UiO : University of Oslo

Christian Agrell

Probabilistic machine learning and phenomenological knowledge

Developments for optimization under uncertainty in safety-critical systems

Thesis submitted for the degree of Philosophiae Doctor

Department of Mathematics
Faculty of Mathematics and Natural Sciences

Group Research and Development
DNV

2021

Series of dissertations submitted to the
Faculty of Mathematics and Natural Sciences, University of Oslo
No. 2421

ISSN 1501-7710

Cover: Hanne Baadsgaard Utigard.
Print production: Reprosentralen, University of Oslo.

Preface

This thesis is submitted in partial fulfillment of the requirements for the degree of *Philosophiae Doctor* at the University of Oslo. The research was conducted at the University of Oslo and at DNV, under the supervision of Professor Arne Huseby and Professor Fred Espen Benth from the University of Oslo, together with Dr. Frank Børre Pedersen and Dr. Simen Eldevik from DNV. This work was supported by the Norwegian Research Council through grant 276282 and DNV Group Research and Development.

In this thesis we explore how phenomenological knowledge can be exploited to alleviate some of the challenges with applying data-driven modelling in safety-critical systems. The thesis is a collection of six papers, concerning development of methods, theory and algorithms associated with probabilistic modelling of safety-critical systems, for applications where phenomenological knowledge and data can be combined. The papers are preceded by an introductory part that provides motivation, background and context for the work.

Acknowledgements

I would like to thank my supervisors Arne Huseby and Fred Espen Benth for all their help and advice with this PhD. Thank you for always taking the time to discuss, and for including me in the risk and stochastics group at the university. A special thanks goes to Kristina Rognlien Dahl, for the close collaboration that led to two of the papers in this thesis.

From DNV I owe a special thanks to Simen Eldevik and Frank Børre Pedersen for their guidance and support throughout the project, and for enabling the project in the first place. During my time in Group Research and Development, I have spent almost every day working on problems that are challenging, interesting and relevant. Problems that I find engaging on both a professional and personal level. This is certainly a privilege and I am deeply grateful for the opportunity. Thank you for making this happen.

I would also like to thank my other colleagues from DNV. In particular Andreas Hafver, Odin Gramstad and Erik Vanem, for our joint efforts and collaboration that have been truly rewarding. I look forward to continued cooperation on new interesting problems in the future.

Finally, on a personal note, I would like to thank my closest ones, Margrethe and our two sons Alfred and Gustav, and the rest of our family and friends for all the love and support.

Christian Agrell
Oslo, August 2021

List of Publications

The following journal papers are part of the thesis:

Paper I

C. Agrell (2019). Gaussian Processes with Linear Operator Inequality Constraints. *Journal of Machine Learning Research.* Vol. 20, no. 135, pp. 1–36.

Paper II

O. Gramstad, C. Agrell, E. Bitner-Gregersen, B. Guo, E. Ruth and E. Vanem (2020). Sequential sampling method using Gaussian process regression for estimating extreme structural response. *Marine Structures.* Vol. 72, 102780.

Paper III

C. Agrell and K. R. Dahl (2021). Sequential Bayesian optimal experimental design for structural reliability analysis. *Statistics and Computing.* Vol. 31, no. 27.

Paper IV

C. Agrell, K. R. Dahl and A. Hafver (2021). Optimal sequential decision making with probabilistic digital twins. Submitted for publication. arXiv: 2103.07405.

Paper V

C. Agrell, S. Eldevik, O. Gramstad and A. Hafver (2021). Risk-based functional black-box optimization – Contribution to the NASA Langley UQ challenge on optimization under uncertainty. *Mechanical Systems and Signal Processing.* Vol. 164, 108266.

Paper VI

A. Hafver, C. Agrell and E. Vanem (2021). Environmental contours as Voronoi cells. Submitted for publication. arXiv: 2008.13480.

The following proceedings have also been produced as part of the doctoral project:

Appendix A

C. Agrell, S. Eldevik, A. Hafver, F. B. Pedersen, E. Stensrud and A. Huseby (2018). Pitfalls of machine learning for tail events in high risk environments. *Safety and Reliability — Safe Societies in a Changing World: Proceedings of ESREL 2018.* pp. 3043–3051, CRC press.

Contents

Chapter 1

Introduction

The aim of this project is to provide some of the tools needed to make data-driven modelling suitable for safety-critical systems. The thesis consists of six papers, concerning development of methods, theoretical results and practical algorithms that combine probabilistic modelling of safety-critical systems with phenomenological knowledge and data.

In this chapter we first discuss the main motivation behind this project. We then present what we mean by a probabilistic model, in particular probabilistic machine learning, and what we mean by phenomenological knowledge. We also expand on the type of applications we are interested in, which ultimately is to help some decision maker act optimally under uncertainty.

In Chapter 2 we introduce some of the background theory and context for the papers, and in Chapter 3 we provide a brief summary of the main contributions of each paper. A discussion of challenges and relevant topics for future research is given in Chapter 4, followed by the papers and two appendices. Appendix A contains a conference paper supplementing the motivation for this project as discussed in Section 1.1. Appendix B contains links to two python packages that have been developed in the project.

1.1 Motivation

Artificial Intelligence (AI) and data-driven decisions based on Machine Learning (ML) are making an impact on an increasing number of industries. As these autonomous and self-learning systems become more and more responsible for making decisions that may ultimately affect the safety of personnel, assets, or the environment, the need to ensure the safe use of AI will be crucial. The use of ML for high-risk and safety-critical applications in particular is problematic. In the paper *Pitfalls of machine learning for tail events in high risk environments*, included in Appendix A, we identify the following challenges:

- there is a reduced tolerance for erroneous predictions due to potentially catastrophic consequences,
- critical consequences often relate to rare events where data is scarce, and
- relevant models are often complex, and a proper treatment of model uncertainty is essential.

However, there are ways to address this – as that there is often additional *phenomenological knowledge* available. This is the knowledge, often causal or physics-based, that underpins our understanding of the underlying mechanisms that drive the things we observe.

The origin of this doctoral project comes from the idea that in ML for physical phenomena, statements such as *"with a larger force comes a higher acceleration"* could be included to improve performance of the ML model, potentially allow for some robust extrapolation, and at the same time reduce uncertainty, or at least provide more realistic uncertainty estimates. In addition, it may provide some additional means of falsification, as there may exist observations that do not agree with the statement, in which case the data is erroneous, or the statement is false and we have learned something fundamentally new about the phenomenon.

In this example, the phenomenological knowledge when stated mathematically comes in the form of a bound on a partial derivative, which may be imposed as a *constraint* on a probabilistic model of the system of interest. Throughout the project, other forms of facilitating the use of phenomenological knowledge in probabilistic modelling of safety-critical systems have been considered. The following sections present these ideas in some more detail.

The project has been carried out in close collaboration with DNV, a global risk management and quality assurance company. All of the papers produced in this project are motivated by experience and challenges from real-world applications.

1.2 Probabilistic modelling of safety-critical systems

A model, in general, is a set of assumptions describing some system under consideration. A *probabilistic model* may be defined likewise, as a set of assumptions that specifies a probability distribution for the relevant quantities of interest. When we are interested in modelling a physical phenomenon, the probabilistic model specifies a stochastic representation of relevant physical quantities and how they interact. And when a probabilistic model includes the representation of some data-generating process, i.e. a stochastic representation of something that could be observed, probabilities given by the probabilistic model can be updated by conditioning on a set of such observations.

The task of conditioning a probabilistic model, where we find a posterior model given a prior, a likelihood and a set of observations, is often termed Bayesian updating or Bayesian inference in the statistics literature, and *probabilistic ML* in the machine learning community. In essence, the term *probabilistic ML* usually implies that there is a focus on obtaining a numerically tractable method for estimation or approximation of the posterior model, and that the end goal is to use the posterior for *prediction*. This probabilistic approach to ML is discussed in more detail in Section 2.2.

A special type of probabilistic model that is particularly relevant for safety-critical systems, and which plays a central role throughout this thesis, is the probabilistic models used for failure probability estimation in structural reliability analysis (SRA). These are probabilistic models of mechanisms that may cause structural failure, for instance that a ship sinks or that a bridge will collapse given a certain load scenario, and they are an important part of modern engineering design and operation of safety-critical systems. These models are often specified by a random variable $\mathbf{X}$, representing some physical object and its environment,

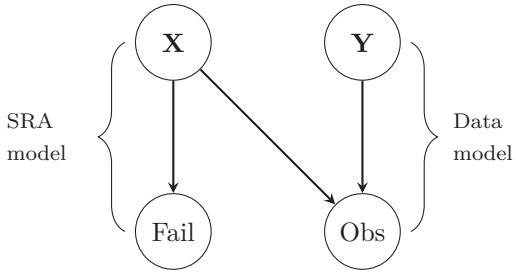


Figure 1.1: Illustration of a model for structural reliability analysis (SRA) combined with a data model. Probabilistic ML is used to obtain $P(\text{Fail}|\text{Obs})$.

together with a causal physics-based model of *failure*, Fail$|\mathbf{X}$. The failure probability $P(\text{Fail})$ can be difficult to estimate, as it is usually a small number, and various special purpose methods have been developed for efficient estimation or approximation of this probability. We discuss some of these in Section 2.1.2.

In this project, we are particularly interested in applications where these kinds of physics-based probabilistic models are connected with data as illustrated in Figure 1.1. Here, a model of some observable quantity is connected with the structural reliability model. The observable quantity depends on the physical state $\mathbf{X}$ and some additional variable $\mathbf{Y}$, representing for instance noise. The model Obs$|\mathbf{X}, \mathbf{Y}$ provides the likelihood of observations, which allows inference about $\mathbf{X}|$Obs, and ultimately the posterior failure probability $P(\text{Fail}|\text{Obs})$.

1.3 Reasoning under uncertainty

The purpose of probabilistic modelling is usually to help some decision maker reason under uncertainty, which typically means that we want to find a decision that is optimal in some way, with respect to future events that may occur and their associated uncertainty. For engineering applications, this often boils down to finding the right balance between cost and safety.

Ideally, the uncertainties associated with the random variable $\mathbf{X}$ and the causal relation Fail$|\mathbf{X}$ in Figure 1.1 are all *aleatory*. This means that all uncertainty is related to inherent variability of the physical phenomenon that is being modelled. In this case, the failure probability $p_f = P_{\mathbf{X}}(\text{Fail})$ (failure probability with respect to the random variable $\mathbf{X}$) is often used for decision making. If $\mathbf{X} = \mathbf{X}(d)$ depends on some deterministic design parameter d that we can choose, we might be interested in minimizing costs associated with the design d, under the constraint that the corresponding failure probability, $p_f(d)$, does not exceed a specified target value.

In many situations we also need to consider *epistemic* uncertainties. Epistemic uncertainty comes from lack of information or knowledge, for instance a physical constant that we do not know the value of. These are uncertainties that *in principle* can be reduced by gathering more information. In a safety-critical

system, a robust decision should be robust with respect to what we do not know, i.e. with respect to epistemic uncertainty. Hence, when both aleatory and epistemic uncertainty is present, there is a need for methodologies that allow for these to be treated separately. When working with failure probabilities, $p_f(d)$ could be viewed as an *epistemic random variable* (e.g. each realization of $p_f(d)$ corresponds to a specific value of the physical constant), and a conservative estimate of $p_f(d)$ could be used for making safety-critical decisions. In Paper III and Paper IV we consider this scenario and present a rigorous formulation of the separation of aleatory and epistemic uncertainty.

The introduction of epistemic uncertainty also gives rise to a new type of optimization problem:

What is the optimal strategy for gathering new information, in order to reduce epistemic uncertainty with respect to some specific objective, when information acquisition comes at a cost?

For instance, performing experiments to infer a physical constant may reduce epistemic uncertainty in $p_f(d)$, which may let us find a more cost efficient design.

If we expand the model illustrated in Figure 1.1 by introducing *cost* or *reward* models, for instance models representing the cost of generating observations, and costs or rewards associated with choices made in the design or operation of the relevant physical asset, the probabilistic model can be used to provide guidance in these kinds of decision-making problems. The papers that are part of this thesis have all been produced with this type of application in mind, for scenarios where additional phenomenological knowledge can be introduced. Ultimately, the purpose of introducing phenomenological knowledge will be to reduce epistemic uncertainty.

1.4 Phenomenological knowledge

In probabilistic modelling of real-world phenomena, data-driven modelling and first principles modelling can be seen as two extremes. For a physical phenomenon, the data-driven approach would be based on observations alone, making as few restrictions or assumptions about the data-generating process as possible, whereas the first principles approach could involve modelling the laws of physics combined with a stochastic representation of relevant random quantities.

Take for instance the example of launching a rocket into space. The purely data-driven approach would be to try and launch a large number of different types of rockets under different conditions to learn about what works best. This is, of course, not how rockets are designed. Instead, we could solve the system of partial differential equations (PDEs) that govern the launch and flight of a rocket under a specific set of conditions *in theory*. To find a robust solution, we would have to account for different weather conditions, variations in material properties, and in general all that is uncertain. We would then obtain a structural reliability model like the one in Figure 1.1, where $\mathbf{X}$ is a random variable representing everything about the physical system that is uncertain, and Fail$|\mathbf{X}$ is determined

from a system of PDEs. The main uncertainty that then remains is related to how well the model Fail|$\mathbf{X}$ fits reality, and we should include data, for instance from controlled experiments, to understand the mismatch between physical simulation and the real world. This is the first principled way in which data-driven and theoretical modelling based on phenomenological knowledge is combined.

A problem that often arises in practice, when dealing with models of physical phenomena that are *computationally expensive* (in money and/or time), is that probabilistic computation becomes infeasible. For instance, sampling based approaches that rely on computing Fail|$\mathbf{x}_i$ for many realizations $\mathbf{x}_i$ are useless if each evaluation of the corresponding computer model takes minutes, or hours or days, which is not uncommon for numerical models of complex physical phenomena. A way to solve this problem is to replace Fail|$\mathbf{X}$ with a probabilistic *surrogate model*. Given a computationally expensive function $f(\mathbf{x})$, we can establish a computationally cheap approximation $\hat{f} \approx f$ using only a small number of evaluations $f(\mathbf{x}_i)$. One of the most common approaches today is to let $\hat{f}$ be a Gaussian process fitted to observations of $f(\mathbf{x}_i)$ (as will be described later in Section 2.2.2). Of course, replacing a function f with some approximation $\hat{f}$ introduces additional uncertainty, and it is important that this uncertainty is quantified. Furthermore, this uncertainty depends on the set of experiments (evaluations of $f(\mathbf{x}_i)$) available, and a very relevant optimization problem is how to best select these experiments.

Going back to the rocket launch example, imagine that we have access to a physics-based computer simulation of the underlying mechanics, which gives us Fail|$\mathbf{X}$, as well as a distribution for $\mathbf{X}$ representing the relevant uncertainties. Assume further that the physics-based computer simulation is too time consuming for probabilistic analysis to be possible, so we replace it with a surrogate model fitted to a finite set of observations D. At this point, the phenomenological knowledge related to the failure mechanism in our model is represented by the dataset D.

The purpose of introducing phenomenological knowledge is to reduce epistemic uncertainty. In this example, the way we might include more phenomenological knowledge is to increase the set D. In Section 1.3 we introduced the problem of adding datapoints to D in an optimal manner, with respect to the potential effect on future decisions and the cost of data collection. This problem, which we call the problem of *optimal information gathering*, is one out of a few different ways of introducing phenomenological knowledge that has been considered in this project. Other methods that have been explored involve the use of model constraints, exploiting the hierarchical structure of physics-based models, and leveraging properties of the set of model inputs that correspond to structural failure. In summary, the following concepts have been studied:

Optimal information gathering
When we know what the model shall be used for, for instance how predictions may affect decisions in the real world, we can study the effect of uncertainty reduction. If we have available a model of some data-generating process, i.e. a model of what we might observe if we decide to run some *experiment*, we can also study the potential uncertainty reduction resulting from experiments. It is then possible to evaluate the chance that running some specific type of experiment is worthwhile. The task of optimizing over this kind of information gathering activities is considered in Paper II, Paper III and Paper IV.

Linear constraints
A relevant form of linear constraint is that of *monotonicity*, which encodes physics-related statements such as *"If you increase the thickness of a steel plate it gets stronger"*, but also, *"kids grow taller"* and *"the probability that you are eligible for a loan should increase as a function of your salary, all else equal"*. All of these statements are of course equivalent to requiring that some partial derivative is nonnegative, and this is something that can be incorporated in probabilistic ML as a *constraint*.

A popular nonparametric probabilistic ML model for dealing with differentiable (potentially latent) functions is the Gaussian process (see Section 2.2.2). A Gaussian process can be viewed as a distribution over functions, and phenomenological knowledge may be included through bounds on derivatives, or some other linear transformation of these functions. In Paper I we develop a numerical procedure for this purpose.

Hierarchical modelling
Physics-based models of failure mechanisms are often hierarchical, given as compositions of multiple models of different physical phenomena that together determine the failure scenario. For instance; loads (wind, waves, or other forces acting on the structure), load effects (stresses, strains), and the structural resistance (material capacity with respect to e.g. stress or strain) may be determined from separate special-purpose models. If some of these are computationally expensive, we need to replace them with surrogates. It is then beneficial to preserve the representation of how these sub-phenomena are interlinked, by combining multiple surrogate models instead of replacing the global model $\text{Fail}|\mathbf{X}$ with a single surrogate. The method in Paper III is developed for this scenario.

Convex failure sets
For many applications, it is reasonable to assume that the failure set is convex. That is, if $\mathbf{x}_1$ and $\mathbf{x}_2$ are two realizations of $\mathbf{X}$ leading to failure, then any interpolation $\lambda\mathbf{x}_1 + (1-\lambda)\mathbf{x}_2$ on the line between $\mathbf{x}_1$ and $\mathbf{x}_2$ will also correspond to a failed state. The theory related to *environmental contours* (see Section 2.1.2) which we consider in Paper VI is developed under this assumption.

Chapter 2

Background

In this chapter we present some relevant background for understanding the papers, and to give some context for seeing how they relate to the main scope of the thesis. We start by presenting the overarching topic of uncertainty quantification (Section 2.1) with a specific focus on structural reliability, which we connect with probabilistic machine learning (Section 2.2), and then end with a discussion on some relevant problems related to optimal decision making under uncertainty (Section 2.3).

2.1 Uncertainty Quantification

In Section 1.4 we considered the example of designing a rocket. We assumed that a numerical model of the physical phenomenon was available, but that the **inputs are uncertain**. This could be related to variability of the inputs or fixed model parameters whose exact values are unknown. But even if the input could be determined exactly, the numerical model does not necessarily provide a true representation of the real physical phenomenon. This form of uncertainty is often called **model uncertainty** related to model discrepancy, and stems from the assumptions and simplification made in the mathematical model used to approximate reality. If we want to estimate the model discrepancy using data, there is usually also **observational uncertainty** involved, which we often accredit to noise.

Additional uncertainties also arise when we try to make use of a mathematical model, as further numerical approximation may be needed. For instance, if model evaluation involves solution of PDEs, we would generally need to rely on numerical schemes such as finite element or finite difference approximations which introduces **numerical errors**. In addition, it is often necessary to introduce surrogate models or emulators to account for the limited number of model evaluations (PDE solves) that can be performed in practice, which introduces additional **interpolation uncertainties**.

A further discussion and categorization of these uncertainties that arise when dealing with simulation of physical phenomena can be found in e.g. (Kennedy and O'Hagan 2001; Vernon et al. 2010). Broadly, the field of Uncertainty Quantification (UQ) can be viewed as the collection of disciplines needed to address all of these uncertainties combined.

> Uncertainty quantification is the rational process by which proximity between predictions and observations is characterized. It can be thought of as the task of determining appropriate uncertainties associated with model-based predictions.

> More broadly, it is a field that combines concepts from applied mathematics, engineering, computational science, and statistics, producing methodology, tools, and research to connect computational models to the actual physical systems they simulate.
>
> — Handbook of Uncertainty Quantification (Ghanem et al. 2017)

UQ is not a mature field, and there is currently no common theoretical foundation for UQ. Traditionally, it has been treated somewhat differently within applied mathematics and statistics, but recent initiatives are trying to bring the two communities closer together (e.g. the *INI Programme on Uncertainty Quantification* 2018). The main differences lie in how computationally expensive models of physical phenomena are treated. Let $y = f(x)$ be a model mapping physical properties of a system, given by the vector x, to an output y which represents some quantity of interest. In physical systems, the relationship between x and y is often modelled using PDEs, and various methods have been developed for the scenario where the variables involved are random. Sullivan 2015 gives an introduction to the relevant theory, and for some recent examples see for instance (Bespalov et al. 2019; Capodaglio et al. 2021; Khan et al. 2018).

Within statistics, uncertainty quantification related to such models has roots in the research on design of computer experiments by Sacks et al. 1989, and the Bayesian treatment by Kennedy and O'Hagan 2001 which considers all of the uncertainties involved when these models are used to simulate real-world phenomena. With the statistical approach, the function $f(x)$ is usually viewed as a black-box model, where the only information available about the mapping $x \to y$ is through a finite set of observations $(x_1, y_1), \ldots, (x_N, y_N)$. Using a computationally cheap approximation $\hat{f} \approx f$, from which a large number of samples ($\gg N$) can be obtained, statistical analysis can then be carried out. The approximation $\hat{f}$ is often called a *surrogate model* or *emulator*[1], and one of the most popular alternatives today is to make use of Gaussian processes as introduced in Section 2.2.2.

The approach taken in this thesis lies somewhere in between these two views, but closest to the statistical alternative. We make use of probabilistic surrogate models, like the Gaussian process, which is agnostic to the type of physical simulation that is implemented in the computer model. We do not assume that the underlying mechanics comes in the form of a PDE that we have knowledge of, but instead try to find other ways of incorporating phenomenological knowledge to enhance the probabilistic surrogate.

2.1.1 Epistemic and aleatory uncertainty

Within UQ it is common to consider two different kinds of uncertainty: Aleatory (stochastic) and epistemic (knowledge-based) uncertainty. We say that uncertainty is epistemic if we foresee the possibility of reducing it through

[1] The name emulator is often reserved for surrogate models that can interpolate between noiseless observations coming from a deterministic computer simulation, and where uncertainty in predictions at untried inputs can be assessed.

gathering more or better information. For instance, uncertainty related to a parameter that has a fixed but unknown value is considered epistemic. Aleatory uncertainty, on the other hand, is the uncertainty which cannot (in the modeler's perspective) be affected by gathering information alone. The characterization of aleatory and epistemic uncertainty depends on the modelling context. It is, for instance, possible to argue that the result of a coin flip is epistemic (given all initial conditions etc.), but this might not be a suitable assumption with respect to the purpose of the model. It is then most relevant to consider the coin flip as aleatory. Kiureghian and Ditlevsen 2009 provide a detailed discussion of the differences between aleatory and epistemic uncertainty, and also show that by not separating between these two types of uncertainty in risk and reliability assessment, one may either over- or underestimate the failure probability by a significant magnitude (depending on the problem at hand). Hence, distinguishing between aleatory and epistemic uncertainty is important for risk assessment.

In the uncertainty quantification literature, aleatory uncertainty is typically modelled via probability theory. However, epistemic uncertainty is represented in many different ways. For instance, Helton et al. 2010 considers four different ways of modelling epistemic uncertainty: Interval analysis, possibility theory, evidence theory (Dempster–Shafer theory) and probability theory.

In this thesis we take a probabilistic approach to both epistemic and aleatory uncertainty. In Paper IV we give a detailed exposition of how the two types of uncertainty can be treated within measure-theoretic probability, and how Bayesian updating of epistemic uncertainty can be formulated.

For the problems addressed in this thesis, one common goal is to make use of phenomenological knowledge to reduce epistemic uncertainty.

2.1.2 Structural reliability analysis and environmental contours

Structural reliability analysis (SRA) is the fundamental building block of modern risk-based engineering methodologies. In probabilistic structural reliability analysis, a model of a physical system and its environment is described by a random variable $\mathbf{X} \in \mathbb{X}$ and a function $g : \mathbb{X} \to \mathbb{R}$. Here $\mathbb{X}$ is an arbitrary measurable space, but we may think of $\mathbb{X} = \mathbb{R}^n$ as the canonical scenario.

A value $\mathbf{x} \in \mathbb{X}$ contains the parameters describing a particular structure, such as the geometry, dimensions and material properties. These quantities may be random, but can be influenced by the designer of the structure. For example, the designer may choose to use a more expensive, but more durable material in order to improve the structural properties of the system. In addition, $\mathbf{X}$ contains the (random) parameters that characterize the systems environment, such as wind speed, wave height etc., and parameters describing how well the model fits reality (model uncertainties).

The failure probability

The function $g(\cdot)$ is called the *limit-state function* (sometimes the *performance function*), with the property that $g(\mathbf{x}) \leq 0$ corresponds to system failure, and

$g(\mathbf{x}) > 0$ corresponds to the system functioning. The set

$$F_g = \{\mathbf{x} \in \mathbb{X} \mid g(\mathbf{x}) \leq 0\} \tag{2.1}$$

is the failure set associated with g. Structural reliability analysis generally revolves around analysis of the *failure probability*

$$P(F_g) = E\left[\mathbf{1}\left(g(\mathbf{X}) \leq 0\right)\right], \tag{2.2}$$

where $E[\cdot]$ denotes the expectation with respect to P and $\mathbf{1}(\cdot)$ is the indicator function.

In most real-world applications one does not attempt to evaluate (2.2) analytically, as this is generally not feasible. Instead, several numerical methods have been developed for approximation and estimation of the failure probability. Two traditional methods are the first- and second-order reliability method (FORM/SORM), where the failure boundary ∂F_g is approximated using a Taylor expansion up to the first or second order (centered at at failure point with high likelihood). Different sampling procedures have also been developed, which often make use of intermediate results obtained from FORM/SORM. See for instance (Madsen et al. 2006).

The need for surrogate models in SRA today

Although non-deterministic methods for structural reliability analysis dates back to the 1950s, it was only a few decades ago that it became possible to implement full probabilistic approaches (Wu 2013). With methods such as Monte Carlo simulation and FORM, evaluation of failure probabilities became possible, but it was still restricted to structural models where a large number of evaluations could be performed in reasonable time. According to Wu 2013, surrogate modelling was first used to address this problem in the 1990s, using polynomial regression which was referred to as the *response surfaces method*. Today, an important part of SRA is the use of probabilistic surrogate models and emulators which are flexible enough that they can represent modern simulations of physical systems, which can be of high fidelity and high computational complexity. This is one of the reasons why SRA is considered as a part of UQ today.

Environmental contours

A natural assumption in many physical systems is that the failure set F_g in (2.1) is convex. That is, if $\mathbf{x}_1, \mathbf{x}_2 \in F_g$ are two realizations of $\mathbf{X}$ leading to failure, then any interpolation $\lambda\mathbf{x}_1 + (1-\lambda)\mathbf{x}_2$ on the line between $\mathbf{x}_1$ and $\mathbf{x}_2$ will also correspond to a failed state.

In this case, if we can find a convex set $\mathcal{B}$ such that $g(\mathbf{x}) \geq 0\ \forall \mathbf{x} \in \mathcal{B}$, it follows from convexity theory that there exists a *supporting hyperplane* Π that separates $\mathcal{B}$ and F_g, i.e. $\mathcal{B} \subseteq \Pi^-$ and $\mathcal{F} \subseteq \Pi^+$, where Π^- and Π^+ are the two half spaces separated by Π. Moreover, it must hold that $P(F_g) \leq P(\mathbf{X} \in \Pi^+)$.

The idea behind environment contours is to find a set $\mathcal{B}$ with the property that $P(\mathbf{X} \in \Pi^+) = p_e$ for all supporting half-spaces Π^+ and some constant target failure probability p_e. The set $\mathcal{B}$ thus depends on the value of p_e and the

distribution of $\mathbf{X}$, which one normally assumes is absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^n$. Then, if one can ensure that any value of $\mathbf{x}$ in $\mathcal{B}$ is safe ($g(\mathbf{x}) > 0$), the failure probability is bounded above by p_e. Figure 2.1 gives a visual illustration for the case where $\mathbf{X}$ is two-dimensional.

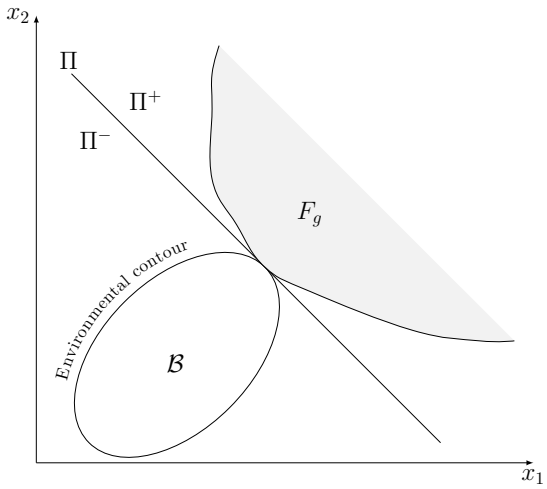


Figure 2.1: Illustration of an environmental contour in $\mathbb{R}^2$. If the exceedance probability for all supporting half-spaces Π^+ of $\mathcal{B}$ is less than p_e, then for any convex failure region F_g that does not intersect the interior of $\mathcal{B}$, the failure probability $P(F_g)$ is also less than p_e.

Note here that we do not rely on specific knowledge about F_g, only the assumption that F_g is convex. In order to be sure that the failure probability is less than p_e we need to validate that the structure *does not fail* within $\mathcal{B}$, but this is a different question than finding the failure set F_g. This is useful when observations of safe behavior is more easily obtainable than that related to failure.

The concept of environmental contours was first introduced by Haver 1980; Haver 1987 as a means to study the joint distribution of significant wave height and wave period of ocean waves (hence the name *environmental*). Since then, efficient numerical procedures for constructing environmental contours have been developed, see for instance (Huseby et al. 2015; Ross et al. 2020). Today, the use of environmental contours is a well-established practice in design of marine structures, and recommended in standards and recommended practices such as DNVGL-RP-C205 2019 and NORSOK N-003 2017.

Because of the historical application towards 2-dimensional wave models, most of the numerical methods developed so far have been restricted to $\mathbf{X} \in \mathbb{R}^2$. But conceptually the use of environment contours is relevant for other application areas as well. In Paper VI we propose a method which is applicable in the n-dimensional case.

2.2 Probabilistic machine learning

Probabilistic machine learning (ML) is concerned with inference and prediction using probabilistic models conditioned on data gathered through experience. It has a natural overlap with statistics, but with some differences with respect to emphasis and terminology. For instance, probabilistic ML tend to focus more on computational developments for dealing with complex models and/or large sets of data, often motivated by some business objective involving prediction.

In probabilistic ML, we work with a probabilistic model of a data-generating process. This model may be designed based on knowledge of the system (e.g. if the data comes from a well understood process), or based on generic function approximation (e.g. a neural network or a Gaussian process), or a combination. This is the same setup used in UQ, and hence connecting these probabilistic models of data and the physical system as in Figure 1.1 is a natural idea.

The probabilistic framework also provides a way to understand what learning is, which gives some conceptual advantages as a normative theory for learning in AI systems (Ghahramani 2015). For instance, if we want to derive a theory for how an AI system should represent and update its beliefs about the world in light of data, the celebrated Cox's theorem shows that if the AI's way of reasoning is not equivalent to probability theory, then some basic logical or "common sense" principles are violated (Cox 1961; Jaynes 2003). The probabilistic framework is therefore considered as one of the principal theoretical and practical approaches to ML and AI today.

2.2.1 The Bayesian formalism

In the Bayesian approach to machine learning, the "learning" part refers to computing the posterior over unknown model parameters in the same way the posterior of any unknown hypothesis is computed using Bayes' theorem (Ghahramani 2015; Murphy 2021).

Let $P(D \mid \theta, M)$ denote a probabilistic model of some data-generating process. Here M denotes the class of all probabilistic models that we are considering, each of which is represented by a value of the unknown parameter θ. For instance, M can denote a certain type of Bayesian network, where the probability distributions of the variables in the network depend on θ. $P(D \mid \theta, M)$ is the likelihood of the parameter θ in model M, and together with this we will need a prior distribution $P(\theta \mid M)$ representing our initial beliefs (or ignorance) about θ. Then, the Bayesian formulation can be simply described as follows (see Ghahramani 2015, p. 13).

Learning: Given data D, *learning* about the parameter θ means finding the posterior

$$P(\theta \mid D, M) = \frac{P(D \mid \theta, M)P(\theta \mid M)}{P(D \mid M)}, \tag{2.3}$$

where

$P(D \mid \theta, M)$ is the likelihood of the parameter θ in model M,
$P(\theta \mid M)$ is the prior probability of θ,
$P(\theta \mid D, M)$ is the posterior of θ given the data D.

With respect to learning we can interpret the denominator $P(D \mid M)$ simply as a term that ensures that (2.3) integrates to one (until we consider model selection below). We can view *learning* as the transformation of prior knowledge or assumptions about $\theta \mid M$, via the data D, into the posterior knowledge represented by $P(\theta \mid D, M)$. This posterior can then be used as a prior when additional data becomes available.

Prediction: An updated model can be used to predict unseen data D^*

$$P(D^* \mid D, M) = \int P(D^* \mid \theta, D, M) P(\theta \mid D, M)\, d\theta. \tag{2.4}$$

Different models can also be compared by considering the Bayesian formulation at the level of M, or by comparing the *marginal likelihood* $P(D \mid M)$.

Model comparison: Given a prior over models $P(M)$ we can evaluate

$$P(M \mid D) = \frac{P(D \mid M) P(M)}{P(D)}, \tag{2.5}$$

$$P(D \mid M) = \int P(D \mid \theta, M) P(\theta \mid M)\, d\theta. \tag{2.6}$$

Here $P(D \mid M)$ is also often called the *model evidence*, and (2.6) provides a preference for simpler models known as Bayesian Ockham's Razor (Jefferys and Berger 1992; Rasmussen and Ghahramani 2001).

2.2.2 Gaussian processes

If θ is a finite vector of numbers, the setup in Section 2.2.1 gives an intuitive illustration of how a probabilistic model can be updated based on data. However, some of the most powerful probabilistic machine learning models are *nonparametric*. In this case we can still think about probabilistic ML in terms of updating θ, but where we have to interpret θ as an infinite-dimensional vector or a function (see for instance the measure-theoretic formulation we present in Paper IV).

A popular nonparametric alternative that plays an important role throughout this thesis is the Gaussian process (GP). A GP is a collection of random variables $\{f(\mathbf{x}) \mid \mathbf{x} \in \mathbb{X}\}$ defined on some index set $\mathbb{X}$, such that for any finite subset $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subseteq \mathbb{X}$, the vector $[f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]$ has a multivariate (possibly degenerate) normal distribution. The GP is a fundamental object of study within many areas of mathematics and statistics, and the available literature is vast. We will assume that $\mathbb{X} = \mathbb{R}^n$ and $f \in \mathbb{R}$, which is the canonical scenario in the

machine learning context (Rasmussen and Williams 2005). In this case, we view a GP as a *distribution over functions*[2] from $\mathbb{R}^n$ to $\mathbb{R}$.

A GP can be defined by a mean function μ and a positive-semidefinite function K called the *covariance function* or *kernel*,

$$\mu(\mathbf{x}) = E[f(\mathbf{x})] : \mathbb{R}^n \to \mathbb{R}, \tag{2.7}$$

$$K(\mathbf{x}, \mathbf{x}') = E[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(\mathbf{x}'))] : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \tag{2.8}$$

and we can write $f \sim \mathcal{GP}(\mu, K)$ for the associated GP f. For any finite set of inputs $\mathbf{x}_1 \dots \mathbf{x}_N$, the vector $\boldsymbol{f} = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)]$ is multivariate Gaussian, $\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{K})$, with mean $\boldsymbol{\mu} = [\mu(\mathbf{x}_1), \dots, \mu(\mathbf{x}_N)]$ and covariance matrix $\boldsymbol{K} = [K(\mathbf{x}_i, \mathbf{x}_j)]$.

Given a set of observations $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_k, y_N)\}$ assume that y_i corresponds to observing $f(\mathbf{x}_i)$ together with additive Gaussian noise, $y_i = f(\mathbf{x}_i) + \varepsilon_i$, where the noise terms ε_i are i.i.d. $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. Then the posterior process, conditioned on the data D, is still Gaussian:

For a new set of input locations, $\mathbf{x}_1^*, \dots, \mathbf{x}_m^*$, let $\boldsymbol{f}^*|D$ denote the posterior process evaluated at each input $\mathbf{x}_j^*$, $\boldsymbol{f}^*|D = [f(\mathbf{x}_1^*), \dots, f(\mathbf{x}_m^*)]|D$. Then $\boldsymbol{f}^*|D \sim \mathcal{N}(\boldsymbol{\mu}_{f^*|D}, \boldsymbol{\Sigma}_{f^*|D})$ with

$$\boldsymbol{\mu}_{f^*|D} = \boldsymbol{\mu}^* + \boldsymbol{K}^*(\boldsymbol{K} + \sigma^2 I)^{-1}(\mathbf{Y} - \boldsymbol{\mu}), \tag{2.9}$$

$$\boldsymbol{\Sigma}_{f^*|D} = \boldsymbol{K}^{**} - \boldsymbol{K}^*(\boldsymbol{K} + \sigma^2 I)^{-1}(\boldsymbol{K}^*)^T. \tag{2.10}$$

Here $\boldsymbol{\mu}^*$, $\boldsymbol{\mu}$ and $\mathbf{Y}$ are vectors with elements $\mu(\mathbf{x}_i^*)$, $\mu(\mathbf{x}_i)$ and y_i respectively, I is the $N \times N$ identity matrix, and $\boldsymbol{K}^*$ and $\boldsymbol{K}^{**}$ have elements $(\boldsymbol{K}^*)_{i,j} = k(\mathbf{x}_i^*, \mathbf{x}_j)$ and $(\boldsymbol{K}^{**})_{i,j} = k(\mathbf{x}_i^*, \mathbf{x}_j^*)$.

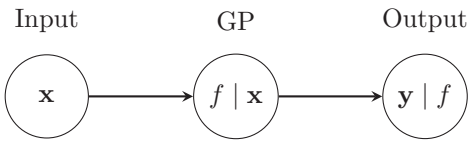


Figure 2.2: Supervised machine learning with Gaussian processes. Given observations $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_k, y_N)\}$ and a likelihood $P(y|f)$, we assign $f|\mathbf{x}$ a GP prior and proceed to infer the function $f|\mathbf{x}$. If the output y_i corresponds to observing $f(\mathbf{x}_i)$ together with additive Gaussian noise, the posterior is still a GP with the mean and covariance as in (2.9)-(2.10).

Figure 2.2 illustrates the probabilistic model for supervised learning with GPs. The conjugate case with Gaussian likelihood, $y|f \sim \mathcal{N}(f, \sigma^2)$, is often applied in UQ. In particular, the noiseless alternative is obtained simply by setting $\sigma = 0$, and the corresponding GP represents an *emulator* which interpolates the data (see Figure 2.3 below).

[2]Given a probability space $(\Omega, \mathcal{F}, P)$ on which the random variables $f(\mathbf{x})$ are defined (for all $\mathbf{x}$), we can write $f(\mathbf{x}, \omega) = f(\mathbf{x})(\omega) : \mathbb{R}^n \times \Omega \to \mathbb{R}$. For each fixed $\omega \in \Omega$ there is an associated function $f(\cdot, \omega) : \mathbb{R}^n \to \mathbb{R}$.

To give some examples of the kernel functions $k(\mathbf{x}, \mathbf{x}')$ often seen in the literature, we can consider two stationary kernels of the form

$$k(\mathbf{x}, \mathbf{x}') = s^2 k(r), \; r = \sqrt{\sum_{i=1}^{n} \left(\frac{x_i - x_i'}{l_i} \right)^2}, \tag{2.11}$$

with variance parameter s^2 and length scale parameters l_i for $i = 1, \ldots, n$. The two popular kernels, the radial basis function (RBF), also called squared exponential kernel, and the Matérn 5/2 kernel are defined through the function $k(r)$ as

$$k_{\text{RBF}}(r) = e^{-\frac{1}{2}r^2} \text{ and } k_{\text{Matérn 5/2}}(r) = (1 + \sqrt{5}r + \frac{5}{3}r^2)e^{-\sqrt{5}r}.$$

Note that these kernels depend on a *hyperparameter* $\psi = [s, l_1, \ldots, l_n]$. Figure 2.3 shows an example of a GP with $\mu = 0$ and two different choices of ψ. Naturally, the quality of the posterior process depends on the treatment of the hyperparameter ψ, and this is also a part of the machine *learning*. In a fully Bayesian approach we could specify a prior over ψ and proceed as discussed in Section 2.2.1, but at the cost of losing the simple analytical expression for the posterior. Because of this, it is common to instead find a plausible value of ψ and act as if this was fixed. In ML one often relies on optimization of hyperparameters using maximum likelihood or cross-validation, see (Rasmussen and Williams 2005), and also (Kennedy and O'Hagan 2001) for a discussion with regards to UQ.

2.2.3 Numerical methods

The probabilistic approach to ML is conceptually very simple, and theoretically appealing, but a fully Bayesian approach poses several practical challenges.

One major obstacle is dealing with the integrals in (2.4) and (2.6), which are generally analytically intractable. For deployment of these models in the real world, there is therefore a need for robust and computationally efficient numerical methods. This is one of the main drivers in current probabilistic ML research.

In Section 2.2.2 we noted that if we want to express uncertainty with respect to GP hyperparameters, in the fully Bayesian spirit, then the analytical posterior computation is no longer tractable and alternative methods are needed (see e.g. (Lalchand and Rasmussen 2020)). This is also true for the case with non-Gaussian likelihood, for instance if the noise is not Gaussian, or if we want to use the GP for *classification* where each output corresponds to a category. From (2.9)-(2.10) we can also observe that exact computation of the posterior involves inversion of an $N \times N$-matrix, which is $\mathcal{O}(N^2)$ in memory and $\mathcal{O}(N^3)$ in time, and this becomes prohibitive if the number of observations, N, is large. One way of dealing with these two problems which is used in practical applications today, is by making use of *sparse variational inference*. Here, a sparse (smaller)

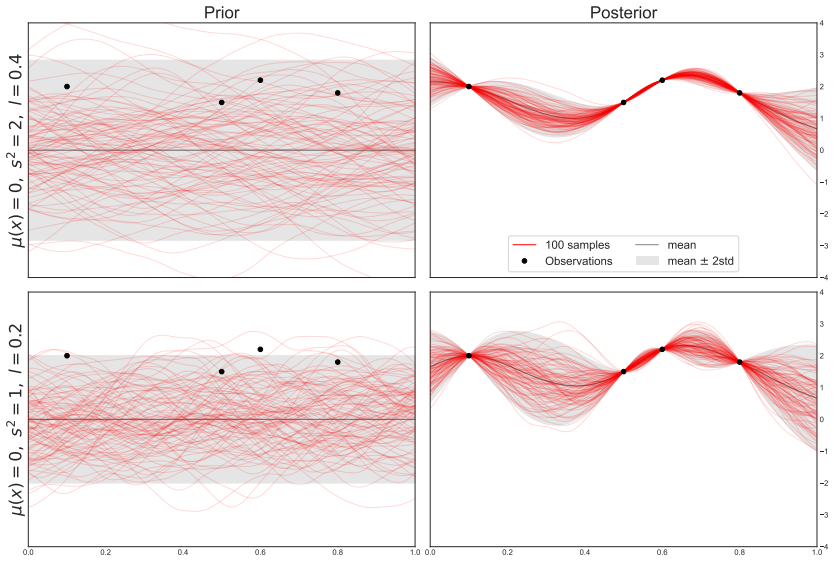


Figure 2.3: Two GPs with $\mu = 0$ and Matérn 5/2 kernels with different hyperparameters ψ. Top row: $\psi = (s^2 = 2, l = 0.4)$, bottom row: $\psi = (s^2 = 1, l = 0.2)$. The observations are assumed noiseless, i.e. $y_i = f(x_i)$.

