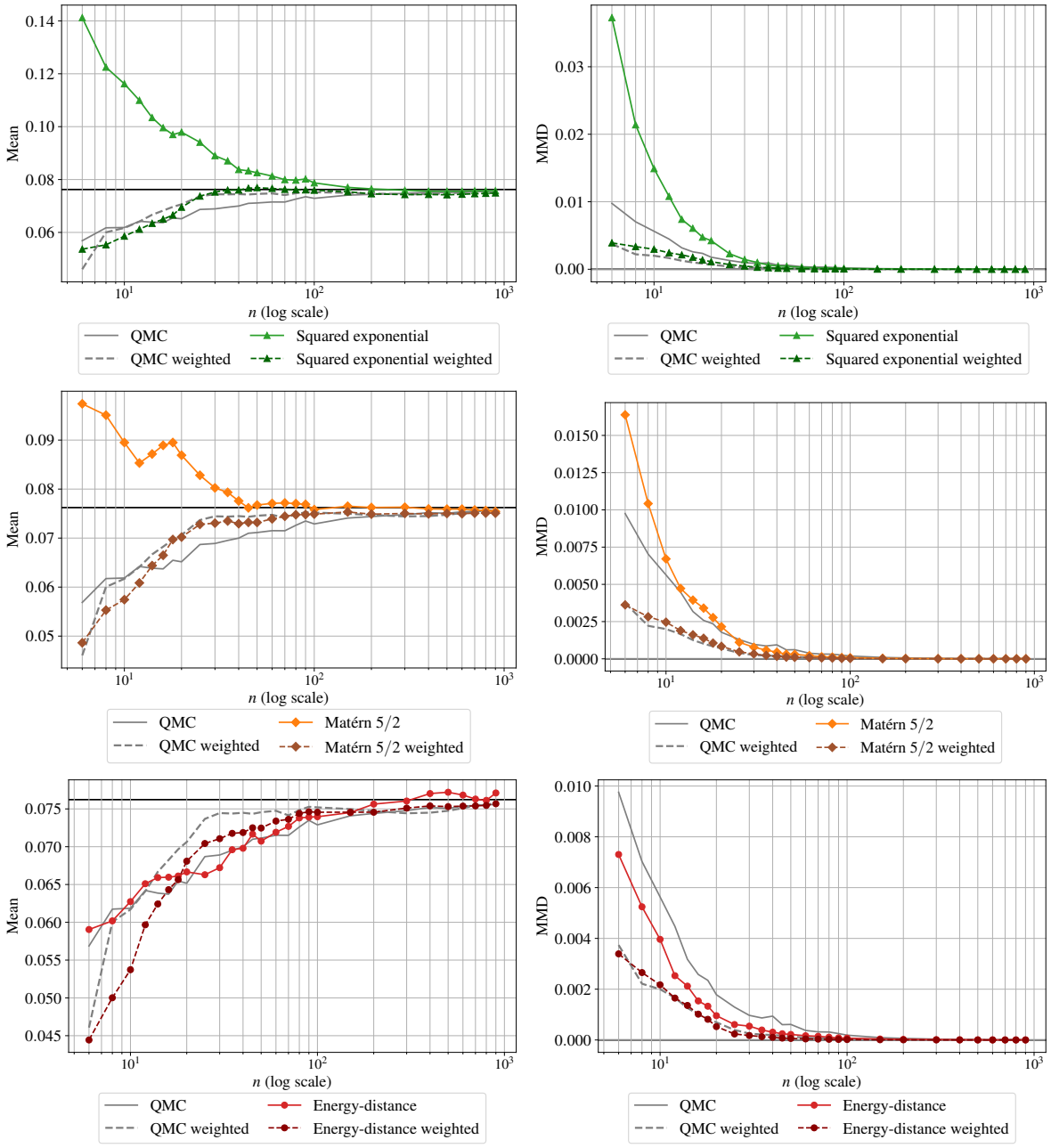

Figure 12: Analytical benchmark results on the toy-case #2

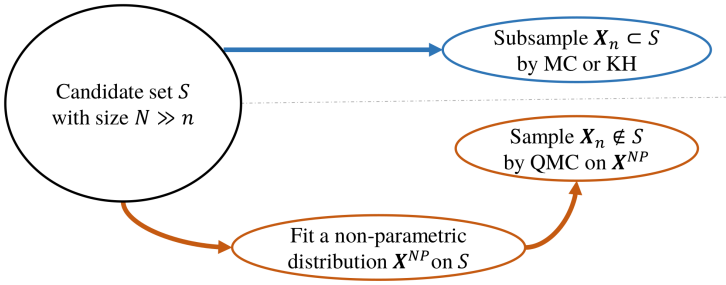

Figure 13: Sampling techniques used for the industrial use case

Appendix A: Multiaxial fatigue damage

This section discusses the damage assessment defined by the standard DNV-GL (2016a). Damage assessment can be divided into four steps:

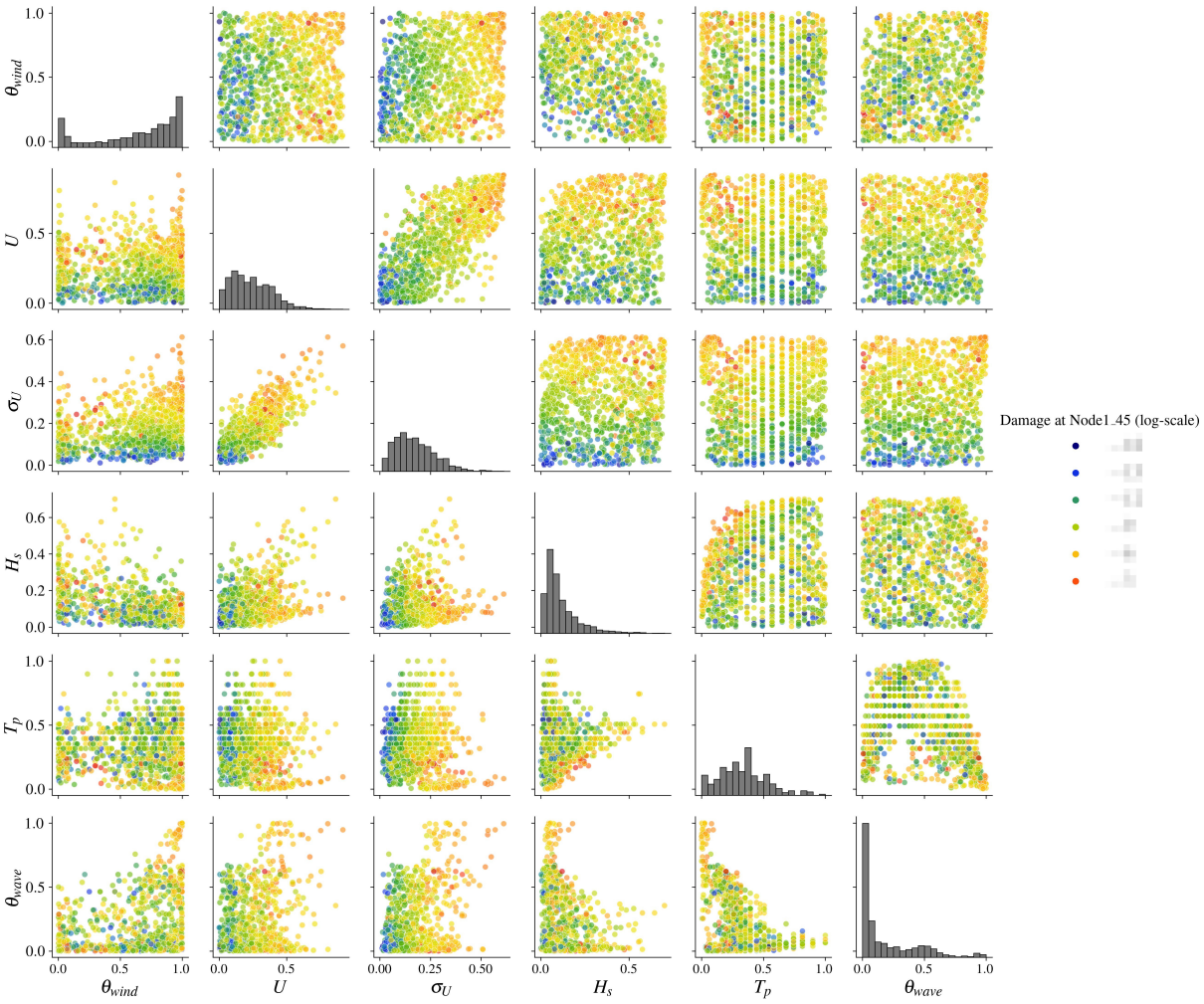

Figure 14: Copulogram of the kernel herding design of experiments with corresponding outputs in color (log-scale) on the Teesside case ($n = 10^3$). The highest values are in red while the lowest values are in blue. Marginals are represented by histograms (diagonal), the dependence structure with scatter plots in the ranked space (upper triangle). Scatter plots on the bottom triangle are set in the physical space.