GP is used to approximate the intractable posterior, through minimizing the Kullback-Leibler divergence between the approximating GP and posterior process (G. Matthews et al. 2016).

The method of variational inference is a general approach to address the intractable integrals that arise in Bayesian modelling. Another popular alternative is to attack the problem with Markov chain Monte Carlo (MCMC), often with implementations of Hamiltonian MC (Hoffman and Gelman 2014). These are examples of general methods and algorithms for approximation or estimation in probabilistic models. *Probabilistic programming* (van de Meent et al. 2018) aims to accelerate the process of applying these methods to any probabilistic model. This is currently an active area of development in the machine learning space, which may lead to a more widespread use of probabilistic modelling in the years to come.

2.3 Optimal decision making under uncertainty

A probabilistic model is first useful when it affects a decision. With reference to the discussion in Section 1.3, the probabilistic model is a tool for reasoning in an uncertain environment, which in the engineering context may relate to finding solutions that are both efficient, in terms of e.g. money or energy, and sufficiently safe. For the probabilistic models discussed in Section 2.2, which can be updated each time new information arrives, at discrete time steps, the discrete-time

sequential decision-making problem is of particular interest. This is the scenario where one gets the opportunity to make a decision, receive information related to the consequences of this decision, and use this to inform the next decision, and so on.

Below we describe in short two classes of decision-making problems that are of particular relevance in this thesis.

2.3.1 Reliability-based design and operation of physical assets

In Section 1.3 we introduced the concept of optimization with respect to a failure probability, where we have access to some deterministic design parameter d that affects both the cost and the failure probability of a given structure. In the structural reliability setting (see Section 2.1.2), assume that the random variable describing the physical system depends on d, $\mathbf{X} = \mathbf{X}(d)$, and that a cost function $c(\mathbf{x})$ and the limit state $g(\mathbf{x})$ are given. Then the cost $c(\mathbf{X}(d))$ and the structural performance $g(\mathbf{X}(d))$ are both random variables depending on d. Of course, there are other ways of describing how these quantities of interest depend on d, but here for simplicity we have packed everything into a single variable $\mathbf{X}$.

We are interested in finding a value of d which is optimal, in some sense, with respect to both cost and structural reliability. Valdebenito and Schuëller 2010 give some examples of different formulations. Some natural variants are obtained by defining the expected cost $C(d) = E[c(\mathbf{X}(d))]$ and the failure probability $p_f(d) = P(g(\mathbf{X}(d)) \leq 0)$ as functions of d, and proceed to minimize $p_f(d)$ under a fixed budget constraint on $C(d)$, or conversely minimize $C(d)$ with some upper bound on $p_f(d)$. Establishing the entire Pareto frontier of optimal combinations would also be of interest, but note that these optimization problems can be numerically challenging as they involve computation of $p_f(d)$.

For certain scenarios it is also necessary to take both aleatory and epistemic uncertainty into account, something that further complicates matters. That is, when $\mathbf{X}$ contains both aleatory and epistemic uncertainty that we want to treat differently. This is something we discuss in great detail in Paper IV, but for simplicity assume for now that $\mathbf{X}$ is purely aleatory and that we instead introduce an additional epistemic variable $\mathbf{e}$ that does not depend on d. We assume that the performance of the system depends on $\mathbf{e}$, i.e. $g = g(\mathbf{x}, \mathbf{e})$ and consequently $p_f = p_f(d, \mathbf{e}) = P(g(\mathbf{X}(d), \mathbf{e}) \leq 0)$. There are different philosophies for how uncertainty with respect to $\mathbf{e}$ should be treated, see for instance (Helton et al. 2010). One simple alternative is to maximize $p_f(d) \leftarrow \max p_f(d, \mathbf{e})$, which introduces an additional optimization step in the design optimization loop.

2.3.2 Optimal information gathering and experimental design

For a probabilistic model of some physical system to be relevant and useful, one needs to bring the epistemic uncertainty down to an acceptable level. This can involve running experiments, physical or in simulation, collecting available data, taking measurements etc. in order to learn more about the physical system and its environment.

One example of epistemic uncertainty is the GP uncertainty in Figure 2.3. Here, we know that there exists some function, that coincides with the four observations at those specific input values, but otherwise is unknown. The GP represents our epistemic uncertainty about this function. Imagine that you get to choose the next input location x, for which the output $f(x)$ of the true function will be revealed. Which value of $\mathbf{x}$ will be the most informative? Of course, this depends on some specified objective. We could aim to minimize the maximum local uncertainty (largest GP standard deviation for $x \in [0, 1]$ in Figure 2.3), or maybe define some global notion of uncertainty by integration with respect to x. These are questions regarding *design of experiments*, a classical area of research in statistics. However allowing for noiseless observations as in Figure 2.3 is a complicating factor. The noiseless case is relevant for deterministic experiments such as those coming from a computer simulation, and we refer to this scenario as *design of computer experiments*, where (Sacks et al. 1989) is a classical reference.

Today, this problem of identifying the optimal experiment to perform, or the optimal data to collect for updating/refining a model, is highly relevant for machine learning applications. We often refer to this as *adaptive* or *active learning*. See for instance (MacKay 1992; Ranjan et al. 2008; Seo et al. 2000) for some different examples. Maybe the most widespread use of this idea today is through *Bayesian optimization* (Brochu et al. 2010), which we could use to find the maximum of the function in Figure 2.3. Where should we evaluate the true function $f(x)$ to find a large value using a small set of function evaluations? We could consider $x \approx 0.6$ which is close to the currently largest observation, or maybe $x \approx 0.3$ where the uncertainty is large and we might get lucky. Balancing these two preferences is called the exploration/exploitation trade-off. One method that is used in Bayesian optimization is based on maximizing the upper confidence bound (UCB) of the GP, for instance by choosing the value of x that maximizes the mean $+2$ standard deviations in Figure 2.3. Another popular alternative is based on maximizing the expected improvement (EI), which is based on how much the observed maximum is expected to increase, given a new function evaluation at x. For the applications considered in this thesis, we are mostly interested in the scenario where we select experiments to provide information related to structural performance, in particular the failure probability. This is somewhat different than the Bayesian optimization scenario, but the core intuitive idea remains the same.

When we consider this type of information gathering, then the order of information does not matter. That is, when we considered the example related to Figure 2.3, the current set of information is the set $D = \{(x_1, f(x_1)), \ldots, (x_4, f(x_4))\}$, and the posterior GP does not depend on any ordering of these observations. We can typically assume this is true in probabilistic modelling, i.e. that we interpret the data D in the likelihood $P(D|\theta, M)$ in (2.3) as a *set*. This means that a model for predicting the next observation given the history, here predicting $f(x_5)$ given D and x_5, is Markovian. That is, our model of the future when conditioned on the present is independent of the past. This is a useful condition, since we can frame the problem of optimal experimental design (or active learning in the ML terminology) using the Bellman

optimality principle. This principle says that an optimal policy chosen at some initial time, must be optimal when the problem is re-solved at a later stage given the state resulting from the initial choice. As a result, the problem of finding an optimal policy for how to select experiments, that considers what may happen multiple steps into the future, can be addressed with dynamic programming (Bertsekas et al. 1995).

The dynamic programming approach does however suffer from the curse of dimensionality, in the sense that it becomes computationally intractable when the space of possible states and/or actions becomes large, and if one wants to consider a horizon beyond looking just a few steps ahead. Because of this, most approaches to optimal experimental design are *myopic*, i.e. short-sighted, meaning that they look only one (or zero) steps ahead. We have introduced briefly the concept of Bayesian optimization, and two alternative approaches called UCB and EI. The EI criterion is an example of a one-step lookahead strategy, as it is based on what we expect will happen next. This corresponds to the Bellman optimal solution if we only look one step into the future. There are also methods based on heuristics, using only what is available at the current time without trying to foresee what will happen next, which corresponds to a zero-horizon approach. The UCB approach is one such example.

A completely different approach to deal with the curse of dimensionality in the dynamic programming formulation, is to attempt to approximate a far-sighted optimal policy instead of computing a short-sighted one exactly. This is the most common route taken within reinforcement learning (RL), where machine learning is applied to find and represent a good policy (Sutton and Barto 2018). A popular alternative today is to make use of deep neural networks, which serve as a very flexible function approximator, to represent either a policy or the value of states and/or actions from which the optimal one can be determined. Within this thesis we have considered both myopic and RL alternatives.

Chapter 3

Summary of papers and main contributions

3.1 Paper I

C. Agrell (2019). Gaussian Processes with Linear Operator Inequality Constraints. *Journal of Machine Learning Research.* Vol. 20 no. 135 pp. 1–36.

This paper is motivated by machine learning applications for high-consequence engineering systems, where a probabilistic approach is essential, and where knowledge related to the physical system often can be framed as a set of linear constraints.

We present an approach for constrained Gaussian Process (GP) regression where we assume that a set of linear transformations of the process are bounded. We consider a GP f over functions on $\mathcal{X} \subset \mathbb{R}^n$ taking values in $\mathbb{R}$, where the process $\mathcal{L}f$ is still Gaussian when $\mathcal{L}$ is a linear operator. Our goal is to model f under the constraint that realizations of $\mathcal{L}f$ are confined to a convex set of functions. In particular, we require that $a \leq \mathcal{L}f \leq b$, given two functions a and b where $a < b$ pointwise. This formulation provides a consistent way of encoding multiple linear constraints, such as shape-constraints based on e.g. boundedness, monotonicity or convexity.

The results needed for stable numerical implementation are derived, together with an efficient sampling scheme for estimating the posterior process. Through a series of numerical examples, we demonstrate that the use of constraints has a significant effect on uncertainty reduction and provides more realistic uncertainty estimates (as the chance of some un-physical outcomes are removed).

3.2 Paper II

O. Gramstad, C. Agrell, E. Bitner-Gregersen, B. Guo, E. Ruth and E. Vanem (2020). Sequential sampling method using Gaussian process regression for estimating extreme structural response. *Marine Structures.* Vol. 72, 102780.

In this paper, we develop a method for sequential design of experiments for a specific maritime application, which involves estimating the maximum (over time) response of a structure that is exposed to a stochastic metocean environment.

The long-term metocean induced extreme response of ships and offshore structures is affected by both the long-term variability of the metocean environment (e.g. waves, wind, currents, etc.) and the short-term variability of the response in a given random sea state (e. g. wave induced bending moment of a ship subject to random waves). Hence, for an accurate estimation of the long-term extreme response, both the long-term variability of the metocean environment and the short-term variability of the response need to be considered. In principle, this can be achieved by running a full long-term analysis, but in practice this is often not feasible when considering nonlinear responses and long return periods like 25 or 100 years, which are typically used in design of ships and offshore structures.

Our proposed methodology uses Gaussian process regression to estimate parameters of the short-term response distribution, based on output from computationally expensive hydrodynamic simulations. We present an adaptive design strategy for sequential updating of the model, focusing on the metocean conditions that contribute the most to the long-term extreme. With this approach, only a limited number of hydrodynamic simulations are needed.

The suggested approach is demonstrated on the problem of estimating the 25-year extreme vertical bending moment on a ship, and we show that a relatively small number of iterations (full hydrodynamic simulations) are needed. The results suggest that the proposed method can be used as an alternative to contour-based methods (see Section 2.1.2) which are commonly used in the offshore industry today, or to other methods that consider a few sea states using accurate numerical simulations, with little or no added complexity or computational effort.

3.3 Paper III

C. Agrell and K. R. Dahl (2021). Sequential Bayesian optimal experimental design for structural reliability analysis. *Statistics and Computing.* Vol. 31, no. 27.

This paper is concerned with optimal design of experiments for investigating the reliability of a structure, in a similar spirit as Paper II, but where we target the failure probability directly.

In many applications of structural reliability analysis, the limit-state function $g(\mathbf{x})$ is practically unknown, as function evaluation involves time consuming numerical simulation or some other form of experiment that is expensive to perform. This is the reality as the analysis of complex engineering systems is becoming more and more involved. The problem we address in this paper is how to optimally design experiments, in a Bayesian decision theoretic fashion, when the goal is to estimate the probability $P(g(\mathbf{X}) \leq 0)$ using a minimal amount of resources. As opposed to existing methods that have been proposed for this purpose, we consider a general structural reliability model given in hierarchical form. We therefore introduce a general formulation of the experimental design

problem, where we distinguish between aleatory uncertainty related to the random variable $\mathbf{X}$ and any additional epistemic uncertainty that we want to reduce through experimentation.

In this paper, we differentiate between aleatory uncertainty that we want to marginalize over (take an expectation with respect to) and additional epistemic uncertainty that we want to reduce by performing experiments. We assume that $\mathbf{X}$ is purely aleatory, and that the additional epistemic uncertainty comes from replacing the true limit-state $g(\mathbf{x})$ with a stochastic process $\xi(\mathbf{x})$. Here, we view ξ as a distribution over functions expressing our uncertainty about g. In measure-theoretic terms, we formulate this by defining two σ-algebras $\mathcal{A}$ and $\mathcal{E}$, representing aleatory and epistemic information respectively. We then assume that $\mathbf{X}$ is $\mathcal{A}$-measurable (purely aleatory) and that the $\xi(\mathbf{x})$ is $\mathcal{E}$-measurable (purely epistemic) for each fixed $\mathbf{x}$. Then $\xi(\mathbf{X})$ is a real-valued random variable which represents the structural performance of the system, and which contains both aleatory uncertainty (from $\mathbf{X}$) and epistemic uncertainty (from imperfect information of structural performance given a fixed value $\mathbf{x}$ of $\mathbf{X}$).

By conditioning on epistemic information we work with the conditional failure probability, $P(\xi(\mathbf{X}) \leq 0 \mid \mathcal{E})$, which is a $\mathcal{E}$-measurable (purely epistemic) random variable. The goal in this paper is to optimize over the types of experiments that can be performed, in order to reduce the epistemic uncertainty with respect to this failure probability. This is the *design of experiments* problem. We evaluate effectiveness of a design strategy through a measure of residual uncertainty, and propose an efficient numerical procedure for approximating this quantity. The method we propose is based on importance sampling combined with the unscented transform (Julier and Uhlmann 2004) for epistemic uncertainty propagation. With this approach we can handle hierarchical SRA models and different types of experiments under the same framework. For instance, there might be different types of experiments involved related to modelling of load effects (e.g. stress and strain) and capacity (material response to stress and strain). We implement a myopic (one-step look ahead) algorithm and demonstrate the effectiveness through a series of numerical experiments. We also discuss an example where the use of short-sighted algorithms (see Section 2.3.2) is problematic, which motivated the work leading up to a later paper (Paper IV).

3.4 Paper IV

C. Agrell, K. R. Dahl and A. Hafver (2021). Optimal sequential decision making with probabilistic digital twins. *Submitted for publication.* arXiv: 2103.07405.

This paper is, in some ways, a continuation of Paper III, but with a broader view than just failure probability estimation. If we allow for different types of experiments that can have different costs, the myopic approach to design of experiments can be problematic as we need to specify "how much a certain amount of uncertainty reduction is worth". It is also possible to construct

examples with delayed reward, i.e. where the effect on uncertainty reduction is not immediate, and then the myopic route is no longer applicable. Based on this, we wanted to study how reinforcement learning could be applied as a non-myopic alternative. We also felt that the way epistemic uncertainty was handled in Paper III could be generalized quite naturally, and we could then study the effect of epistemic uncertainty reduction in a more generic setting using the same type of framework. Instead of focusing on failure probability estimation specifically, we therefore consider a rather generic setup of a probabilistic model of a physical system, where the epistemic uncertainty is updated in discrete time.

We call this the *probabilistic digital twin* (PDT), the probabilistic analogue of the *digital twin* that consist of simulation models and data associated with a specific physical system. In the paper we treat epistemic uncertainty like in Paper III, by assuming that we have available a σ-algebra $\mathcal{E}$ representing epistemic information, and discuss how epistemic uncertainty can be updated in a probabilistic model as new information becomes available. We consider the general problem of sequential decision making, where we use the probabilistic model (the PDT) to *plan*, i.e. find an optimal action or policy for achieving some real-world objective, and propose a generic approximate solution using deep reinforcement learning together with neural networks defined on sets.

3.5 Paper V

C. Agrell, S. Eldevik, O. Gramstad and A. Hafver (2021). Risk-based functional black-box optimization – Contribution to the NASA Langley UQ challenge on optimization under uncertainty. *Mechanical Systems and Signal Processing.* Vol. 164, 108266.

In this paper we present an approach to solve the NASA Langley UQ challenge problem on optimization under uncertainty (Crespo and Kenny 2020).

> NASA missions often involve the development of new vehicles and systems that must be designed to operate in harsh domains with a wide array of operating conditions. These missions involve high-consequence and safety-critical systems for which quantitative data is either very sparse or prohibitively expensive to collect. Limited heritage data may exist, but is also usually sparse and may not be directly applicable to the system of interest, making UQ extremely challenging. NASA modeling and simulation standards require estimates of uncertainty and descriptions of any processes used to obtain these estimates. The NASA Langley Research Center has developed a UQ challenge problem in an effort to focus a community of researchers towards common goals. While the problem formulation is written in a discipline-independent framework, the underlying application is consistent with the complexities of realistic systems.
>
> — Crespo and Kenny 2020

The UQ problem deals with reliability optimization of a dynamical system and inference using functional (time series) data. With respect to Figure 1.1, $\mathbf{X}$ is here a random vector containing both aleatory and epistemic components, and the observations are time series corresponding to a subsystem. A small set of observations and a numerical black-box model of the physical system was provided. The UQ challenge involved inferring an appropriate uncertainty model for $\mathbf{X}$, followed by reliability analysis, uncertainty reduction, and reliability- and risk-based design optimization.

We took a Bayesian nonparametric approach to establish a distribution representing aleatory uncertainty, and to determine plausible values of the epistemic quantity. Similar to the method developed in Paper III, we make use of the unscented transform (Julier and Uhlmann 2004) for efficient approximate uncertainty propagation (computation of posterior moments after nonlinear transformation).

According to NASA, the dynamic system at the core of this challenge problem is highly relevant to a wide variety of systems and tasks related to e.g. aircraft gust alleviation, aeroelastic control, flutter suppression, and spacecraft precision pointing. The final results from the challenge were presented at the 2020 European Safety and Reliability (ESREL) conference, where we were able to achieve the most stable and energy efficient solution.

3.6 Paper VI

A. Hafver, C. Agrell and E. Vanem (2020). Environmental contours as Voronoi cells. *Submitted for publication.* arXiv: 2008.13480.

In this paper, we show that convex environmental contours (see Section 2.1.2) may be regarded as boundaries of Voronoi cells. This geometric interpretation leads to new theoretical insights and suggests a simple novel construction algorithm that guarantees the desired probabilistic properties. The method is illustrated with examples in two and three dimensions, but the results extend to environmental contours in arbitrary dimensions.

Inspired by the Voronoi-Delaunay duality in the numerical discrete scenario, we also derive an analytical representation where the environmental contour is considered as a differentiable manifold, and a criterion for its existence is established.

Chapter 4

Discussion

Here we end with some final reflections, discussing some identified limitations and a few ideas for further research.

4.1 Working with constraints

In Paper I we take a Bayesian approach to the inclusion of constraints. That is, given some data D and a constraint C, we are interested in $P(\cdot \mid C, D)$. The objective in Paper I is, with some abuse of notation, to establish a posterior distribution over functions $P(f \mid C, D)$, where $P(f)$ is given by a Gaussian process, D contains observations of f (potentially including noise), and C represents a set of linear constraints applied to f.

For this kind of Bayesian computation to be possible, it is necessary that the probability that the constraint holds, $P(C|D)$, is not too small. Otherwise, the method in Paper I "breaks", in the sense that there is no numerically tractable way of sampling from a posterior distribution that agrees with both C and D. In the paper we discuss a few ways of investigating this issue, by computing the *constraint probability* $P(C|D)$ directly, and by studying a regularization parameter that can be interpreted as noise applied to observations of constraints.

This kind of computation answer questions such as *"given data of the form 'f + noise', how likely is it that f is monotone?"*. As a form of hypothesis testing, this can provide useful insights. In the paper included in Appendix A we also discuss how this kind of analysis could play a role when data and constraints are combined in probabilistic modelling. A related interesting question is whether it is possible to automatically discover constraints that hold with high probability, given some set of observations.

The combination of nonparametric function approximation and constraints that are supposed to hold globally poses some computational challenges. In Paper I we try to find a small finite subset of the domain of f where the constraints are enforced, such that the constraint holds globally with high probability, which is sufficient for practical applications. Other alternatives for how to enforce constraints have also been proposed (a summary is given in Section 4 of Paper I), which are all based on some form of discretization, so it seems that for high-dimensional applications, completely new ideas will be needed.

In Paper I we also discuss the problem of optimizing the hyperparameters of a Gaussian process when constraints are applied. We later found that it was rather straightforward to derive the necessary equations for the Expectation Maximization (EM) algorithm, which provides a much more stable alternative than direct maximum likelihood estimation. It could be of interest to see how

this compares with the current practice of optimizing the unconstrained Gaussian process alone.

4.2 The unscented transform

In some of the papers we make use of the unscented transform (UT) for fast approximation of moments (mean and covariance) of a random variable after nonlinear transformation. UT is commonly applied in the context of Kalman filtering, and it is based on the general idea that *it is easier to approximate a probability distribution than an arbitrary nonlinear transformation* (Julier and Uhlmann 2004; Uhlmann 1995).

Intuitively, given a random variable $\mathbf{X}$ with finite mean and covariance, we may define a set of *weighted sigma-points* $\{(v_i, \mathbf{x}_i)\}$, such that if $\{(v_i, \mathbf{x}_i)\}$ was considered as a discrete probability distribution, then its mean and covariance would coincide with $\mathbf{X}$. For any nonlinear transformation $\mathbf{Y} = f(\mathbf{X})$, if $\mathbf{X}$ was discrete we could compute the mean and covariance of $\mathbf{Y}$ exactly. The UT approximation is the result of such computation, where we make use of a small set of weighted points in the place of $\mathbf{X}$. Specifically, for a random variable $\mathbf{X}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, a set of sigma-points for $\mathbf{X}$ is a set of weighted samples $\{(v_1, \mathbf{x}_1), \ldots, (v_k, \mathbf{x}_k)\}$ such that

$$\boldsymbol{\mu} = \sum_{i=1}^{k} v_i \mathbf{x}_i, \quad \boldsymbol{\Sigma} = \sum_{i=1}^{k} v_i (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T. \tag{4.1}$$

If $\mathbf{y} = f(\mathbf{x})$ is any (generally nonlinear) transformation, the UT approximation of the mean and covariance of $\mathbf{Y} = f(\mathbf{X})$ are then obtained as

$$\widehat{E}[\mathbf{Y}] = \sum_{i=1}^{k} v_i \mathbf{y}_i, \quad \widehat{\mathrm{Cov}}[\mathbf{Y}] = \sum_{i=1}^{k} v_i (\mathbf{y}_i - \widehat{E}[\mathbf{Y}])(\mathbf{y}_i - \widehat{E}[\mathbf{Y}])^T, \tag{4.2}$$

where $\mathbf{y}_i = f(\mathbf{x}_i)$.

Naturally, the selection of appropriate sigma-points is essential for UT to be successful. It is important to note that, although we may view the sigma-points as *weighted samples*, v_i and $\mathbf{x}_i$ are usually fixed or given by some deterministic procedure. Moreover, the definition of sigma-points given in (4.1) does not require that the weights are nonnegative and sum to one. Although this conflicts with the intuition of approximating $\mathbf{X}$ with a discrete random variable, the unscented transform still makes sense as a procedure for approximating statistics after nonlinear transformation. Also, different sigma points can of course be selected for the approximation of mean and covariance in (4.2).

Since the introduction of UT to Kalman filters in the 1990s, many different alternatives to sigma-point selection have been proposed (Menegaz et al. 2015). One of the most common versions used in Kalman filtering, the method by van der Merwe 2004, produces a set of $2n + 1$ points $\mathbf{x}_i$ with corresponding weights for the case where $\mathbf{X}$ is n-dimensional. In Paper III (Section 4.2 and Appendix

C) we present a way to apply this method to continuous random variables with values in $\mathbb{R}^n$. We make use of the same procedure for the numerical experiments also in Paper IV and Paper V. The main benefit with this approach is that only $2 \cdot \dim(\mathbf{X}) + 1$ evaluations of the function $f(\cdot)$ is required in order to approximate the moments of $f(\mathbf{X})$. Unless $\mathbf{X}$ is high-dimensional, this is much smaller than for most methods based on Monte Carlo estimation. Also, the UT approximation is extremely simple to implement, and we do not need to deal with computation of gradients which is often needed for alternative methods that are based on approximation of $f(\cdot)$.

But how suitable is it in practice? Of course, it is easy to construct pathological examples where the UT approximation breaks down, for instance by considering functions where $\{f(\mathbf{x}_i), \ldots, f(\mathbf{x}_k)\}$ is not representative for the global behavior of $f(\cdot)$. Or if we just let $f(\cdot)$ be highly irregular, then we do not expect UT to work well. In this thesis, we achieved satisfactory results with the method by van der Merwe 2004 in all of our numerical experiments, and no other alternative was investigated. One reason for this is that, for the specific applications of UT that we have considered, we were mainly interested in using it as a proxy for representing the "amount of variability" caused by a nonlinear transformation. For this, a weighted sum of $f(\cdot)$ applied to a small space-filling set of inputs seems reasonable, but the interpretation of the result as an approximation of mean or covariance is maybe less important. On the other hand, by introducing a bit of randomization, and some assumptions on the regularity of $f(\cdot)$, it seems plausible that some theoretical guarantees can be achieved, and maybe such results are already available. It would be interesting to investigate this further.

4.3 New applications of environmental contours

Today, most applications of environmental contours are restricted to two-dimensional contours. It is a well-established practice in the design of marine structures, where the two dimensions correspond to significant wave height and wave period of ocean waves. When an environment contour like the one in Figure 2.1 has been established, the task of the designer is to verify that the structure will survive any environment represented by the contour. If $\mathcal{B}$ is the set in Figure 2.1, the environmental contour is the boundary $\partial\mathcal{B}$. The prescribed upper bound on the failure probability is achieved by verifying that

$$\min_{\mathbf{x} \in \mathcal{B}} g(\mathbf{x}) > 0, \tag{4.3}$$

where the function $g(\mathbf{x})$ is the relevant limit-state. In design of marine structures, this is often achieved by evaluating $g(\mathbf{x})$ for some few *worst-case* inputs $\mathbf{x} \in \partial\mathcal{B}$, from which (4.3) follows from known properties of the physical mode of failure.

One important takeaway from this approach, is that the reliability of the system is assessed using only observations of *safe* behavior, i.e. that $g > 0$. Extending this capability to other application areas was some of the motivation

for Paper VI, where we consider how to construct environmental contours in higher dimensions.

However, the question still remains of how to *verify* an n-dimensional contour. Given the set $\mathcal{B} \subset \mathbb{R}^n$ and a set of k observations, $D = \{g(\mathbf{x}_1), \ldots, g(\mathbf{x}_k)\}$, can we conclude whether (4.3) holds? If we represent the unknown limit-state $g(\cdot)$ using a Gaussian process, the probability that (4.3) holds is the constraint probability $P(C|D)$ from Paper I, that we discussed in Section 4.1, with the linear inequality constraint $g > 0$ on $\mathcal{B}$. In Paper I we also present an approach for identifying the input locations $\mathbf{x} \in \mathcal{B}$ where the constraint, i.e. the statement $g(\mathbf{x}) > 0$, is most uncertain. This becomes a design strategy for reducing epistemic uncertainty with respect to (4.3). It could be interesting to see if combining these ideas from Paper I and Paper VI could open up for some new applications of environment contours.

Another interesting intersection between Paper I and Paper VI, is that a criterion for an environmental contour to have the desired property with respect to exceedance probability of all supporting half-spaces, assuming that the contour set itself is convex, is also given in the form of a linear constraint. This is a linear constraint involving the second-order derivatives of the *percentile function* corresponding to the input distribution under consideration, which has to be estimated before an environmental contour can be constructed. It could be of interest to see whether a constrained Gaussian process could be applied in this estimation.

4.4 Trading rigour for speed

The probabilistic approach to machine learning and AI is theoretically very appealing, but poses some numerical challenges for applications that involve non-trivial models or large sets of data. In the discussion of ML applied to safety-critical systems, it is tempting to refer to the theoretical guarantees promised by the Bayesian framework. But in practice, with the methods available today, a tradeoff between rigour and numerical efficiency has to be made.

For instance, in Paper I we developed a sampling scheme for posterior computation which is exact in the limit. We selected this approach over approximation methods such as Laplace approximations, variational Bayesian inference, expectation propagation etcetera. However, we chose to not consider a fully Bayesian approach to hyperparameter estimation, which is generally much more numerically demanding.

It is important to note that the motivation for Paper I was to develop a method suitable for *prediction*, and that this is different than when we consider problems related to *optimization*. In Paper III, Paper IV and Paper V we make use of the unscented transform, which introduces the issues discussed in Section 4.2, and in Paper IV we even rely on reinforcement learning with artificial neural networks to search for an optimal solution. But the key here is that we *search*, and with these methods we choose to optimize over many alternatives approximately instead of a few exactly. In the optimization game

everything is allowed, as long as there is a way to *verify* that the proposed solution is acceptable. For instance, like in the examples we provide in Paper III with respect to failure probability estimation. Hence, for further development of the type of models and algorithms considered in this thesis, it seems reasonable to stick with approaches where it is possible to "flip the switch" from efficiency to accuracy when needed.

Bibliography

Bertsekas, D. P. et al. (1995). *Dynamic programming and optimal control*. Vol. 1. 2. Athena scientific Belmont, MA.

Bespalov, A. et al. (2019). "Goal-oriented error estimation and adaptivity for elliptic PDEs with parametric or uncertain inputs". In: *Computer Methods in Applied Mechanics and Engineering* vol. 345, pp. 951 –982.

Brochu, E., Cora, V. M., and Freitas, N. de (2010). *A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning*. arXiv: 1012.2599.

Capodaglio, G., Gunzburger, M., and Wynn, H. P. (2021). "Approximation of Probability Density Functions for PDEs with Random Parameters Using Truncated Series Expansions". In: *Vietnam Journal of Mathematics*, pp. 2305–2228.

Cox, R. T. (1961). *The Algebra of Probable Inference*. Johns Hopkins University Press, Baltimore, MD.

Crespo, L. G. and Kenny, S. P. (2020). "The NASA Langley Challenge on Optimization Under Uncertainty". In: *Proc. ESREL 2020*.

DNVGL-RP-C205 (2019). *Environmental Conditions and Environmental Loads*. September 2019. DNVGL-RP-C205. DNV GL.

G. Matthews, A. G. de et al. (2016). "On Sparse Variational Methods and the Kullback-Leibler Divergence between Stochastic Processes". In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics* vol. 51, pp. 231–239.

Ghahramani, Z. (2015). "Probabilistic machine learning and artificial intelligence". In: *Nature* vol. 521, no. 7553, pp. 452–459.

Ghanem, R., Owhadi, H., and Higdon, D. (2017). *Handbook of Uncertainty Quantification*. Springer International Publishing.

Haver, S. (1980). *Analysis of uncertainties related to the stochastic modelling of ocean waves*. Tech. rep. UR-80-09. Norges tekniske høgskole.

— (1987). "On the joint distribution of heights and periods of sea waves". In: *Ocean Engineering* vol. 14, pp. 359–376.

Helton, J. et al. (2010). "Representation of analysis results involving aleatory and epistemic uncertainty". In: *International Journal of General Systems* vol. 39, pp. 605–646.

Hoffman, M. and Gelman, A. (2014). "The No-U-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo". In: *Journal of Machine Learning Research* vol. 15, pp. 1593–1623.

Huseby, A. B., Vanem, E., and Natvig, B. (2015). "Alternative environmental contours for structural reliability analysis". In: *Structural Safety* vol. 54, pp. 32 –45.

INI Programme on Uncertainty Quantification (2018). https://www.newton.ac.uk/files/reports/scientific/unq.pdf. Catherine Powell, Peter Challenor, Max Gunzburger and Henry Wynn. Online; accessed 28 December 2020.

Jaynes, E. T. (2003). *Probability theory: The logic of science*. Cambridge University Press.

Jefferys, W. H. and Berger, J. O. (1992). "Ockham's Razor and Bayesian Analysis". In: *American Scientist* vol. 80, no. 1, pp. 64–72.

Julier, S. and Uhlmann, J. (2004). "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE* vol. 92, pp. 401 –422.

Kennedy, M. C. and O'Hagan, A. (2001). "Bayesian calibration of computer models". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* vol. 63, no. 3, pp. 425–464.

Khan, A., Powell, C., and Silvester, D. (2018). "Robust preconditioning for stochastic Galerkin formulations of parameter-dependent linear elasticity equations". In: *SIAM Journal on Scientific Computing* vol. 41.

Kiureghian, A. D. and Ditlevsen, O. (2009). "Aleatory or epistemic? Does it matter?" In: *Structural Safety* vol. 31, no. 2. Risk Acceptance and Risk Communication, pp. 105 –112.

Lalchand, V. and Rasmussen, C. E. (2020). "Approximate Inference for Fully Bayesian Gaussian Process Regression". In: *Proceedings of The 2nd Symposium on Advances in Approximate Bayesian Inference* vol. 118, pp. 1–12.

MacKay, D. (1992). "Information-Based Objective Functions for Active Data Selection". In: *Neural Computation* vol. 4, pp. 590–604.

Madsen, H., Krenk, S., and Lind, N. (2006). *Methods of Structural Safety*. Dover Civil and Mechanical Engineering Series. Dover Publications.

Meent, J.-W. van de et al. (2018). *An Introduction to Probabilistic Programming*. arXiv: 1809.10756.

Menegaz, H. M. T. et al. (2015). "A Systematization of the Unscented Kalman Filter Theory". In: *IEEE Transactions on Automatic Control* vol. 60, no. 10, pp. 2583–2598.

Merwe, R. van der (2004). "Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models". PhD thesis. OGI School of Science and Engineering.

Murphy, K. P. (2021). *Probabilistic Machine Learning: An introduction*. MIT Press.

NORSOK N-003 (2017). *NORSOK Standard N-003:2017. Action and action effects*. Edition 3.

Ranjan, P., Bingham, D., and Michailidis, G. (2008). "Sequential Experiment Design for Contour Estimation From Complex Computer Codes". In: *Technometrics* vol. 50.

Rasmussen, C. and Ghahramani, Z. (2001). "Occam's Razor". In: *Proc. Advances in Neural Information Processing Systems* vol. 13, pp. 294–300.

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press.

Ross, E. et al. (2020). “On environmental contours for marine and coastal design”. In: *Ocean Engineering* vol. 195, p. 106194.

Sacks, J. et al. (1989). “Design and Analysis of Computer Experiments”. In: *Statist. Sci.* vol. 4, no. 4, pp. 409–423.

Seo, S. et al. (2000). “Gaussian Process Regression: Active Data Selection and Test Point Rejection”. In: *Proceedings of the International Joint Conference on Neural Networks* vol. 3, 241 –246 vol.3.

Sullivan, T. J. (2015). *Introduction to Uncertainty Quantification*. Vol. 63.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. Second edition. Adaptive Computation and Machine Learning. The MIT Press.

Uhlmann, J. (1995). “Dynamic map building and localization : New theoretical foundations.” PhD thesis. University of Oxford.

Valdebenito, M. and Schuëller, G. (2010). “A survey on approaches for reliability-based optimization”. In: *Structural and Multidisciplinary Optimization* vol. 42, pp. 645–663.

Vernon, I., Goldstein, M., and Bower, R. G. (2010). “Galaxy formation: a Bayesian uncertainty analysis”. In: *Bayesian Anal.* vol. 5, no. 4, pp. 619–669.

Wu, B. (2013). *Reliability Analysis of Dynamic Systems: Efficient Probabilistic Methods and Aerospace Applications*. Elsevier and Shanghai Jiao Tong University Press aerospace series. Elsevier Science.

Papers

Paper I

Gaussian Processes with Linear Operator Inequality Constraints

Christian Agrell

Journal of Machine Learning Research (2019) Vol. 20 no. 135 pp. 1–36.

Journal of Machine Learning Research 20 (2019) 1-36 Submitted 1/19; Revised 7/19; Published 8/19

Gaussian Processes with Linear Operator Inequality Constraints

Christian Agrell CHRISAGR@MATH.UIO.NO
Department of Mathematics
University of Oslo
P.O. Box 1053 Blindern, Oslo N-0316, Norway

Group Technology and Research
DNV GL
P.O. Box 300, 1322 Høvik, Norway

Editor: Andreas Krause

Abstract

This paper presents an approach for constrained Gaussian Process (GP) regression where we assume that a set of linear transformations of the process are bounded. It is motivated by machine learning applications for high-consequence engineering systems, where this kind of information is often made available from phenomenological knowledge. We consider a GP f over functions on $\mathcal{X} \subset \mathbb{R}^n$ taking values in $\mathbb{R}$, where the process $\mathcal{L}f$ is still Gaussian when $\mathcal{L}$ is a linear operator. Our goal is to model f under the constraint that realizations of $\mathcal{L}f$ are confined to a convex set of functions. In particular, we require that $a \leq \mathcal{L}f \leq b$, given two functions a and b where $a < b$ pointwise. This formulation provides a consistent way of encoding multiple linear constraints, such as shape-constraints based on e.g. boundedness, monotonicity or convexity. We adopt the approach of using a sufficiently dense set of virtual observation locations where the constraint is required to hold, and derive the exact posterior for a conjugate likelihood. The results needed for stable numerical implementation are derived, together with an efficient sampling scheme for estimating the posterior process.

Keywords: Gaussian processes, Linear constraints, Virtual observations, Uncertainty Quantification, Computer code emulation

1. Introduction

Gaussian Processes (GPs) are a flexible tool for Bayesian nonparametric function estimation, and widely used for applications that require inference on functions such as regression and classification. A useful property of GPs is that they automatically produce estimates on prediction uncertainty, and it is often possible to encode prior knowledge in a principled manner in the modelling of prior covariance. Some early well-known applications of GPs are within spatial statistics, e.g. meteorology (Thompson, 1956), and in geostatistics (Matheron, 1973) where it is known as *kriging*. More recently, GPs have become a popular choice within probabilistic machine learning (Rasmussen and Williams, 2005; Ghahramani, 2015). Since the GPs can act as interpolators when observations are noiseless, GPs have also become the main approach for uncertainty quantification and analysis involving computer experiments (Sacks et al., 1989; Kennedy and O'Hagan, 2001).

Often, the modeler performing function estimation has prior knowledge, or at least hypotheses, on some properties of the function to be estimated. This is typically related to the function shape with respect to some of the input parameters, such as boundedness, monotonicity or convexity. Various methods have been proposed for imposing these types of constraints on GPs (see Section 4.1 for a short review). For engineering and physics based applications, constraints based on integral operators and partial differential equations are also relevant (Jidling et al., 2017; Särkkä, 2011). What the above constraints have in common is that they are linear operators, and so any combination of such constraints can be written as a single linear operator. For instance, the constraints $a_1(\mathbf{x}) \leq f(\mathbf{x}) \leq b_1(\mathbf{x})$, $\partial f/\partial x_i \leq 0$ and $\partial^2 f/\partial x_j^2 \geq 0$ for some function (or distribution over functions) $f : X \to Y$, can be written as $a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) \leq b(\mathbf{x})$ for $a(\mathbf{x}) = [a_1(\mathbf{x}), -\infty, 0]$, $b(\mathbf{x}) = [b_1(\mathbf{x}), 0, \infty]$ and $\mathcal{L} : Y^X \to (Y^X)^3$ being the linear operator $\mathcal{L}f = [f, \partial f/\partial x_i, \partial^2 f/\partial x_j^2]$.

The motivation for including constraints is usually to improve predictions and to obtain a reduced and more realistic estimate on the uncertainty, the latter having significant impact for risk-based applications. For many real-world systems, information related to constraints in this form is often available from phenomenological knowledge. For engineering systems, this is typically knowledge related to some underlying physical phenomenon. Being able to make use of these constraint in probabilistic modelling is particularly relevant for high-consequence applications, where obtaining realistic uncertainty estimates in subsets of the domain where data is scarce is a challenge. Furthermore, information on whether these types of constraints are likely to hold given a set of observations is also useful for explainability and model falsification. For a broader discussion see (Agrell et al., 2018; Eldevik et al., 2018).

In this paper, we present a model for estimating a function $\mathrm{f} : \mathbb{R}^{n_x} \to \mathbb{R}$ by a constrained GP (CGP) $f|D, a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) \leq b(\mathbf{x})$. Here D is a set of observations of $(\mathbf{x}_j, y_j)$, possibly including additive white noise, and $f \sim \mathcal{GP}(\mu(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$ is a GP with mean $\mu(\mathbf{x})$ and covariance function $K(\mathbf{x}, \mathbf{x}')$ that are chosen such that existence of $\mathcal{L}f$ is ensured. Due to the linearity of $\mathcal{L}$, both $\mathcal{L}f|D$ and $f|D, \mathcal{L}f$ remain Gaussian, and our approach is based on modelling $f|D, \mathcal{L}f$ under the constraint $a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) \leq b(\mathbf{x})$. To model the constraint that $a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) \leq b(\mathbf{x})$ for all inputs $\mathbf{x}$, we take the approach of using a finite set of input locations where the constraint is required to hold. That is, we require that $a(\mathbf{x}_v) \leq \mathcal{L}f(\mathbf{x}_v) \leq b(\mathbf{x}_v)$ for a finite set of inputs $\{\mathbf{x}_v\}$ called the set of *virtual observation locations*. With this approach the CGP is not guaranteed to satisfy the constraint on the entire domain, but a finite set of points $\{\mathbf{x}_v\}$ can be found so that the constraint holds globally with sufficiently high probability.

The model presented in this paper is inspired by the research on shape-constrained GPs, in particular (Wang and Berger, 2016; Da Veiga and Marrel, 2012, 2015; Riihimki and Vehtari, 2010; Golchi et al., 2015; Maatouk and Bay, 2017; López-Lopera et al., 2018). We refer to Section 4 for further discussion on these alternatives. In the case where $\mathcal{L} = \partial/\partial x_i$, our approach is most similar to that of Wang and Berger (2016), where the authors make use of a similar sampling scheme for noiseless GP regression applied to computer code emulation. Many of the approaches to constrained GPs, including ours, rely on the constraint to be satisfied at a specified set of virtual locations. The use of virtual constraint observations may seem *ad hoc* at first, as the set of virtual observation locations has to be dense enough to ensure that the constraint holds globally with sufficiently high probability. Inversion

of the covariance matrix of the joint GP may therefore be of concern, both because this scales with the number of observations cubed and because there is typically high serial correlation if there are many virtual observations close together. The general solution is then to restrict the virtual observation set to regions where the probability of occurrence of the constraint is low (Riihimki and Vehtari, 2010; Wang and Berger, 2016). According to Wang and Berger (2016), when they followed this approach in their experiments, they found that only a modest number of virtual observations were typically needed, that these points were usually rather disperse, and the resulting serial correlation was not severe. We draw the same conclusion in our experiments. There is also one benefit with the virtual observation approach, which is that implementation of constraints that only hold on subsets of the domain is straightforward.

For practical use of the model presented in this paper, we also pay special attention to numerical implementation. The computations involving only real observations or only virtual observations are separated, which is convenient when only changes to the constraints are made such as in algorithms for finding a sparse set of virtual observation locations or for testing/validation of constraints. We also provide the algorithms based on Cholesky factorization for stable numerical implementation, and an efficient sampling scheme for estimating the posterior process. These algorithms are based on derivation of the exact posterior of the constrained Gaussian process using a general linear operator, and constitutes the main contribution of this paper.

The paper is structured as follows: In Section 2 we state the results needed on GP regression and GPs under linear transformations. Our main results are given in Section 3, where we introduce the constrained GP (CGP) and present the model for GP regression under linear inequality constraints. In particular, given some training data, we derive the posterior predictive distribution of the CGP evaluated at a finite set of inputs, which is a compound Gaussian with a truncated Gaussian mean (Section 3.1). Section 3.2 presents an algorithm for sampling from the posterior, and parameter estimation is addressed in Section 3.3. Section 3.4 and Section 3.5 are dedicated to optimization of the set of virtual observation locations needed to ensure that the constraint holds with sufficiently high probability. Some relevant alternative approaches from the literature on GP's under linear constraints are discussed in Section 4, followed up by numerical examples considering monotonicity and boundedness constraints. A Python implementation is available at `https://github.com/cagrell/gp_constr`, together with the code used for the examples. We end with some concluding remarks in Section 5.

2. Gaussian Processes and Linear Operators

We are interested in GP regression on functions $\mathrm{f} : \mathbb{R}^{n_x} \to \mathbb{R}$ under the additional inequality constraint $a(\mathbf{x}) \leq \mathcal{L}\mathrm{f}(\mathbf{x}) \leq b(\mathbf{x})$ for some specified functions $a(\mathbf{x})$ and $b(\mathbf{x})$, and the class of linear operators $\{\mathcal{L}|\mathcal{L}\mathrm{f} : \mathbb{R}^{n_x} \to \mathbb{R}^{n_c}\}$. Here n_x and n_c are positive integers, and the subscripts are just used to indicate the relevant underlying space over $\mathbb{R}$. We will make use of the properties of GPs under linear transformations given below.

2.1 Gaussian Process Regression

We consider a Gaussian process $f \sim \mathcal{GP}(\mu(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$ given as a prior over functions $\mathrm{f} : \mathbb{R}^{n_x} \to \mathbb{R}$, which is specified by its mean and covariance function

$$\begin{aligned} \mu(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] : \mathbb{R}^{n_x} \to \mathbb{R}, \\ K(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(\mathbf{x}'))] : \mathbb{R}^{n_x \times n_x} \to \mathbb{R}. \end{aligned} \tag{1}$$

Let $\mathbf{x}$ denote a vector in $\mathbb{R}^{n_x}$ and X the $N \times n_x$ matrix of N such input vectors. The distribution over the vector $\mathbf{f}$ of N latent values corresponding to X is then multivariate Gaussian with

$$\mathbf{f}|X \sim \mathcal{N}(\mu(X), K(X, X)),$$

where $K(X, X')$ denotes the Gram matrix $K(X, X')_{i,j} = K(\mathbf{x}_i, \mathbf{x}'_j)$ for two matrices of input vectors X and X'. Given a set of observations $Y = [y_1, \ldots, y_N]^T$, and under the assumption that the relationship between the latent function values and observed output is Gaussian, $Y|\mathbf{f} \sim \mathcal{N}(\mathbf{f}, \sigma^2 I_N)$, the predictive distribution for new observations X^* is still Gaussian with mean and covariance

$$\begin{aligned} \mathbb{E}[\mathbf{f}^*|X^*, X, Y] &= \mu(X^*) + K(X^*, X)[K(X, X) + \sigma^2 I_N]^{-1}(Y - \mu(X)), \\ \mathrm{cov}(\mathbf{f}^*|X^*, X, Y) &= K(X^*, X^*) - K(X^*, X)[K(X, X) + \sigma^2 I_N]^{-1} K(X, X^*). \end{aligned} \tag{2}$$

Here $\mathbf{f}^*|X^*$ is the predictive distribution of $f(X^*)$ and $\mathbf{f}^*|X^*, X, Y$ is the predictive posterior given the data X, Y. For further details see e.g. Rasmussen and Williams (2005).

2.2 Linear Operations on Gaussian Processes

Let $\mathcal{L}$ be a linear operator on realizations of $f \sim \mathcal{GP}(\mu(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$. As GPs are closed under linear operators (Rasmussen and Williams, 2005; Papoulis and Pillai, 2002), $\mathcal{L}f$ is still a GP [1]. We will assume that the operator produces functions with range in $\mathbb{R}^{n_c}$, but where the input domain $\mathbb{R}^{n_x}$ is unchanged. That is, the operator produces functions from $\mathbb{R}^{n_x}$ to $\mathbb{R}^{n_c}$. This type of operators on GPs has also been considered by Särkkä (2011) with applications to stochastic partial differential equations. The mean and covariance of $\mathcal{L}f$ are given by applying $\mathcal{L}$ to the mean and covariance of the argument:

$$\begin{aligned} \mathbb{E}[\mathcal{L}f(\mathbf{x})] &= \mathcal{L}\mu(\mathbf{x}) : \mathbb{R}^{n_x} \to \mathbb{R}^{n_c}, \\ \mathrm{cov}(\mathcal{L}f(\mathbf{x}), \mathcal{L}f(\mathbf{x}')) &= \mathcal{L}K(\mathbf{x}, \mathbf{x}')\mathcal{L}^T : \mathbb{R}^{n_x \times n_x} \to \mathbb{R}^{n_c \times n_c}, \end{aligned} \tag{3}$$

1. We assume here that $\mathcal{L}f$ exists. For instance, if $\mathcal{L}$ involves differentiation then the process f must be differentiable. See e.g. (Adler, 1981) for details on proving existence.

and the cross-covariance is given as

$$\begin{aligned}\operatorname{cov}(\mathcal{L}f(\mathbf{x}), f(\mathbf{x}')) &= \mathcal{L}K(\mathbf{x}, \mathbf{x}') : \mathbb{R}^{n_x \times n_x} \to \mathbb{R}^{n_c}, \\ \operatorname{cov}(f(\mathbf{x}), \mathcal{L}f(\mathbf{x}')) &= K(\mathbf{x}, \mathbf{x}')\mathcal{L}^T : \mathbb{R}^{n_x \times n_x} \to \mathbb{R}^{n_c}.\end{aligned} \tag{4}$$

The notation $\mathcal{L}K(\mathbf{x}, \mathbf{x}')$ and $K(\mathbf{x}, \mathbf{x}')\mathcal{L}^T$ is used to indicate when the operator acts on $K(\mathbf{x}, \mathbf{x}')$ as a function of $\mathbf{x}$ and $\mathbf{x}'$ respectively. That is, $\mathcal{L}K(\mathbf{x}, \mathbf{x}') = \mathcal{L}K(\mathbf{x}, \cdot)$ and $K(\mathbf{x}, \mathbf{x}')\mathcal{L} = \mathcal{L}K(\cdot, \mathbf{x}')$. With the transpose operator the latter becomes $K(\mathbf{x}, \mathbf{x}')\mathcal{L}^T = (\mathcal{L}K(\cdot, \mathbf{x}'))^T$. In the following sections we make use of the predictive distribution (2), where observations correspond to the transformed GP under $\mathcal{L}$.

3. Gaussian Processes with Linear Inequality Constraints

Following Section 2.1 and Section 2.2, we let $f \sim \mathcal{GP}(\mu(\mathbf{x}), K(\mathbf{x}, \mathbf{x}'))$ be a GP over real valued functions on $\mathbb{R}^{n_x}$, and $\mathcal{L}$ a linear operator producing functions from $\mathbb{R}^{n_x}$ to $\mathbb{R}^{n_c}$. The matrix X and the vector Y will represent N noise perturbed observations: $y_i = f(\mathbf{x}_i) + \varepsilon_i$ with ε_i i.i.d. $\mathcal{N}(0, \sigma^2)$ for $i = 1, \ldots, N$.

We would like to model the posterior GP conditioned on the observations X, Y, and on the event that $a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) \leq b(\mathbf{x})$ for two functions $a(\mathbf{x}), b(\mathbf{x}) : \mathbb{R}^{n_x} \to (\mathbb{R} \cup \{-\infty, \infty\})^{n_c}$, where $a_i(\mathbf{x}) < b_i(\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^{n_x}$ and $i = 1, \ldots, n_c$. To achieve this approximately, we start by assuming that the constraint $a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) \leq b(\mathbf{x})$ only holds at a finite set of inputs $\mathbf{x}_1^v, \ldots, \mathbf{x}_S^v$ that we refer to as *virtual observation locations*. Later, we will consider how to specify the set of virtual observation locations such that the constraint holds for any $\mathbf{x}$ with sufficiently high probability. Furthermore, we will also assume that *virtual observations* of the transformed process, $\mathcal{L}f(\mathbf{x}_i^v)$, comes with additive white noise with variance σ_v^2. We can write this as $a(X^v) \leq \mathcal{L}f(X^v) + \varepsilon^v \leq b(X^v)$, where $X^v = [\mathbf{x}_1^v, \ldots, \mathbf{x}_S^v]^T$ is the matrix containing the virtual observation locations and ε^v is a multivariate Gaussian with diagonal covariance of elements σ_v^2.

We will make use of the following notation: Let $\widetilde{C}(X^v) \in \mathbb{R}^{S \times n_c}$ be the matrix with rows $(\widetilde{C}(X^v))_i = \mathcal{L}f(\mathbf{x}_i^v) + \varepsilon_i^v$ for i.i.d. $\varepsilon_i^v \sim \mathcal{N}(\mathbf{0}, \sigma_v^2 I_{n_c})$, and let $C(X^v)$ denote the event $C(X^v) := \cap_{i=1}^S \{a(\mathbf{x}_i^v) \leq (\widetilde{C}(X^v))_i \leq b(\mathbf{x}_i^v)\}$. $C(X^v)$ thus represents the event that the constraint $a(\mathbf{x}) \leq \mathcal{L}f(\mathbf{x}) + \varepsilon^v \leq b(\mathbf{x})$ is satisfied for all points in X^v, and it is defined through the latent variable $\widetilde{C}(X^v)$.

In summary, the process we will consider is stated as

$$f | X, Y, X^v, C(X^v) := f | f(X) + \varepsilon = Y, a(X^v) \leq \mathcal{L}f(X^v) + \varepsilon^v \leq b(X^v),$$

where f is a Gaussian process, X, Y is the training data and X^v are the locations where the transformed process $\mathcal{L}f + \varepsilon^v$ is bounded. The additive noise ε and ε^v are multivariate Gaussian with diagonal covariance matrices of elements σ^2 and σ_v^2 respectively.

Here we assume that observations of all parts of $\mathcal{L}f$ comes with i.i.d. white noise with variance σ_v^2. The reason for this is mainly for numerical stability, where we in computations will choose a tiny variance to approximate noiseless observations. Similarly, σ^2 may be chosen as a fixed small number for interpolation in the standard GP regression setting. In the following derivations, the results for exact noiseless observations can be obtained by setting the relevant variance to zero.

We also assume that any sub-operator of $\mathcal{L}$ is constrained at the same set of virtual locations X^v. This is mainly for notational convenience, and this assumption will be relaxed in Section 3.5. In the following, we let N_v denote the total number of virtual observation locations. Here $N_v = S \cdot n_c$ for now, whereas we will later consider $N_v = \sum_{i=1}^{n_c} S_i$ where the i-th sub-operator is associated with S_i virtual observation locations.

3.1 Posterior Predictive Distribution

Our goal is to obtain the posterior predictive distribution $\mathbf{f}^*|X^*, X, Y, X^v, C(X^v)$. That is: the distribution of $\mathbf{f}^* = f(X^*)$ for some new inputs X^*, conditioned on the observed data $Y = f(X) + \varepsilon$ and the constraint $a(X^v) \leq \mathcal{L}f(X^v) + \varepsilon^v \leq b(X^v)$.

To simplify the notation we write $\mathbf{f}^*|Y, C$, excluding the dependency on inputs X, X^* and X^v (as well as any hyperparameter of the mean and covariance function). The posterior predictive distribution is given by marginalizing over the latent variable $\widetilde{C}$:

$$
\begin{aligned}
p(\mathbf{f}^*, C|Y) &= p(\mathbf{f}^*|C, Y)p(C|Y), \\
p(\mathbf{f}^*|C, Y) &= \int_{a(X^v)}^{b(X^v)} p(\mathbf{f}^*|\widetilde{C}, Y)p(\widetilde{C}|Y)d\widetilde{C}, \\
p(C|Y) &= \int_{a(X^v)}^{b(X^v)} p(\widetilde{C}|Y)d\widetilde{C},
\end{aligned}
$$

where the limits correspond to the hyper-rectangle in $\mathbb{R}^{N_v}$ given by the functions $a(\cdot)$ and $b(\cdot)$ evaluated at each $\mathbf{x}^v \in X^v$. The predictive distribution and the probability $p(C|Y)$ are given in Lemma 1. $p(C|Y)$ is of interest, as it is the probability that the constraint holds at X^v given the data Y.

In the remainder of the paper we will use the shortened notation $\mu^* = \mu(X^*)$, $\mu = \mu(X)$, $\mu^v = \mu(X^v)$ and $K_{X,X'} = K(X, X')$. For vectors with elements in $\mathbb{R}^{n_c}$, such as $\mathcal{L}\mu^v$, we interpret this elementwise. E.g. $\mathcal{L}\mu^v(X^v)$ is given by the column vector $[\mathcal{L}\mu(\mathbf{x}_1^v)_1, \ldots, \mathcal{L}\mu(\mathbf{x}_1^v)_{n_c}, \ldots, \mathcal{L}\mu(\mathbf{x}_S^v)_1, \ldots, \mathcal{L}\mu(\mathbf{x}_S^v)_{n_c}]$.

We start by deriving the posterior predictive distribution $\mathbf{f}^*$ at some new locations X^*. The predictive distribution is represented by a Gaussian, $\mathbf{f}^*|Y, C \sim \mathcal{N}(\mu(\mathbf{C}), \Sigma)$, for some fixed covariance matrix Σ and a mean $\mu(\mathbf{C})$ that depends on the random variable $\mathbf{C} = \widetilde{C}|Y, C$. The variable $\widetilde{C} = \mathcal{L}f(X^v) + \varepsilon^v$ remains Gaussian after conditioning on the observations Y, i.e. $\widetilde{C}|Y \sim \mathcal{N}(\nu_c, \Sigma_c)$ with some expectation ν_c and covariance matrix Σ_c that can be computed using (3, 4). Applying the constraints represented by the event C on the random variable $\widetilde{C}|Y$ just means restricting $\widetilde{C}|Y$ to lie in the hyper-rectangle defined by the bounds $a(X^v)$ and $b(X^v)$. This means that $\mathbf{C} = \widetilde{C}|Y, C$ is a truncated multivariate Gaussian, $\mathbf{C} \sim \mathcal{TN}(\nu_c, \Sigma_c, a(X^v), b(X^v))$. The full derivation of the distribution parameters of $\mathbf{C}$ and $\mathbf{f}^*|Y, C$ are given in Lemma 1 below, whereas Lemma 2 provides an alternative algorithmic representation suitable for numerical implementation.

Lemma 1 *The predictive distribution $\mathbf{f}^*|Y, C$ is a compound Gaussian with truncated Gaussian mean:*

$$
\mathbf{f}^*|Y, C \sim \mathcal{N}(\mu^* + A(\boldsymbol{C} - \mathcal{L}\mu^v) + B(Y - \mu), \Sigma), \tag{5}
$$

$$\boldsymbol{C} = \widetilde{C}|Y, C \sim \mathcal{TN}(\mathcal{L}\mu^v + A_1(Y-\mu), B_1, a(X^v), b(X^v)), \tag{6}$$

where $\mathcal{TN}(\cdot,\cdot,a,b)$ *is the Gaussian* $\mathcal{N}(\cdot,\cdot)$ *conditioned on the hyper-rectangle* $[a_1,b_1]\times\cdots\times[a_k,b_k]$*, and*

$$\begin{array}{ll} A_1 = (\mathcal{L}K_{X^v,X})(K_{X,X}+\sigma^2 I_N)^{-1}, & B_1 = \mathcal{L}K_{X^v,X^v}\mathcal{L}^T + \sigma_v^2 I_{N_v} - A_1 K_{X,X^v}\mathcal{L}^T, \\ A_2 = K_{X^*,X}(K_{X,X}+\sigma^2 I_N)^{-1}, & B_2 = K_{X^*,X^*} - A_2 K_{X,X^*}, \\ & B_3 = K_{X^*,X^v}\mathcal{L}^T - A_2 K_{X,X^v}\mathcal{L}^T, \end{array}$$
$$A = B_3 B_1^{-1}, \quad B = A_2 - AA_1, \quad \Sigma = B_2 - AB_3^T.$$

Moreover, the probability that the unconstrained version of $\boldsymbol{C}$ *falls within the constraint region,* $p(C|Y)$*, is given by*

$$p(C|Y) = p\left(a(X^v) \le \mathcal{N}(\mathcal{L}\mu^v + A_1(Y-\mu), B_1) \le b(X^v)\right), \tag{7}$$

and the unconstrained predictive distribution is

$$\mathbf{f}^*|Y \sim \mathcal{N}(\mu^* + A_2(Y-\mu), B_2).$$

The derivation in Lemma 1 is based on conditioning the multivariate Gaussian $(\mathbf{f}^*, Y, \widetilde{C})$, and the proof is given in Appendix A. For practical implementation the matrix inversions involved in Lemma 1 may be prone to numerical instability. A numerically stable alternative is given in Lemma 2 below.

In the following lemma, $Chol(K)$ is the lower triangular Cholesky factor of a matrix K. We also let $R = (P \setminus Q)$ denote the solution to the linear system $PR = Q$ for matrices P and Q, which may be efficiently computed when P is triangular using forward or backward substitution.

Lemma 2 *Let* $L = Chol(K_{X,X}+\sigma^2 I_N)$*,* $v_1 = L \setminus K_{X,X^v}\mathcal{L}^T$ *and* $v_2 = L \setminus K_{X,X^*}$*.*

Then the matrices in Lemma 1 can be computed as

$$\begin{array}{ll} A_1 = (L^T \setminus v_1)^T, & B_1 = \mathcal{L}K_{X^v,X^v}\mathcal{L}^T + \sigma_v^2 I_{N_v} - v_1^T v_1, \\ A_2 = (L^T \setminus v_2)^T, & B_2 = K_{X^*,X^*} - v_2^T v_2, \\ & B_3 = K_{X^*,X^v}\mathcal{L}^T - v_2^T v_1. \end{array}$$

Moreover, B_1 *is symmetric and positive definite. By letting* $L_1 = Chol(B_1)$ *and* $v_3 = L_1 \setminus B_3^T$ *we also have*

$$A = (L_1^T \setminus v_3)^T, \quad B = A_2 - AA_1, \quad \Sigma = B_2 - v_3^T v_3.$$

The proof is given in Appendix B. The numerical complexity of the procedures in Lemma 2 is $n^3/6$ for Cholesky factorization of $n \times n$ matrices and $mn^2/2$ for solving triangular systems where the unknown matrix is $n \times m$. In the derivation of Lemma 1 and Lemma 2, the order of operations was chosen such that the first Cholesky factor $L = Chol(K_{X,X}+\sigma^2 I_N)$ only depends on X. This is convenient in the case where the posterior $\mathbf{f}^*|Y, C$ is calculated multiple times for different constraints C or virtual observations X^v, but where the data X, Y remain unchanged.

3.2 Sampling from the Posterior Distribution

In order to sample from the posterior we can first sample from the constraint distribution (6), and then use these samples in the mean of (5) to create the final samples of $\mathbf{f}^*|Y, C$.

To generate k samples of the posterior at M new input locations, $[\mathbf{x}_1^*, \ldots, \mathbf{x}_M^*]^T = X^*$, we use the following procedure

Algorithm 3 *Sampling from the posterior distribution*

1. *Find a matrix Q s.t. $Q^TQ = \Sigma \in \mathbb{R}^{M\times M}$, e.g. by Cholesky or a spectral decomposition.*
2. *Generate $\widetilde{C}_k$, a $N_v \times k$ matrix where each column is a sample of $\widetilde{C}|Y, C$ from the distribution in* (6).
3. *Generate U_k, a $M \times k$ matrix with k samples from the standard normal $\mathcal{N}(\boldsymbol{0}, I_M)$.*
4. *The $M \times k$ matrix where each column in a sample from $\mathbf{f}^*|Y, C$ is then obtained by*

$$[\mu^* + B(Y - \mu)] \oplus_{col} [A(-\mathcal{L}\mu^v \oplus_{col} \widetilde{C}_k) + QU_k],$$

where $\oplus_{col}$ means that the $M \times 1$ vector on the left hand side is added to each column of the $M \times k$ matrix on the right hand side.

This procedure is based on the well-known method for sampling from multivariate Gaussian distributions, where we have used the property that in the distribution of $\mathbf{f}^*|Y, C$, only the mean depends on samples from the constraint distribution.

The challenging part of this procedure is the second step where samples have to be drawn from a truncated multivariate Gaussian. The simplest approach is by rejection sampling, i.e. generating samples from the normal distribution and rejection those that fall outside the bounds. In order to generate m samples with rejection sampling, the expected number of samples needed is $m/p(C|Y)$, where the acceptance rate is the probability $p(C|Y)$ given in (7). If the acceptance rate is low, then rejection sampling becomes inefficient, and an alternative approach such as Gibbs sampling (Kotecha and Djuric, 1999) is typically used. In our numerical experiments (presented in Section 4.2) we made use of a new method based on simulation via minimax tilting by Botev (2017), developed for high-dimensional exact sampling. Botev (2017) prove strong efficiency properties and demonstrate accurate simulation in dimensions $d \sim 100$ with small acceptance probabilities ($\sim 10^{-100}$), that take about the same time as one cycle of Gibbs sampling. For higher dimensions in the thousands, the method is used to accelerate existing Gibbs samplers by sampling jointly hundreds of highly correlated variables. In our experiments, we experienced that this method worked well in cases where Gibbs sampling was challenging. A detailed comparison with other sampling alternatives for an application similar to ours is also given in (López-Lopera et al., 2018). An important observation in Algorithm 3 is that for inference at a new set of input locations X^*, when the data X, Y and virtual observation locations X^v are unchanged, the samples generated in step *2* can be reused.

3.3 Parameter Estimation

To estimate the parameters of the CGP we make use of the marginal maximum likelihood approach (MLE). We define the marginal likelihood function of the CGP as

$$L(\theta) = p(Y, C|\theta) = p(Y|\theta)p(C|Y, \theta), \tag{8}$$

i.e. as the probability of the data Y and constraint C combined, given the set of parameters represented by θ. We assume that both the mean and covariance function of the GP prior (1) $\mu(\mathbf{x}|\theta)$ and $K(\mathbf{x}, \mathbf{x}'|\theta)$ may depend on θ. The log-likelihood, $l(\theta) = \ln p(Y|\theta) + \ln p(C|Y, \theta)$, is thus given as the sum of the unconstrained log-likelihood, $\ln p(Y|\theta)$, which is optimized in unconstrained MLE, and $\ln p(C|Y, \theta)$, which is the probability that the constraint holds at X^v given in (7).

In (Bachoc et al., 2018) the authors study the asymptotic distribution of the MLE for shape-constrained GPs, and show that for large sample sizes the effect of including the constraint in the MLE is negligible. But for small or moderate sample sizes the constrained MLE is generally more accurate, so taking the constraint into account is beneficial. However, due to the added numerical complexity in optimizing a function that includes the term $\ln p(C|Y, \theta)$, it might not be worthwhile. Efficient parameter estimation using the full likelihood (8) is a topic of future research. In the numerical experiments presented in this paper, we therefore make use of the unconstrained MLE. This also makes it possible to compare models with and without constraints in a more straightforward manner.

3.4 Finding the Virtual Observation Locations

For the constraint to be satisfied locally at any input location in some bounded set $\Omega \subset \mathbb{R}^{n_x}$ with sufficiently high probability, the set of virtual observation locations X^v has to be sufficiently dense. We will specify a target probability $p_{\text{target}} \in [0, 1)$ and find a set X^v, such that when the constraint is satisfied at all virtual locations in X^v, the probability that the constraint is satisfied for any $\mathbf{x}$ in Ω is at least p_{target}. The number of virtual observation locations needed depends on the smoothness properties of the kernel, and for a given kernel it is of interest to find a set X_v that is effective in terms of numerical computation. As we need to sample from a truncated Gaussian involving cross-covariances between all elements in X^v, we would like the set X^v to be small, and also to avoid points in X^v close together that could lead to high serial correlation.

Seeking an optimal set of virtual observation locations has also been discussed in (Wang and Berger, 2016; Golchi et al., 2015; Riihimki and Vehtari, 2010; Da Veiga and Marrel, 2012, 2015), and the intuitive idea is to iteratively place virtual observation locations where the probability that the constraint holds is low. The general approach presented in this section is most similar to that of Wang and Berger (2016). In Section 3.5 we extend this to derive a more efficient method for multiple constraints.

In order to estimate the probability that the constraint holds at some new location $\mathbf{x}^* \in \Omega$, we first derive the posterior distribution of the constraint process.

Lemma 4 *The predictive distribution of the constraint $\mathcal{L}f(\boldsymbol{x}^*)$ for some new input $\boldsymbol{x}^* \in \mathbb{R}^{n_x}$, condition on the data Y is given by*

$$\mathcal{L}f(\boldsymbol{x}^*)|Y \sim \mathcal{N}(\mathcal{L}\mu^* + \widetilde{A}_2(Y - \mu), \widetilde{B}_2), \tag{9}$$

and when $\mathcal{L}f(\boldsymbol{x}^*)$ is conditioned on both the data and virtual constraint observations, X, Y and $X^v, C(X^v)$, the posterior becomes

$$\mathcal{L}f(\boldsymbol{x}^*)|Y, C \sim \mathcal{N}(\mathcal{L}\mu^* + \widetilde{A}(\boldsymbol{C} - \mathcal{L}\mu^v) + \widetilde{B}(Y - \mu), \widetilde{\Sigma}). \tag{10}$$

Here L, v_1, A_1, B_1 and L_1 are defined as in Lemma 2 , $\boldsymbol{C}$ is the distribution in (6) *and*

$$\begin{aligned} \widetilde{v}_2 = L \setminus K_{X,\boldsymbol{x}^*}\mathcal{L}^T, \quad & \widetilde{B}_2 = \mathcal{L}K_{\boldsymbol{x}^*,\boldsymbol{x}^*}\mathcal{L}^T - \widetilde{v}_2^T\widetilde{v}_2, \\ \widetilde{A}_2 = (L^T \setminus \widetilde{v}_2)^T, \quad & \widetilde{B}_3 = \mathcal{L}K_{\boldsymbol{x}^*,X^v}\mathcal{L}^T - \widetilde{v}_2^T v_1, \\ & \widetilde{v}_3 = L_1 \setminus \widetilde{B}_3^T, \\ \widetilde{A} = (L_1^T \setminus \widetilde{v}_3)^T, \quad & \widetilde{B} = \widetilde{A}_2 - \widetilde{A}A_1, \quad \widetilde{\Sigma} = \widetilde{B}_2 - \widetilde{v}_3^T\widetilde{v}_3. \end{aligned}$$

The proof is given in Appendix D. The predictive distribution in Lemma 4 was defined for a single input $\mathbf{x}^* \in \mathbb{R}^{n_x}$, and we will make use of the result in this context. But we could just as well consider an input matrix X^* with rows $\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots$, where the only change in Lemma 4 is to replace $\mathbf{x}^*$ with X^*. In this case we also note that the variances, $\mathrm{diag}(\widetilde{\Sigma})$, is more efficiently computed as $\mathrm{diag}(\widetilde{\Sigma}) = \mathrm{diag}(\mathcal{L}K_{X^*,X^*}\mathcal{L}^T) - \mathrm{diag}(\widetilde{v}_2^T\widetilde{v}_2) - \mathrm{diag}(\widetilde{v}_3^T\widetilde{v}_3)$ where we recall that $\mathrm{diag}(v^T v)_i = \sum_j v_{i,j}^2$ for $v^T = [v_{i,j}]$.

Using the posterior distribution of $\mathcal{L}f$ in Lemma 4 we define the constraint probability $p_c : \mathbb{R}^{n_x} \to [0, 1]$ as

$$p_c(\mathbf{x}) = P\left(a(\mathbf{x}) - \nu < \xi(\mathbf{x}, X^v) < b(\mathbf{x}) + \nu\right), \tag{11}$$

where $\xi(\mathbf{x}, X^v) = \mathcal{L}f(\mathbf{x}^*)|Y$ for $X^v = \emptyset$ and $\xi(\mathbf{x}, X^v) = \mathcal{L}f(\mathbf{x}^*)|Y, C$ otherwise. The quantity ν is a non-negative fixed number that is included to ensure that it will be possible to increase p_c using observations with additive noise. When we use virtual observations $\widetilde{C}(\mathbf{x}) = \mathcal{L}f(\mathbf{x}^*) + \varepsilon^v$ that come with noise $\varepsilon^v \sim \mathcal{N}(0, \sigma_v^2)$, we can use $\nu = \max\{\sigma_v\Phi^{-1}(p_{\text{target}}), 0\}$ where $\Phi(\cdot)$ is the normal cumulative distribution function. Note that σ_v, and in this case ν, will be small numbers included mainly for numerical stability. In the numerical examples presented in this paper this noise variance was set to 10^{-6}.

In the case where $X^v = \emptyset$, computation of (11) is straightforward as $\xi(\mathbf{x}, X^v)$ is Gaussian. Otherwise, we will rely on the following estimate of $p_c(\mathbf{x})$:

$$\hat{p}_c(\mathbf{x}) = \frac{1}{m}\sum_{j=1}^{m} P\left(a(\mathbf{x}) - \nu < (\mathcal{L}f(\mathbf{x})|Y, C_j) < b(\mathbf{x}) + \nu\right), \tag{12}$$

where $C_1, \ldots, C_m$ are m samples of $\mathbf{C}$ given in (6).

We outline an algorithm for finding a set of virtual observation locations X^v, such that the probability that the constraint holds locally at any $\mathbf{x} \in \Omega$ is at least p_{target} for some specified set $\Omega \subset \mathbb{R}^{n_x}$ and $p_{\text{target}} \in [0, 1)$. That is, $\min_{\mathbf{x}\in\Omega} p_c(\mathbf{x}) \geq p_{\text{target}}$. The algorithm can be used starting with no initial virtual observation locations, $X^v = \emptyset$, or using some pre-defined set $X^v \neq \emptyset$. The latter may be useful e.g. if the data X, Y is updated, in which case only a few additions to the previous set X^v might be needed.

Algorithm 5 *Finding locations of virtual observations X^v s.t. $\hat{p}_c(\boldsymbol{x}) \geq p_{target}$ for all $\boldsymbol{x} \in \Omega$.*

1. *Compute $L = Chol(K_{X,X} + \sigma^2 I_N)$.*

2. Until convergence do:

(a) If $X^v \neq \emptyset$ compute A_1 and B_1 as defined in Lemma 2, and generate m samples $C_1, \ldots, C_m$ of $\boldsymbol{C}$ given in (6).

(b) If $X^v = \emptyset$ compute $(\boldsymbol{x}^*, p^*) = (\arg\min p_c(\boldsymbol{x}), p_c(\boldsymbol{x}^*))$. Otherwise compute $(\boldsymbol{x}^*, p^*) = (\arg\min \hat{p}_c(\boldsymbol{x}), \hat{p}_c(\boldsymbol{x}^*))$ with $\hat{p}_c$ defined as in (12), using the samples generated in step (a).

(c) Terminate if $p^* \geq p_{target}$, otherwise update $X^v \to X^v \cup \{\boldsymbol{x}^*\}$.

The rate of convergence of Algorithm 5 relies on the probability that the constraint holds initially, $P\left(a(\mathbf{x}) < (\mathcal{L}f(\mathbf{x})|Y) < b(\mathbf{x})\right)$, and for practical application one may monitor p^* as a function of the number of virtual observation locations, $|X^v|$, to find an appropriate stopping criterion.

With the exception of low dimensional input $\mathbf{x}$, the optimization step $\mathbf{x}^* = \arg\min \hat{p}_c(\mathbf{x})$ is in general a hard non-convex optimization problem. But with respect to how $\mathbf{x}^*$ and p^* are used in the algorithm, some simplifications can be justified. First, we note that when computing $\hat{p}_c(\mathbf{x})$ with (12) for multiple $\mathbf{x} = \mathbf{x}_1, \mathbf{x}_2, \ldots$, the samples $C_1, \ldots, C_m$ are reused. It is also not necessary to find the the absolute minimum, as long as a *small enough* value is found in each iteration. Within the global optimization one might therefore decide to stop after the first occurrence of $\hat{p}_c(\mathbf{x})$ less than some threshold value. With this idea one could also search over finite candidate sets $\Omega \subset \mathbb{R}^{n_x}$, using a fixed number of random points in $\mathbb{R}^{n_x}$. This approach might produce a larger set X^v, but where the selection of $\mathbf{x}^*$ is faster in each iteration. Some of the alternative strategies for locating $\mathbf{x}^*$ in Algorithm 5 are studied further in our numerical experiments in Section 4.2.

With the above algorithm we aim to impose constraints on some bounded set $\Omega \subset \mathbb{R}^{n_x}$. Here Ω has to be chosen with respect to both training and test data. For a single boundedness constraint, it might be sufficient that the constraint only holds at the points $\mathbf{x} \in \mathbb{R}^{n_x}$ that will be used for prediction. But if we consider constraints related to monotonicity (see Example 1, Section 4.2), dependency with respect to the latent function's properties at the training locations is lost with this strategy. In the examples we give in this paper we consider a convex set Ω, in particular $\Omega = [0, 1]^{n_x}$, and assume that training data, test data and any input relevant for prediction lies within Ω.

3.5 Separating Virtual Observation Locations for Sub-operators

Let $\mathcal{L}$ be a linear operator defined by the column vector $[\mathcal{F}_1, \ldots, \mathcal{F}_k]$, where each $\mathcal{F}_i$ is a linear operator leaving both the domain and range of its argument unchanged, i.e. $\mathcal{F}_i$ produces functions from $\mathbb{R}^{n_x}$ to $\mathbb{R}$, subjected to an interval constraint $[a_i(\mathbf{x}), b_i(\mathbf{x})]$. Until now we have assumed that the constrain holds at a set of virtual observation locations X^v, which means that $a_i(X^v) \leq \mathcal{F}_i f(X^v) \leq b_i(X^v)$ for all $i = 1, \ldots, k$.

However, it might not be necessary to constrain each of the sub-operators $\mathcal{F}_i$ at the same points $\mathbf{x}^v \in X^v$. Intuitively, constraints with respect to $\mathcal{F}_i$ need only be imposed at locations where $p(\mathcal{F}_i f(\mathbf{x}) \notin [a_i(\mathbf{x}), b_i(\mathbf{x})])$ is large. To accommodate this we let X^v be the concatenation of the matrices $X^{v,1}, \ldots, X^{v,k}$ and define $\mathcal{L}^T f(X^v) = [\mathcal{F}_1^T f(X^{v,1}), \ldots, \mathcal{F}_k^T f(X^{v,1})]^T$. This is equivalent to removing some of the rows in $\mathcal{L}(\cdot)(X^v)$, and all of the results in this paper still apply. In this setting we can improve the algorithm in Section 3.4 for finding the

set of virtual observation locations by considering each sub-operator individually. This is achieved using the estimated partial constraint probabilities, $p_{c,i}(\mathbf{x})$, that we defined as in (11) by considering only the i-th sub-operator. We may then use the estimate

$$\hat{p}_{c,i}(\mathbf{x}) = \frac{1}{m}\sum_{j=1}^{m} P\left(a_i(\mathbf{x}) - \nu < (\mathcal{L}f(\mathbf{x})|Y, C_j)_i < b_i(\mathbf{x}) + \nu\right), \tag{13}$$

where $(\mathcal{L}f(\mathbf{x})|Y, C_j)_i$ is the univariate Normal distribution given by the i-th row of $(\mathcal{L}f(\mathbf{x})|Y, C_j)$, and $C_1, \ldots, C_m$ are m samples of $\mathbf{C}$ given in (6) as before. Algorithm 5 can then be improved by minimizing (13) with respect to both $\mathbf{x}$ and $i = 1, \ldots k$. The details are presented in Appendix C, Algorithm 7.

3.6 Prediction using the Posterior Distribution

For the unconstrained GP in this paper where the likelihood is given by Gaussian white noise, the posterior mean and covariance is sufficient to describe predictions as the posterior remains Gaussian. It is also known that in this case there is a correspondence between the posterior mean of the GP and the optimal estimator in the Reproducing Kernel Hilbert Space (RKHS) associated with the GP (Kimeldorf and Wahba, 1970). This is a Hilbert space of functions defined by the positive semidefinite kernel of the GP. Interestingly, a similar correspondence holds for the constrained case. Maatouk et al. (2016) show that for constrained interpolation, the Maximum *A Posteriori* (MAP) or mode of the posterior is the optimal constrained interpolation function in the RKHS, and also illustrate in simulations that the unconstrained mean and constrained MAP coincide only when the unconstrained mean satisfies the constraint. This holds when the GP is constrained to a convex set of functions, which is the case in this paper where we condition on linear transformations of a function restricted to a convex set.

3.7 An Alternative Approach based on Conditional Expectations

Da Veiga and Marrel (2012, 2015) propose an approach for approximating the first two moments of the constrained posterior, $\mathbf{f}^*|Y, C$, using conditional expectations of the truncated multivariate Gaussian. This means, in the context of this paper, that the first two moments of $\mathbf{f}^*|Y, C$ are computed using the first two moments of the latent variable $\mathbf{C}$. To apply this idea using the formulation of this paper, we can make use of the following result.

Corollary 6 *Let the matrices A, B, Σ and the truncated Gaussian random variable $\boldsymbol{C}$ be as defined in Lemma 1, and let ν, Γ be the expectation and covariance of $\boldsymbol{C}$. Then the expectation and covariance of the predictive distribution $\boldsymbol{f}^*|Y, C$ are given as*

$$\begin{aligned} \mathbb{E}(\boldsymbol{f}^*|Y, C) &= \mu^* + A(\nu - \mathcal{L}\mu^v) + B(Y - \mu), \\ cov(\boldsymbol{f}^*|Y, C) &= \Sigma + A\Gamma A^T. \end{aligned} \tag{14}$$

Moreover, if $\widetilde{A}$, $\widetilde{B}$ and $\widetilde{\Sigma}$ are the matrices defined in Lemma 4, then the expectation and variance of the predictive distribution of the constraint $\mathcal{L}f(\boldsymbol{x}^*)|Y, C$ are given as

$$\begin{aligned} \mathbb{E}(\mathcal{L}f(\boldsymbol{x}^*)|Y, C) &= \mathcal{L}\mu^* + \widetilde{A}(\nu - \mathcal{L}\mu^v) + \widetilde{B}(Y - \mu), \\ var(\mathcal{L}f(\boldsymbol{x}^*)|Y, C) &= \widetilde{\Sigma} + \widetilde{A}\Gamma\widetilde{A}^T. \end{aligned} \tag{15}$$

The results follows directly from the distributions derived in Lemmas 1 and 4, and moments of compound distributions. A proof is included in Appendix E for completeness.

Da Veiga and Marrel (2012, 2015) make use of a Genz approximation (Genz, 1992, 1997) to compute ν, Γ for inference using (14). They also introduce a crude but faster correlation-free approximation that can be used in the search for virtual observation locations. With this approach, (15) is used where ν, Γ are computed under the assumption that $\mathrm{cov}(\widetilde{C}|Y)$ is diagonal. We can state this approximation as follows:

$$\nu_i \approx m_i + s_i \frac{\phi(\widetilde{a_i}) - \phi(\widetilde{b_i})}{\Phi(\widetilde{b_i}) - \Phi(\widetilde{a_i})}, \quad \Gamma_{i,i} \approx s_i^2 \left[1 + \frac{\widetilde{a_i}\phi(\widetilde{a_i}) - \widetilde{b_i}\phi(\widetilde{b_i})}{\Phi(\widetilde{b_i}) - \Phi(\widetilde{a_i})} - \left(\frac{\phi(\widetilde{a_i}) - \phi(\widetilde{b_i})}{\Phi(\widetilde{b_i}) - \Phi(\widetilde{a_i})} \right)^2 \right],$$

where m_i is the i-th component of $\mathbb{E}(\widetilde{C}|Y) = \mathcal{L}\mu^v + A_1(Y - \mu)$, $s_i = \sqrt{\mathrm{cov}(\widetilde{C}|Y)_{i,i}} = \sqrt{(B_1)_{i,i}}$, $\widetilde{a_i} = (a(X^v)_i - m_i)/s_i$, $\widetilde{b_i} = (b(X^v)_i - m_i)/s_i$, ϕ and Φ are the pdf and cdf of the standard normal distribution and Γ is diagonal with elements $\Gamma_{i,i}$. We will make use of these approximations in some of the examples in Section 4.2 for comparison.

3.8 Numerical Considerations

For numerical implementation, we discuss some key considerations with the proposed model. One of the main issues with implementation of GP models in terms of numerical stability is related to covariance matrix inversion, which is why alternatives such as Cholesky factorization are recommended in practice. This does however not alleviate problems related to ill-conditioned covariance matrices. This is a common problem in computer code emulation (zero observational noise) in particular, where training points might be 'too close to each other' in terms of the covariance function, leaving the covariance matrix close to degenerate as some of the observations become redundant. A common remedy is to introduce a 'nugget' term on the diagonal entries of the covariance matrix, in the form of additional white noise on the observations. This means using a small $\sigma > 0$ instead of $\sigma = 0$ in Equation (2), even when the observations are noiseless. In terms of matrix regularization this is equivalent to Tikhonov regularization. See for instance Ranjan et al. (2010) and Andrianakis and Challenor (2012) which give a detailed discussion and recommendations for how to choose appropriate value for σ. In practice, a fixed small value is often used without further analysis, as long as the resulting condition number is not too high. This approach can be justified since the use of a nugget term has a straightforward interpretation, as opposed to other alternatives such as pseudoinversion. In our experiments on noiseless regression we fix $\sigma^2 = 10^{-6}$, as the error introduced by adding a variance of 10^{-6} to the observations is negligible.