1. Compute the equivalent Von Mises stress time series;
2. Identify the stress cycles using rainflow counting;
3. Define the Stress - Number of cycles curve corresponding to the material;
4. Compute the damage using Miner's rule.

Since stress is multiaxial in the local coordinate system, the equivalent Von Mises stress is computed, turning a multiaxial stress into an equivalent uniaxial stress. This first step is necessary since fatigue laws are mostly established for uniaxial stresses. The following expression gives the Von Mises stress for a Cauchy stress tensor:

$$\sigma_{\text{VM}} = \sqrt{\frac{1}{2}\left[(\sigma_{11}-\sigma_{22})^2 + (\sigma_{22}-\sigma_{33})^2 + (\sigma_{33}-\sigma_{11})^2\right] + 3\left(\sigma_{12}^2 + \sigma_{23}^2 + \sigma_{31}^2\right)}. \tag{28}$$

Note that in the mechanical model, the hypothesis of "plane strain" is considered, meaning that the Cauchy stress tensor $\underline{\underline{\sigma}}$ can be written as below:

$$\underline{\underline{\sigma}} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & 0 \\ \sigma_{21} & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{pmatrix}, \tag{29}$$

which simplifies the expression of the Von Mises stress:

$$\sigma_{\text{VM}} = \sqrt{\frac{1}{2}\left[(\sigma_{11}-\sigma_{22})^2 + (\sigma_{22}-\sigma_{33})^2 + (\sigma_{33}-\sigma_{11})^2\right] + 3\sigma_{12}^2}. \tag{30}$$