Similarly, for the virtual observations used in this paper we make use of the noise parameter σ_v to avoid ill-conditioning of the matrix B_1 defined in Lemma 1. B_1 is the covariance matrix of the transformed GP, $\widetilde{C}|Y$, and B_1^{-1} together with $(K_{X,X} + \sigma^2 I_N)^{-1}$ are needed for all the posterior computations that involve constraints. The virtual noise parameter σ_v has a similar interpretation as σ, but where the artificial added noise acts on observations of the transformed process. Here $\sigma_v = 0$ means that the constraints are enforced with probability 1, $\sigma_v > 0$ implies that the constraints are enforced in a soft way, and $\sigma_v \to \infty$ provides no constraint at all. In the numerical examples presented in this

paper, a fixed value $\sigma_v^2 = 10^{-6}$ has been used to approximate hard constraints with an error we find negligible.

As for computational complexity, we may start by first looking at the operations involved in computing the posterior predictive distribution at M inputs $\mathbf{x}_1^*, \ldots, \mathbf{x}_M^*$ (including covariances), using Lemma 2. We first make note of the operations needed in the unconstrained case, i.e. standard GP regression with Gaussian noise, for comparison. If there are $N \geq M$ observations in the training set, then the complexity is dominated by the Cholesky factorization $L = Chol(K_{X,X} + \sigma^2 I_N)$, which require an order of N^3 operations and N^2 in memory. The Cholesky factor may be stored for subsequent predictions. Then, to compute the posterior predictive distribution at M new inputs, the number of operations needed is dominated by matrix multiplication and solving triangular systems, of orders NM^2 and N^2M. When a number N_v of virtual observation locations are included, we are essentially dealing with the same computations as the standard GP regression, but with $N + N_v$ number of observations. I.e. the computations involved are of order $(N + N_v)^3$ in time and $(N + N_v)^2$ in memory. The order of operations in Lemma 2 was chosen such that the Cholesky factor L that only depends on the training data can be reused. For a new set X^v of size N_v, the computations needed for prediction at M new locations X^* will only require the Cholesky factorization $L_1 = Chol(B_1)$ of order N_v^3. When both L and L_1 are stored, the remaining number of operations will be of order N^2M or N_v^2M for solving triangular systems, and NM^2, N_vM^2 or NMN_v for matrix multiplications.

In order to sample from the posterior using Algorithm 3, some additional steps are required. After the computations of Lemma 2 we continue to factorize the $M \times M$ covariance matrix Σ and generate samples from the truncated Gaussian $\widetilde{C}|Y, C$. The complexity involved in sampling from this N_v-dimensional truncated Gaussian depends on the sampling method of choice, see Section 3.2. We can combine k of these samples with k samples from a standard normal $\mathcal{N}(\mathbf{0}, I_M)$ to obtain samples of the final posterior, using an order of $MN_vk + M^2k$ operations. The total procedure of generating k samples at $M \leq N$ new inputs is therefore dominated by matrix operations of order $(N + N_v)^3$, MN_vk and M^2k, together with the complexity involved with sampling from a N_v-dimensional truncated Gaussian. For subsequent prediction it is convenient to here also reuse the samples generated from the truncated Gaussian, together with results that only involve X and X^v. This means storing matrices of size $N_v \times k$, $N \times N$ and $N_v \times N_v$. The remaining computations are then dominated by operations of order N^2M, N_v^2M, NM^2, N_vM^2, NMN_v, MN_vk, and M^2k. In the algorithms used to find virtual observation locations, Algorithm 5 and 7, we make sure to reuse computations that only involve the training data in each iteration of $N_v = 1, 2, \ldots$. This means that in addition to the previously stated operations, we need to perform Cholesky factorization of order N_v^3 and generate samples from a N_v-dimensional truncated Gaussian. This is initially very cheap, but becomes the main numerical challenge when N_v grows large. As the purpose of these algorithms is to find a small set X^v, that also avoids sampling issues due to serial correlation, we found it useful to output the minimal constraint probability p^* found in each iteration to reveal if the stopping criterion used (in terms of p_{target} or a maximum number of iterations) was unrealistic in practice.

4. Gaussian Process Modelling with Boundedness and Monotonicity Constraints

In this section we present some examples related to function estimation where we assume that the function and some of its partial derivatives are bounded. This is the scenario considered in the literature on shape-constrained GPs, and alternative approaches to GPs under linear constraints are usually presented in this setting. We start by a brief discussion on related work, followed by some numerical experiments using boundedness and monotonicity constraints. The numerical experiments were performed using the Python implementation available at `https://github.com/cagrell/gp_constr`.

4.1 Related Work

We give a brief overview of some alternative and related approaches to constrained GPs. For the approaches that rely on imposing constraints at a finite set of virtual observation locations, we recall that the constraint probability can be used in the search for a suitable set of virtual observation locations. The constraint probability is the probability that the constraint holds at an arbitrary input $\mathbf{x}$, $p_c(\mathbf{x})$ given in (11). Some key characteristics of the approaches that make use of virtual observations are summarized in Table 1.

The related work most similar to the approach presented in this paper is that of Wang and Berger (2016) and Da Veiga and Marrel (2012, 2015). Wang and Berger (2016) make use of a similar sampling scheme for noiseless GP regression applied to computer code emulation. A Gibbs sampling procedure is used for inference and to estimate the constraint probability $p_c(\mathbf{x})$ in the search for virtual observation locations. The approach of Da Veiga and Marrel (2012, 2015) is based on computation of the posterior mean and covariance of the constrained GP, using the equations that are also restated in this paper in Corollary 6. They make use of a Genz approximation for inference (Genz, 1992, 1997), and also introduce a crude but faster correlation-free approximation that can be used in the search for virtual observation locations. The approach of Da Veiga and Marrel (2012, 2015) is discussed further in the numerical experiments below, where we illustrate the idea in Example 1 and in Example 2 study an approximation of the posterior constrained GP using the constrained moments with a Gaussian distribution assumption. A major component in (Da Veiga and Marrel, 2012, 2015), (Wang and Berger, 2016) and this paper is thus computation involving the truncated multivariate Gaussian. Besides the choice of method for sampling from this distribution, the main difference with our approach is that we leverage Cholesky factorizations and noisy virtual observations for numerical stability.

A different approach that also make use of virtual observations is that of Riihimki and Vehtari (2010), where a *probit* likelihood is used to represent interval observations of the derivative process to impose monotonicity. They then make use of Expectation Propagation (EP) to approximate the posterior with a multivariate Gaussian. As pointed out by Golchi et al. (2015), the Gaussian assumption is questionable if the constraint (in this case monotonicity) does not hold with high probability a priori. Golchi et al. (2015) proceeds to develop a fully Bayesian procedure for application to computer experiments by the use of Sequentially Constrained Monte Carlo Sampling (SCMC). A challenge with this approach however is that finding a suitable set of virtual observation locations is difficult. Our experience, in agreement with (Wang and Berger, 2016; Da Veiga and Marrel, 2012,

	Virtual obs. likelihood	**Inference strategy**	**Strategy for finding X^v**
Agrell (2019)	Indicator + noise	Sampling (Minimax tilting)	Based on estimating $p_c(\mathbf{x})$ from samples
Wang and Berger (2016)	Indicator	Sampling (Gibbs)	Based on estimating $p_c(\mathbf{x})$ from samples
Da Veiga and Marrel (2012, 2015)	Indicator	Moment approximation (Genz)	Based on approximating $p_c(\mathbf{x})$ assuming Gaussian posterior distribution
Riihimki and Vehtari (2010)	Probit	Expectaion Propagation	Based on approximating $p_c(\mathbf{x})$ assuming Gaussian posterior distribution
Golchi et al. (2015)	Probit	SCMC	NA

Table 1: Summary of alternative approaches that make use of virtual observations. The table compares the likelihood used for virtual observations, the method used for inference and to determine the set of virtual observation locations X^v.

2015; Riihimki and Vehtari, 2010), is that for practical applications in more than a few dimensions, such a strategy is essential to avoid numerical issues related to high serial correlation, and also to reduce the number of virtual observation locations needed. It is also worth noting that a strategy that decouples computation involving training data and virtual observation locations from inference at new locations is beneficial. For the approaches discussed herein that rely on sampling/approximation related to the truncated multivariate Gaussian, the samples/approximations can be stored and reused as discussed in Section 3.8.

There are also some approaches to constrained GPs that are not based on the idea of using virtual observations. An interesting approach by Maatouk and Bay (2017), that is also followed up by López-Lopera et al. (2018), is based on modelling a conditional process where the constraints hold in the entire domain. They achieve this through finite-dimensional approximations of the GP that converge uniformly pathwise. With this approach, sampling from a truncated multivariate Gaussian is also needed for inference, in order to estimate the coefficients of the finite-dimensional approximation that arise from discretization of the input space. The authors give examples in 1D and 2D, but note that due to the structure of the approximation, the approach will be time consuming for practical applications in higher dimensions. There are also other approaches that consider special types of shape constraints, but where generalization seems difficult. See for instance (Abrahamsen and Benth, 2001; Yoo and Kyriakidis, 2006; Michalak, 2008; Kleijnen and Beers, 2013; Lin and Dunson, 2014; Lenk and Choi, 2017).

4.2 Numerical Experiments

In this section we will make us of the following constraints:

- $a_0(\mathbf{x}) \leq f(\mathbf{x}) \leq b_0(\mathbf{x})$
- $a_i(\mathbf{x}) \leq \partial f/\partial x_i(\mathbf{x}) \leq b_i(\mathbf{x})$

for all $\mathbf{x}$ in some bounded subset of $\mathbb{R}^{n_x}$, and $i \in \mathcal{I} \subset \{1, \ldots, n_x\}$. Without loss of generality we assume that the constrains on partial derivatives are with respect to the first k components of $\mathbf{x}$, i.e. $\mathcal{I} = \{1, \ldots, k\}$ for some $k \leq n_x$.

As the prior GP we will assume a constant mean $\mu = 0$ and make use of either the RBF or Matérn 5/2 covariance function. These are stationary kernels of the form

$$K(\mathbf{x}, \mathbf{x}') = \sigma_K^2 k(r), \; r = \sqrt{\sum_{i=1}^{n_x} \left(\frac{x_i - x_i'}{l_i} \right)^2}, \tag{16}$$

with variance parameter σ_K^2 and length scale parameters l_i for $i = 1, \ldots, n_x$. The radial basis function (RBF), also called squared exponential kernel, and the Matérn 5/2 kernel are defined through the function $k(r)$ as

$$k_{\text{RBF}}(r) = e^{-\frac{1}{2}r^2} \text{ and } k_{\text{Matérn } 5/2}(r) = (1 + \sqrt{5}r + \frac{5}{3}r^2)e^{-\sqrt{5}r}.$$

In general, the kernel hyperparameters σ_K^2 and l_i are optimized together with the noise variance σ through MLE. In the examples that consider noiseless observations, the noise variance is not estimated, but set to a small fixed value as discussed in Section 3.8. With the above choice of covariance function, existence of the transformed GP is ensured. In fact, the resulting process is infinitely differentiable using the RBF kernel (see Adler, 1981, Theorem 2.2.2) and twice differentiable with the Matérn 5/2. These prior GP alternatives were chosen as they are the most commonly used in the literature, and thus a good starting point for illustrating the effect of including linear constraints. We note that although it is not in general possible to design mean and covariance functions that produce GPs that satisfy the constraints considered in this paper, one could certainly ease numerical computations by selecting a GP prior based on the constraint probability $p(C|Y, \theta)$ in (7), and for instance make us of a mean function that is known to satisfy the constraint.

If we let $\mathcal{F}^0 f = f$, $\mathcal{F}^i f = \partial f/\partial x_i$, and $X^{v,i}$ be the set of S_i virtual observations corresponding to the i-th operator $\mathcal{F}^i$, then we can make use of the formulation in Section 3.5 and equations from Appendix C to obtain

$$\mathcal{L}\mu^v = [\mu \mathbf{1}_{S_0}, \mathbf{0}_{S_{[1,k]}}]^T,$$

where $\mathbf{1}_{S_1}$ is the vector $[1, \ldots, 1]^T$ of length S_1 and $\mathbf{0}_{S_{[1,k]}}$ is the vector $[0, \ldots, 0]^T$ of length $S_{[1,k]} = -S_0 + \sum S_i$. Furthermore,

$$\begin{aligned}
K_{X,X^v}\mathcal{L}^T &= \left[K_{X,X^{v,0}}, (K^{1,0}_{X^{v,1},X})^T, \ldots, (K^{k,0}_{X^{v,k},X})^T \right], \\
K_{X^*,X^v}\mathcal{L}^T &= \left[K_{X^*,X^{v,0}}, (K^{1,0}_{X^{v,1},X^*})^T, \ldots, (K^{k,0}_{X^{v,k},X^*})^T \right], \\
\mathcal{L}K_{X^v,X^v}\mathcal{L}^T &= \begin{bmatrix}
K_{X^{v,0},X^{v,0}} & (K^{1,0}_{X^{v,1},X^{v,0}})^T & \cdots & (K^{k,0}_{X^{v,k},X^{v,0}})^T \\
K^{1,0}_{X^{v,1},X^{v,0}} & K^{1,1}_{X^{v,1},X^{v,1}} & \cdots & K^{1,k}_{X^{v,1},X^{v,k}} \\
\vdots & \vdots & \ddots & \vdots \\
K^{k,0}_{X^{v,k},X^{v,0}} & K^{k,1}_{X^{v,k},X^{v,1}} & \cdots & K^{k,k}_{X^{v,k},X^{v,k}}
\end{bmatrix},
\end{aligned}$$

where we have used the notation

$$K^{i,0}(\mathbf{x},\mathbf{x}') = \frac{\partial}{\partial x_i}K(\mathbf{x},\mathbf{x}') \text{ and } K^{i,j}(\mathbf{x},\mathbf{x}') = \frac{\partial^2}{\partial x_i \partial x_i'}K(\mathbf{x},\mathbf{x}').$$

The use of constraints related to boundedness and monotonicity is illustrated using three examples of GP regression. Example 1 considers a function $\mathrm{f}:\mathbb{R}\to\mathbb{R}$ subjected to boundedness and monotonicity constraints. In Example 2 a function $\mathrm{f}:\mathbb{R}^4\to\mathbb{R}$ is estimated under the assumption that information on whether the function is monotone increasing or decreasing as a function of the first two inputs is known, i.e. $\mathrm{sgn}(\partial \mathrm{f}/\partial x_1)$ and $\mathrm{sgn}(\partial \mathrm{f}/\partial x_2)$ are known. In Example 3 we illustrate how monotonicity constraints in multiple dimensions can be used in prediction of pressure capacity of pipelines.

4.2.1 **Example 1**: Illustration of Boundedness and Monotonicity in 1D

As a simple illustration of imposing constraints in GP regression, we first consider the function $\mathrm{f}:\mathbb{R}\to\mathbb{R}$ given by $\mathrm{f}(x) = \frac{1}{3}[\tan^{-1}(20x-10) - \tan^{-1}(-10)]$. We assume that the function value is known at 7 input locations given by $x_i = 0.1 + 1/(i+1)$ for $i = 1,\ldots,7$. First, we assume that the observations are noiseless, i.e. $\mathrm{f}(x_i)$ is observed for each x_i. Estimating the function that interpolates at these observations is commonly referred to as *emulation*, which is relevant when dealing with data from computer experiments. Our function $\mathrm{f}(x)$ is both bounded and increasing on all of $\mathbb{R}$. In this example we will constrain the GP to satisfy the conditions that for $x\in[0,1]$, we have that $\mathrm{d}\mathrm{f}/\mathrm{d}x \geq 0$ and $a(x) \leq \mathrm{f}(x) \leq b(x)$ for $a(x) = 0$ and $b(x) = \frac{1}{3}\ln(30x+1) + 0.1$. The function is shown in Figure 1 together with the bounds and the 7 observations.

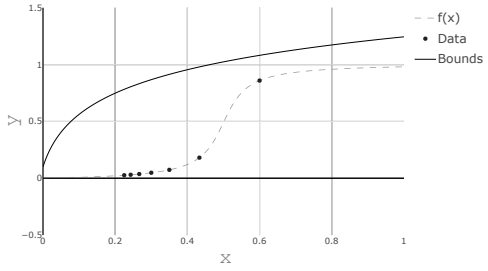


Figure 1: Function to emulate in Example 1

We select an RBF kernel (16) with parameters $\sigma_K = 0.5$ (variance) and $l = 0.1$ (length scale). To represent noiseless observations we set $\sigma^2 = 10^{-6}$, where σ^2 is the noise variance in the Gaussian likelihood. The assumed noise on virtual observations will also be set to 10^{-6}. To illustrate the effect of adding constraints we show the constrained GP using only boundedness constraint, only monotonicity constraint and finally when both constraints are imposed simultaneously. Figure 2 shows the resulting GPs. Algorithm 7 was used with a target probability $p_{target} = 0.99$ to determine the virtual observation locations that are indicated in the figures, and the posterior mode was computed by maximizing a Gaussian

kernel density estimator over the samples generated in Algorithm 3. For both constraints, 17 locations was needed for monotonicity and only 3 locations was needed to impose boundedness when the virtual locations for both constraints where optimized simultaneously. This is reasonable, as requiring $f(0) > 0$ is sufficient to ensure $f(x) > 0$ for $x \geq 0$ when f is increasing, and similarly requiring $f(x^v) < b(x^v)$ for some few points $x^v \in [0.6, 1]$ should suffice. But note that Algorithm 7 finds the virtual observation locations for both constraints simultaneously. Here $x^v = 0$ for boundedness was first identified, followed by some few points for monotonicity, followed by a new point x^v for boundedness etcetera.

For illustration purposes none of the hyperparameters of the GP were optimized. Moreover, for data sets such as the one in this example using plug-in estimates obtained from MLE generally not appropriate due to overfitting. Maximizing the marginal likelihood for the unconstrained GP gives a very poor model upon visual inspection ($\sigma_K = 0.86, l = 0.26$). However, it was observed that the estimated parameters for the constrained model (using Eq. (8)) gives estimates closer to the selected prior which seems more reasonable ($\sigma_K = 0.42, l = 0.17$), and hence the inclusion of the constraint probability, $p(C|Y, \theta)$, in the likelihood seems to improve the estimates also for the unconstrained GP.

We may also assume that the observations come with Gaussian white noise, which in terms of numerical stability is much less challenging than interpolation. Figure 3 shows the resulting GPs fitted to 50 observations. The observations were generated by sampling $x_i \in [0.1, 0.8]$ uniformly, and y_i from f$(x_i)+\varepsilon_i$ where ε_i are i.i.d. zero mean Gaussian with variance $\sigma^2 = 0.04$. Both GPs were optimized using plug-in estimates of hyperparameters (σ_K, l, σ^2) given by maximizing the marginal likelihood. These are $(\sigma_K = 0.34, l = 0.32, \sigma^2 = 0.053)$ for the constrained case and $(\sigma_K = 0.34, l = 0.23, \sigma^2 = 0.040)$ for the unconstrained case. We observe that the estimated noise variance is larger in the constrained model than the unconstrained where this estimate is exact.

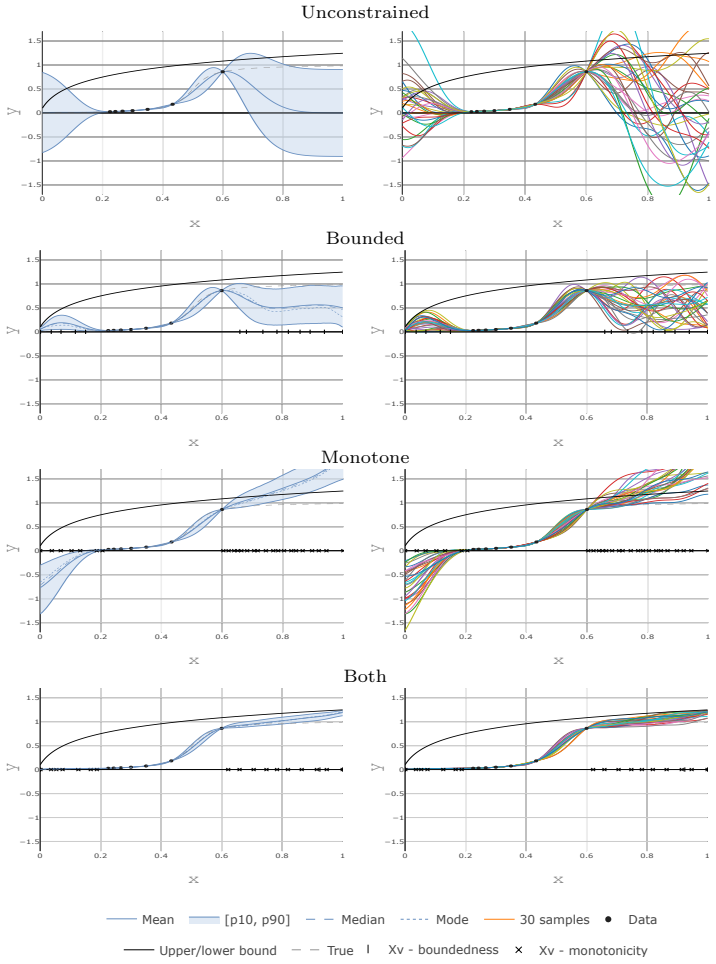


Figure 2: The GP with parameters $\sigma_K = 0.5$ (variance) and $l = 0.1$ (length scale) used in Example 1. The virtual observation locations are indicated by markers on the x-axis.

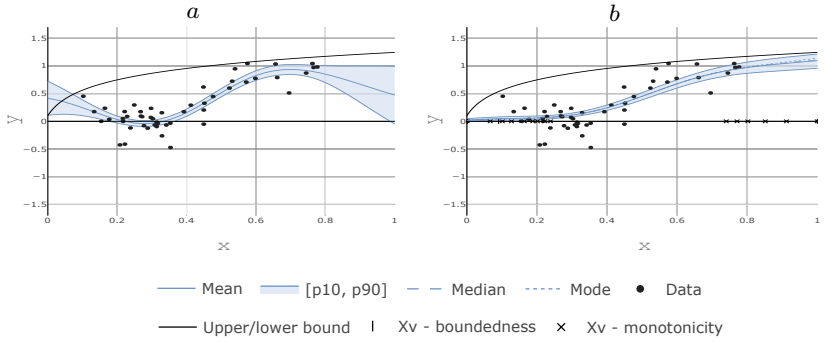


Figure 3: Unconstrained (a) and constrained (b) GPs fitted to 50 observations with Gaussian noise. The predictive distributions are shown, i.e. the distribution of $f(x)$ where $y = f(x) + \varepsilon$.

Da Veiga and Marrel (2015) propose to use estimates of the posterior mean and variance of $\mathcal{L}f(\mathbf{x})|Y, C$ to estimate the constraint probability $p_c(\mathbf{x})$ assuming a Gaussian distribution. They also introduce the faster correlation-free approximation, where the parameters are estimated under the assumption that observations of $\mathcal{L}f(\mathbf{x})|Y$ at different input locations $\mathbf{x}$ are independent (see Section 3.7). In Figure 4 we plot estimates of $p_{c,i}(\mathbf{x})$, for the boundedness and monotonicity constraint individually, using the approach in this paper (13) and the two moment based approximations. The plots were generated first after a total of 5 and then 10 virtual observations locations had been included in the model with both constraints. As we are mainly interested in finding $\mathbf{x}^* = \arg\min p_{c,i}(\mathbf{x})$, Figure 4 indicates that the moment based approximations are appropriate initially. However, as more virtual observation locations are included, the correlation-free assumption becomes questionable. But it could still serve as a useful starting point, and in a strategy based on checking the approximation error from time to time, it should still be possible to take advantage of the computational savings offered by the correlation-free approximation.

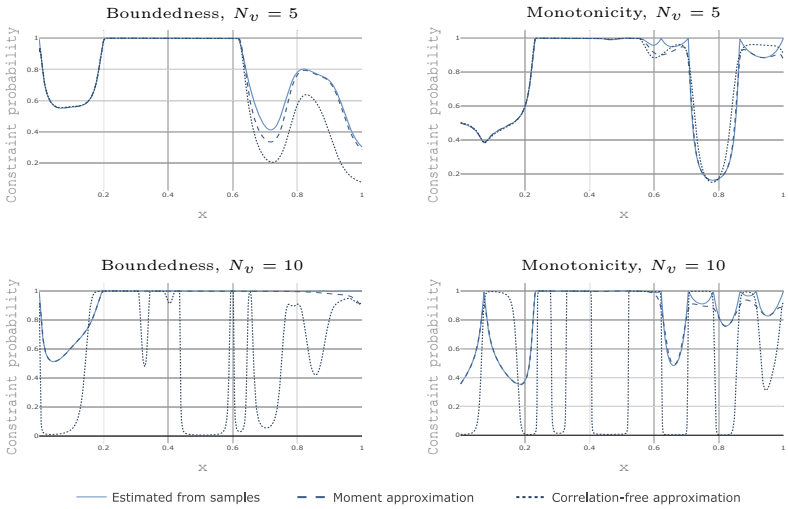


Figure 4: Constraint probability $p_c(\mathbf{x})$ computed using the estimate (13) together with the moment based approximations from Da Veiga and Marrel (2015). The constraint probability is shown for monotonicity and boundedness, where N_v is the total number of virtual observation locations used in the model.

4.2.2 **Example 2**: 4D Robot Arm Function

In this example we consider emulation of a function $\mathrm{f} : \mathbb{R}^4 \to \mathbb{R}$, where we assume that the sign of the first two partial derivatives, $\mathrm{sgn}(\partial \mathrm{f}/\partial x_1)$ and $\mathrm{sgn}(\partial \mathrm{f}/\partial x_2)$, are known. The function to emulate is

$$\mathrm{f}(\mathbf{x}) = \sum_{i=1}^{m} L_i \cos\left(\sum_{j=1}^{i} \tau_j\right),$$

for $m = 2$, and $\mathbf{x} = [L_1, L_2, \tau_1, \tau_2]$. The function is inspired by the robot arm function often used to test function estimation (An and Owen, 2001). Here $\mathrm{f}(\mathbf{x})$ is the y-coordinate of a two dimensional robot arm with m line segments of length $L_i \in [0, 1]$, positioned at angle $\tau_i \in [0, 2\pi]$ with respect to the horizontal axis. The constraints on the first two partial derivatives thus implies that it is known whether or not the arm will move further away from the x-axis, as a function of the arm lengths, L_1 and L_2, for any combination of τ_1 and τ_2.

In this experiment we first fit an unconstrained GP using 40 observations taken from a Latin hypercube sample over the input space $[0, 1]^2 \times [0, 2\pi]^2$. A Matérn 5/2 covariance function is used with plug-in MLE hyperparameters. Then, a total of 80 virtual observation locations are found using the procedure in Algorithm 7, where we search over a finite candi-

date set of 1000 locations in the minimization of the constraint probability. We repeat this procedure 100 times and report performance using the predictivity coefficient Q^2, predictive variance adequation (PVA) and the average width of 95% confidence intervals (AWoCI).

Given a set of tests $y_1, \ldots, y_{n_{test}}$ and predictions $\hat{y}_1, \ldots, \hat{y}_{n_{test}}$, Q^2 is defined as

$$Q^2 = 1 - \sum_{i=1}^{n_{test}} (\hat{y}_i - y_i)^2 / \sum_{i=1}^{n_{test}} (\bar{y} - y_i)^2,$$

where $\bar{y}$ is the mean of $y_1, \ldots, y_{n_{test}}$. In our experiments the predictions $\hat{y}_i$ are given by the posterior mean of the GP. The PVA criterion is defined as

$$\text{PVA} = \left| \log \left(\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \frac{(\hat{y}_i - y_i)^2}{\hat{\sigma}_i^2} \right) \right|,$$

where $\hat{\sigma}_i^2$ is the predictive variance. This criterion evaluates the quality of the predictive variances and to what extent confidence intervals are reliable. The smaller the PVA is, the better (Bachoc, 2013). In addition to this criterion, it is also useful to evaluate the size of confidence intervals. For this we compute the average width of 95% confidence intervals

$$\text{AWoCI} = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (p_{0.975}^{(i)} - p_{0.025}^{(i)}),$$

where $p_{0.975}^{(i)}$ and $p_{0.025}^{(i)}$ are the predicted 97.5% and 2.5% percentiles.

The result of 100 predictions for one single experiment is shown in Figure 5. As expected, the estimated prediction uncertainty is reduced significantly using the constrained model, and single predictions given by the posterior mean are also improved. In Table 2 we summarize the results from running 100 of these experiments. In each experiment, Q^2, PVA and AWoCI was computed from prediction at 1000 locations sampled uniformly in the domain. We also report the probability that the constraint holds in the unconstrained GP, $p(C|Y)$ given in (7), and the CPU time in seconds used to generate 10^4 samples from the posterior on an Intel® Core™ i5-7300U 2.6GHz CPU. For comparison, we also include predictions from moment-based approximations using the approach of Da Veiga and Marrel (2012, 2015). We study in particular their approach for finding the set of virtual observation locations, as discussed in Section 3.7 and illustrated in the previous example. In total, the following alternatives are considered:

1. **Unconstrained**: The initial GP without constraints.

2. **Constrained**: The constrained GP using the approach presented in this paper.

3. **Moment approx. 1**: Using the sampling scheme of this paper for inference, but where the moment based approximation is used in the search for virtual observation locations.

4. **Moment approx. 2**: Using moment approximation for both inference and searching for virtual observation locations. This is one of the procedures from Da Veiga and Marrel (2012, 2015).

5. **Correlation-free approx.**: Same as **Moment approx. 1** but where the correlation-free approximation is used in the search for virtual observation locations.

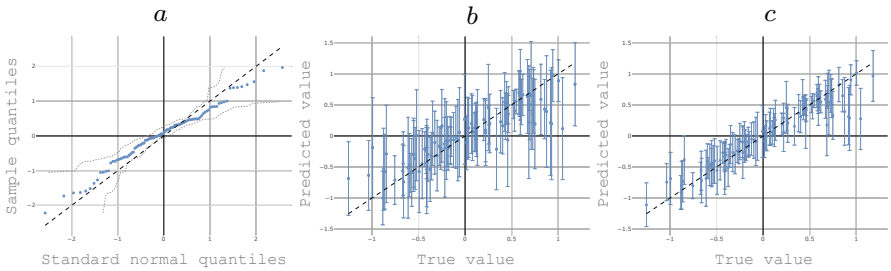


Figure 5: Figure a shows a qq-plot with 95% confidence band of 100 normalized residuals $(y_i - \mu_i)/(\sigma_i)$, where μ_i and σ_i^2 are the mean and variance of the predictive distribution of the unconstrained GP. In Figure b, predictions vs the true function value is shown together with a $[0.025, 0.975]$ (95%) percentile interval for the unconstrained GP. The same type of figure is shown in c for the constrained GP.

In Table 2 we see that the use of constraints is beneficial in terms of both a higher Q^2 (better predictive performance) and a smaller PVA (higher quality of predictive variances). With the exception of 'Moment approx. 2', the inclusion of constraints provides significant uncertainty reduction as the width of 95% confidence intervals (AWoCI) are reduce by almost a factor of 2 on average. A box plot showing AWoCI from the 100 experiments is also shown in Figure 6. We see that the different approaches for estimating the constraint probability, $p_c(\mathbf{x})$, in the search for virtual observation locations work equally well. The Gaussian assumption on the posterior $\mathbf{f}^*|Y, C$ on the other hand is not optimal, as it tends to overestimate the uncertainty in this example.

	$p(C\|Y)$	T_s	PVA	Q^2	AWoCI
Unconstrained			3.03	0.7558	0.99
Constrained	4.1E-34	24.8	2.85	0.8842	0.54
Moment approx. 1	2.4E-36	25.2	2.84	0.8844	0.54
Moment approx. 2	2.4E-36	25.2	2.84	0.8844	0.83
correlation-free approx.	8.6E-37	21.1	2.91	0.8775	0.55

Table 2: Average values from 100 experiments of the robot arm function. T_s is the CPU time in seconds used to generate 10^4 samples.

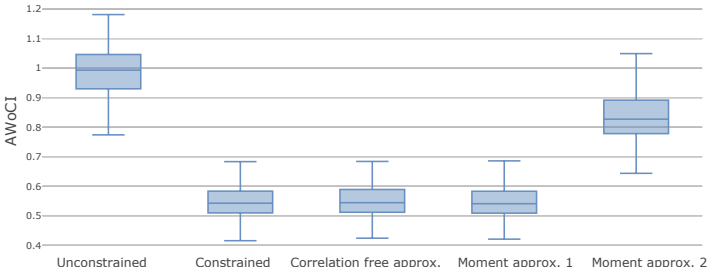


Figure 6: Average width of confidence intervals (AWoCI) from 100 experiments of the robot arm function.

4.2.3 **Example 3**: Pipeline Pressure Capacity

In this example we consider a model for predicting the pressure capacity of a steel pipeline with defects due to corrosion. As corrosion is one of the major threats to the integrity of offshore pipelines, experiments are carried out to understand how metal loss due to corrosion affects a pipeline's capacity with respect to internal pressure (Sigurdsson et al., 1999; Amaya et al., 2019). These include full scale burst tests and numerical simulation through Finite Element Analysis (FEA). Results from this type of experiments serve as the basis for current methodologies used in the industry for practical assessment of failure probabilities related to pipeline corrosion, such as ASME B31G or DNVGL-RP-F101. We consider experiments related to a single rectangular shaped defect, which is essential to these methodologies.

To simulate synthetic experiments of the burst capacity of a pipeline with a rectangular defect, we will use the simplified capacity equation given in in (RP-F101 DNV GL, 2017). The maximum differential pressure (capacity in MPa) the pipeline can withstand without

bursting is in the simplified equation given as

$$P_{cap}(\sigma_u, D, t, d, l) = 1.05\frac{2t\sigma_u}{D-t}\frac{1-d/t}{1-\frac{d/t}{Q}}, \quad Q = \sqrt{1+0.31\frac{l^2}{Dt}},$$

where $\sigma_u \in [450, 550]$ (MPa) is the ultimate tensile strength of the material, $D \in [10t, 50t]$ (mm) and $t \in [5, 30]$ (mm) are the outer diameter and wall thickness of the pipeline, and $d \in [0, t]$ (mm) and $l \in [0, 1000]$ (mm) are the depth and length of the rectangular defect.

From the physical phenomenon under consideration, we know that the capacity of the pipeline will decrease if the size of the defect were to increase. Similarly, we know that the pipeline capacity increases with a higher material strength or wall thickness, and decreases as a function of the diameter, all else kept equal. In the form of partial derivatives we can express this information as: $\frac{\partial P_{cap}}{\partial d} < 0$, $\frac{\partial P_{cap}}{\partial l} < 0$, $\frac{\partial P_{cap}}{\partial \sigma_u} > 0$, $\frac{\partial P_{cap}}{\partial t} > 0$ and $\frac{\partial P_{cap}}{\partial D} < 0$.

For convenience we will transform the input variables to the unit hypercube. Let $\mathbf{x}$ denote the transformed input vector $\mathbf{x} = [x_1, \ldots, x_5]$, where $x_1 = (\sigma_u - 450)/(550 - 450)$, $x_2 = (D/t - 10)/(50 - 10)$, $x_3 = (t - 5)/(30 - 5)$, $x_4 = d/t$ and $x_5 = l/1000$. We will make use of the function

$$f(\mathbf{x}) = P_{cap}(\mathbf{x}) \text{ for } \mathbf{x} \in [0, 1]^5,$$

and assume that the burst capacity observed in an experiment is $f(\mathbf{x}) + \varepsilon$, where ε is a zero mean Normal random variable with variance $\sigma^2 = 4$. The constraints on the partial derivatives after the transformation becomes: $\frac{\partial f}{\partial x_1} > 0$, $\frac{\partial f}{\partial x_2} < 0$, $\frac{\partial f}{\partial x_3} > 0$, $\frac{\partial f}{\partial x_4} < 0$ and $\frac{\partial f}{\partial x_5} < 0$ for $\mathbf{x} \in [0, 1]^5$.

In this example we thus have five constraints available, represented by bounds on the partial derivative of $f(\mathbf{x})$ w.r.t. x_i for $i = 1, \ldots, 5$. Besides studying the effect of including all five constraints, we will test some different alternatives using a smaller number of constraints, and also lower input dimensions. To simulate a lower dimensional version of the capacity equation, we can consider only the fist n_x input variables and keep the remaining variables fixed. We consider $n_x = 3, 4$ and 5 where we fix $x_i = 0.5$ for all $i > n_x$. For each of these scenarios we will consider n_x and $n_x - 1$ number of constraints. We let n_c denote the number of constraints, where using n_c constraints means that the bound on $\partial f/\partial x_i$ is included for $i = 1, \ldots, n_c$.

In each experiment we start by generating a training set of $N = 5n_x$ or $N = 10n_x$ LHS samples from $[0, 1]^{n_x}$. As in the previous example in Section 4.2.2, we fit a zero mean GP using a Matérn 5/2 covariance function and plug-in hyperparameters by MLE. We search over a candidate set consisting of 2500 uniform samples from $[0, 1]^{n_x}$ iteratively to update the set of virtual observation locations, until the constraint probability at all locations in the candidate set, and for each constraint, is at least 0.7. To check whether this is a reasonable stopping criterion we finish by minimizing the constraint probability for each constraint, using the differential evolution (Storn and Price, 1997) global optimization algorithm available in (SciPy Jones et al., 2001–).

Table 3 shows the results for different combinations of input dimensionality n_x, number of constraints n_c and number of training samples N, where the results in each row is computed from 100 experiments. As in the previous example we report $p(C|Y)$, PVA, Q^2 and AWoCI, and the CPU time spent generating samples for prediction (T_s). We also report the average CPU time used in the search for a new virtual observation location and

n_x	n_c	N	N_v	T_v	$p(C\|Y)$	$p_{c,\min}$	T_s	PVA	Q^2	AWoCI
3	2	15	3.6	0.6	2.6E-01	0.79	0.05	0.94 (0.89)	0.95 (0.95)	3.9 (6.2)
3	2	30	3.5	0.6	2.5E-01	0.78	0.04	0.89 (0.87)	0.97 (0.97)	3.0 (4.8)
3	3	15	5.8	0.9	1.2E-01	0.74	0.09	1.47 (1.23)	0.95 (0.95)	3.7 (6.1)
3	3	30	3.9	0.9	2.2E-01	0.76	0.04	0.79 (0.79)	0.97 (0.97)	3.1 (5.0)
4	3	20	11.8	0.9	1.5E-02	0.67	0.19	1.40 (1.29)	0.87 (0.92)	5.5 (9.4)
4	3	40	11.7	0.9	6.6E-03	0.71	0.18	0.51 (0.52)	0.97 (0.97)	4.1 (6.9)
4	4	20	13.6	1.2	6.9E-03	0.65	0.49	1.56 (1.31)	0.91 (0.91)	5.5 (9.6)
4	4	40	12.8	1.2	2.7E-03	0.69	0.19	0.50 (0.48)	0.97 (0.97)	4.0 (6.7)
5	4	25	14.8	1.2	6.3E-03	0.66	0.22	1.03 (1.08)	0.85 (0.83)	8.3 (14.3)
5	4	50	17.4	1.2	1.2E-03	0.66	0.26	0.73 (0.78)	0.90 (0.90)	6.8 (11.5)
5	5	25	15.5	1.5	3.1E-03	0.65	0.24	1.12 (1.10)	0.82 (0.81)	8.4 (14.4)
5	5	50	20.2	1.6	1.1E-03	0.61	0.35	0.67 (0.77)	0.90 (0.90)	6.5 (11.3)

Table 3: Average values from 100 experiments with input dimensionality n_x, number of constraints n_c and number of training samples N. Values in parenthesis correspond to the unconstrained model. Here $p_{c,\min}$ is the minimum of the constraint probability for any constraint over the entire domain after a total of N_v virtual observation locations have been included. T_v is the average CPU time in seconds used to find each of the N_v points using 10^3 samples, and T_s is the CPU time in seconds used to generate 10^4 samples of the final model for prediction.

the minimum constraint probability, $p_{c,\min} = \min_{i=1,\dots n_c} \min_{\mathbf{x}\in[0,1]^{n_x}} \hat{p}_{c,i}(\mathbf{x})$ (13), computed with differential evolution. Here we make use of 10^3 samples to compute the estimate $\hat{p}_{c,i}(\mathbf{x})$, whereas 10^4 samples are used for the final prediction.

From Table 3 we first notice that the number of virtual observation locations (N_v) determined by the searching algorithm is fairly low. One might interpret this as an indication that the unconstrained GP produces samples that are likely to agree with the monotonicity constraints, except for at a few locations. As a result, computation that involve sampling from the truncated multivariate Gaussian is efficient. Still, we see that inclusion of the constraints has an effect on uncertainty estimates as the AWoCI is reduced by a factor of around 1.6 in each experiment, whereas PVA and Q^2 are fairly similar for the unconstrained and constrained model overall. We also notice that the smallest constraint probability found in the domain using a global optimization technique is reduced when the number of constraints or dimensionality is increased. This is expected, as we only considered a finite candidate set and not the entire domain when searching for the location minimizing the constraint probability. Hence, if we really want to achieve a minimal constraint probability larger than 0.7 in 5 dimensions, more than 2500 samples in the candidate set would be needed with this strategy, or a global optimizer could be used to identify the remaining virtual observation locations needed.

For the application considered in this example, where uncertainty in the prediction is key to risk assessment, we argue that the effect the constraints have on uncertainty estimates makes the inclusion of constraints worthwhile. Modern engineering methodologies that

make use of capacity predictions as the one illustrated in this example are usually derived in the context of Structural Reliability Analysis (SRA), where the capacity is combined with a probabilistic representation of load (in this case differential pressure) to estimate the probability of failure (Madsen et al., 2006).

Alternative methods based on conservative estimates to ensure sufficient safety margin between load and capacity are also common. For the application considered herein, this would typically mean using a lower percentile instead of the posterior mean in order to represent a conservative capacity. The inclusion of constraints can therefore help to avoid unnecessary conservatism due to unphysical scenarios, that are not realistic but have positive probability in the unconstrained model.

Finally, we note that the constraints used in this example are not from differentiating the equation used as stand-in for experiments, but from knowledge related to the underlying physical phenomenon. The constraints therefore remain applicable, were the experiments to come from physical full-scale tests. This naturally also holds in applications to computer code emulation, where we would set the noise term ε to zero in this example if we were to assume that the capacity experiments came from a numerical (FEA) simulation. With results from this type of numerical simulation, a noise parameter is usually added to the simulation output as well, to represent model uncertainty as the numerical simulation is not a perfect representation of the real physical phenomenon. Very often the model uncertainty is represented by a univariate Gaussian. An interesting alternative here is to instead account for the model uncertainty as observational noise in the GP, where the use of constraints may help to obtain a more realistic model uncertainty as well.

5. Discussion

The model presented in this paper provides a consistent approach to GP regression under multiple linear constraints. The computational framework used is based on a sampling scheme which is exact in the limit. However, sampling strategies like the one in this paper can be too numerically demanding as opposed to approximation methods such as Laplace approximations, variational Bayesian inference, expectation propagation etcetera. The choice of using a sampling-based approach came from the author's intended use, which relates to machine learning for high-risk and safety-critical engineering applications (Agrell et al., 2018). For these applications, a proper treatment of uncertainty with respect to risks and the overall reliability of the system under consideration is essential. Making predictions based on past observations in this setting is challenging, as the consequence of wrong predictions may be catastrophic. In addition, critical consequences often relate to infrequent or low probability events, where relevant data is naturally scarce. However, there is usually additional knowledge available, and todays methods for assessing risk tend to rely heavily on understanding the underlying physical phenomenon. We gave an example in Section 4.2.3 considering prediction of the burst capacity of a pipeline, that may serve as a component in a larger model of system reliability. Such models are often graphical, e.g. Bayesian networks, that are derived from known causal dependencies. In this scenario it is essential that the accuracy of numerical estimation- or approximation methods can be assessed. In the case where simulation-based methods cannot be used due to computational limitations, they still serve as a useful benchmark that can help in the development and assessment of

suitable approximation-based algorithms. As for the simulation scheme in this paper, the only computational burden lies in sampling from a truncated multivariate Gaussian. As this is a fairly general problem, multiple good samplers exist for this purpose. We found the method of Botev (2017) to work particularly well for our applications, as it provides exact sampling in a relevant range of dimensions where many alternative sampling schemes fail. Based on a comparison made by López-Lopera et al. (2018), we see that the method based on Hamiltonian Monte Carlo by Pakman and Paninski (2012) may also be appropriate.

As we discuss briefly in Section 3.3, estimation of hyperparameters becomes challenging when the term $p(C|Y, \theta)$ enters the likelihood. Moreover, as our approach is based on the use of virtual observation locations, we are aware that the task of estimating or optimizing model hyperparameters in general is not well defined. This is because the likelihood depends both on the hyperparameters and the set of virtual observation locations (Eq. 8). This problem is neglected in the literature on shape-constrained GPs, where it is either assumed that the virtual observation locations are known a priori (for low input dimension selecting a space filling sufficiently dense design is unproblematic), or the hyperparameters are addressed independently of these. To our knowledge the problem of simultaneously estimating hyperparameters and virtual observation locations has not yet been addressed. A rather simplistic approach is to iterate between estimating hyperparameter and the set of virtual observation locations. However, for higher input dimensions this might be problematic altogether, in which case sparse approximations may be needed to deal with a large set of virtual observation locations. In this setting, it might be more fruitful to view the virtual observation locations as additional hyperparameters, in a model approximating the posterior corresponding to an sufficiently dense set of virtual observation locations, e.g. as in the inducing points framework for scaling GPs to large data sets (de G. Matthews et al., 2016). This is a topic of further research.

With the approach in this paper, we make use of the probability $p(C|Y)$, which is interesting in its own for investigating whether constraints such as e.g. monotonicity are likely to hold given a set of observations. Alternatively, inference on the constraint noise parameter σ_v can provide similar type of information. Ideally, we choose a small fixed value for σ_v to avoid numerical instability, as discussed in Section 3.8. But in extreme cases, with conflicting constraints or observations that contradict constraints with high probability, the model may still experience numerical issues. We argue that models that 'break' under these circumstances are preferred as it reveals that either 1) there is something wrong with the observations, or 2) there is something wrong with the constraints and hence our knowledge of the underlying phenomenon (Agrell et al., 2018). It would nevertheless be better if more principled ways of investigating such issues were available. In our experiments we observed that the conditional likelihood, $p(Y|C)$, in general is decreasing as a function of σ_v, whereas this was not the case for an invalid constraint assuming a monotonic *decreasing* function in Example 1. Hence, σ_v might provide useful information in this manner. The estimated partial constraint probabilities $\hat{p}_{c,i}(\mathbf{x})$ can also be useful for revealing such issues, for instance by monitoring the intermediate minimum values p_i^* computed in Algorithm 7 as new virtual observation locations are added.

Finally, we note that as the model presented in this paper relies on conditioning on a transformed GP with values in $\mathbb{R}^{n_c}$, it could be extended to multi-output GPs over functions

$\mathrm{f}:\mathbb{R}^{n_x}\to\mathbb{R}^{n_y}$ in a natural way. But for non-Gaussian likelihoods, or applications with large or high-dimensional data, other approximation based alternatives are needed.

Acknowledgments

This work has been supported by grant 276282 from the Norwegian Research Council and DNV GL Group Technology and Research. The research is part of an initiative on applying constraints based on phenomenological knowledge in probabilistic machine learning for high-risk applications, and the author would like to thank colleagues at DNV GL and the University of Oslo for fruitful discussions on the topic. A special thanks to Arne B. Huseby, Simen Eldevik, Andreas Hafver, and the editor and reviewers of JMLR for insightfull comments that have greatly improved the paper.

Appendix A. Proof of Lemma 1

Proof. We start by observing that $(\mathbf{f}^*,\widetilde{C},Y)$ is jointly Gaussian with mean and covariance

$$\mathbb{E}([\mathbf{f}^*,\widetilde{C},Y]^T)=[\mu^*,\mathcal{L}\mu^v,\mu]^T, \tag{17}$$

$$\operatorname{cov}([\mathbf{f}^*,\widetilde{C},Y]^T)=\begin{bmatrix} K_{X^*,X^*} & K_{X^*,X^v}\mathcal{L}^T & K_{X^*,X} \\ \mathcal{L}K_{X^v,X^*} & \mathcal{L}K_{X^v,X^v}\mathcal{L}^T+\sigma_v^2 I_{N_v} & \mathcal{L}K_{X^v,X} \\ K_{X,X^*} & K_{X,X^v}\mathcal{L}^T & K_{X,X}+\sigma^2 I_N \end{bmatrix}. \tag{18}$$

By first conditioning on Y we obtain

$$\begin{matrix}\mathbf{f}^*\\ \widetilde{C}\end{matrix}\,\middle|\, Y\sim\mathcal{N}\left(\begin{bmatrix}\mu^*+A_2(Y-\mu)\\ \mathcal{L}\mu^v+A_1(Y-\mu)\end{bmatrix},\begin{bmatrix}B_2 & B_3\\ B_3^T & B_1\end{bmatrix}\right), \tag{19}$$

for $A_1=(\mathcal{L}K_{X^v,X})(K_{X,X}+\sigma^2 I_N)^{-1}$, $A_2=K_{X^*,X}(K_{X,X}+\sigma^2 I_N)^{-1}$, $B_1=\mathcal{L}K_{X^v,X^v}\mathcal{L}^T+\sigma_v^2 I_{N_v}-A_1K_{X,X^v}\mathcal{L}^T$, $B_2=K_{X^*,X^*}-A_2K_{X,X^*}$, and $B_3=K_{X^*,X^v}\mathcal{L}^T-A_2K_{X,X^v}\mathcal{L}^T$.

Conditioning on $\widetilde{C}$ then gives

$$\mathbf{f}^*|Y,\widetilde{C}\sim\mathcal{N}\left(\mu^*+A(\widetilde{C}-\mathcal{L}\mu^v)+B(Y-\mu),\Sigma\right), \tag{20}$$

for $A=B_3B_1^{-1}$, $B=A_2-AA_1$ and $\Sigma=B_2-AB_3^T$.

Similarly, we may derive $\widetilde{C}|Y$ by observing that the joint distribution of $\widetilde{C},Y$ is given by removing the first row in (17) and the first row and column in (18). Hence,

$$\widetilde{C}|Y\sim\mathcal{N}\left(\mathcal{L}\mu^v+A_1(Y-\mu),B_1\right). \tag{21}$$

The constrained posterior of $\widetilde{C}$ is obtained by applying the constraint C to the posterior, and hence $\widetilde{C}|Y,C$ becomes a truncated Gaussian with the same mean and variance as in (21), and the bounds $a(X^v)$ and $b(X^v)$ given by C. Similarly, $\mathbf{f}^*|Y,C$ is obtained by replacing $\widetilde{C}$ in (20) with $\widetilde{C}|Y,C$. Finally, the probability $p(C|Y)$ is just the probability that $\widetilde{C}|Y$ given in (21) falls within the bounds given by C, and the unconstrained distribution remains the same as (2). ■

Appendix B. Proof of Lemma 2

Proof. The equations in Lemma 2 can be verified by simply inserting L, v_1 and v_2 and check against the expressions in Lemma 1. We show this for A_1 and B_1, and the results for the remaining matrices are proved by applying the same procedures. In order to factorize B_1, we use that B_1 is the covariance matrix of a Gaussian random variable (see Equation 21 in Appendix A), and must therefore be symmetric and positive definite.

To show that $A_1 = (L^T \setminus v_1)^T$ we use that $v_1 = L \setminus K_{X,X^v}\mathcal{L}^T \Rightarrow Lv_1 = K_{X,X^v}\mathcal{L}^T$. Hence,

$$\begin{aligned}
&A_1 = (L^T \setminus v_1)^T \\
&\Rightarrow L^T A_1^T = v_1 = L \setminus K_{X,X^v}\mathcal{L}^T \\
&\Rightarrow LL^T A_1^T = K_{X,X^v}\mathcal{L}^T \\
&\Rightarrow A_1 = ((LL^T)^{-1}K_{X,X^v}\mathcal{L}^T)^T = (\mathcal{L}K_{X^v,X})(K_{X,X} + \sigma^2 I_N)^{-1},
\end{aligned}$$

where we have used that $(K_{X,X^v}\mathcal{L}^T)^T = \mathcal{L}K_{X^v,X}$ and $LL^T = K_{X,X} + \sigma^2 I_N$.

To show that $B_1 = \mathcal{L}K_{X^v,X^v}\mathcal{L}^T + \sigma_v^2 I_{N_v} - v_1^T v_1$ we need to show that $v_1^T v_1 = A_1 K_{X,X^v}\mathcal{L}^T$, which is trivial

$$\begin{aligned}
v_1^T v_1 &= (L^{-1}K_{X,X^v}\mathcal{L}^T)^T(L^{-1}K_{X,X^v}\mathcal{L}^T) \\
&= \mathcal{L}K_{X^v,X}(LL^T)^{-1}K_{X,X^v}\mathcal{L}^T \\
&= A_1 K_{X,X^v}\mathcal{L}^T.
\end{aligned}$$

■

Appendix C. Algorithm for Finding Virtual Observation Locations based on Individual Sub-operators

We present the details of the algorithm for finding virtual observation locations introduced in Section 3.5. Here we let $\mathcal{L}$ be a linear operator defined by the column vector $[\mathcal{F}_1, \ldots, \mathcal{F}_k]$, where $\mathcal{F}_i$ produces functions from $\mathbb{R}^{n_x}$ to $\mathbb{R}$, subjected to an interval constraint $[a_i(\mathbf{x}), b_i(\mathbf{x})]$. We would like to impose constraints related to the i-th sub-operator only at locations where $p(\mathcal{F}_i f(\mathbf{x}) \notin [a_i(\mathbf{x}), b_i(\mathbf{x})])$ is not sufficiently small. For this we let X^v be the concatenation of the matrices $X^{v,1}, \ldots, X^{v,k}$ and define $\mathcal{L}^T f(X^v) = [\mathcal{F}_1^T f(X^{v,1}), \ldots, \mathcal{F}_k^T f(X^{v,1})]^T$. The matrices needed to make use of Lemma 1 and Lemma 2 are $\mathcal{L}\mu^v$, $K_{X,X^v}\mathcal{L}^T$, $K_{X^*,X^v}\mathcal{L}^T$, and $\mathcal{L}K_{X^v,X^v}\mathcal{L}^T$. Using that $\mathcal{F}_i f(X^v) = \mathcal{F}_i f(X^{v,i})$, these are given by

$$\mathcal{L}\mu^v = \begin{bmatrix} \mathcal{F}_1\mu(X^{v,1}) \\ \vdots \\ \mathcal{F}_k\mu(X^{v,k}) \end{bmatrix}, \quad K_{X,X^v}\mathcal{L}^T = \begin{bmatrix} K_{X,X^{v,1}}\mathcal{F}_1^T \\ \vdots \\ K_{X,X^{v,k}}\mathcal{F}_k^T \end{bmatrix},$$

where $K_{X^*,X^v}\mathcal{L}^T$ also is given by the above equation for $X = X^*$. Finally, $\mathcal{L}K_{X^v,X^v}\mathcal{L}^T$ is the block matrix with blocks

$$(\mathcal{L}K_{X^v,X^v}\mathcal{L}^T)_{i,j} = \mathcal{F}_i K_{X^{v,i},X^{v,j}}\mathcal{F}_j^T.$$

We want to improve the algorithm in Section 3.4 for finding the set of virtual observation locations by considering each sub-operator individually. To do this we make use estimated partial constraint probabilities (given in (13) and restated below).

$$\hat{p}_{c,i}(\mathbf{x}) = \frac{1}{m}\sum_{j=1}^{m} P\left(a_i(\mathbf{x}) - \nu < (\mathcal{L}f(\mathbf{x})|Y, C_j)_i < b_i(\mathbf{x}) + \nu\right),$$

where $(\mathcal{L}f(\mathbf{x})|Y, C_j)_i$ is the univariate Normal distribution given by the i-th row of $(\mathcal{L}f(\mathbf{x})|Y, C_j)$ and $C_1, \ldots, C_m$ are m samples of $\mathbf{C}$ given in (6) as before. For the individual sub-operators $\mathcal{F}_i$, the set of virtual observations X_i^v needed to ensure that $\hat{p}_{c,i}(\mathbf{x}) \geq p_{\text{target}}$ can then be found using the following algorithm.

Algorithm 7 *Finding locations of virtual observations X_i^v s.t. $\hat{p}_{c,i}(\boldsymbol{x}) \geq p_{target}$ for all $\boldsymbol{x} \in \Omega$ and all sub-operators $\mathcal{F}_1, \ldots, \mathcal{F}_k$.*

1. *Compute $L = Chol(K_{X,X} + \sigma^2 I_N)$.*
2. *Until convergence do:*
   (a) *If $X^v \neq \emptyset$ compute A_1 and B_1 as defined in Lemma 2, and generate m samples $C_1, \ldots, C_m$ of $\boldsymbol{C}$ given in (6).*
   (b) *If $X^v = \emptyset$ compute $(\boldsymbol{x}_i^*, p_i^*) = (\arg\min p_{c,i}(\boldsymbol{x}), p_{c,i}(\boldsymbol{x}^*))$. Otherwise compute $(\boldsymbol{x}_i^*, p_i^*) = (\arg\min \hat{p}_{c,i}(\boldsymbol{x}), \hat{p}_{c,i}(\boldsymbol{x}^*))$, for all $i = 1, \ldots, k$ with $\hat{p}_{c,i}$ defined as in (13) using the samples generated in step (a).*
   (c) *Let $(\boldsymbol{x}^*, p^*, j)$ correspond to the smallest probability: $p^* = p_j^* = min_i\ p_i^*$.*
   (d) *Terminate if $p^* \geq p_{target}$, otherwise update $X_j^v \to X_j^v \cup \{\boldsymbol{x}^*\}$.*

Appendix D. Proof of Lemma 4

Proof. This follows exactly from the proofs of Lemma 1 and Lemma 2 by replacing $\mathbf{f}^* \to \mathcal{L}f(\mathbf{x}^*)$, which implies $\mu^* \to \mathcal{L}\mu^*$, $K_{X^*,X} \to \mathcal{L}K_{\mathbf{x}^*,X}$, $K_{X^*,X^*} \to \mathcal{L}K_{\mathbf{x}^*,\mathbf{x}^*}\mathcal{L}^T$ and $K_{X^*,X^v}\mathcal{L}^T \to \mathcal{L}K_{\mathbf{x}^*,X^v}\mathcal{L}^T$.

■

Appendix E. Proof of Corollary 6

Proof. We show the derivation of the expectation and covariance of $\mathbf{f}^*|Y, C$ as the derivations for $\mathcal{L}f(\mathbf{x}^*)|Y, C$ are equivalent. From Lemma 1 we have that

$$\mathbf{f}^*|Y, C \sim \mathcal{N}(\mu^* + A(\mathbf{C} - \mathcal{L}\mu^v) + B(Y - \mu), \Sigma).$$

If we let ν, Γ be the expectation and covariance of $\mathbf{C}$, then

$$\begin{aligned}\mathbb{E}[\mathbf{f}^*|Y, C] = \mathbb{E}_{\mathbf{C}}\left[\mathbb{E}[\mathbf{f}^*|Y, \mathbf{C}]\right] &= \mathbb{E}_{\mathbf{C}}\left[\mu^* + A(\mathbf{C} - \mathcal{L}\mu^v) + B(Y - \mu)\right] \\ &= \mu^* + A(\nu - \mathcal{L}\mu^v) + B(Y - \mu),\end{aligned}$$

and

$$\begin{aligned}\operatorname{cov}[\mathbf{f}^*|Y,C] &= \mathbb{E}_{\mathbf{C}}\left[\operatorname{cov}[\mathbf{f}^*|Y,\mathbf{C}]\right] + \operatorname{cov}_{\mathbf{C}}[\mathbb{E}[\mathbf{f}^*|Y,\mathbf{C}]] \\ &= \mathbb{E}_{\mathbf{C}}[\Sigma] + \operatorname{cov}_{\mathbf{C}}[\mu^* + A(\mathbf{C} - \mathcal{L}\mu^v) + B(Y-\mu)] \\ &= \Sigma + \operatorname{cov}_{\mathbf{C}}[A\mathbf{C}] = \Sigma + A\Gamma A^T.\end{aligned}$$

■

References

Petter Abrahamsen and Fred Espen Benth. Kriging with inequality constraints. *Mathematical Geology*, 33(6):719–744, Aug 2001. ISSN 1573-8868. doi: 10.1023/A:1011078716252. URL `https://doi.org/10.1023/A:1011078716252`.

Robert J. Adler. *The Geometry of Random Fields*. Wiley series in probability and mathematical statistics. Probability and mathematical statistics. J. Wiley, 1981. ISBN 9780471278443.

Christian Agrell, Simen Eldevik, Andreas Hafver, Frank Børre Pedersen, Erik Stensrud, and Arne Huseby. Pitfalls of machine learning for tail events in high risk environments. In Stein Haugen, Anne Barros, Coen van Gulijk, Trond Kongsvik, and Jan Erik Vinnem, editors, *Safety and Reliability Safe Societies in a Changing World - Proceedings of ESREL 2018*. CRC Press, june 2018. ISBN 978-0815386827.

Rafael Amaya, Mauricio Sanchez-Silva, Emilio Bastidas-Arteaga, Franck Schoefs, and Felipe Munoz. Reliability assessments of corroded pipelines based on internal pressure - A review. *Engineering Failure Analysis*, 98, 01 2019. doi: 10.1016/j.engfailanal.2019.01.064.

Jian An and Art Owen. Quasi-regression. *Journal of Complexity*, 17(4):588 – 607, 2001. ISSN 0885-064X. doi: https://doi.org/10.1006/jcom.2001.0588. URL `http://www.sciencedirect.com/science/article/pii/S0885064X01905886`.

Ioannis Andrianakis and Peter G. Challenor. The effect of the nugget on gaussian process emulators of computer models. *Computational Statistics & Data Analysis*, 56(12):4215 – 4228, 2012. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2012.04.020. URL `http://www.sciencedirect.com/science/article/pii/S0167947312001879`.

François Bachoc. Cross validation and maximum likelihood estimations of hyper-parameters of gaussian processes with model misspecification. *Computational Statistics & Data Analysis*, 66:55 – 69, 2013. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2013.03.016. URL `http://www.sciencedirect.com/science/article/pii/S0167947313001187`.

François Bachoc, Agnes Lagnoux, and Andrés F. López-Lopera. Maximum likelihood estimation for gaussian processes under inequality constraints. working paper or preprint, August 2018. URL `https://hal.archives-ouvertes.fr/hal-01864340`.

Zdravko I. Botev. The normal law under linear restrictions: simulation and estimation via minimax tilting. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(1):125–148, 2017. doi: 10.1111/rssb.12162. URL `https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12162`.

Sébastien Da Veiga and Amandine Marrel. Gaussian process modeling with inequality constraints. *Annales de la faculté des sciences de Toulouse Mathématiques*, 21(3):529–555, 4 2012. URL http://eudml.org/doc/250989.

Sébastien Da Veiga and Amandine Marrel. Gaussian process regression with linear inequality constraints. working paper or preprint, 10 2015. URL https://hal.archives-ouvertes.fr/hal-01515468.

Alexander G. de G. Matthews, James Hensman, Richard Turner, and Zoubin Ghahramani. On sparse variational methods and the kullback-leibler divergence between stochastic processes. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 231–239. PMLR, 09–11 May 2016. URL http://proceedings.mlr.press/v51/matthews16.html.

DNV GL. Reccomended Practice: Corroded pipelines DNVGL-RP-F101. *DNV GL, Høvik, Norway*, 2017. URL http://rules.dnvgl.com/docs/pdf/dnvgl/RP/2017-05/DNVGL-RP-F101.pdf.

Simen Eldevik, Christian Agrell, Andreas Hafver, and Frank B. Pedersen. AI + Safety: Safety implications for artificial intelligence and why we need to combine casual- and data-driven models. 08 2018. URL https://ai-and-safety.dnvgl.com/. [Online position paper by DNV GL Group Technology and Research; https://ai-and-safety.dnvgl.com/, posted 28-August-2018].

Alan Genz. Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, 1(2):141–149, 1992. doi: 10.1080/10618600.1992.10477010.

Alan Genz. Comparison of methods for the computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, 11, 04 1997. doi: 10.1198/106186002394.

Zoubin Ghahramani. Probabilistic machine learning and artificial intelligence. *Nature*, 521(7553):452–459, 2015. doi: 10.1038/nature14541. URL https://doi.org/10.1038/nature14541.

Shirin Golchi, D R. Bingham, H Chipman, and David Campbell. Monotone emulation of computer experiments. *SIAM/ASA Journal on Uncertainty Quantification*, 3:370–392, 01 2015.

Carl Jidling, Niklas Wahlström, Adrian Wills, and Thomas B Schön. Linearly constrained gaussian processes. pages 1215–1224. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/6721-linearly-constrained-gaussian-processes.pdf.

Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. URL http://www.scipy.org/.

Marc C. Kennedy and Anthony O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464, 2001. doi: 10.1111/1467-9868.00294. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/1467-9868.00294.

George S. Kimeldorf and Grace Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *Ann. Math. Statist.*, 41(2):495–502, 04 1970. doi: 10.1214/aoms/1177697089. URL https://doi.org/10.1214/aoms/1177697089.

Jack P. C. Kleijnen and Wim C. M. Van Beers. Monotonicity-preserving bootstrapped kriging metamodels for expensive simulations. *JORS*, 64:708–717, 2013.

Jayesh H. Kotecha and Petar M. Djuric. Gibbs sampling approach for generation of truncated multivariate gaussian random variables. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No.99CH36258)*, volume 3, pages 1757–1760 vol.3, March 1999. doi: 10.1109/ICASSP.1999.756335.

Peter Lenk and Taeryon Choi. Bayesian analysis of shape-restricted functions using gaussian process priors. *Statistica Sinica*, 27:43–69, 2017.

Lizhen Lin and David B. Dunson. Bayesian monotone regression using gaussian process projection. *Biometrika*, 101(2):303–317, 2014. doi: 10.1093/biomet/ast063. URL http://dx.doi.org/10.1093/biomet/ast063.

Andrés López-Lopera, François Bachoc, Nicolas Durrande, and Olivier Roustant. Finite-dimensional gaussian approximation with linear inequality constraints. *SIAM/ASA Journal on Uncertainty Quantification*, 6(3):1224–1255, 2018. doi: 10.1137/17M1153157. URL https://doi.org/10.1137/17M1153157.

Hassan Maatouk and Xavier Bay. Gaussian process emulators for computer experiments with inequality constraints. *Mathematical Geosciences*, 49(5):557–582, Jul 2017. ISSN 1874-8953. doi: 10.1007/s11004-017-9673-2. URL https://doi.org/10.1007/s11004-017-9673-2.

Hassan Maatouk, Laurence Grammont, and Xavier Bay. Generalization of the kimeldorf-wahba correspondence for constrained interpolation. *Electronic Journal of Statistics*, 10 (1):1580–1595, 2016.

Henrik O. Madsen, Steen Krenk, and Niels C. Lind. *Methods of Structural Safety*. Dover Civil and Mechanical Engineering Series. Dover Publications, 2006. ISBN 9780486445977. URL https://books.google.no/books?id=e8sZjD7so-AC.

Georges Matheron. The intrinsic random functions and their applications. *Advances in Applied Probability*, 5(3):439–468, 1973. ISSN 00018678. URL http://www.jstor.org/stable/1425829.

Anna Michalak. A gibbs sampler for inequality-constrained geostatistical interpolation and inverse modeling. *Water Resour. Res*, 44, 09 2008. doi: 10.1029/2007WR006645.

Ari Pakman and Liam Paninski. Exact hamiltonian monte carlo for truncated multivariate gaussians. *Journal of Computational and Graphical Statistics*, 23, 08 2012. doi: 10.1080/10618600.2013.788448.

Athanasios Papoulis and S. Unnikrishna Pillai. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill Higher Education, 4 edition, 2002.

Pritam Ranjan, Ronald Haynes, and Richard Karsten. A computationally stable approach to gaussian process interpolation of deterministic computer simulation data. *Technometrics*, 53, 03 2010. doi: 10.1198/TECH.2011.09141.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. ISBN 026218253X.

Jaakko Riihimki and Aki Vehtari. Gaussian processes with monotonicity information. *Journal of Machine Learning Research - Proceedings Track*, 9:645–652, 01 2010.

Jerome Sacks, William J. Welch, Toby J. Mitchell, and Henry P. Wynn. Design and analysis of computer experiments. *Statist. Sci.*, 4(4):409–423, 11 1989. doi: 10.1214/ss/1177012413. URL `https://doi.org/10.1214/ss/1177012413`.

Simo Särkkä. Linear operators and stochastic partial differential equations in gaussian process regression. In Timo Honkela, Włodzisław Duch, Mark Girolami, and Samuel Kaski, editors, *Artificial Neural Networks and Machine Learning – ICANN 2011*, pages 151–158, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-21738-8.

Gudfinnur Sigurdsson, Espen H. Cramer, Ola H. Bjørnøy, B. Fu, and D. Ritchie. Background to DNV RP-F101 Corroded pipelines. In *Proceedings of the* 18^{th} *international conference on offshore mechanics and arctic engineering, OMAE, Newfoundland, Canada.* American Society of Mechanical Engineers, U.S., 1999.

Rainer Storn and Kenneth Price. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4): 341–359, Dec 1997. ISSN 1573-2916. doi: 10.1023/A:1008202821328. URL `https://doi.org/10.1023/A:1008202821328`.

Philip Duncan Thompson. Optimum smoothing of two-dimensional fields. *Tellus*, 8(3):384–393, 1956. doi: 10.1111/j.2153-3490.1956.tb01236.x. URL `https://onlinelibrary.wiley.com/doi/abs/10.1111/j.2153-3490.1956.tb01236.x`.

Xiaojing Wang and James O. Berger. Estimating shape constrained functions using gaussian processes. *SIAM/ASA Journal on Uncertainty Quantification*, 4:1–25, 01 2016.

Eun-Hye Yoo and Phaedon C. Kyriakidis. Area-to-point kriging with inequality-type data. *Journal of Geographical Systems*, 8(4):357–390, Oct 2006. ISSN 1435-5949. doi: 10.1007/s10109-006-0036-7. URL `https://doi.org/10.1007/s10109-006-0036-7`.

Paper II

Sequential sampling method using Gaussian process regression for estimating extreme structural response

II

Odin Gramstad, Christian Agrell, Elzbieta Bitner-Gregersen, Bingjie Guo, Eivind Ruth, Erik Vanem

Marine Structures (2020) Vol. 72 102780.

Marine Structures 72 (2020) 102780

Contents lists available at ScienceDirect

Marine Structures

journal homepage: http://www.elsevier.com/locate/marstruc

Sequential sampling method using Gaussian process regression for estimating extreme structural response

Odin Gramstad [a,*], Christian Agrell [a], Elzbieta Bitner-Gregersen [a], Bingjie Guo [a], Eivind Ruth [b], Erik Vanem [a]

[a] DNV GL Group Technology and Research, Høvik, Norway
[b] DNV GL Maritime, Hydrodynamics and Stability, Høvik, Norway

ARTICLE INFO

Keywords:
Extreme response
Wave environment
Structural reliability
Response based methods

ABSTRACT

A methodology for estimating extreme response statistics for marine structures, that takes both the long-term variability of the metocean environment and the short-term variability of response into account is presented. The proposed methodology uses Gaussian process regression to estimate parameters of the short-term response distribution, based on output from computationally expensive hydrodynamic simulations. We present an adaptive design strategy for sequential updating of the model, focusing on the metocean conditions that contribute the most to the long-term extreme. With this approach, only a limited number of hydrodynamic simulations are needed.

The suggested approach is demonstrated on the problem of estimating the 25-year extreme vertical bending moment on a ship. We show that a relatively small number of iterations (full hydrodynamic simulations) are needed to converge toward the "exact" results obtained by running a large number of simulations covering the entire range of sea states.

The results suggest that the proposed method can be used as an alternative to contour-based methods or other methods that consider a few sea states using accurate numerical simulations, with little or no added complexity or computational effort.

1. Introduction

The long-term metocean induced extreme response of ships and offshore structures is affected by both the *long-term* variability of the metocean environment (e.g. waves, wind, currents, etc) and the *short-term* variability of the response in a given random sea state (e.g. wave induced bending moment of a ship subject to random waves). Hence, for an accurate estimation of the long-term extreme response, both the long-term variability of the metocean environment and the short-term variability of the response need to be considered. In principle, this can be achieved by running a full long-term analysis, but in practice this is often not feasible when considering nonlinear responses and long return periods like 25 or 100 years, which are typically used in design of ship and offshore structures.

A commonly used practice in the offshore industry is to use contour-based methods, for which a contour corresponding to the desired probability level is found in the metocean parameter space. Then, short-term simulations are run for selected points along the contour, from which the maximum responses over a duration of e.g. 3 h are found. Note that due to the stochastic nature of the

* Corresponding author.
E-mail address: Odin.Gramstad@dnvgl.com (O. Gramstad).

https://doi.org/10.1016/j.marstruc.2020.102780
Received 20 September 2019; Received in revised form 3 April 2020; Accepted 20 April 2020

environmental loads the 3-h maximum is a random variable defined by some probability distribution. Typically, the median (50% percentile) or the mode (most probable maximum) of the distribution is estimated for each case, and the maximum value over all the selected combinations of environmental parameters along the contour is selected as the long-term extreme response.

The obvious drawback of this approach is that the short-term variability of the response is not accounted for. Several approaches have been proposed to include this uncertainty [1–3]. Winterstein et al. [4] suggested to use an "inflated" return period based on the FORM omission factors introduced by Madsen [5]. Other approaches suggest increasing the short-term percentile of the response derived along the contours. That is, instead of the mode or median of the distribution, to use a higher fractile level, or apply a correction factor to the mode or median. Values reported in the literature are fractiles 75%–90% for 100-year response, and multiplying factors 1.1 to 1.3 [2,6]. The problem of applying such methods is that the appropriate fractile levels and multipliers are case-specific, and depend of the type of structure and structural response. A recent investigation of different methods to correct for the lack of short-term variability in the context of line tensions of offshore units was presented in Ref. [3].

An alternative to contour based methods is to run a full long-term analysis taking both long- and short-term variability into account, by applying a linear approximation to the problem so that the computational effort allows for a long-term analysis, and combining it with selected nonlinear response calculations. As suggested by Baarholm and Moan [7] in the context of ship vertical bending moment, one can identify the most important sea states in a scatter diagram by using linear analysis. Then nonlinear simulations can be performed on selected sea states to calculate the long-term responses. These can then be used to apply a correction factor to the linear results.

In this paper we propose an alternative method for estimating extreme responses, that takes both the long- and short-term variability into account, and uses an active learning/experimental design approach based on Gaussian Process (GP) regression to limit the number of time-consuming simulations needed in the analysis. For experimental design (i.e. deciding which experiments/simulations to perform), the Gaussian process framework has proved particularly useful [8–11], especially for applications involving complex computer simulations [12,13]. In the context of extreme response statistics, an approach based on Gaussian processes was recently presented by Mohamad and Sapsis [14]. In the present paper, we utilize Gaussian Process regression models to represent the distribution parameters describing the extreme response under consideration, and repeated sampling from the distributions is utilized to estimate the long-term responses. A similar approach was applied by Wang et al. [15], who also used a GP/Kriging based model to study the long-term extreme of an FPSO mooring system.

The proposed method is applied to a real problem of estimating the 25-year extreme ship vertical bending moment (VBM) on an LNG tanker, taking nonlinearities in both the wave description and the ship-response into account. The same case study was considered in a recent paper [16], also using GP-models to estimate the long-term extreme VBM. In the present paper, however, we address two limitations of the approach in Vanem et al. [16], by including short-time variability as well as applying an active learning approach, which is shown to drastically reduce the number of simulations needed for obtaining accurate results.

The paper is organized as follows. First, in section 2, a brief introduction to Gaussian process regression is given. In section 3, the proposed sampling method using GP regression to estimate the long-term extreme response is presented. The case study, for which the methodology is applied to estimate the 25-year extreme wave induced bending moment on an LNG-tanker is described in section 4. Finally, some discussion and conclusions are given in section 5.

2. Gaussian process regression

A Gaussian process $f \sim \mathcal{GP}(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}'))$, given as a prior over functions $\mathrm{f} : \mathbb{R}^D \to \mathbb{R}$, is specified by its mean and covariance function

$$\mu(\boldsymbol{x}) = \mathbb{E}[f(\boldsymbol{x})] : \mathbb{R}^D \to \mathbb{R}, \tag{1a}$$

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}[(f(\boldsymbol{x}) - \mu(\boldsymbol{x}))(f(\boldsymbol{x}') - \mu(\boldsymbol{x}'))] : \mathbb{R}^{D \times D} \to \mathbb{R}. \tag{1b}$$

The defining property of the Gaussian process is that for any finite collection $[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$ of points $\boldsymbol{x}_i \in \mathbb{R}^D$, the distribution of $[f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_n)]$ is multivariate Gaussian with mean $[\mu(\boldsymbol{x}_1), \ldots, \mu(\boldsymbol{x}_n)]$ and covariance $\boldsymbol{K}$ given as the Gram matrix $\boldsymbol{K}_{i,j} = \boldsymbol{k}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. One important feature of the multivariate normal distribution is that if a set of random variables are jointly multivariate normally distributed, then the *marginal* distribution of subsets of the random variables will be Gaussian. Moreover, if some of the variables are observed, the conditional distribution of the remaining variables will also be normal. In GP-regression, these properties are exploited to make predictions of $f(\boldsymbol{x}^*)$ at unobserved points $\boldsymbol{x}^*$ in the input space.

More specifically, given some training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ and a likelihood function $p(y|f)$, we want to infer the distribution of $f(\boldsymbol{x}^*)|\mathscr{D}$. This is trivial when we assume that observations come with additive white noise (Gaussian likelihood function), which means that $y_i = f(\boldsymbol{x}_i) + \varepsilon_i$ where the noise term ε_i follows a normal distribution. We will assume that ε_i are i.i.d. zero-mean Gaussian with common variance c^2. The predictive posterior distribution at n new inputs, $\boldsymbol{f}^* \,|\, \mathscr{D} = [f(\boldsymbol{x}_1^*), \ldots f(\boldsymbol{x}_n^*)] \,|\, \{y_i = f(\boldsymbol{x}_i) + \varepsilon_i\}_{i=1}^N$, is then given in closed form:

$$\boldsymbol{f}^* \,|\, \mathscr{D} \sim N\left(\mu_{\boldsymbol{f}^*|\mathscr{D}}, \Sigma_{\boldsymbol{f}^*|\mathscr{D}}\right), \tag{2}$$

with mean and covariance

$$\mu_{f^*|\mathscr{D}} = \boldsymbol{\mu}_* + \boldsymbol{K}_* \left(\boldsymbol{K} + c^2 I\right)^{-1} (\boldsymbol{y} - \boldsymbol{\mu}), \tag{3a}$$

$$\Sigma_{f^*|\mathscr{D}} = \boldsymbol{K}_{**} - \boldsymbol{K}_* \left(\boldsymbol{K} + c^2 I\right)^{-1} \boldsymbol{K}_*^T. \tag{3b}$$

Here $\boldsymbol{\mu}_*$, $\boldsymbol{\mu}$ and $\boldsymbol{y}$ are vectors with elements $\mu(\boldsymbol{x}_i^*)$, $\mu(\boldsymbol{x}_i)$and y_i respectively, I is the $N \times N$ identity matrix, and $\boldsymbol{K}_*$ and $\boldsymbol{K}_{**}$ have elements $(\boldsymbol{K}_*)_{ij} = k(\boldsymbol{x}_i^*, \boldsymbol{x}_j)$ and $(\boldsymbol{K}_{**})_{ij} = k(\boldsymbol{x}_i^*, \boldsymbol{x}_j^*)$.

In order to train a Gaussian process model, one needs to specify a mean function $\mu(\boldsymbol{x})$ and the type of covariance function $k(\boldsymbol{x},\boldsymbol{x}')$. In this study we have used the Matérn 3/2 kernel, which can be written in the form

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma^2 \left(1 + \sqrt{3}\, r\right) \mathrm{e}^{-\sqrt{3}r} \quad \text{where} \quad r = \sqrt{\sum_{i=1}^{D} \frac{(x_i - x_i')^2}{l_i}}. \tag{4}$$

The kernel variance σ^2, correlation length-scales l_i and the white noise variance c^2 are hyperparameters of the GP model that can be estimated from the training data, typically by maximum likelihood or Bayesian methods [17]. We could also include additional hyperparameters from the mean function, e.g. $\mu(\boldsymbol{x}|\boldsymbol{\beta}) = \boldsymbol{x}^T\boldsymbol{\beta}$for some unknown vector $\boldsymbol{\beta}$, but in this paper we do not make use of mean functions that come with additional hyperparameters. For a more comprehensive introduction to Gaussian process regression, see e.g. Ref. [18].

In the application presented in this paper, we represent multiple variables using GP regression models. The most straightforward way to achieve this is to fit independent GP models for the desired output variables, corresponding to fitting independent covariance kernels for each case. A limitation of this approach is that one does not take into account possible dependencies between the output variables. In the case that the considered output variables are statistically dependent, the ability to model this dependency may lead to an improved model. The construction of multi-output GPs is based on the intuitive idea that if $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ are multivariate Gaussian, then the joint distribution $[\boldsymbol{f}_1, \boldsymbol{f}_2]$ is also multivariate Gaussian. In particular, if $\boldsymbol{f}_1 \sim N(\boldsymbol{\mu}_1, \boldsymbol{K}_1)$ and $\boldsymbol{f}_2 \sim N(\boldsymbol{\mu}_2, \boldsymbol{K}_2)$, then

$$\begin{bmatrix} \boldsymbol{f}_1 \\ \boldsymbol{f}_2 \end{bmatrix} \sim N\left(\begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \begin{bmatrix} \boldsymbol{K}_1 & \boldsymbol{K}_{1,2} \\ \boldsymbol{K}_{1,2}^T & \boldsymbol{K}_2 \end{bmatrix} \right),$$

where $\boldsymbol{K}_{1,2}$ is the cross-covariance between $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$. This extends naturally to the case where $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ are GPs, with mean and covariance *functions*. When the cross-covariance $\boldsymbol{K}_{1,2} = 0$, this multi-output GP over functions $\mathrm{f} : \mathbb{R}^D \to \mathbb{R}^2$ is equivalent to two independent GPs modeling two functions $\mathrm{f}_1, \mathrm{f}_2 : \mathbb{R}^D \to \mathbb{R}$. The relevant theory related to multi-output GPs thus revolves around designing a valid (positive definite) kernel function that can transfer information between the processes. This usually goes under the name of *multi-task learning*, *vector-valued learning* or *transfer learning* within machine learning, or *cokriging* in statistics. We refer to Ref. [19] for a detailed overview. The model we will consider in this paper is a type of a linear model for coregionalization (LCM) for 2-dimensional output, based on the assumption that two processes, f_1 and f_2, are both linear transformations of a common set of latent GPs:

$$f_d(\boldsymbol{x}) = \mu_d(\boldsymbol{x}) + \sum_{q=1}^{Q} \sum_{i=1}^{R_q} a_{d,q}^i u_q^i(\boldsymbol{x}), \tag{5}$$

for $d = 1, 2$. Here $a_{d,q}^i$ are scalar coefficients (hyperparameters) and u_q^i are zero-mean GPs with covariance functions $\mathrm{cov}[u_q^i(\boldsymbol{x}), u_{q'}^{i'}(\boldsymbol{x}')] = k_q(\boldsymbol{x},\boldsymbol{x}')$ if $i = i'$ and $q = q'$. In the case study presented in section 4 we consider a natural extension of the scenario with independent GPs for our LCM model, by selecting $Q = 2$ and $R_q = 1$ where $k_q(\boldsymbol{x},\boldsymbol{x}')$ for $q = 1, 2$ are both of the Matérn 3/ 2 class (4) but with different parameters σ^2 and l_i.

3. Estimating long-term extreme response by direct sampling

3.1. Definition of the long-term extreme response

We consider a dynamical system whose extreme value y over a time-period T_s, for a given state $\boldsymbol{x} = (x_1, \cdots, x_n)$ follows some probability distribution $g_{Y|X}(y|\boldsymbol{x})$. It is further assumed that the states $\boldsymbol{X}$ follow a joint probability distribution $f_X(\boldsymbol{x})$.

For example, taking the case study that is presented in section 4, $f_X(\boldsymbol{x}) = f_{H_s,T_z}(h_s, t_z)$ is the long-term distribution of sea states with respect to significant wave height H_s and zero-crossing wave period T_z, and $g_{Y|X}(y|\boldsymbol{x})$ is the short-term distribution of maximum vertical ship bending moment (VBM) during a sea-state duration $T_s = 3$ hours.

If both $f_X(\boldsymbol{x})$ and $g_{Y|X}(y|\boldsymbol{x})$ are known, the marginal distribution of Y for a random state can be found by integrating over all states $\boldsymbol{X}$,

$$g_Y(y) = \int g_{Y|X}(y|\boldsymbol{x}) f_X(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}. \tag{6}$$

In practice, evaluating this integral directly may be difficult. In many cases $g_{Y|X}(y|\boldsymbol{x})$ is not known analytically, but samples from the underlying distribution may be obtained, for example by numerical simulation of the underlying dynamical system. Again, taking the case study presented herein as an example, obtaining a sample from $g_{Y|X}(y|\boldsymbol{x})$ means running a hydrodynamic numerical simulation of a

ship in random waves, from which the maximum VBM can be calculated.