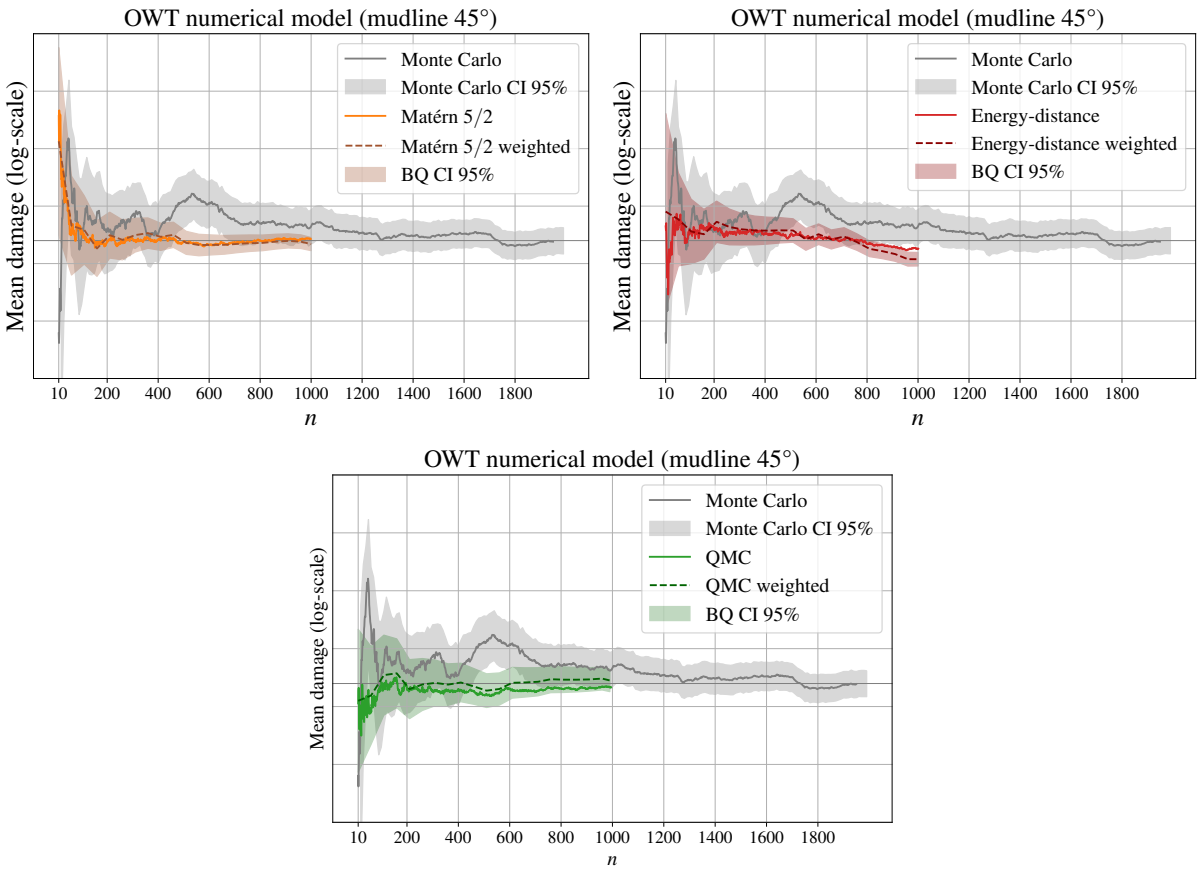

Figure 15: Mean estimation convergence (at the mudline, azimuth $\theta = 45$ deg.) on the Teesside case

After computing the equivalent Von Mises stress time serie, one can identify the fatigue loading cycles. The usual method to identify fatigue stress cycles is rainflow counting (Dowling, 1972). Fatigue stress cycles are only defined by their amplitude (also called "range"), regardless of their mean value or their chronology. Rainflow counting returns a list of stress ranges identified denoted by s in what follows. The "Stress - Number of cycle" curve (also called "Wöhler curve") allows one to estimate the number of similar stress cycles necessary to reach a fatigue ruin for a defined stress cycle amplitude. A well admitted simplification of the Wöhler curve is to consider it as log-linear on two parts. The Wöhler curve is generally defined as:

$$\log(N_c) = \log(a) - m \log(s). \tag{31}$$

One can therefore introduce the following function for each segment of the Wöhler curve:

$$N_c = W(s) = \begin{cases} a_1 s^{-m_1}, & \text{for } s \in [s_{\min}, s_e] \\ a_2 s^{-m_2}, & \text{for } s \in [s_e, s_{\max}] \end{cases}$$

Where N_c is the predicted number of cycles to failure for stress range s, m is the negative inverse slope of the Wöhler curve, $\log(a)$ is the intercept of log N-axis by the Wöhler curve, $s_{\min}$ is the minimal (resp. maximal) stress range identified by the rainflow counting, and s_e is the stress range axis of the intersection of the two log-lines formed by the Wöhler curve. Also interpreted as the endurance limit of the material. Note that, according to the standards DNV-GL (2016a), the Wöhler curve is altered for welded joints by taking into account the weld plate's thickness. In our case the considered Wöhler curve is defined by values given in the second line from Table 5, i.e.:

$$N_c = W(s) = \begin{cases} a_1 \left(s (\frac{t}{t_{\text{ref}}})^h \right)^{-m_1}, & \text{for } s \in [s_{\min}, s_e] \\ a_2 \left(s (\frac{t}{t_{\text{ref}}})^h \right)^{-m_2}, & \text{for } s \in [s_e, s_{\max}] \end{cases} \tag{32}$$

With t_{ref} the reference thickness (for tubular welded joints $t_{\text{ref}} = 25$ mm); t the plate thickness, and h the thickness exponent.

Damage is a physical measure of the structure's fatigue. An approach to compute the damage, is to consider the fatigue contribution of each stress cycle according to the Wöhler curve. Miner's rule defines a damage by summing

Table 5: Wöhler curve numeric values for tubular joints (source: DNV-GL (2016a))

Environment	m_1	$log(a_1)$	m_2	$log(a_2)$	h
Air	3.0	12.48	5.0	16.13	0.25
Seawater with cathodic protection	3.0	12.18	5.0	16.13	0.25
Seawater free corrosion	3.0	12.03	3.0	12.03	0.25

the fatigue contributions of the stress cycles:

$$y = \sum_{j=1}^{k} \frac{1}{N_{\mathrm{c}}^{(j)}} = \sum_{j=1}^{k} \frac{1}{W\left(s^{(j)}\right)} \tag{33}$$

When using Miner's rule, an alternative is to use bins to classify the stress ranges as an histogram. Our intuition is that choosing a number of bins too small will lead to a poor approximation of the damage. DNV-GL (2016a) standards recommend to use at least 20 bins for damage computation. Few tests were performed on a defined simulation model, for a fixed input stress time serie. A reference damage was computed by applying the Miner's rule to all stress range identified by rainflow counting ($\{n_i = 1\}_{i=1}^{k}$). Then, for various numbers of bins, stress ranges were classified and Miner's rule applied to these classes. The conclusion is that this method leads to a relative error around 1% for 500 bins.