We now assume that the quantity of interest is the N_y-year extreme value of Y, denoted Y_{N_y} in the following. Assuming that $g_Y(y)$ is known explicitly from (6), this may be estimated as the $1-1/(365.25\cdot 24\cdot N_y/T_s)$-fractile of $g_Y(y)$, or as the mode of the corresponding extreme value distribution of the N_y-year maximum, defined in terms of its cumulative distribution function $G_{Y_{N_y}}(y) = G_Y(y)^{365.25\cdot 24\cdot N_y/T_s}$.

In the case that $g_Y(y)$ is not known, but that samples from both $f_{\boldsymbol{X}}(\boldsymbol{x})$ and $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$ can be obtained, the distribution of the N_y-year extreme value can be estimated by the following sampling approach, summarized in Algorithm 1.

Algorithm 1 Direct sampling method for estimating the N_y-year extreme value of Y

1. Draw $N_s = 365.25\cdot 24\cdot N_y/T_s$ random states $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_{N_s}\}$ from $f_{\boldsymbol{X}}(\boldsymbol{x})$.
2. For each state $\boldsymbol{x}_j$, draw a corresponding y_j from $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$.
3. Find the maximum of all samples: $y_{N_y}^{(max)} = \max\{y_1,\cdots,y_{N_s}\}$.
4. Repeat step 1–3 to obtain N samples of the N_y-maximum.

Note that this procedure takes into account the variability of both the states $\boldsymbol{X}$ and the dynamical response Y. For the case study presented below this means that the variability in both the wave-environment description (i.e. accounting for that every N_y-year period will have different realizations of sea states) and the short-term variability of the ship response (i.e. that a given sea state may give different maximum VBM, due to randomness of the waves) are accounted for.

3.2. The probabilistic surrogate model

The problem of applying the procedure outlined in Algorithm 1 in practice, is typically that obtaining a very large number of samples from $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$ is too computationally expensive. The crude Monte Carlo procedure given in Algorithm 1 will require running $N\times N_s$ numerical simulations of the relevant dynamical system. We could make use of variance reduction techniques such as importance sampling to reduce the required number of samples, but the total number of samples would still be much larger than what we can achieve within reasonable time in practice.

To overcome such a problem it is common to use environmental contours, where a contour in the $\boldsymbol{X}$-space is constructed to represent the desired probability level, and that samples from $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$ are obtained only for a few states along this contour. However, with the contour approach some of the inherent randomness is lost, since contribution to the long-term extreme may come from a wide range of states, in particular if short-term variability of the dynamical system is important.

Here we suggest an alternative approach, which aims to estimate the extreme response by a limited number of samples from the distribution $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$, but where the variability of $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$ is properly accounted for. The suggested procedure relies on the assumption that the distribution $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$ can be modeled by some parametric distribution $\widehat{g}(y;\boldsymbol{\theta})$ with distribution parameters $\boldsymbol{\theta}(\boldsymbol{x}) = (\theta_1(\boldsymbol{x}),\cdots,\theta_m(\boldsymbol{x}))$. Again taking the case study presented in 4 as an example, $\boldsymbol{\theta}(\boldsymbol{x})$ might be the parameters of a Weibull distribution representing the structural response in sea-states $\boldsymbol{x} = (H_s,T_z)$. Assuming that drawing random samples from $\widehat{g}(y;\boldsymbol{\theta})$ is trivial, the problem reduces to obtaining the distribution parameters $\boldsymbol{\theta}(\boldsymbol{x})$ for all possible values for $\boldsymbol{x}$.

The following approach is based on the simple idea of representing the unknown distribution parameters $\boldsymbol{\theta}(\boldsymbol{x})$ by a surrogate model that is constructed using Gaussian Process (GP) regression, which is trained on samples from $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$ for a limited number of inputs $\boldsymbol{x}$. The use of a probabilistic surrogate model is a common technique when dealing with expensive computer models. The purpose of a surrogate, is to serve as an approximation that is cheap to evaluate numerically. The *probabilistic* component is needed to express uncertainty in the approximation, i.e. given some input $\boldsymbol{x}$, the output of the surrogate should be a distribution representing possible outcomes of running the expensive computer model. Surrogates based on Gaussian Processes have this property, and are used extensively for problems related to uncertainty quantification (UQ) involving expensive computer simulations [12,13]. For a broader overview of relevant methods from the UQ literature, we refer to Ref. [20], where the theory related to *uncertainty propagation* is most relevant for the type of problem we address in this paper. We also note that most of the UQ literature is dealing with deterministic computer models. This is not the case for the model we consider in this paper, as our numerical model of the dynamical system produces samples from $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x})$, from which we estimate the parameter $\boldsymbol{\theta}(\boldsymbol{x})$. We make use of a Gaussian Process surrogate, and can in general handle both cases. In particular, the observational noise in our GP model is represented by the constant variance c^2, and letting $c=0$ would correspond to the deterministic alternative (although a small fixed $c\approx 0$ is usually used for numerical stability). In our non-deterministic situation, we infer the observational variance c^2 directly from the data. Typically, the parameters used in training of the GP model are estimated from the raw data by e.g. maximum-likelihood or Bayesian inference. In our implementation we rely on maximum-likelihood (plug-in) estimates of all the GP parameters. However, if we had any prior knowledge about the uncertainties of the distribution parameters, a better alternative would be to include this in the GP model explicitly.

In many practical problems there exists an approximation of the underlying dynamical system, for which approximate distribution parameters $\tilde{\boldsymbol{\theta}}(\boldsymbol{x})\approx\boldsymbol{\theta}(\boldsymbol{x})$ are known from theory, or can be efficiently calculated by simulating a simplified model. Typically, such an approximation can be obtained as a linearization of the full model, and may be useful as prior information to the GP-model. Hence, in the following we assume that the distribution parameters are represented as

$$\boldsymbol{\theta}(\boldsymbol{x}) \sim \mathcal{GP}(\tilde{\boldsymbol{\theta}}(\boldsymbol{x}), k(\boldsymbol{x},\boldsymbol{x}')), \tag{7}$$

where $\tilde{\theta}(x)$ is a known (or easy to calculate) function obtained from an approximate model, and $k(x,x')$ is the covariance function (4). The training data for the GP is a set of observations $\mathscr{D} = \{\theta_i, x_i\}_{i=1}^{M}$, where θ_i is estimated using samples[1] from $g_{Y|X}(y|x)$ at $x = x_i$. To estimate the parameter $\theta(x)$ at any x, we use the posterior predictive distribution (2), which we denote $p(\theta|\mathscr{D},x)$. To represent the marginal distribution of maximum vertical ship bending moment during a sea-state duration of $T_s = 3$ hours, $g_Y(y)$ given in (6), we then have the following approximation

$$\hat{g}_Y(y) = \int \hat{g}(y|\theta)p(\theta|\mathscr{D},x)f_X(x)\,\mathrm{d}\theta\mathrm{d}x. \tag{8}$$

By comparing the distributions (6) and (8), we see that the approximation relies on two assumptions. First, that the conditional distribution $g_{Y|X}(y|x)$ can be modeled by a parametric distribution $\hat{g}(y|\theta(x))$, and second, that the parameter $\theta(x)$ for any x can be inferred from a finite set of observations $\mathscr{D}$. For the latter, we will present an approach for updating set of observations $\mathscr{D}$ iteratively, so that this can be justified using only a moderate number of observations. We note also that if no approximate model (prior mean) is available, $\tilde{\theta}(x) = 0$ can be used, together with a few initial runs to initialize the GP-model. The effect of having a prior mean on the convergence of the method is further discussed in section 4.4.2.

In order for the approximation (8) to be accurate, the distribution $p(\theta|\mathscr{D},x)$ must be well represented by the GP model. However, in particular for small number of observations $\mathscr{D}$, accurate estimation of the uncertainty of the distribution parameters $\theta(x)$ may be difficult. This may again lead to inaccuracies in the estimated extreme response, when the uncertainty in the distribution parameters is integrated out, as is the case in (8).

Another alternative is to ignore the uncertainty in the distribution parameters and use the predictive mean of the GP models instead of sampling from the GP. That is, instead of (8), we may use

$$\hat{g}_Y(y) = \int \hat{g}(y|\mathbb{E}[\theta|\mathscr{D},x])f_X(x)\,\mathrm{d}x. \tag{9}$$

The effect of including the uncertainty in the distribution parameters is briefly discussed in section 4.4.1.

3.3. Sequential sampling strategy

The update of the GP model is done in a sequential manner so that samples from $g_{Y|X}(y|x)$ are obtained in regions in the x-space that influence the end result the most. In other words, the sampling points are chosen so that the surrogate model is most accurate in regions of the x-space that contribute to the long-term extreme value of Y.

We make use of a one-step lookahead (myopic) strategy, where the inputs x are selected sequentially as maximizers of a specified *acquisition function*. See for instance Refs. [8–10,21] that discuss this type of strategies in more detail, although for different objectives than the one presented in this paper.

We propose an acquisition that makes a balance between exploring new regions in the x-space where the response is unknown, and areas that are expected to have a large influence on the long-term extreme response. It is defined as a weighted product of the predictive variance of $\theta(x)$ that comes from the Gaussian Process, and the probability density of $x|\hat{Y}(x) = \hat{Y}_{N_y}$. Here $\hat{Y}(x)$ is the response given by the probabilistic surrogate (8), with corresponding marginal N_y-year extreme value $\hat{Y}_{N_y}$. A detailed description of the acquisition function, and the complete sequential sampling approach for estimating the N_y-year extreme value of Y, is summarized in Algorithm 2.

Algorithm 2 Sequential sampling method for estimating the N_y-year extreme value of Y

1. Initialize $\theta(x)$ as a GP-model (7) using the prior mean $\tilde{\theta}(x)$ and chosen covariance function $k(x,x')$. In the case that no good linear prior is known, let $\tilde{\theta}(x) = 0$ and instead use a few selected initial runs (samples from $g_{Y|X}(y|x)$) to initialize the GP model.

2a) Draw $N_s = 365.25 \cdot 24 \cdot N_y/T_s$ random states $\{x_1, \cdots, x_{N_s}\}$ from $f_X(x)$.

2b) For each state x_j, find the corresponding distribution parameters θ_j from the GP model (7) either by random sampling from the GP (corresponding to (8)) or by using the predictive mean of the GP model (corresponding to (9)), and draw corresponding y_j from $\hat{g}(y;\theta_j)$.

2c) Find the maximum of all samples: $y_{N_y}^{(max)} = \max\{y_1, \cdots, y_{N_s}\}$.

3. Repeat steps 2a)-2c) N times to obtain an ensemble $\mathscr{C}$ of N N_y-year maxima together with a set $\mathscr{S}$ of states $x^{(max)}$ in which these maxima occurred:

$$\mathscr{C} = \left\{y_{N_y,1}^{(max)}, \cdots, y_{N_y,N}^{(max)}\right\}, \qquad \mathscr{S} = \left\{x_1^{(max)}, \cdots, x_N^{(max)}\right\}. \tag{10}$$

4. From the sets $\mathscr{C}$ and $\mathscr{S}$, estimate the empirical distribution of the N_y-year maxima $c(y_{N_y}^{(max)})$and the multivariate distribution $s(x^{(max)})$ of states in which the N_y-year maximum occurs. These distributions can typically be estimated using kernel density estimation.

5. Choose a new sea state for training of the GP model $\theta(x)$, based on the uncertainty of the GP estimates and the estimated distribution $s(x^{(max)})$. That is, we select a new training point as follows:

(*continued on next page*)

[1] The samples generated by running a $T_s = 3$ hour time-domain simulation in sea state $x = x_i$.

(*continued*)

$$\boldsymbol{x}^{(new)} = \underset{\boldsymbol{x}}{\operatorname{argmax}}\ s(\boldsymbol{x})^{1-\nu}\left|\boldsymbol{\sigma}_{\boldsymbol{\theta}}(\boldsymbol{x})\right|^{\nu}, \tag{11}$$

where $\boldsymbol{\sigma}_{\boldsymbol{\theta}} = (\boldsymbol{\sigma}_{\theta_1}, \cdots, \boldsymbol{\sigma}_{\theta_m})$ is the vector of standard deviations of the GP estimates for the distribution parameters. These uncertainties will typically be small close to the existing training data points, and larger in regions far from existing training data. Hence, (11) represents a trade-off between training in regions that have a high probability of contributing to the extreme value, and to reduce the uncertainty of the parameter estimates $\boldsymbol{\theta}$. The parameter $\nu \in [0, 1]$ may be chosen to give desired weighting between reducing uncertainty in the most important regions and exploring "new" regions.

6. Obtain a sample from the underlying distribution $g_{Y|X}(y|\boldsymbol{x})$ for the state selected in point 5, and update (train) the GP models with this new observation.
7. Repeat step 2 to 6 until the change in subsequent estimates of the distribution $c(y_{N_y}^{(max)})$ is smaller than some desired tolerance.

4. Case study: long-term extreme vertical bending moment on a tanker

The ultimate strength of the hull girder is one of the most important characteristics for reliability of a ship structure. It determines the reliability against the most critical structural failure mode identified as sagging failure of a ship in severe weather conditions. The potential consequence is total loss of the ship, cargo and crew. Thus, the accurate estimation of the long-term extreme wave bending moment contributing to dimensioning the hull girder is very important for ship structural design and reliability assessment [22].

The use of a linear analysis for the wave bending moment in extreme weather is a simplification. However, it is difficult to conclude on a "correct" model uncertainty to account for non-linear effects; e.g. due to green water on deck or slamming. Many investigations have been performed to find a measure for nonlinear correction of linear calculations, but the results show large scatter. This uncertainty has significant impact on probability of ship structure failure as demonstrated e.g. by Hørte et al. [22].

Some methods for taking the nonlinear effects into account require only one time domain simulation in a selected wave trace, which is assumed to induce the extreme wave loads, often called a most likely extreme response (MLER) wave. Other approaches require time domain simulation in a variety of conditions [23]. Often numerical calculations are supported by experimental tests. During the development of the IACS Common Structural Rules for Tankers (CSR) non-linear effects of wave vertical bending moment were accounted for in the Structural Reliability Analysis (SRA) through introduction of a nonlinear uncertainty factor [22].

As mentioned in the introduction, many methods such as contour-based methods [4], typically ignore the short-term variability of the response. However, short-term variability can be accounted for in an ad-hoc manner when using contour methods [2,24].

In the following, we use the methodology described in the previous section, and show that we are able to take the short-time variability into account in a consistent manner and achieve accurate results in similar computational time and number of nonlinear simulation as when using contour-based methods. The basic parameters of the LNG tanker considered in the case study are given in Table 1.

Using the notation introduced in the previous section on the problem of estimating the 25-year extreme VBM, we define $\boldsymbol{X} = (H_s, T_z)$, where H_s is the significant wave height and T_z is the zero-crossing wave period associated with a random sea state. The long-term distribution of sea states is then $f_{\boldsymbol{X}}(\boldsymbol{x}) = f_{H_s,T_z}(h_s, t_z)$. For a given sea state $\boldsymbol{X} = (H_s, T_z)$ we let $g_{Y|\boldsymbol{X}}(y|\boldsymbol{x}) = g_{Y|\boldsymbol{X}}(y|h_s, t_z)$ represent the probability distribution of maximum VBM during 3 h. More details are given below.

4.1. Long-term wave description

In the following we consider two different cases for the long-term wave description. For the first case (hereafter referred to as "NAS") it is assumed that the long-term distribution $f_{H_s,T_z}(h_s, t_z)$ of sea states is modeled by a marginal three-parameter Weibull distribution for H_s and a conditional log-normal distribution for T_z, see Ref. [25], as follows:

$$f_{H_s}(h_s) = \frac{\beta}{\alpha}\left(\frac{h_s-\gamma}{\alpha}\right)^{\beta-1}\exp\left[-\left(\frac{h_s-\gamma}{\alpha}\right)^{\beta}\right], \tag{12}$$

$$f_{T_z|H_s}(t_z|h_s) = \frac{1}{\sigma(h_s)t_z\sqrt{2\pi}}\exp\left[-\frac{(\ln t_z-\mu(h_s))^2}{2\sigma(h_s)^2}\right], \tag{13}$$

Table 1
Parameters of the LNG tanker used in this study.

	Unit	Value
Length (Lpp)	[m]	186.90
Breadth (B)	[m]	30.38
Depth (D)	[m]	18.20
Draught (T)	[m]	8.40
Displacement (Δ)	[kg]	35 674 800
COG_x	[m]	94.8675
COG_y	[m]	0
COG_z	[m]	8.26

where

$$\mu = E(\ln T_z) = a_1 + a_2 h_s^{a_3}, \quad \sigma = std(\ln T_z) = b_1 + b_2\, e^{b_3 h_s}. \tag{14}$$

The distribution parameters for the case considered in this study are taken from the North-Atlantic scatter diagram [2], which are fitted to about 37 years of visual observations from the North-Atlantic [26].

The second case (hereafter referred to as "DATA") is based on a dataset of wave hindcast simulations from one location in the North Atlantic Ocean. The data consist of three sets of 30-year data for the period January 1971–December 2000 corresponding to three ensemble members from wave model runs using the WAM wave model with wind input from the EC-EARTH climate model, as described in Ref. [27]. The data are taken from the location 57.62 °N, 20.28°W. It is assumed that the three model runs are independent and equally likely realizations of the ocean wave climate on this location, so that the combined data represents 90 years of wave climate data for a stationary wave climate. Note that for this case, we do not use a parametric distribution for the wave environment distribution, but use bootstrap sampling directly from the dataset.

4.2. Short-term description of VBM

The 3-h extreme bending moment is calculated with the improved nonlinear version of the time-domain sea-keeping code WASIM [28], using wave input from the nonlinear wave model DNV GL HOSM. The nonlinear WASIM is a 3-D panel method, which considers nonlinear Froude-Krylov and hydrostatic forces [29]. The DNV GL HOSM code implements the higher order spectral method (HOSM), described in West et al. [30], with nonlinear calculation of water particle kinematics, using the H_2-operator method presented in Bateman et al. [31]. The nonlinear order in calculation of waves and kinematics is in this study set to $M = 5$. Verification of WASIM combined with DNV GL HOSM has been presented in Refs. [16,29,32].

As mentioned above, for a given sea state $\boldsymbol{X} = (H_s, T_z)$ we let $g_{Y|X}(y|h_s, t_z)$ represent the probability distribution of maximum VBM during 3 h. Naturally, the distribution $g_{Y|X}(y|h_s, t_z)$ is not known, but samples of the distribution is obtained by running a WASIM simulation.

Following the methodology outlined in section 3, we assume that $g_{Y|X}(y|h_s, t_z)$ can be approximated by a distribution $\hat{g}(y;\boldsymbol{\theta})$. A common statistical model for representing VBM for a ship in random waves is the Weibull distribution, and in the following we will assume that the individual VBM maxima can be described by a two-parameter Weibull distribution with probability density function $h(m)$ and cumulative distribution function $H(m)$, given as

$$h(m) = \frac{\beta}{\alpha}\left(\frac{m}{\alpha}\right)^{\beta-1} \mathrm{e}^{-\left(\frac{m}{\alpha}\right)^{\beta}}, \qquad H(m) = 1 - \mathrm{e}^{-\left(\frac{m}{\alpha}\right)^{\beta}}, \tag{15}$$

where distribution parameters $\alpha = \alpha(H_s, T_z)$ and $\beta = \beta(H_s, T_z)$ are functions of sea state parameters.

Given the distribution of individual VBM maxima (15), it is easy to show that the extreme VBM during a sea-state duration T_s has cumulative distribution function and probability density function given respectively as

$$\widehat{G}(y) = H(y)^{N_w} = \left(1 - \mathrm{e}^{-\left(\frac{y}{\alpha}\right)^{\beta}}\right)^{N_w} \quad \text{and} \quad \hat{g}(y) = N_w H(y)^{N_w - 1} h(y), \tag{16}$$

where $N_w = T_s / T_z^{(vbm)}$ is the number of individual response cycles during the sea state duration. Using the notation introduced in section 3, we let $\hat{g}(y;\boldsymbol{\theta})$ be defined by (16) with $\boldsymbol{\theta} = (\alpha(H_s, T_z), \beta(H_s, T_z), T_z^{(vbm)}(H_s, T_z))$. We also note that random samples from the distribution $\hat{g}(y)$ can easily be obtained as

$$Y = \widehat{G}^{-1}(U) = \alpha\left(-\ln\left(1 - U^{1/N_w}\right)\right)^{1/\beta}, \tag{17}$$

where U is a uniformly distributed random variate on $(0, 1)$, and where $\widehat{G}^{-1}$ is the inverse function of $\widehat{G}$.

A priori the distribution parameters α, β and $T_z^{(vbm)}$ as function of sea state parameters (H_s, T_z) are unknown, and must be estimated by running nonlinear WASIM simulations. However, in the case of linear random waves and linear ship response, subject to the relevant response amplitude operator (RAO) of the ship, it is known that $\alpha = \sqrt{2}\sigma_r(H_s, T_z)$, $\beta = 2$ and $T_z^{(vbm)} = 2\pi\sqrt{m_0^{(vbm)}/m_2^{(vbm)}}$, where $\sigma_r = \sqrt{m_0^{(vbm)}}$ and where

$$m_j^{(vbm)} = \int \omega^j H(\omega)^2 S(\omega) \mathrm{d}\omega. \tag{18}$$

$H(\omega)$ and $S(\omega)$ are the RAO and wave spectrum, as functions of the angular wave frequency ω, respectively. Note also that under the linear assumption, (15) reduces to a Rayleigh distribution with scale parameter σ_r.

4.3. Gaussian process regression models

Applying the method suggested in section 3, and using the Gaussian process regression framework described in section 2, we introduce GP regression models to represent the distribution parameters $\boldsymbol{\theta} = (\alpha(H_s, T_z), \beta(H_s, T_z), T_z^{(vbm)}(H_s, T_z))$ using (7) with

$$\tilde{\boldsymbol{\theta}} = \left(\sqrt{2}\sigma_r, 2, 2\pi\sqrt{m_0^{(vbm)}/m_2^{(vbm)}}\right) \tag{19}$$

As mentioned in section 2 we consider both the case that all three output variables are modeled independently, and the more advanced approach where the dependence between α and β are included in the GP model by using the LCM model [19] available in the GPy software [33]. We have not considered the possible dependence between $T_z^{(vbm)}$ and α and β. Thus, $T_z^{(vbm)}$ is modeled independently in both cases.

For comparison purposes we have also considered the simpler approach adopted in Ref. [16]. In this case, one GP-model is used to represent the most probable extreme value, x_c, of the vertical bending moment in a sea state, which for a given Weibull distribution with parameters α and β is given by the mode x_c of the extreme distribution $\hat{g}$:

$$x_c = \alpha(\log N_w)^{\frac{1}{\beta}}. \tag{20}$$

Then a GP-model is used to represent the most probable extreme as a function of H_s and T_z. For any sea state (H_s, T_z), the GP is used to find the most probable extreme VBM in this sea state. Note that a limitation of this approach compared to the methodology presented in this paper is that it does not take the short-term variability of the response into account. Note also that in Vanem et al. [16], the GP-model for the most probable extreme was fitted using all available WASIM runs (95 in total), but herein we adopt the same

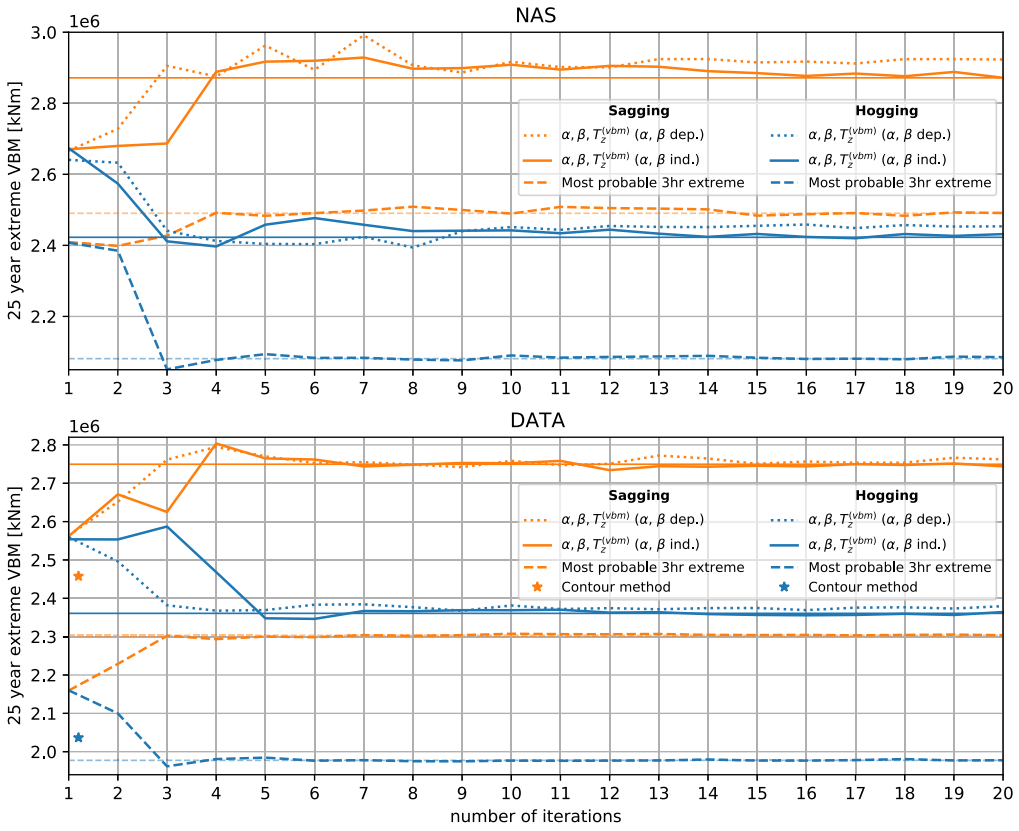


Fig. 1. Estimated mean 25-year extreme VBM at different iteration steps, for hogging and sagging for NAS (upper) and DATA (lower) cases. Linear result corresponds to the value at iteration 1 (for which the hogging and sagging moments are the same). Dotted and solid lines show the results with and without modeling the dependency between α and β. Dashed lines show the results using the most probable extreme (no short-term variability), and thin horizontal lines show the "exact" results obtained by using all 95 available WASIM simulations in fitting of the models. The corresponding results of contour method [16] using the most probable maximum are shown for the DATA case.

sequential update of the GP-models also for this case.

4.4. Results

Using the methodology described above, and applying Algorithm 2 with $N = 1000$ and $\nu = 0.5$ we obtain, for each iteration, $N = 1000$ samples of the $N_y = 25$-year extreme VBM. In the following we let the mean of the $N = 1000$ realizations of the 25-year period be an estimate of the $N_y = 25$-year extreme VBM. Alternatively, one could, for example, have used the mode or median instead of the mean.

The resulting estimates of the 25-year extreme VBM as a function of the number of iterations, are shown in Fig. 1. The dotted and solid lines show the results with and without modeling of the dependency between α and β, respectively. The dashed lines show the results of the approach for which the short-term variability is ignored (using the most probable maximum in each sea state).

The "true" values of the extreme VBM, shown by horizontal lines in Fig. 1, are in the following comparisons taken as the results obtained by fitting GP models on all available WASIM runs, consisting of 95 runs evenly distributed over the entire scatter diagram, see also Vanem et al. [16], for more details.

For the DATA-case, the results obtained by Vanem et al. [16] using contour-based method on the same dataset are indicated by ⋆-symbols. The contour based results were obtained by first fitting a 3-parameter Weibull/conditional lognormal distribution to the (H_s, T_z) dataset, and then using the most probable maximum over selected sea states along the 25-year contour line.

As seen from the figures, the iterative method converges quite rapidly to the "exact" results that is obtained by running a much larger number of WASIM simulations. We also note that the method of Vanem et al. [16] that does not take the short-term variability into account significantly underestimates the extreme VBM. Hence, the present results highlights the importance of including the short-term variability of the response. Also, we stress that by applying the active learning approach, with a sequential update of the GP-models, accurate results may be obtained only by a few nonlinear simulations.

The effect of modeling the dependence between the output variables can be assessed by considering the difference between the dotted and solid lines in Fig. 1. Generally, the difference between the two approaches is small, and they converge towards the same results for large number of iterations. Since the noise parameter c^2 is shown to be quite small, and since all observations come from the same locations in the input space (isotopic data), this is not unexpected. This is related to the phenomenon of *autokrigability*, which

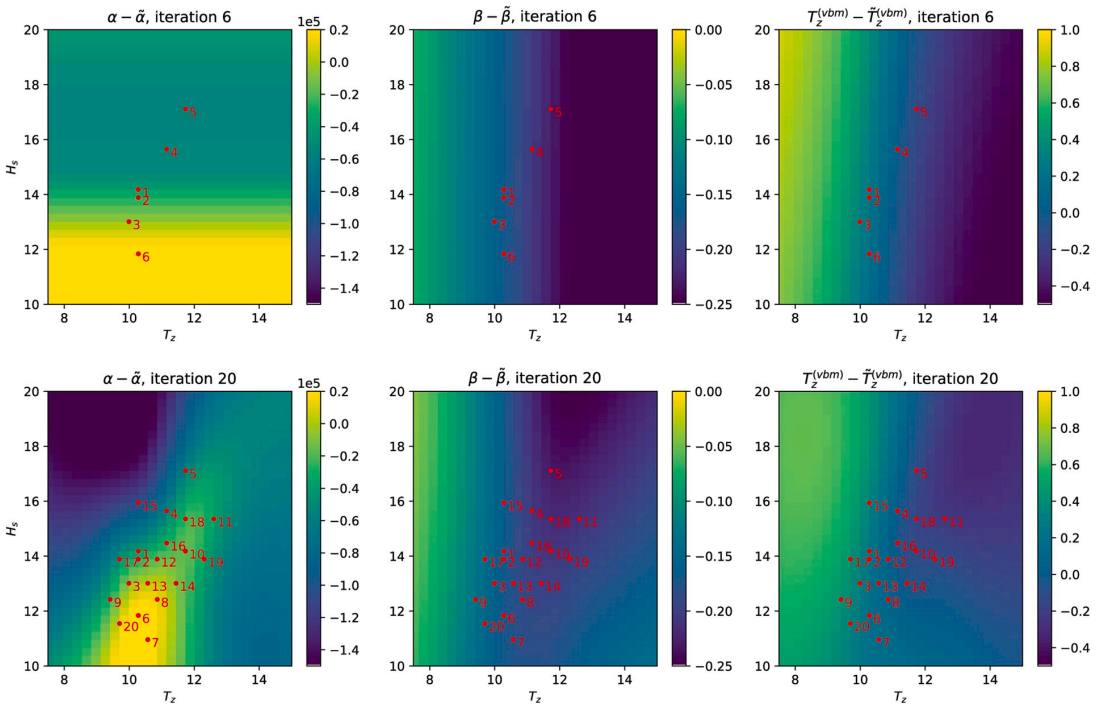


Fig. 2. Fitted GP models $\alpha(H_s, T_z)$, $\beta(H_s, T_z)$ and $T_z^{(vbm)}(H_s, T_z)$ for the NAS-case after 6 (upper) and 20 (lower) iterations. Red dots show the sea states used for the training (fitting) of the models with their order in the sequential approach indicated. Note that the linear results (prior mean) is subtracted so that the plots show the difference from linear results. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

means that when $Q = 1$ in (5), then predictions are equivalent to independent prediction over each output if observations are noise-free and isotopic.

The fitted GP models for $\alpha(H_s, T_z)$, $\beta(H_s, T_z)$ and $T_z^{(vbm)}(H_s, T_z)$ after 6 and 20 iterations are shown in Fig. 2. Note that although the fitted GP surfaces after 6 and 20 iterations differ significantly for H_s-T_z combinations for which there are no training data, the important part of the H_s-T_z is reasonable well represented already after 6 iterations. This is also reflected in the fact that the resulting estimates of the 25-year extreme VBM have converged after about 5–10 iterations, as seen from Fig. 1.

An advantage of the present sampling approach, where the extreme value is estimated from a large number (here $N = 1000$) realization of the considered 25-year period, is that one also obtains information about the distribution of the considered extreme value. The estimated distributions of the 25-year extreme VBM are shown in Fig. 3. Again, we observe good agreement with the "exact" results after six iterations. We observe a slightly better agreement in the distributions after six iterations for the DATA case than for the NAS case. This may be explained by the fact that for the DATA-case a dataset of sea states are used directly, and hence sampling from this dataset can only yield sea states present in the dataset. For the NAS-case, however, sampling is performed from a parametric distribution, which means that in some of the $N = 1000$ realizations of the 25-year period will give extreme sea states for which the GP model is more inaccurate.

As mentioned previously, several methods have been suggested to correct for the lack of short-time variability in e.g. contour based methods. Such approaches are also mentioned in DNV GL recommended practice [2] and the NORSOK standard [24]. One approach is to use a higher percentile of the response distribution instead of using the mode or the median. To investigate this further, we have run the method of Vanem et al. [16] that does not take the short-term variability into account and that was shown to underestimate the extreme VBM significantly, but replaced the most probable maximum with the 90% percentile of the 3-h maximum VBM. The results

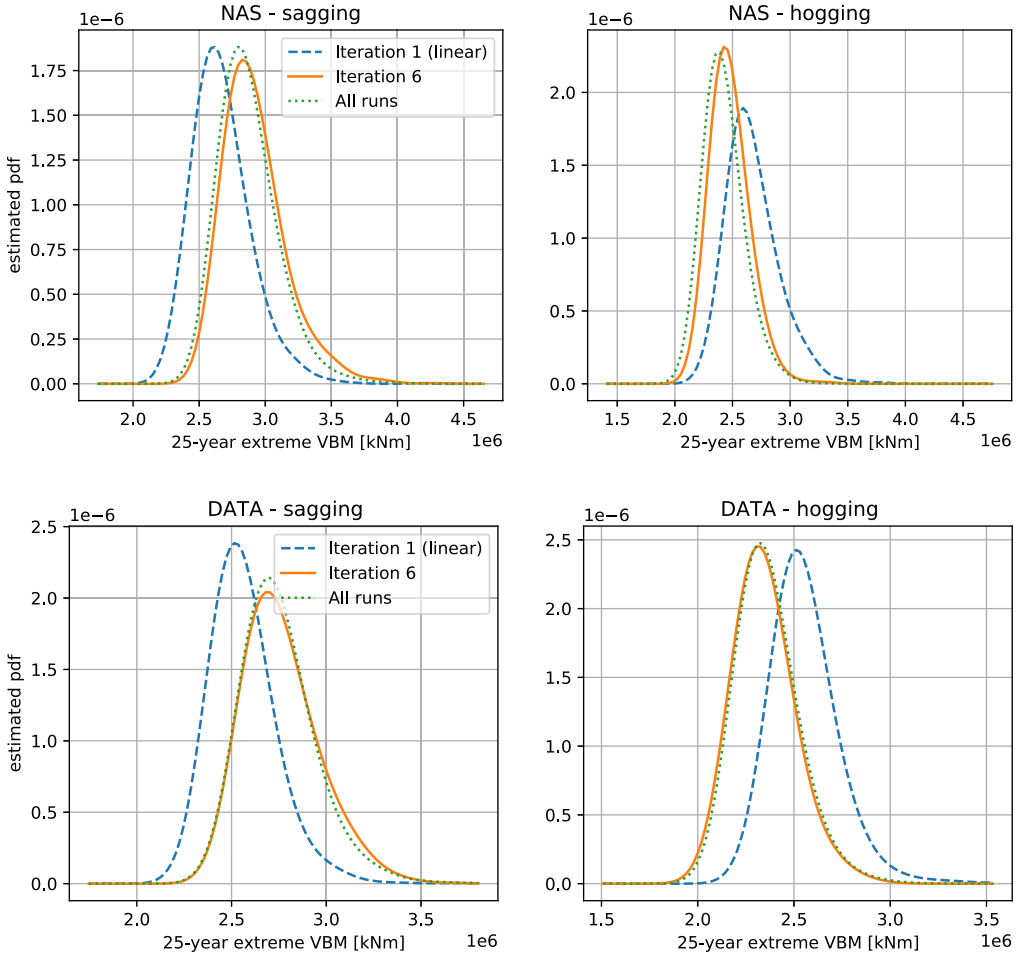


Fig. 3. Estimated distributions of the 25-year extreme VBM for the NAS (upper) and DATA (lower) cases, based on $N = 1000$ realization of the 25-year period, for iteration 1 (linear prior), iteration 6 as well as the result obtained by using all 95 WASIM runs.

are shown in Fig. 4. As seen from the figure, when using the 90% percentile, the results are much closer to the results of the full sampling approach. However, the agreement depend from case to case, and the DATA case is still somewhat underestimated when using the 90% percentile.

4.4.1. The effect of including uncertainty in the distribution parameters

In the above results, the distribution parameters α, β and $T_z^{(vbm)}$ are for each sea state taken as the predictive mean of the GP models, corresponding to (9). This has the consequence that the uncertainty of the distribution parameters are not included in the sampling approach itself, but only in the decision criterion (11) for the next sea state for which to include in the training of the GP models. However, since the GP models also provide the variance of the distribution parameters for each sea state, an alternative is to use random samples from the GP models instead of using the predictive mean. The effect of this on the 25-year extreme VBM is shown in Fig. 5. As seen from the figure, this mainly influence the beginning of the iteration process, likely where the uncertainty of the GP estimates is larger.

4.4.2. Effect of linear prior

In the results presented above the prior information provided by the linear analysis is utilized in the fitting of the GP models, in the sense that the linear results is used as the prior mean in the GP models. This ensures a good initial guess for the first steps of the iteration process and also provides reasonable results for sea states far from existing training data. However, in some cases good prior information may no be available. In such cases, a natural approach is to fit the GP models directly on the nonlinear results. Then, the GP models should be initialized by results from a few selected sea states.

The effect of initializing the GP-models by four selected sea states instead of using the linear model as a prior is shown in Fig. 6, which shows the result with and without the linear prior. Note that in the case of no linear prior, iteration 1 corresponds to using GP

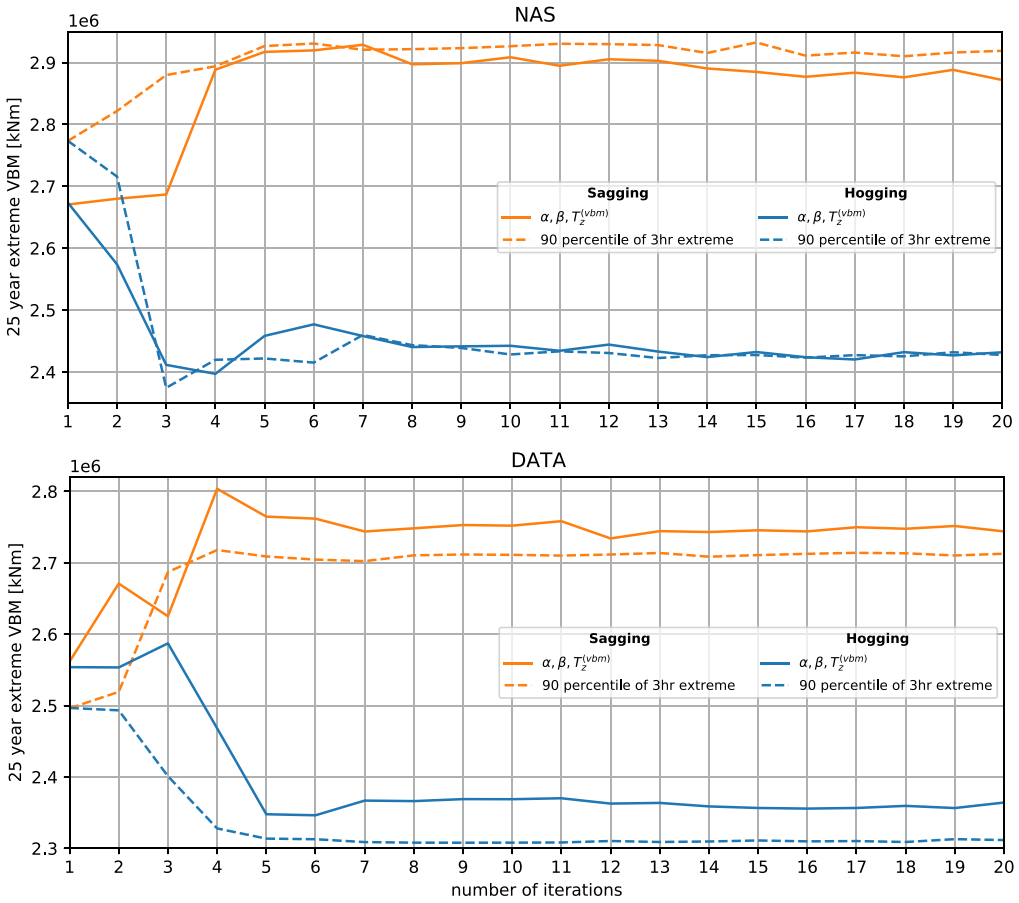


Fig. 4. Estimated mean 25-year extreme VBM at different iteration steps, for hogging and sagging, for the NAS (upper) and DATA (lower) cases, using the full sampling approach (solid lines) compared to the results using the 90% percentile of the 3-h extreme distribution (dashed line).

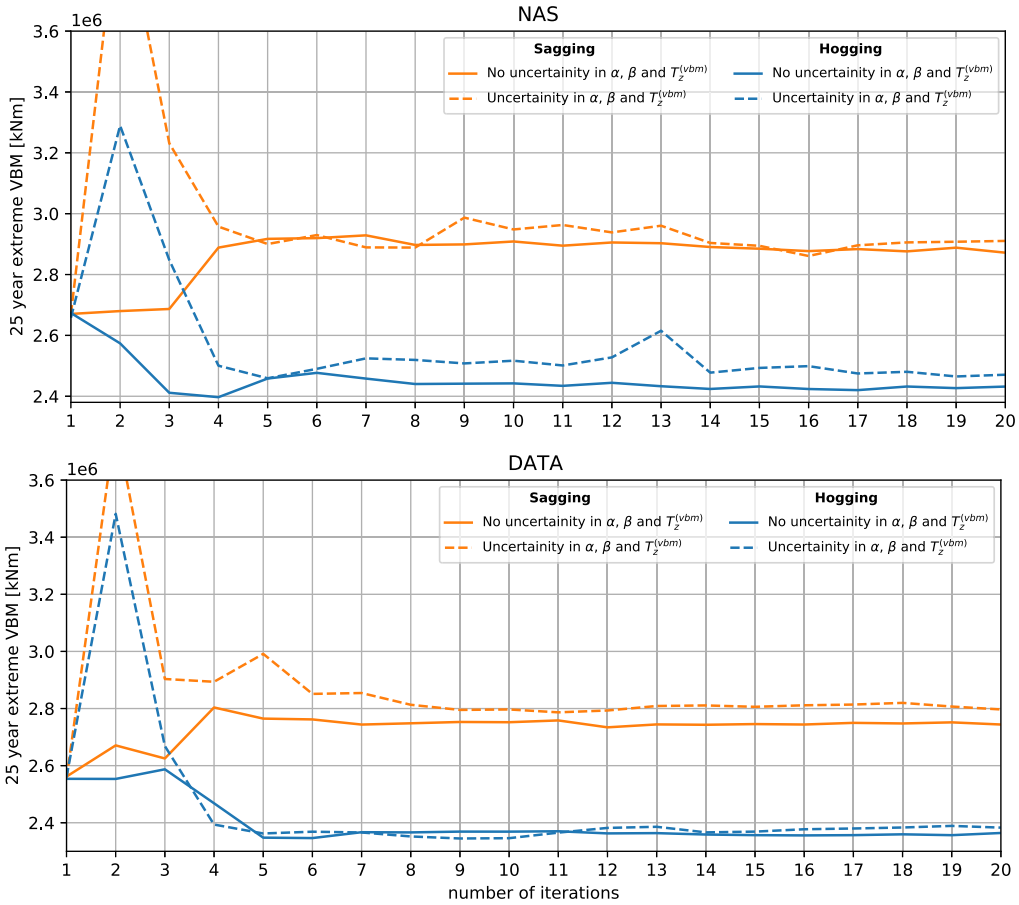


Fig. 5. Estimated mean 25-year extreme VBM at different iteration steps, for hogging and sagging, with and without taking the uncertainty in the distribution parameters into account. That is, using the predictive mean of the GP model versus random sampling from the GP models. Horizontal lines show the corresponding "exact" results where all 95 WASIM runs are used to fit the GP models.

models trained on the following four sea states $(H_s = 10, T_z = 8), (H_s = 10, T_z = 12), (H_s = 16, T_z = 10), (H_s = 16, T_z = 14)$.

As is clearly seen from Fig. 6, the effect of using the linear prior information is small, and good convergence is obtained when initializing the GP model with four initial points rather than using the linear prior as the GP mean. This shows that for the application considered here, the proposed methodology does not rely on the existence of a good linear approximation of the system. However, we note that it might still be reasonable to include a (computationally cheap) prior model for quality assurance, as disagreement with the prior may reveal issues with the (expensive) model. But one should not expect a significant performance increase, in terms of the number of simulations needed.

5. Discussion and conclusions

This paper considers the estimation of long term extreme response statistics for marine structures. A new approach is presented for which the underlying short-term response is described by some chosen probability distribution, and where the parameters of the distribution, as functions of environmental parameters, are represented using Gaussian process regression models. The proposed method has two main advantages over commonly used alternative methods, such as contour-based methods. Firstly, the short-term variability of the response is fully taken into account by applying repeated sampling from both the underlying long-term distribution of the environment and the short-term distribution of the response. Secondly, the Gaussian process approach allows for a sequential update of the surfaces describing the model parameters as functions of environmental parameters, with the model automatically suggesting a new point for which to obtain an accurate sample of the response statistics, which typically means performing a time-domain hydrodynamic simulation.

The methodology is applied to a case study of estimating the 25-year extreme wave induced vertical bending moment of a ship. We

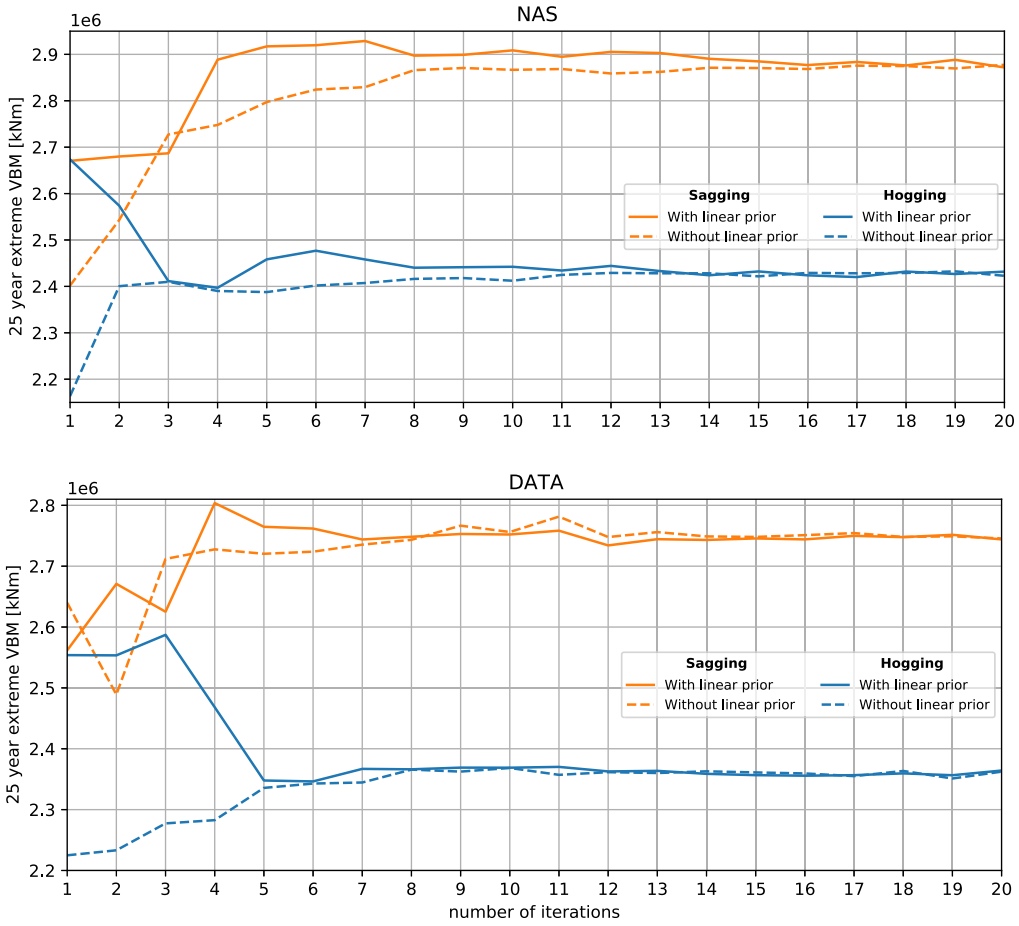


Fig. 6. Estimated mean 25-year extreme VBM at different iteration steps, for hogging and sagging, with and without using the linear results as prior mean in the GP models.

show that the proposed method converges in just a few iterations to the "exact" result obtained from running a large number of hydrodynamic time-domain simulations over the entire scatter diagram. This means that the proposed method is comparable to e.g. contour methods, when it comes to computational cost.

Comparison with Vanem et al. [16], which also uses GP-regression to represent the short term response as a function of sea state parameters, but that does not consider the short-term variability, shows that not including the short-term variability leads to significant underestimation of the extreme response. This result is well known, and we also confirm that the lack of short-term variability can be compensated for by using a higher percentile of the response statistics. For the present case study, we show that by using the 90% percentile of the distribution of 3-h maxima yield results much closer to the analysis taking the short-term variability into account. However, the actual percentile to use will be case specific, and is difficult to determine a priori. The method proposed herein provides the short-term variability explicitly, with little added computational cost, as the bottleneck of the analysis is still the hydrodynamic sea-keeping simulations.

Finally, as a numerical model of the dynamical system is not the ground truth, one might foresee the possibility of combining numerical simulations with measurements from full-scale or laboratory tests. From the relevant literature dealing with this scenario, see for instance Refs. [13,34,35], the natural way forward would be to extend the probabilistic surrogate model presented in this paper with an additional Gaussian Process, to capture the discrepancy between numerical simulations and the real physical phenomenon.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The work presented in this paper has been carried out within the research project ECSADES, with support from the Research Council of Norway (RCN) under the MARTEC II ERA-NET initiative; project no. 249261/O80, and also support from grant 276282 from RCN.

References

[1] Winterstein SR, Ude TC, Bazzurro P, Cornell CA. Ocean environment contours for structural response analysis and experiment design. In: Probabilistic mechanics & structural reliability; 1996.
[2] DNV GL. Recommended practice DNVGL-RP-C205. DNV GL: Environmental Conditions and Environmental Loads; 2019.
[3] Derbanne Q, de Hauteclocque G, Dumont M. How to account for short-term and long-term variability in the prediction of the 100 Years response?. In: 36th international conference on ocean, offshore and arctic engineering (OMAE), Madrid, Spain; 2017. p. 57656. https://doi.org/10.1115/omae2017-61701. V03AT02A023.
[4] Winterstein S, Ude T, Cornell C, Bjerager P, Haver S. Environmental parameters for extreme response: inverse FORM with omission factors. In: Proc. 6th international conference on structural safety and reliability; 1993.
[5] Madsen HO. Omission sensitivity factors. Struct Saf 1988;5:35–45.
[6] Winterstein SR, Engebretsen K. Reliability-based prediction of design loads and responses for floating ocean structures. In: Proc. 17th international conference on offshore mechanics and arctic engineering (OMAE), Lisbon, Portugal; 1998.
[7] Baarholm GS, Moan T. Estimation of nonlinear long-term extremes of hull girder loads in ships. Mar struct 2000;13(6):495–516. https://doi.org/10.1016/S0951-8339(00)00060-5. ISSN 0951-8339, http://www.sciencedirect.com/science/article/pii/S0951833900000605.
[8] MacKay DJC. Information-based objective functions for active data selection. Neural Comput 1992;4(4):590–604. https://doi.org/10.1162/neco.1992.4.4.590. ISSN 0899-7667.
[9] Seo Sambu, Wallat M, Graepel T, Obermayer K. Gaussian process regression: active data selection and test point rejection. In: Proceedings of the IEEE-INNS-ENNS international joint conference on neural networks. IJCNN 2000, vol. 3. Neural Computing: New Challenges and Perspectives for the New Millennium; 2000. p. 241–6. https://doi.org/10.1109/IJCNN.2000.861310. 1098-7576.
[10] Ranjan P, Bingham D, Michailidis G. Sequential experiment design for contour estimation from complex computer codes. Technometrics 2008;50. https://doi.org/10.1198/004017008000000541.
[11] Brochu E, Cora VM, De Freitas N. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. 2010. CoRR abs/1012.2599.
[12] Sacks J, Welch WJ, Mitchell TJ, Wynn HP. Design and analysis of computer experiments. Stat Sci 1989;4(4):409–23. https://doi.org/10.1214/ss/1177012413.
[13] Kennedy MC, O'Hagan A. Bayesian calibration of computer models. J Roy Stat Soc B 2001;63(3):425–64. https://doi.org/10.1111/1467-9868.00294. https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/1467-9868.00294.
[14] Mohamad MA, Sapsis TP. Sequential sampling strategy for extreme event statistics in nonlinear dynamical systems. Proc Natl Acad Sci Unit States Am 2018;115(44):11138–43. https://doi.org/10.1073/pnas.1813263115. ISSN 0027-8424, https://www.pnas.org/content/115/44/11138.
[15] Wang A, Huang S, Barltrop N. Long term extreme analysis of FPSO mooring systems based on Kriging metamodel. In: Proceedings of the ASME 2014 33rd international conference on offshore mechanics and arctic engineering (OMAE), 1B. American Society of Mechanical Engineers (ASME); 2014. https://strathprints.strath.ac.uk/52690/.
[16] Vanem E, Guo B, Ross E, Jonathan P. Comparing different contour methods with response-based methods for extreme ship response analysis. Mar Struct 2020;69:102680. https://doi.org/10.1016/j.marstruc.2019.102680. ISSN 0951-8339, http://www.sciencedirect.com/science/article/pii/S0951833919303958.
[17] Williams CK, Rasmussen CE. Gaussian processes for regression. In: Touretzky Ds, Mozer Mc, Hasselmo ME, editors. Advances in neural information processing systems, vol. 8. MIT Press; 1996, ISBN 9780262201070.
[18] Rasmussen CE, Williams CKI. Gaussian processes for machine learning. The MIT Press; 2006, ISBN 026218253X.
[19] Álvarez AM, Rosasco L, Lawrence ND. Kernels for vector-valued functions: a review. Found Trends Mach Learn 2012;4. https://doi.org/10.1561/2200000036.
[20] Ghanem R, Owhadi H, Higdon D. Handbook of uncertainty quantification. 2017. https://doi.org/10.1007/978-3-319-12385-1.
[21] Bect J, Ginsbourger D, Li L, Picheny V, Vazquez E. Sequential design of computer experiments for the estimation of a probability of failure. Stat Comput 2012;22(3):773–93. https://doi.org/10.1007/s11222-011-9241-4. ISSN 1573-1375.
[22] Hørte T, Wang G, White N. Calibration of the hull girder ultimate capacity criterion for double hull tankers. In: 10th international symposium on practical design of ships and other floating structures. Houston, Texas: United States of America; 2007.
[23] Soares CG, Fonseca N, Pascoal R. Long term prediction of non-linear vertical bending moments on a fast monohull. Appl Ocean Res 2004;26(6):288–97. https://doi.org/10.1016/j.apor.2005.08.002. ISSN 0141-1187, http://www.sciencedirect.com/science/article/pii/S0141118705000313 [advanced Methods to Predict Wave-Induced Loads for High-Speed Ships].
[24] Norsok, Norsok Standard N-003. Action and action effects. third ed. 2017. 2017.
[25] Bitner-Gregersen EM. Joint met-ocean description for design and operations of marine structures. Appl Ocean Res 2015;51:279–92. ISSN 0141-1187, http://www.sciencedirect.com/science/article/pii/S0141118715000085.
[26] Bitner-Gregersen E, Cramer E, Korbijn F. Environmental description for long-term load response of ship structures. In: Proceedings of ISOPE-95 conference; 1995.
[27] Aarnes OJ, Reistad M, Breivik Ø, Bitner-Gregersen E, Eide LI, Gramstad O, Magnusson AK, Natvig B, Vanem E. Projected changes in significant wave height toward the end of the 21st century: northeast Atlantic. J Geophys Res Oceans 2017;122(4):3394–403. https://doi.org/10.1002/2016JC012521. ISSN 2169-9291.
[28] DNV GL, SESAM User Manual–WASIM. Wave loads on vessels with forward speed, revision 8, valid from wasim version 5. 3 edn. 2015.
[29] Guo B, Bitner-Gregersen E, Sun H, Helmers JB. Statistical analysis of ship response in extreme seas. Ocean Eng 2016;119:154–64.
[30] West BJ, Brueckner KA, Janda RS, Milder DM, Milton RL. A new numerical method for surface hydrodynamics. J Geophys Res 1987;92(C11):11803–24. https://doi.org/10.1029/JC092iC11p11803. ISSN 2156-2202.
[31] Bateman WJD, Swan C, Taylor PH. On the calculation of the water particle kinematics arising in a directionally spread wavefield. J Comput Phys 2003;186(1):70–92. ISSN 0021-9991, http://www.sciencedirect.com/science/article/pii/S0021999103000123.
[32] Guo B, Gramstad O, Vanem E, Bitner-Gregersen E. Study on the effect of climate changes on ship responses based on nonlinear simulations. J Offshore Mech Arctic Eng 2019;141:1–13. 041605.
[33] GPy GPy. A Gaussian process framework in python. http://github.com/SheffieldML/GPy. since 2012.
[34] Vernon I, Goldstein M, Bower R. Galaxy formation: Bayesian history matching for the observable universe. Stat Sci 2014;29(1):81–90.
[35] Jones MJ, Hansen HF, Zeeberg AR, Randell D, Jonathan P. Uncertainty quantification in estimation of ocean environmental return values. Coast Eng 2018;141:36–51.

Paper III

Sequential Bayesian optimal experimental design for structural reliability analysis

Christian Agrell, Kristina Rognlien Dahl

Statistics and Computing (2021) Vol. 31 no. 27.

III

Statistics and Computing (2021) 31:27
https://doi.org/10.1007/s11222-021-10000-2

Sequential Bayesian optimal experimental design for structural reliability analysis

Christian Agrell[1,2] · **Kristina Rognlien Dahl**[1]

Received: 1 July 2020 / Accepted: 1 February 2021

Abstract
Structural reliability analysis is concerned with estimation of the probability of a critical event taking place, described by $P(g(\mathbf{X}) \leq 0)$ for some n-dimensional random variable $\mathbf{X}$ and some real-valued function g. In many applications the function g is practically unknown, as function evaluation involves time consuming numerical simulation or some other form of experiment that is expensive to perform. The problem we address in this paper is how to optimally design experiments, in a Bayesian decision theoretic fashion, when the goal is to estimate the probability $P(g(\mathbf{X}) \leq 0)$ using a minimal amount of resources. As opposed to existing methods that have been proposed for this purpose, we consider a general structural reliability model given in hierarchical form. We therefore introduce a general formulation of the experimental design problem, where we distinguish between the uncertainty related to the random variable $\mathbf{X}$ and any additional epistemic uncertainty that we want to reduce through experimentation. The effectiveness of a design strategy is evaluated through a measure of residual uncertainty, and efficient approximation of this quantity is crucial if we want to apply algorithms that search for an optimal strategy. The method we propose is based on importance sampling combined with the unscented transform for epistemic uncertainty propagation. We implement this for the myopic (one-step look ahead) alternative, and demonstrate the effectiveness through a series of numerical experiments.

Keywords Optimal experimental design · Structural reliability · Probability of failure · Epistemic and aleatory uncertainty · Unscented transform

1 Introduction

In order to ensure sufficient reliability of engineered systems, such as buildings, ships, offshore structures, aircraft or technological products, uncertainties with respect to the system's capabilities and the system's environment must be accounted for. In probabilistic structural reliability analysis, this is achieved through a probabilistic modelof the system and its environment. A primary objective with such a model is to estimate the probability that the system will fail (e.g. collapse, sink, crash or explode).[1]

A probabilistic structural reliability model is commonly defined through a *performance function* (also called a *limit-state function*) $g(\mathbf{X})$ depending on some random variable $\mathbf{X}$. Here, $g(\mathbf{X}) < 0$ corresponds to system failure, and $g(\mathbf{X}) \geq 0$ corresponds to the system functioning. Typically, $\mathbf{X}$ contains the parameters describing a particular structure, such as the geometry, dimensions and material properties. These quantities may be random, but can be influenced by the designer of the structure. For example, the designer may choose to use a more expensive, but more durable material in order to improve the structural properties of the system. In addition, $\mathbf{X}$ contains the (random) parameters that characterize the sys-

Christian Agrell
chrisagr@math.uio.no; christian.agrell@dnvgl.com

Kristina Rognlien Dahl
kristrd@math.uio.no

[1] Department of Mathematics, University of Oslo, Oslo, Norway

[2] DNV GL Group Technology and Research, Høvik, Norway

[1] This is rarely interpreted as a frequentist probability. As the model is not the real world, it is common to design models such that the failure probability can be interpreted as a conservative estimate, or as a consistent measure of robustness for comparison with other 'acceptable' systems.

Published online: 09 March 2021

tems environment, such as wind speed, wave height etc., and parameters describing how well the model fits reality (model uncertainties). Given $\mathbf{X}$ and the function $g(\cdot)$, the *probability of failure* is defined as the probability $P(g(\mathbf{X}) < 0)$. Modern engineering requirements for safe design and operation of such systems are usually given as an upper bound on this probability (Madsen et al. 2006).

Hence, for many practical applications, the failure probability computation is an important task. This is often challenging for complex systems, as a computationally feasible stochastic model of the complete system and its environment is not available. To capture this in our modelling framework, we consider additional *epistemic* uncertainties, i.e. uncertainties due to limited data or knowledge that *in principle* can be reduced by gathering more information.

1.1 Epistemic and aleatory uncertainty

The concept of epistemic uncertainty is commonly used in uncertainty quantification (UQ) and in reliability analysis. One often considers two different kinds of uncertainty: Aleatory (stochastic) and epistemic (knowledge-based) uncertainty. Aleatory uncertainty is uncertainty which cannot be removed by collecting more or better information. For instance, the result of throwing a dice is an example of aleatory information, because there is a range of possible outcomes even if we understand the experimental setup. Epistemic uncertainty, on the other hand, is uncertainty which can be affected by collecting more and/or better information. For example, if a quantity or parameter has a definite value, but this value is unknown to us, then the uncertainty considered epistemic. Likewise, uncertainty about the form of a model for a physical phenomenon is epistemic, because more research or experiments could be performed to improve the model.

We note that this characterization of uncertainties will have to depend on the relevant modelling and decision-making context. Given the aleatory example of throwing a dice, one could argue that given sufficient information about initial conditions together with a detailed physics model, it should be possible to predict the outcome (and the uncertainty is therefore epistemic). But based on the modelling context this may not be relevant or a realistic assumption at all. See for instance (Der Kiureghian and Ditlevsen 2009) for a broader discussion.

The following example illustrates that a random variable may contain both epistemic and aleatory uncertainty.

Example 1 Consider two experiments:

Experiment 1 Consider a fair dice that is to be thrown, and denote the outcome $\mathbf{A}$. Since the distribution of dice throwing is known ($P(\mathbf{A} = i) = 1/6$ for i=1, …, 6), the uncertainty in the random variable $\mathbf{A}$ is (purely) aleatory.

Experiment 2 Consider another dice that has been thrown, but where the dice has been covered so that the result is not visible. Now there is uncertainty about the value of the hidden dice. Call this random variable $\mathbf{E}$. The uncertainty in $\mathbf{E}$ is (purely) epistemic because it could be reduced by gathering more information (removing the cover from the dice).

Assume the (random) quantity of interest is the sum, $\mathbf{S} = \mathbf{A} + \mathbf{E}$, of the result of the two die. If $\mathbf{E}$ is given, then the remaining uncertainty is the aleatory (stochastic) uncertainty in throwing a dice. Without knowing the value of the hidden dice, the uncertainty in the sum $\mathbf{S}$ is both aleatory and epistemic.

In Example 1 we had the option of uncovering the second dice, an *experiment* that would remove all epistemic uncertainty in $\mathbf{S}$. Generally, we will consider experiments that reduces (but not necessarily completely removes) epistemic uncertainty. For instance, in the context of Example 1, an experiment that would reveal whether $\mathbf{E}$ was an even or odd number, or whether $\mathbf{E} > 1$. Or, the sum $\mathbf{S}$ (but not the value of $\mathbf{A}$) from a few repeated throws of the aleatory die, from which inference about $\mathbf{E}$ could be made. Reducing epistemic uncertainty usually comes at a cost, where the more informative experiments are more expensive. In this paper we are interested in how to decide on which experiments to perform, where the cost and potential effect of experiments are balanced in an optimal manner.

In the context of structural reliability modelling, the epistemic uncertainty usually comes from one of the two reasons:

1. The function $g(\cdot)$ or the distribution of $\mathbf{X}$ depends on parameters that we do not know the value of.
2. Evaluating $g(\mathbf{x})$ at some single realization $\mathbf{x}$ of $\mathbf{X}$ is *expensive* in terms of money and/or time.

The second part comes from the complex physical nature of failure mechanisms, where experiments are needed to evaluate the function $g(\mathbf{x})$. This includes numerical computer simulations and physical experiments in a laboratory, which are both time consuming and expensive. Hence, due to the limited number of experiments that can be performed in practice, any method for estimating $P(g(\mathbf{X}) < 0)$ that relies on a large number of evaluations of $g(\cdot)$ is practically infeasible. This problem is usually solved by replacing the performance function $g(\cdot)$ with a computationally cheap *surrogate model* or *emulator*[2], constructed from a small set of experiments.

[2] The word *emulator* is often used for a surrogate model that can interpolate between noiseless observations coming from a deterministic computer simulation.

When the surrogate model is a stochastic process (viewed as a distribution over functions), we can quantify the added epistemic uncertainty that comes from this simplification.

We will assume that epistemic uncertainty is introduced in a structural reliability model, and that there is a way to reduce this uncertainty by performing experiments.

The problem we address in this paper is how to optimally estimate $P(g(\mathbf{X}) < 0)$ using as little resources as possible. In particular, we want to find an optimal strategy for the scenario where we can perform experiments sequentially, i.e. where each experiment may depend on the preceding ones.

Remark 1 (Why separate between epistemic and aleatory uncertainties?) Note that if there is no epistemic uncertainty in our model, then there is no incentive for performing experiments to collect more information, since the uncertainty cannot be reduced no matter what experiment we do. Hence, for our problem formulation to make sense, it is crucial to know that there is epistemic uncertainty present. In Example 1 we can consider the conditional probability $P(\mathbf{S}|\mathbf{E} = \mathbf{e})$, which for any fixed realization $\mathbf{e}$ is a property of the aleatory uncertainty alone. When we do not know the value of $\mathbf{E}$, the quantity $P(\mathbf{S}|\mathbf{E})$ becomes a random variable of purely epistemic uncertainty. We are going to treat the failure probability $P(g(\mathbf{X}) < 0)$ in this way, by conditioning on epistemic information, so that we can study the potential effect of experiments.

Furthermore, Der Kiureghian and Ditlevsen (2009) show that by not separating between these two types of uncertainty in risk and reliability assessment, one may either over- or underestimate the failure probability by a significant magnitude (depending on the problem at hand), and conclude that distinguishing between aleatory and epistemic uncertainty in risk assessment is important. This is also supported by the examples we present in Sect. 6.

1.2 Hierarchical modelling

The scenario where $g(\cdot)$ is replaced by a surrogate model created from a finite set of observations $\{g(\mathbf{x}_i)\}_{i=1}^n$ has already been studied extensively (Bect et al. 2012; Echard et al. 2011; Bichon et al. 2008; Sun et al. 2017; Jian et al. 2017; Perrin 2016; Schueremans and Gemert 2005). The most common approach is to approximate $g(\cdot)$ using a Gaussian process, and make use of the convenient fact that a surrogate model given by the posterior predictive distribution of the Gaussian process has a closed form solution. However, structural reliability models are often hierarchical, and the reason why $g(\cdot)$ is expensive comes from one or more expensive *sub-components*[3]. An example is shown in Fig. 1, where

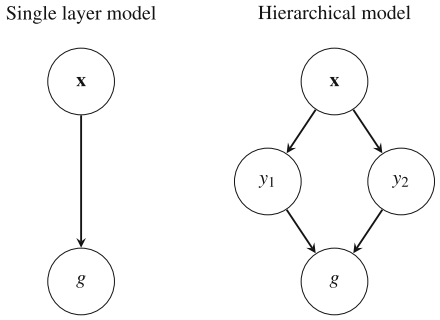


Fig. 1 Left: Single layer model. Right: Example of an hierarchical (2 layer) model where $g(\mathbf{x}) = g(y_1(\mathbf{x}), y_2(\mathbf{x}))$

$g(\mathbf{x}) = g(y_1(\mathbf{x}), y_2(\mathbf{x}))$. Assume here that $\mathbf{x} \in \mathbb{R}^m$, then the index set of the Gaussian process approximation of $g(\mathbf{x})$ is m-dimensional. Naturally, the number of experiments needed is highly dependent on m. If $g(\mathbf{x})$ is expensive, then this must be because one (or more) of the functions, $y_1(\mathbf{x})$, $y_2(\mathbf{x})$ or $g(y_1, y_2)$ is expensive. Very often, the effective domains[4] of these functions have dimensionality much smaller than m, so fitting a Gaussian process to observations of $g(\mathbf{x})$ is not very efficient. There is also some practical inconvenience here, which is that some of the expensive sub-components (for instance load models) may be applicable in different structural reliability models, so there is a potential for re-use if we create a surrogate model for, say $y_1(\mathbf{x})$, instead of $g(\mathbf{x})$. Kyzyurova et al. (2018) also consider a similar scenario and give some examples, for the 2-layer case where each component is replaced by a Gaussian process emulator.

In this paper we will work with hierarchical models (not necessarily with the structure illustrated in Fig. 1), where we assume that some of the intermediate variables are stochastic processes with epistemic (potentially reducible) uncertainty. Note that this also covers the case where we just introduce additional epistemic variables into the model. Actually, in the approximate numerical solution we propose in this paper, these two problems become equivalent. Moreover, as Gaussianity generally is lost in the hierarchical setting, we will only make assumptions on existence of second order moments of the stochastic processes used as surrogates. We will present a general formulation of the problem of finding an optimal strategy for performing experiments based on Bellman's principle of optimality, and discuss some alternative routes for solving such problems. For the myopic (one step look-ahead) strategy, we propose an efficient numerical procedure, based on finite-dimensional approximation of the

[3] For instance, $g(\mathbf{x})$ is often a function of a structures capacity and the effect of loads acting on the structure, where each of which are determined from separate types of experiments.

[4] If for instance $y_1(\mathbf{x}) : \mathbb{R}^m \to \mathbb{R}$ depends only on $x_1, \ldots, x_n$ for $n \leq m$, the effective domain of y_1 is n-dimensional.

stochastic processes and uncertainty propagation using the unscented transform.

1.3 Structure and main contributions of the paper

The structure of the remaining part of the paper is as follows: Through Sects. 2 and 3 we develop the Bayesian optimal experimental design problem for a general structural reliability model. We introduce a framework for separation of aleatory and epistemic uncertainties using conditional expectations, from which we can express any type of experiment associated with a structural reliability problem. For the purpose of estimating a failure probability, we consider three alternative optimization objectives, and in Sect. 3 we discuss how the experimental design problem may be tackled using dynamic programming and the one-step lookahead approximation. Optimization problems of this form will involve evaluation of a *measure of residual uncertainty*, and in Sect. 4 we present an approach for approximating this quantity. We implement this in Sect. 5 to develop an efficient numerical procedure for the one-step lookahead case, which we illustrate through a series of examples in Sect. 6. Finally, our concluding remarks are given in Sect. 7, and some supporting material used throughout the paper is included in the Appendices.

2 Problem formulation

Given a probabilistic surrogate of a structural reliability model, we are interested in how to optimally improve the model for failure probability estimation, given a fixed experimental budget. More generally, given a structural reliability model with epistemic uncertainty (e.g. as introduced when using a surrogate), and a set of possible experiments than can be performed, we want to select the experiments in an optimal manner. The choice of experiment is called a *decision*, $d \in \mathbb{D}$ where $\mathbb{D}$ is a space of feasible decisions. Note that this set may include different kinds of decisions, such as performing computer experiments, lab experiments or performing physical measurements in the field.

In the following subsections we present a rigorous formulation of the Bayesian optimal experimental design problem for structural reliability analysis. Here we will need a way to express uncertainty about the *performance function* used in structural reliability models, and a way to model uncertainty about future outcomes of potential experiments that can be made. For this purpose we will define a model (ξ, δ), where

- ξ is a stochastic representation of the *performance function* $g(\mathbf{x})$ evaluated at some fixed input $\mathbf{x}$.
- $\delta(d)$ is a predictive model of experimental outcomes given a decision d. In other words, δ models the data generating process of potential experiments.

We will consistently write $\mathbf{X}$ as a random variable with values in $\mathbb{X} \subseteq \mathbb{R}^m$, and let $\mathbf{x}$ be a deterministic realization. ξ and δ are stochastic processes, indexed over inputs $\mathbf{x}$ and decisions d, respectively. In structural reliability analysis, we are interested in the random variable $g(\mathbf{X})$, and likewise we will consider $\xi(\mathbf{X})$, but now where $\xi(\mathbf{x})$ is also random for any fixed $\mathbf{x}$. Here, for notational convenience, we suppress the $\omega \in \Omega$ when referring to the random variable $\mathbf{X} : \Omega \to \mathbb{X}$ or the stochastic process $\xi(\mathbf{x}) : \mathbb{X} \times \Omega \to \mathbb{R}$. That is, we define the notation $\xi(\mathbf{X}) := \xi(\mathbf{X}(\omega), \omega)$ to describe the random variable $\xi(\mathbf{X}) : \Omega \to \mathbb{R}$.

Remark 2 Note that $\xi(\mathbf{x})$ is a stochastic representation of the performance function $g(\cdot)$. When making decisions d, we aim to reduce the uncertainty in $\xi(\mathbf{X})$, where also the input $\mathbf{X}$ is random. Hence, the process $\delta(d)$ is linked with $\xi(\mathbf{X})$ through its reduction of uncertainty (see Sect. 2.3).

As the purpose of performing experiments will be to provide information about ξ, note that ξ and δ are generally not independent. A detailed description of how (ξ, δ) is constructed is provided in the following subsections.

2.1 Structural reliability analysis

Let $\mathbb{X} \subseteq \mathbb{R}^m$, and let $\mathbf{X}$ be a random variable on the probability space $(\Omega, \mathscr{F}, P)$ with values in $\mathbb{X}$ and $g : \mathbb{X} \to \mathbb{R}$ a measurable function. We call g the *performance function* or *limit state*, with the associated failure set

$$F_g = \{\mathbf{x} \in \mathbb{X} \mid g(\mathbf{x}) \leq 0\}.$$

In structural reliability analysis, we are interested in estimating the *failure probability*, which we here denote $\bar{\alpha}$. It is defined as

$$\bar{\alpha}(g) = P(F_g) = E\left[\mathbf{1}\left(g(\mathbf{X}) \leq 0\right)\right], \tag{1}$$

where $E[\cdot]$ denotes the expectation with respect to P and $\mathbf{1}(\cdot)$ is the indicator function.

In most real-world cases it is difficult to derive an analytical expression for the failure probability. To overcome this, several approximation and simulation methods have been suggested, see e.g. Madsen et al. (2006) or Huang et al. (2017). Two traditional methods are the first- and second-order reliability method (FORM/SORM), where the failure boundary is approximated at a specific point using a Taylor expansion up to the first and second order, respectively. Different sampling procedures have also been developed, which often make use of intermediate results obtained from

FORM/SORM. Other relevant techniques involve the construction of environmental contours and the estimation of buffered failure probabilities as in (Dahl and Huseby 2019). In this paper, our focus is different from these methods in the sense that we are mainly interested in how to estimate the failure probability as well as possible, given a limited experimental budget. To do so, we need to separate between different kinds of uncertainty in our model.

2.2 Separating epistemic and aleatory uncertainties

Ideally, the uncertainty related to the random variable $g(\mathbf{X})$ in (1) is *aleatory*, in the sense that that it relates to inherent variability of the physical phenomenon that is being modelled, but in reality we must also include *epistemic* uncertainty due to lack of information or knowledge. For instance, assume that $g(\mathbf{x}, \mathbf{e})$ depends on the aleatory variable $\mathbf{x}$ and some fixed but unknown parameter $\mathbf{e}$. Assume further that $\mathbf{X}$ is the aleatory random variable representing variability in $\mathbf{x}$, $\mathbf{E}$ is the epistemic random variable representing our belief about $\mathbf{e}$, and that $\mathbf{X}$ and $\mathbf{E}$ are independent with laws P_x and P_e. It is then relevant to view the failure probability as a random quantity with epistemic uncertainty, $\alpha(\mathbf{E}) = \int \mathbf{1}\left(g(\mathbf{x}, \mathbf{E}) \leq 0\right) P_x(d\mathbf{x})$. For engineering applications, one would then typically be interested in some specified upper percentile values of $\alpha(\mathbf{E})$, i.e. ensuring that the epistemic uncertainty is under control.

In the following, we will assume that we have a performance function $\xi(\cdot)$ that depends on a strictly aleatory random variable $\mathbf{X}$, and some other random quantity with epistemic uncertainty. We will need to formulate this with a bit of generality, in order to cover the different ways epistemic uncertainty can be introduced in a structural reliability model.

As in Sect. 2.1 we will work with $(\Omega, \mathscr{F}, P)$ as the global probability space, capturing all forms of uncertainty. We then let $\mathscr{A}$ and $\mathscr{E}$ be two sub σ-algebras representing, respectively, aleatory and epistemic information i.e.,

$$\mathscr{A}, \mathscr{E} \subseteq \mathscr{F}. \tag{2}$$

Though all uncertainty in our model is assumed to be either epistemic or aleatory, Example 1 illustrates that random variables may contain both aleatory and epistemic information.

We will assume that $\mathbf{X}$ is $\mathscr{A}$-measurable. Furthermore, for any $\mathbf{x} \in \mathbb{X}$ we assume that $\xi(\mathbf{x})$ is $\mathscr{E}$-measurable. That is, $\xi : \mathbb{X} \times \Omega \to \mathbb{R}$ is a stochastic process indexed by $\mathbf{x} \in \mathbb{X}$ (this is also called a random field), and $\xi(\mathbf{X})$ is a real-valued random variable. We will write $\xi(\cdot)$ instead of $g(\cdot)$ whenever epistemic uncertainty has been introduced, as for instance in the canonical case where a deterministic performance function $g(\cdot)$ is approximated with a probabilistic surrogate $\xi(\cdot)$.

We can now define the failure probability with epistemic uncertainty as the $\mathscr{E}$-measurable random variable

$$\alpha(\xi) = E\left[\mathbf{1}\left(\xi(\mathbf{X}) \leq 0\right) \mid \mathscr{E}\right]. \tag{3}$$

Note that (3) coincides with (1) in the case where the performance function is not affected by epistemic uncertainty, and in general as $\bar{\alpha}(\xi) = E\left[\alpha(\xi)\right]$ because

$$\begin{aligned} E\left[\alpha(\xi)\right] &= E\left[E\left[\mathbf{1}\left(\xi(\mathbf{X}) \leq 0\right)\right] \mid \mathscr{E}\right] \\ &= E\left[\mathbf{1}\left(\xi(\mathbf{X}) \leq 0\right)\right] \\ &= \bar{\alpha}(\xi), \end{aligned} \tag{4}$$

where the second equality uses the double expectation property.

In the following we will just write α or $\bar{\alpha}$ without the dependency on ξ when there is no risk of confusion.

Example 2 Assume ξ is a deterministic function of the aleatory random variable $\mathbf{X}$ and epistemic random variable $\mathbf{E}$, both defined on $(\Omega, \mathscr{F}, P)$. Then $\mathscr{A} = \sigma(\mathbf{X})$ and $\mathscr{E} = \sigma(\mathbf{E})$, i.e., the σ-algebras generated by the random variables $\mathbf{X}$ and $\mathbf{E}$, respectively.

Note that the converse of Example 2 also holds true, as we can always view ξ as a deterministic function applied to two random variables $\mathbf{X}$ and $\mathbf{E}$. That is, where $\xi(\mathbf{x}, \mathbf{e})$ is a deterministic function for $\mathbf{x}$ and $\mathbf{e}$ fixed, and we can write the stochastic process $\xi(\mathbf{x}, \omega)$ as $\xi(\mathbf{x}, \mathbf{E})$. It is sometimes useful to think of ξ in this way. In particular, the numerical approximation we propose later in this paper is based on obtaining a finite-dimensional approximation of $\mathbf{E}$.

Example 3 Let g be given as in the hierarchical model in Fig. 1, and $\mathbf{X}$ a random variable defined on some measure space $(\Omega_x, \mathscr{F}_x, P_x)$. Assume that y_1 and y_2 are expensive to evaluate, so we replace them with surrogate models in the form of two stochastic processes $\widetilde{y}_1$ and $\widetilde{y}_2$ defined on another measure space $(\Omega_y, \mathscr{F}_y, P_y)$. Note that we assume that both $\widetilde{y}_1$ and $\widetilde{y}_2$ are defined on the same measure space. Then, the measure space for the experimental design problem is given by $(\Omega, \mathscr{F}, P) = (\Omega_x \times \Omega_y, \mathscr{F}_x \otimes \mathscr{F}_y, P_x \times P_y)$, $\mathscr{A} = \mathscr{F}_x$ and $\mathscr{E} = \mathscr{F}_y$ (up to isomorphism), and we would write $\xi(\mathbf{x}) = g(\widetilde{y}_1(\mathbf{x}), \widetilde{y}_2(\mathbf{x}))$.

2.3 Decisions, outcomes and experiments

We are interested in the case where the epistemic uncertainty in α can be reduced by running experiments. For instance, in Example 2 the epistemic variable $\mathbf{E}$ could be a fixed but unknown parameter, and maybe additional measurements could be performed to reduce the uncertainty in $\mathbf{E}$. Or in Example 3, additional experiments could be performed to infer the values of y_1 or y_2 at some given input $\mathbf{x}'$, in order to reduce uncertainty in the surrogate models $\widetilde{y}_1$ and $\widetilde{y}_2$.

These are examples of possible *decisions* we could make to reduce epistemic uncertainty. We will let $\mathbb{D}$ denote the set of all possible decisions, and $\mathbb{O}$ the set of all possible *outcomes*. For any decision $d \in \mathbb{D}$, the corresponding outcome is uncertain a priori, and in order to evaluate the potential impact of a decision we will need to specify (possibly subjectively) a distribution representing the possible outcomes. We will let $\delta(d)$ denote the random outcome of a decision $d \in \mathbb{D}$ with values in $\mathbb{O}$. For any realization $o \in \mathbb{O}$ of $\delta(d)$, we will refer to the pair (d, o) as an *experiment*.

In our modelling framework, we will assume that $\xi(\mathbf{x})$ as defined in Sect. 2.2 is provided together with $(\Omega, \mathscr{F}, P)$ and the sub σ-algebras $\mathscr{A}$ and $\mathscr{E}$, and that a decision process $\delta(d)$ is given where $\delta(d)$ is $\mathscr{E}$-measurable for any $d \in \mathbb{D}$. Table 1 gives an overview of the notation we have introduced so far, in order to define the problem of optimal experimental design for structural reliability analysis.

Example 4 Continuing from Example 3, assume that noise perturbed observations of y_1 can be made. Let $d(\mathbf{x}) = \{\text{observe } y_1(\mathbf{x})\}$, and define $\mathbb{D}$ as the union of such events for all $\mathbf{x}$. If we assume that observations come with additive noise, $o(\mathbf{x}) = y_1(\mathbf{x}) + \epsilon(\mathbf{x})$, for some specified noise process ϵ, then we can let $\delta(d(\mathbf{x})) = \widetilde{y_1}(\mathbf{x}) + \epsilon(\mathbf{x})$. In a similar fashion, $\mathbb{D}$ and $\delta(d)$ could be extended to include observations of y_2 as well.

We will note that the noise-free alternative to Example 4, i.e. the case where $\epsilon \equiv 0$, is a common scenario when dealing with deterministic computer simulations. Another related scenario that is also of relevance here, is that of muiltifidelity modelling (Fernandez et al. 2017), in which case inaccurate estimates of $y_1(\mathbf{x})$ could be available at the same time, but at a lower cost.

2.4 Sequential model updating

Now, having defined a random variable $\mathbf{X}$ and the two processes $\{\xi(\mathbf{x})\}_{\mathbf{x}\in\mathbb{X}}$ and $\{\delta(d)\}_{d\in\mathbb{D}}$, we want to perform a sequence of experiments, $(d_0, o_0), (d_1, o_1), \ldots$, and update ξ and δ accordingly.

We let $I_k := \{(d_0, o_0), \ldots, (d_{k-1}, o_{k-1})\}$ denote the information or history up to the kth experiment, and define $\mathscr{E}_k$ as the σ-algebra generated by $\mathscr{E}$ and I_k. Hence, $\mathscr{E}_k$ is all the information regarding epistemic quantities that is available after k experiments. We introduce the notation $P_k(\cdot)$ and $E_k[\cdot]$ to denote the conditional distribution $P(\cdot \mid \mathscr{E}_k)$ and conditional expectation $E[\cdot \mid \mathscr{E}_k]$ given the updated information $\mathscr{E}_k$. For convenience we define $I_0 = \emptyset$, so that we can use the index $k = 0$ with these definitions for the scenario before any experiment has been made. We will write ξ_k and δ_k as the updated processes $\xi \mid I_k$ and $\delta \mid I_k$ corresponding to P_k. Per definition,

$$(\xi_{k+1}, \delta_{k+1}) = (\xi_k, \delta_k) \mid d_k, o_k = (\xi_0, \delta_0) \mid I_k, d_k, o_k.$$

In the following example, we show how this sequential update can be done via Bayes' theorem.