Acknowledgments We are grateful for the technical assistance from various contributors to the HIPERWIND European project, A. Lovera, N. Dimitrov, and M. Capaldo. Moreover, we would like to thank S. Da Veiga, L. Pronzato and M. Munoz Zuniga for they fruitful discussions about the several topics found in this paper.

Funding Statement 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.

Competing Interests None.

Data Availability Statement Replication data and code of the analytical benchmark can be found in the GitHub repository: `https://github.com/efekhari27/ctbenchmark`.

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

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

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., Dutfoy, 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., editors, *Springer Handbook on Uncertainty Quantification*, pages 2001–2038. Springer.

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 *Advances in Neural Information Processing Systems*.

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 *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pages 109 – 116. AUAI Press.

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., 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, N. E. (1972). Fatigue Failure Predictions for Complicated Stress-Strain Histories. *Journal of Materials, JMLSA*, 7:71 – 87.

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*, pages 837–846.

Fekhari, E., Iooss, B., Muré, J., Pronzato, L., and Rendas, J. (2023). Model predictivity assessment: incremental test-set selection and accuracy evaluation. In Salvati, N., Perna, C., Marchetti, S., and Chambers, R., editors, *Studies in Theoretical and Applied Statistics, SIS 2021, Pisa, Italy, June 21-25*. Springer Cham.

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, K. M., Rasch, M., Schölkopf, B., and Smola, A. (2006). A kernel method for the two-sample-problem. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, page 513–520.

Hansen, M. and Henriksen, L. (2013). *Basic DTU Wind Energy controller*. Number 0028 in DTU Wind Energy E. DTU Wind Energy.

Huchet, Q. (2019). *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*, pages 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*. 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. In *Advances in Neural Information Processing Systems*, volume 32.

Kanagawa, M., Hennig, P., Sejdinovic, D., and Sriperumbudur, B. (2018). Gaussian processes and kernel methods: A review on connections and equivalences. preprint.

Kaplan, Z., Li, Y., Nakayama, M., and Tuffin, B. (2019). Randomized quasi-monte carlo for quantile estimation. In *2019 Winter Simulation Conference (WSC)*, pages 428–439.

Kim, T., Natarajan, A., Lovera, A., Julan, 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 HIPERWIND 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. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38, pages 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.

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.

Mak, S. and Joseph, V. (2018). Support points. *The 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, W. J. and Caflisch, R. E. (1995). Quasi-monte carlo integration. *Journal of Computational Physics*, 122(2):218–230.

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.

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.

Nagler, T., Schellhase, C., and Czado, C. (2017). Nonparametric estimation of simplified vine copula models: comparison of methods. *Dependence Modeling*, 5:99–120.

Oates, C. J. (2021). Minimum Discrepancy Methods in Uncertainty Quantification. Lecture Notes at École Thématique sur les Incertitudes en Calcul Scientifique (ETICS21), https://www.gdr-mascotnum.fr/etics.html.

O'Hagan, A. (1991). Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29:245–260.

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. (2021). Validation design I: construction of validation designs via kernel herding. preprint, https://arxiv.org/abs/2112.05583.

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*. MIT Press.

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. *The Annals of Statistics*, 41:2263–2291.

Slot, R. M., Sørensen, J. D., Sudret, B., Svenningsen, L., and Thøgersen, M. L. (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. *J. Mach. Learn. Res.*, 11:1517–1561.

Székely, G. J. and Rizzo, M. L. (2013). Energy statistics: A class of statistics based on distances. *Journal of Statistical Planning and Inference*, 143:1249 – 1272.

Teixeira, R., O'Connor, A., Nogal, M., Nandakumar, K., and Nichols, J. (2017). Analysis of the design of experiments of offshore wind turbine fatigue reliability design with Kriging surfaces. *Procedia Structural Integrity*, 5:951 – 958.

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. (to appear in 2023). A joint probability distribution model for multivariate wind and wave conditions. In *Proceedings of the ASME 2023 42th International Conference on Ocean, Offshore and Arctic Engineering*.

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.