Example 5 Let $k \in \mathbb{N}$. Assume (ξ, δ) admits a joint probability density at any finite subset of $\mathbb{X} \times \mathbb{D}$ with respect to P_k, which we write $p_k(\xi, \delta)$ for short. E.g. $p_k(\xi)$ means

$$P_k\left(\left(\xi(\mathbf{x}^{(1)}), \ldots, \xi(\mathbf{x}^{(n)})\right) = \left(\xi^{(1)}, \ldots, \xi^{(n)}\right)\right)$$

for some $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)} \in \mathbb{X}$ and $\xi^{(1)}, \ldots, \xi^{(n)} \in \mathbb{R}$. Then $p_k(\xi) = p_0(\xi_k)$, $p_k(\delta) = p_0(\delta_k)$, and the update of the probabilities is done by using Bayes' theorem:

$$\begin{aligned} p_{k+1}(\xi) &= p_k(\xi \mid d_k, o_k) = \frac{p_k(o_k \mid \xi, d_k)\, p_k(\xi)}{p_k(o_k \mid d_k)}, \\ p_{k+1}(\delta) &= p_k(\delta \mid d_k, o_k) = \frac{p_k(o_k \mid d_k, \delta)\, p_k(\delta)}{p_k(o_k \mid d_k)}, \end{aligned} \tag{5}$$

where $p_k(\cdot \mid \cdot)$ is the relevant density with respect to P_k.

Example 6 For a specific problem there will typically be simpler ways of updating the model than the generic formulation given in the previous example. Continuing again from Examples 3 and 4, assume $\delta(d) = \delta(\mathbf{x}, \widetilde{y_1}, \widetilde{y_2})$ corresponds to observing $\widetilde{y_1}(\mathbf{x}) + \epsilon_1(\mathbf{x})$ or $\widetilde{y_2}(\mathbf{x}) + \epsilon_2(\mathbf{x})$. Then $\widetilde{y_1}$ and $\widetilde{y_2}$ can be updated directly, and we let $\xi \mid I_k = g(\widetilde{y_1} \mid I_k, \widetilde{y_2} \mid I_k)$ and $\delta \mid I_k = \delta(\mathbf{x}, \widetilde{y_1} \mid I_k, \widetilde{y_2} \mid I_k)$.

In fact, if $\widetilde{y_1}$ and $\widetilde{y_2}$ and the noise terms ϵ_1 and ϵ_2 are all Gaussian processes, then $\widetilde{y_1} \mid I_k$ and $\widetilde{y_2} \mid I_k$ are also Gaussian and closed form representations are available (see Appendix A). Note that in this case the model update could include updating the Gaussian process hyperparameters as well.

2.5 Optimization objective

Following the formulation of Bect et al. (2012, 2019), a strategy for uncertainty reduction starts with a *measure of residual uncertainty* for the quantity of interest after k experiments. This is a functional

$$H_k = \mathscr{H}(P_k) \tag{6}$$

of the conditional distribution P_k. In this paper we will consider three specific alternatives for H_k.

Assume k experiments have been performed, resulting in the updated probabilistic model (ξ_k, δ_k). The updated failure probability according to (3) can then be defined as

$$\alpha_k = \alpha(\xi_k) = E_k[\mathbf{1}(\xi(\mathbf{X}) \le 0)], \quad \bar{\alpha}_k = E[\alpha_k]. \tag{7}$$

Table 1 Overview of the framework for the optimal experimental design problem for structural reliability analysis

Symbol	Description	Type
$\mathbf{X}$	Parameters describing structure and environment	$\mathbb{R}^m$-valued random variable
$\mathbf{x}$	Deterministic realization of $\mathbf{X}$	values in $\mathbb{R}^m$
$g(\mathbf{x})$	Performance function of structure	real-valued
F_g	Failure set of $g(\cdot)$	subset of $\mathbb{R}^m$
$\bar{\alpha}(g)$	The failure probability, $P(F_g)$	values in [0, 1]
$\xi(\mathbf{x})$	Stochastic approximation of $g(\cdot)$	real-valued stochastic process
$\alpha(\xi)$	The failure probability with epistemic uncertainty	values in [0, 1]
d	Decision	contained in set of decisions $\mathbb{D}$
o	Outcome of experiment	contained in set of outcomes $\mathbb{O}$
(d, o)	Summary of an experiment	contained in $\mathbb{D} \times \mathbb{O}$
$\delta(d)$	Model of experiment outcomes	$\mathbb{O}$-valued stochastic process
$\mathbf{E}$	Parameters for epistemic uncertainty, independent of $\mathbf{X}$	random variable
$\mathscr{A}$	Aleatory information	σ-algebra
$\mathscr{E}$	Epistemic information	σ-algebra
(ξ, δ)	The model	$(\mathbb{R} \times \mathbb{O})$-valued

As we are interested in reducing uncertainty in α, a natural optimization objective is to minimize $\mathrm{Var}(\alpha_k) = E\left[(\alpha_k - \bar{\alpha}_k)^2\right]$. However, computation of $\mathrm{Var}(\alpha_k)$ can be problematic in practice. Most of the proposed methods for design of experiments in (non-hierarchical) structural reliability models therefore make use of alternative *heuristic* optimization objectives. That is, some alternative function $H_k(\cdot)$ that is easier to compute than $\mathrm{Var}(\alpha_k)$, and where the design that minimizes $H_k(\cdot)$ hopefully also performs well with respect to $\mathrm{Var}(\alpha_k)$.

Bect et al. (2012) present a few such criteria, some of which will also be considered in this paper. Let

$$\begin{aligned} p_k(\mathbf{X}) &= P_k(\xi(\mathbf{X}) \leq 0), \\ \gamma_k(\mathbf{X}) &= p_k(\mathbf{X})(1 - p_k(\mathbf{X})). \end{aligned} \tag{8}$$

Observe that

$$\begin{aligned} \mathrm{Var}(\mathbf{1}\left(\xi_k(\mathbf{x}) \leq 0\right)) &= E\left[(\mathbf{1}\left(\xi_k(\mathbf{x}) \leq 0\right))^2\right] - \\ &\quad E\left[\mathbf{1}\left(\xi_k(\mathbf{x}) \leq 0\right)\right]^2 \\ &= E\left[(\mathbf{1}\left(\xi_k(\mathbf{x}) \leq 0\right))\right] - p_k(\mathbf{x})^2 \\ &= p_k(\mathbf{x}) - p_k(\mathbf{x})^2 \\ &= \gamma_k(\mathbf{x}), \end{aligned} \tag{9}$$

and also that $\gamma_k(\mathbf{x})/2$ is the probability that two i.i.d. samples from $\xi_k(\mathbf{x})$ have the same sign. Hence, γ_k provides a measure of how accurate $\xi_k(\mathbf{x})$ is around the critical value $\xi_k = 0$. We will introduce two measures of residual uncertainty based on taking the expectation of γ_k with respect the distribution of $\mathbf{X}$, which we denote $P_{\mathbb{X}}$. In total, we will consider the following three alternativesfor H_k:

$$\begin{aligned} H_{1,k} &= E_k\left[(\alpha - \bar{\alpha})^2\right], \\ H_{2,k} &= \int_{\mathbb{X}} \gamma_k \, \mathrm{d}P_{\mathbb{X}} = E\left[\gamma_k\right], \\ H_{3,k} &= \left(\int_{\mathbb{X}} \sqrt{\gamma_k} \, \mathrm{d}P_{\mathbb{X}}\right)^2 = E\left[\sqrt{\gamma_k}\right]^2. \end{aligned} \tag{10}$$

Here $H_{2,k}$ and $H_{3,k}$ can also be motivated by realizing that they serve as upper bounds on $H_{1,k}$. In fact, $H_{1,k} \leq H_{3,k} \leq H_{2,k}$ (see Proposition 3 in Bect et al. 2012).

For optimal design of experiments we will consider loss functions given by the above measures of residual uncertainty, potentially in combination with an additional penalty term that represents the cost of performing a given experiment. In the Bayesian decision-theoretic framework, given such a loss function depending on a *policy* for selecting experiments π, we can evaluate the policy by looking n-steps ahead. For instance, a relevant loss function for minimizing uncertainty in α after n additional experiments, following after the current experiment k, could be given as $J_k(\pi) = E_k\left[H_{1,k+n}\right]$ where $\mathscr{E}_{k+n}$ corresponds to following the policy π. The additional notation introduced with respect to the measure of residual uncertainty and sequential model updating is summarized in Table 2.

3 Modelling information and experimental design

In this section, we introduce the experimental design framework and explain how the development of information is modelled in this context. In the following, let $k =$

Table 2 Overview of the framework for the optimal experimental design problem for structural reliability analysis with sequential model updating

Symbol	Description	Type
I_k	Information up to k'th experiment	Sequence of decisions and outcomes
$\mathscr{E}_k$	Information given ξ and I_k	σ-algebra
P_k	Conditional probability given ξ_k	Values in [0, 1]
ξ_k	Update of ξ given I_k	Stochastic process indexed by k
δ_k	Update of δ given I_k	Stochastic process indexed by k
$H_k = \mathscr{H}(P_k)$	Measure of residual uncertainty	Functional from space of probability distributions to $\mathbb{R}$
α_k	Updated epistemic failure probability	Values in [0, 1]
$\bar{\alpha}_k$	Updated expected failure probability	Values in [0, 1]

$0, 1, \ldots, K-1$ be the *experiment index* which keeps track of the number of performed experiments.

3.1 The dynamic programming formulation

Huan and Marzouk (2016) introduce a general framework for sequential optimal experimental design: Let the *state*[5] *of the system* after experiment $k-1$ be denoted by s_k. The input (decided by the experimental designers) to experiment k is denoted by d_k. We want to determine a *policy*

$$\pi := (\pi_0, \pi_1, \ldots, \pi_{K-1})$$

where $d_k = \pi_k(s_k)$. That is, given the current state of the system, the policy is a function which tells the experimental designer the input to the next experiment.

From each experiment, we get *observations* o_k. These observations may include measurement noise and modelling errors. Associated to each experiment, we have a *stage reward* $R_k(s_k, o_k, d_k)$. The stage reward reflects the cost of doing the experiment (measured in e.g. money or time) plus any additional benefits or penalties of doing the experiment (measured in the same unit). Furthermore, we have a *terminal reward* $R_K(s_K)$ only depending on the final state of the system.

In order to model the development of the system of experiments, we have the *system dynamics*:

$$s_{k+1} = \mathscr{V}(s_k, d_k, o_k)$$

where $\mathscr{V}(\cdot)$ is some function specifying the transition from a current state to a new state based on the performed experiment. The optimal experimental design problem can then be formulated as follows:

[5] In (Huan and Marzouk 2016) the *state* is written as $s_k = (s_k^{(b)}, s_k^{(p)})$, where $s_k^{(b)}$ denotes the *uncertainty state* and $s_k^{(p)}$ denotes the *physical state* that describes any additional deterministic decision-relevant variables. Herein we will not write s_k specifically in this form.

Maximize

$$E_{o_0,\ldots,o_{K-1}}\left[\sum_{k=0}^{K-1} R_k(s_k, o_k, \pi_k(s_k)) + R_K(s_K)\right] \tag{11}$$

such that $s_{k+1} = \mathscr{V}(s_k, d_k, o_k)$,

and the maximization is done over all policies π that do not look into the future (in the sense that information about future results of experiments are used in current policy making). That is, when deciding policy π_k, only what is known up to experiment $k-1$ can be used. Another way of saying this is that the policy π should be *adapted* to the filtration generated by the processes $\{s_k\}$, $\{o_k\}$ and $\{d_k\}$.

To adapt this framework to the experimental design problem for structural reliability analysis, we write

$$s_k = (\xi_k, \delta_k, I_k), \quad d_k = \pi_k(s_k), \quad o_k = \delta_k(d_k), \tag{12}$$

and where the dynamics $s_{k+1} = \mathscr{V}(s_k, d_k, o_k)$ is given by updating ξ_k, δ_k and I_k with respect to the experiment (d_k, o_k) as described in Sect. 2.4.

Remark 3 Note that the expectation in (11) is with respect to future outcomes $o_0, \ldots, o_{K-1}$ which a priori are uncertain, and where each outcome o_k depends on the previous outcomes $o_0, \ldots, o_{k-1}$. An equivalent formulation can be given in terms of conditional expectations. Let each reward be defined by backwards induction:

$$R_k = \max_d E_k\left[R_{k+1} | d_k = d\right], \quad k = K-1, \ldots, 0,$$

where $R_K = R_K(s_k)$ only depends on the final state of the system. Then, the policy defined by selecting for each k the decision

$$\begin{aligned} d_k^* &= \arg\max_{d\in\mathbb{D}} E_k\left[R_{k+1} | d_k = d\right] \\ &= \arg\max_{d\in\mathbb{D}} E_k\left[\max E_{k+1} \cdots E_K R_K | d_k = d\right] \end{aligned}$$

is optimal. This corresponds with the formulation used by Bect et al. (2012).

Problem (11) is a dynamic programming problem. Though theoretically optimal, such problems are known for suffering form the so-called *curse of dimensionality*. That is, possible sequences of design and observation realizations grow exponentially with the dimension of the state space. According to Defourny et al. (2011), the curse of dimensionality implies that dynamic programming can only be solved numerically for state spaces embedded in $\mathbb{R}^d$ with $d \leq 10$. Therefore, such problems can often only be solved approximately via *approximate dynamic programming*, see (Huan and Marzouk 2016). Note also that this type of formulation is based on a *Markovianity* assumption, i.e., that there is no memory in the dynamics of the system. This assumption is necessary in order to perform the simplification to only having dependency on the current state of the system in Remark 3. If the system is not Markovian, in the sense that the decision at any time depends not only on the current state of the system, but also on some of the previous states, we cannot solve the experimental design problem by backwards induction. The reason for this is that the Bellman equation, which backwards induction is based on, does not hold in this case. In such cases, the experimental design problem can for instance be solved via the maximum principle, see e.g. Dahl et al. (2016) for an example of systems with memory in continuous time.

Remark 4 An alternative solution method to dynamic programming for problem (11) is to use a scenario tree based approach, see Defourny et al. (2011). Scenario tree based approaches are not sensitive to curse of dimensionality based on the state space, but based on the number of experiments. Hence, a scenario based approach can be attempted whenever there are few experiments (less than or equal 10), but potentially a large dimensional state space. If the number of experiments is large (greater than 10), but the state space dimension is small (less than or equal 10), dynamic programming is a viable solution method. If both the state space dimension and the number of experiments is large, one can try approximate dynamic programming (see Huan and Marzouk (2016)) or a one-step lookahead (myopic)[6] formulation as an alternative to the dynamic programming one. In Sect. 3.2, we consider such a one-step lookahead formulation.

Note that problem (11) is maximization problem of a reward, but can trivially be transformed to a minimization problem with some loss function $L_k = -R_k$ instead. For the application considered in this paper, we are interested in minimization problems associated with the residual uncertainty described in Sect. 2.5.

[6] Some authors, for instance Huan and Marzouk (2016) and Bect et al. (2012), remark that all strategies which consider fewer terms in the summation are *myopic*. Other authors use *myopic* only in the case where no future decisions are taken into account, i.e. the horizon is zero. When we say myopic in this paper, we mean one-step lookahead.

Example 7 Let $\lambda(d_k)$ denote the cost of decision d_k. A relevant set of loss functions could then be: $L_k(s_k, d_k, o_k) = 0$ for $k < K$ and $L_K = H_K \cdot \sum_{k<K} \lambda(d_k)$, where $H_K = H_{1,k}$, $H_{2,k}$ or $H_{3,k}$ as described in Sect. 2.5. Or, letting $L_k(s_k, d_k, o_k) = \eta^k \lambda(d_k) H_k$ for $k < K$ where η is some discount factor, $\eta \in (0, 1)$, would produce a similar but more greedy policy. Another relevant alternative is to define $L_K = \sum_{k<k^*} \lambda(d_k)$ as the sum of costs up to the iteration k^* where some target level, $H_k < H^*$ for $k > k^*$, has been reached.

3.2 The one-step lookahead formulation

As mentioned in Sect. 3.1, the dynamic programming formulation suffers from the curse of dimensionality. An approximation to the dynamic programming formulation which mends this problem, is the *myopic formulation* or *one-step lookahead*. This corresponds to truncating the dynamic programming sum in (11) and only looking at one time-step ahead.

In this section, we define the the *one-step lookahead optimal decision* $d \in \mathbb{D}$ at step k as the minimizer of the following function

$$J_{i,k}(d) = \lambda(d) E_{k,d}\left[H_{i,k+1}\right] \text{ for } i = 1, 2, \text{ or } 3. \tag{13}$$

Here $H_{i,k}$ are the measures of residual uncertainty defined in Sect. 2.5, and $E_{k,d}$ represents the conditional expectation with respect to $\mathscr{E}_k$ with $d_k = d$. Hence, $E_{k,d}\left[H_{i,k+1}\right]$ represents how desirable decision d is for reducing the expected remaining uncertainty in α at experiment $k+1$, if the next experiment is performed with input d. We let $\lambda(d)$ be a deterministic function representing the cost associated with decision d, and we will refer to a function $J_{i,k}(d)$ as the *acquisition* function for myopic (one-step lookahead) design. Other ways of introducing additional rewards or penalties associated with an experiment are of course also possible. In fact, there is no particular reason why we write (13) as a *product* of cost and the measure of residual uncertainty, besides emphasizing that $J_{i,k}(d)$ should be a function of these two terms.

Remark 5 We have assumed here that a total number K of experiments that are to be performed, where we want to perform each experiment optimally. But in practice it is relevant to consider stopping before the Kth experiment, when some objective has been reached, or when the potential gain of new experiments diminishes. Section 5.3 we introduce a criterion for stopping when the variance in the failure probability is sufficiently low.

4 Approximating the measure of residual uncertainty

Assume k experiments have been performed, resulting in the updated probabilistic model (ξ_k, δ_k). A simple method for estimating the measures of residual uncertainty described in Sect. 2.5, is by a double-loop Monte Carlo simulation: Let $N_1, N_2 \in \mathbb{N}$ and let $h_{i,j}^{(k)} = \mathbf{1}\left(\xi_{k,j}(\mathbf{x}_i) \leq 0\right)$, where $\mathbf{x}_1, \ldots, \mathbf{x}_{N_1}$ are N_1 i.i.d. samples of $\mathbf{X}$ and $\xi_{k,1}(\mathbf{x}_i), \ldots \xi_{k,N_2}(\mathbf{x}_i)$ are N_2 i.i.d. performance functions sampled from ξ_k and evaluated at each $\mathbf{x}_i$. Then $H_{1,k}$ can be obtained as the sample variance of the N_2 samples of the form $\hat{\alpha}_{k,j} = \frac{1}{N_1}\sum_i h_{i,j}^{(k)}$. Similarly, $H_{2,k}$ and $H_{3,k}$ can be estimated from $\hat{p}_k(\mathbf{x}_i) = \frac{1}{N_2}\sum_j h_{i,j}^{(k)}$.

This approach is problematic for several reasons. First of all, $\hat{\alpha}_{k,j}$ is an unbiased estimator of the failure probability $\alpha_{k,j} = \alpha(\xi_{k,j})$ corresponding to the deterministic performance function $\xi_{k,j}$. When $\alpha_{k,j}$ is small, the variance of this estimator is $\text{var}(\hat{\alpha}_{k,j}) = \alpha_{k,j}(1-\alpha_{k,j})/N_1 \approx \alpha_{k,j}/N_1$. If we want to achieve an accuracy, of say $\sqrt{\text{var}(\hat{\alpha}_{k,j})} < 0.1\alpha_{k,j}$, and $\alpha_{k,j} = 10^{-m}$, then the number of samples required would be approximately $N_1 = 10^{m+2}$. The failure probabilities considered in structural reliability analysis can typically be in the range from 10^{-6} to 10^{-2}.

When N_1 is large, it can also be a practical challenge to obtain the samples $\xi_{k,j}(\mathbf{x}_1), \ldots, \xi_{k,j}(\mathbf{x}_{N_1})$ simultaneously for a fixed j. Moreover, the total number of samples needed to evaluate the measures of residual uncertainty $H_{i,k}$ is $N_1 N_2$, and we are interested in optimization over $H_{i,k}$ that will require multiple simulations of this kind.

In this section we present a procedure for efficient approximation of the measures of residual uncertainty. We will start by introducing a finite-dimensional approximation of $\xi_k(\mathbf{x})$, given as a deterministic function $\hat{\xi}_k(\mathbf{x}, \mathbf{E})$ depending on $\mathbf{x}$ and a finite-dimensional $\mathscr{E}_k$-measurable random variable $\mathbf{E}$. Then, in Sect. 4.2 we consider how the mean and variance, $E[f(\mathbf{E})]$ and $\text{var}(f(\mathbf{E}))$, can be approximated for any $\mathscr{E}_k$-measurable function $f(\mathbf{e})$ using the unscented transform. In Sects. 4.3 and 4.4 we present an importance sampling scheme for the case where $f(\mathbf{e})$ is defined in terms of an expectation over $\mathbf{X}$. Finally, in Sect. 4.5 we consider the case where $f(\mathbf{e}) = \alpha(\hat{\xi}_k(\mathbf{X}, \mathbf{e}))$, which provides the approximations $\hat{\alpha}_k = f(\mathbf{E})$ and $\hat{H}_{1,k} = \text{var}(f(\mathbf{E}))$, and where approximations of $H_{2,k}$ and $H_{3,k}$ are obtained in a similar manner.

In summary, this kind of approximation which we will refer to as UT-MCIS from now on, makes use of the unscented transform (UT) for epistemic uncertainty propagation and Monte Carlo simulation with importance sampling (MCIS) for aleatory uncertainty propagation. The motivation behind this specific setup is that a technique such as MCIS is needed to obtain low variance estimates of $\alpha(\hat{\xi}_k(\mathbf{X}, \mathbf{e}))$, which will typically be a small number. The sampling scheme we propose is also designed to be efficient in the case where subsequent estimates corresponding to perturbations of $\alpha(\hat{\xi}_k(\mathbf{X}, \mathbf{e}))$ are needed, which is relevant for estimation of e.g. $\alpha(\hat{\xi}_{k+1}(\mathbf{X}, \mathbf{e}))$ or $\alpha(\hat{\xi}_k(\mathbf{X}, \mathbf{e}'))$ for some $\mathbf{e}' \neq \mathbf{e}$ if $\alpha(\hat{\xi}_k(\mathbf{X}, \mathbf{e}))$ has already been estimated. As for epistemic uncertainty propagation, when $\alpha(\hat{\xi}_k(\mathbf{x}, \mathbf{E}))$ is viewed as an $\mathscr{E}_k$-measurable random variable, the UT alternative which is both simpler and more efficient seems like a viable alternative, in particular for the purpose of optimization with respect to future decisions.

4.1 The finite-dimensional approximation of ξ_k

In our framework, we have defined ξ_k as a $\mathscr{E}_k$-measurable stochastic process indexed by $\mathbf{x} \in \mathbb{X}$ (often called a random field), and we view ξ_k as a distribution over some (generally infinite-dimensional) space of functions. The special case where $\xi_k = \xi_k(\mathbf{x}, \mathbf{E})$ for some finite-dimensional $\mathscr{E}_k$-measurable random variable $\mathbf{E}$ can be very useful for simulation. That is, if samples $\mathbf{e}_j$ of $\mathbf{E}$ can be generated efficiently, then random functions $\xi_{k,j}(\mathbf{x}) = \xi_k(\mathbf{x}, \mathbf{e}_j)$ can be sampled as well. As long as ξ_k is square integrable, such a representation of ξ_k is always available from the Karhunen-Loéve transform:

$$\xi_k(\mathbf{x}) - E[\xi_k(\mathbf{x})] = \sum_{i=1}^{\infty} E_i \phi_i(\mathbf{x}),$$

where the functions ϕ_i are deterministic and E_i are uncorrelated random variables with zero mean. The canonical ordering of the terms $E_i\phi_i(\mathbf{x})$ also provides a suitable method for approximating $\xi_k(\mathbf{x})$, by truncating the sum at some finite $i = M$, and we could then let $\mathbf{E} = (E_1, \ldots, E_M)$ (see for instance Wang 2008).

But obtaining the Karhunen-Loéve transform can also be challenging. Because of this, we present an extremely simple approximation, that just relies on computation of the first two moments of ξ_k. We let $\mathbf{E}$ be a 1-dimensional random variable with $E[\mathbf{E}] = 0$ and $E\left[\mathbf{E}^2\right] = 1$, and define

$$\hat{\xi}_k(\mathbf{x}) = E[\xi_k(\mathbf{x})] + \mathbf{E}\sqrt{\text{var}(\xi_k(\mathbf{x}))}. \tag{14}$$

This is indeed a very crude approximation, as essentially we assume that the values of ξ_k at any set of inputs $\mathbf{x}$ are fully correlated. But for probabilistic surrogates used in structural reliability models, this is actually not that unreasonable, and as it turns out, for the examples we consider in Sect. 6 it seems sufficient.

Remark 6 Note that to update the approximate model $\hat{\xi}_k(\mathbf{x})$ in (14) given some new experiment (d_k, o_k), we only need to update the mean and variance functions. This is in line

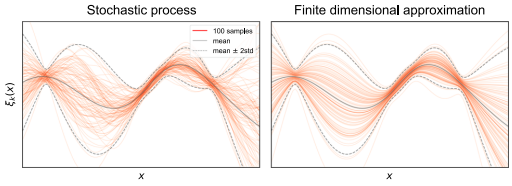


Fig. 2 Illustration of the finite-dimensional approximation (14)

with the numerically efficient Bayes linear approach (Goldstein and Wooff 2007), where random variables are specified only through the first two moments, and where the Bayesian updating given some experiment corresponds to computation of an adjusted mean and covariance. An application of the Bayes linear theory to sequential optimal design of experiments can be found in (Jones et al. 2018).

We note also that in the case where Gaussian processes are used as surrogate models, the classical and linear Bayesian approaches are computationally equivalent. Moreover, in the following section we will introduce the unscented transform for approximation of the updated/adjusted moments, and as a consequence the complete prior probability specification of $\mathbf{E}$ becomes less relevant.

In the case where we are dealing with a hierarchical model, it might not be convenient to compute $E\left[\xi_k(\mathbf{x})\right]$ and $\text{var}(\xi_k(\mathbf{x}))$. If $\xi_k(\mathbf{x}) = g(\mathbf{Y}_k(\mathbf{x}))$ where $\mathbf{Y}_k(\mathbf{x})$ is a stochastic process with values in $\mathbb{R}^n$ for any $\mathbf{x} \in \mathbb{X}$, we would instead approximate $\mathbf{Y}_k$ with

$$\hat{\mathbf{Y}}_k = E\left[\mathbf{Y}_k\right] + L\mathbf{E}, \tag{15}$$

where $\mathbf{E}$ is n-dimensional with $E\left[\mathbf{E}\right] = 0$, $E[\mathbf{E}\mathbf{E}^T] = I$, and the matrix L satisfies $LL^T = (\mathbf{Y}_k - E\left[\mathbf{Y}_k\right])(\mathbf{Y}_k - E\left[\mathbf{Y}_k\right])^T$. The approximation of ξ_k is then obtained as $\hat{\xi}_k(\mathbf{x}) = g(\hat{\mathbf{Y}}_k(\mathbf{x}))$. The same goes for the scenario with more than two layers in the hierarchy, for instance $\xi_k(\mathbf{x}) = g(\mathbf{Z}_k(\mathbf{Y}_k(\mathbf{x})))$, where we would approximate both $\mathbf{Z}_k(\mathbf{y})$ and $\mathbf{Y}_k(\mathbf{x})$. In any case, we end up with a finite-dimensional random variable $\mathbf{E}$, and we can define the approximation $\hat{\xi}_k(\mathbf{x}, \mathbf{E})$.

4.2 The unscented transform for epistemic uncertainty propagation

The unscented transform (UT) is a very efficient method for approximating the mean and covariance of a random variable after nonlinear transformation. UT is commonly applied in the context of Kalman filtering, and it is based on the general idea that *it is easier to approximate a probability distribution than an arbitrary nonlinear transformation* (Uhlmann 1995; Julier and Uhlmann 2004). Intuitively, given any finite-dimensional random variable $\mathbf{E}$ we may define a set of *weighted sigma-points* $\{(v_i, \mathbf{e}_i)\}$, such that if $\{(v_i, \mathbf{e}_i)\}$ was considered as a discrete probability distribution, then its mean and covariance would coincide with $\mathbf{E}$. For any nonlinear transformation $\mathbf{Y} = f(\mathbf{E})$, if $\mathbf{E}$ was discrete we could compute the mean and covariance of $\mathbf{Y}$ exactly. The UT approximation is the result of such computation, where we make use of a small set of weighted points $\{(v_i, \mathbf{e}_i)\}$.

Specifically, let $\mathbf{E}$ be a finite-dimensional random variable with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. A set of sigma-points for $\mathbf{E}$ is a set of weighted samples $\{(v_1, \mathbf{e}_1), \ldots, (v_n, \mathbf{e}_n)\}$ such that

$$\boldsymbol{\mu} = \sum_{i=1}^{n} v_i \mathbf{e}_i, \quad \boldsymbol{\Sigma} = \sum_{i=1}^{n} v_i (\mathbf{e}_i - \boldsymbol{\mu})(\mathbf{e}_i - \boldsymbol{\mu})^T. \tag{16}$$

If $\mathbf{y} = f(\mathbf{e})$ is any (generally nonlinear) transformation, the UT approximation of the mean and covariance of $\mathbf{Y} = f(\mathbf{E})$ are then obtained as

$$\begin{aligned} \widehat{E}[\mathbf{Y}] &= \sum_{i=1}^{n} v_i \mathbf{y}_i, \\ \widehat{\text{Cov}}[\mathbf{Y}] &= \sum_{i=1}^{n} v_i (\mathbf{y}_i - \widehat{E}[\mathbf{Y}])(\mathbf{y}_i - \widehat{E}[\mathbf{Y}])^T, \end{aligned} \tag{17}$$

where $\mathbf{y}_i = f(\mathbf{e}_i)$.

Naturally, the selection of appropriate sigma-points is essential for UT to be successful. It is important to note that, although we may view the sigma-points as *weighted samples*, v_i and $\mathbf{e}_i$ are fixed or given by some deterministic procedure. Moreover, the definition of sigma-points given in (16) does not require that the weights are nonnegative and sum to one. Although this conflicts with the intuition of approximating $\mathbf{E}$ with a discrete random variable, the unscented transform still makes sense as a procedure for approximating statistics after nonlinear transformation.

Since the introduction of UT to Kalman filters in the 1990's, many different alternatives to sigma-point selection have been proposed (Menegaz et al. 2015). These mostly focus on applications where $\mathbf{E}$ follows a multivariate Gaussian distribution, but we do not see this as a restriction since we will assume that $\mathbf{E}$ can be represented as a transformation $\mathbf{E} = \mathscr{T}^{-1}(\boldsymbol{U})$ of a multivariate Gaussian variable $\boldsymbol{U}$. For the applications considered in this paper, we will let $\{(v_i, \mathbf{u}_i)\}$ denote a set of sigma-points that are appropriate for the multivariate standard normal $\boldsymbol{U} \sim \mathcal{N}(0, I)$ where $\dim(\boldsymbol{U}) = \dim(\mathbf{E})$. If $\mathscr{T}$ is the corresponding isoprobabilistic transformation, i.e. $\mathscr{T}(\mathbf{E}) \sim \mathcal{N}(0, I)$ (see Appendix B.1), we will use $\{(v_i, \mathscr{T}^{-1}(\mathbf{u}_i))\}$ as a set of sigma-points for $\mathbf{E}$. Equivalently, we could also view this as taking the UT approximation of $\boldsymbol{U}$ under a different transformation given by $f \circ \mathscr{T}$. For the numerical examples we present in this paper, we have made use of the the method developed by Merwe (2004), which produces a set of $n = 2 \cdot \dim(\mathbf{E}) + 1$

points $\mathbf{e}_i$ with corresponding weights[7]. Determining sigma-points with this procedure is quite straightforward, and the details are given in Appendix C. We note again that for any structural reliability model, as long as we do not change dimensionality of $\mathbf{E}$, determining the sigma-points is a one-time computation, and any subsequent UT approximation of $\mathbf{Y} = f(\mathbf{E})$, for some nonlinear transformation $f(\cdot)$, is computationally very efficient.

Remark 7 Note that it is not necessary that the sigma points used in the approximation of the mean and covariance in (17) are the same. In fact, the method presented in Appendix C makes use of two different sets of weights for these approximations. As this is not of any relevance for the remaining part of this paper, we will keep writing $\{v_i, \mathbf{e}_i\}$ as a single set of sigma-points to simplify the notation.

4.3 Generating samples in $\mathbb{X}$

In order to estimate the measures of residual uncertainty, we will need a set of samples of $\mathbf{X}$. We will generate a finite set of 3-tuples $\{(\mathbf{x}_i, w_i, \hat{\eta}_i)\}$, where $\{(\mathbf{x}_i, w_i)\}$ are weighted samples in $\mathbb{X}$ suitable for obtaining importance sampling estimates of failure probabilities, and $\hat{\eta}_i$ is a number describing how influential a given sample $(\mathbf{x}_i, w_i)$ is expected to be in such an estimate. In other words, $\{\mathbf{x}_i\}$ should be constructed to "cover the relevant regions in $\mathbb{X}$", and for estimation we will only make use of a subset of $\{(\mathbf{x}_i, w_i)\}$. The relevant subset will be determined from the *measure of insignificance* $|\hat{\eta}_i|$, where we will only consider samples $(\mathbf{x}_i, w_i)$ where $|\hat{\eta}_i|$ is below some threshold. We start by describing how the weighted samples $\{(\mathbf{x}_i, w_i)\}$ are generated.

4.3.1 Importance sampling

The general idea behind importance sampling is that if we select some random variable $\boldsymbol{Q} \geq 0$ with law $P_{\boldsymbol{Q}}$, such that $E_{P_{\mathbf{X}}}[Q] = 1$ and $\boldsymbol{Q} \neq 0$ $P_{\mathbf{X}}$-almost surely, then

$$E_{P_{\mathbf{X}}}[f(\mathbf{X})] = E_{P_{\boldsymbol{Q}}}[f(\mathbf{X})/\boldsymbol{Q}], \tag{18}$$

for any $\mathscr{A}$-measurable function $f(\mathbf{x})$. This is often useful for estimation, for instance when sampling from $P_{\mathbf{X}}$ is difficult, and in the case where we can find a $\boldsymbol{Q}$ such that estimates with respect to the right hand side of (18) are better (have lower variance) than estimating $E_{P_{\mathbf{X}}}[f(\mathbf{X})]$ directly.

[7] Other alternatives for sigma-point selection could also be applied, potentially with better performance. The method by Merwe (2004) depends on a set of parameters, and it could also be relevant to refine or *learn* the appropriate parameter values as in (Turner and Rasmussen 2010). However, in our current implementation we have only considered the fixed set of sigma-points given in Appendix C.

In the case where $\mathbf{X}$ admits a probability density $p_{\mathbf{X}}$, we can let $q_{\mathbf{X}}$ be any density function such that $q_{\mathbf{X}}(\mathbf{x}) > 0$ whenever $p_{\mathbf{X}}(\mathbf{x}) > 0$. Let $\mathbf{x}_1, \ldots, \mathbf{x}_N$ be i.i.d. samples generated according to $q_{\mathbf{X}}$, and define $w_i = p_{\mathbf{X}}(\mathbf{x}_i)/q_{\mathbf{X}}(\mathbf{x}_i)$. The importance sampling estimate of $E_{P_{\mathbf{X}}}[f(\mathbf{X})]$ with respect to the proposal density $q_{\mathbf{X}}$ is then obtained as

$$E_{P_{\mathbf{X}}}[f(\mathbf{X})] = E_{P_Q}\left[f(\mathbf{X})\frac{p_{\mathbf{X}}(\mathbf{X})}{q_{\mathbf{X}}(\mathbf{X})}\right] \approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i)w_i\,. \tag{19}$$

We now assume that the stochastic limit state can be written as $\xi_k(\mathbf{x}, \mathbf{E})$ for some finite-dimensional random variable $\mathbf{E}$, and for any deterministic performance function $\xi_k(\mathbf{x}, \mathbf{e})$ we will write $\alpha_k(\mathbf{e}) = \alpha(\xi_k(\mathbf{X}, \mathbf{e}))$ as the corresponding failure probability. An importance sampling estimate of $\alpha_k(\mathbf{e})$ is then given by (19) with $f(\mathbf{x}) = \mathbf{1}(\xi_k(\mathbf{x}, \mathbf{e}) \leq 0)$, that is

$$\hat{\alpha}_k(\mathbf{e}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(\xi_k(\mathbf{x}_i, \mathbf{e}) \leq 0)\, w_i\,. \tag{20}$$

In order to obtain a good estimate of $\alpha_k(\mathbf{e})$, we would like the proposal distribution $q_{\mathbf{X}}$ to produce samples such that there is an even balance between the samples where $\xi_k(\mathbf{x}, \mathbf{e}) \leq 0$ and $\xi_k(\mathbf{x}, \mathbf{e}) > 0$, where at the same time $p_{\mathbf{X}}$ is as large as possible. One way to achieve this is to generate samples in the vicinity of points on the surface $\xi_k(\mathbf{x}, \mathbf{e}) = 0$ with (locally) maximal density. A point with this property is called a *design point*[8] or *most probable failure point* in the structural reliability literature. We will let $q_{\mathbf{X}}$ represent a mixture of distributions, centered around different design points that are appropriate for different values of $\mathbf{e}$. The full details are given in Appendix B, where we also describe a simpler alternative than can be used in the case where design point searching is difficult or not appropriate.

4.3.2 The measure of insignificance $|\eta_i|$

Assume $\{(\mathbf{x}_i, w_i)\}$ is a set of samples capable of providing a satisfactory estimate of $\alpha_k(\mathbf{e})$, and we now want to estimate $\alpha_k(\mathbf{e}')$ for some new value $\mathbf{e}'$. If we know that the sign of $\xi_k(\mathbf{x}_i, \mathbf{e})$ and $\xi_k(\mathbf{x}_i, \mathbf{e}')$ will coincide for many of the samples $\mathbf{x}_i$, then the estimate of $\alpha_k(\mathbf{e}')$ can be obtained more efficiently by not computing all the terms in the sum (20). This is typically the case when $\mathbf{e}$ and $\mathbf{e}'$ are both sampled from $\mathbf{E}$. It is also true in the case where we want to estimate $\alpha_{k+1}(\mathbf{e}')$ given some new experiment (d_k, o_k), if we assume that updating with respect to (d_k, o_k) has local effect (i.e. there are always regions in $\mathbb{X}$ where $\xi_{k+1}(\mathbf{x}) \approx \xi_k(\mathbf{x})$), or if

[8] The most common definition of a design point is that it is the point on the limit state surface with maximal density after transformation to the standard normal space. See Appendix B.1

the experiment is carried out to reduce the uncertainty in the level set $\xi_k = 0$ (which is what we intend to do).

In other words, we consider some perturbation of the performance function $\xi_k(\mathbf{x}, \mathbf{e})$, and we are interested in identifying the samples $\mathbf{x}_i$ where $\mathbf{1}(\xi_k(\mathbf{x}_i, \mathbf{e}) \leq 0)$ does not change under the perturbation. For this purpose we define the function

$$\eta(\mathbf{x}, \xi) = E[\xi(\mathbf{x})] / \sqrt{\mathrm{var}(\xi(\mathbf{x}))}, \tag{21}$$

and let $\eta_i = \eta(\mathbf{x}_i, \xi_k)$ be defined with respect to the relevant process ξ_k. Here η_i describes how uncertain $\xi_k(\mathbf{x}_i)$ is around the critical value $\xi_k = 0$, in the sense that if $|\eta_i|$ is small (close to zero) then $\xi_k(\mathbf{x}_i) > 0$ and $\xi_k(\mathbf{x}_i) \leq 0$ may both be probable outcomes. Conversely, if $|\eta_i|$ is large then either $P(\xi_k(\mathbf{x}_i) \leq 0) \approx 0$ or $P(\xi_k(\mathbf{x}_i) \leq 0) \approx 1$, and the input $\mathbf{x}_i$ is *insignificant* as it is unnecessary to keep track of changes in $\mathbf{1}(\xi_k(\mathbf{x}_i) \leq 0)$. We will use η_i to *prune* the sample set $\{(\mathbf{x}_i, w_i)\}$, by only considering the samples where $|\eta_i|$ is below a given threshold τ. Although this is an intuitive idea, we may also justify the definition of η and selection of a threshold τ more formally by making use of the following proposition.

Proposition 41 *Given any process $\xi(\boldsymbol{x})$, let $\eta(\boldsymbol{x}) = \eta(\boldsymbol{x}, \xi)$ be defined as in (21) and let $\tau > \sqrt{2}$. Assume $\xi^{(1)}$ and $\xi^{(2)}$ are two i.i.d. random samples from $\xi(\boldsymbol{x})$. Then,*

$$P\left(\mathbf{1}\left(\xi^{(1)} \leq 0\right) \neq \mathbf{1}\left(\xi^{(2)} \leq 0\right) \,\middle|\, |\eta| \geq \tau\right) \leq \frac{2}{\tau^2}\left(1 - \frac{1}{\tau^2}\right). \tag{22}$$

Proof Let $p = P(\xi(\mathbf{x}) \leq 0)$ and $\gamma(p) = p(1-p)$ for short (note also that this is (8) for $\xi = \xi_k$), and observe that $P\left(\mathbf{1}\left(\xi^{(1)} \leq 0\right) \neq \mathbf{1}\left(\xi^{(2)} \leq 0\right)\right) = 2\gamma(p)$. Assume first that $\eta > 0$. Then $E[\xi] > 0$ and by Chebyshev's one-sided inequality we get

$$\eta = \tau \Rightarrow p \leq \frac{\mathrm{var}(\xi(\mathbf{x}))}{(\mathrm{var}(\xi(\mathbf{x})) + E[\xi(\mathbf{x})]^2)} \leq \frac{1}{\tau^2},$$

and as $\tau > \sqrt{2}$ we also get $p \leq 1/2$. Since $\gamma(p)$ is increasing for $p \in [0, 1/2]$, we must have $\gamma(p) \leq \gamma(1/\tau^2)$.

Conversely, if $-\tau = \eta < 0$ then $p \geq 1 - 1/\tau^2 \geq 1/2$, and as $\gamma(p)$ is decreasing for $p \in [1/2, 1]$ we have that $\gamma(p) \leq \gamma(1 - 1/\tau^2) = \gamma(1/\tau^2)$. Hence, combining both cases we get $|\eta| = \tau \Rightarrow \gamma(p) \leq \gamma(1/\tau^2)$, and (22) is proved by observing that $\gamma(1/(\tau + \varepsilon)^2) \leq \gamma(1/\tau^2)$ for any $\varepsilon > 0$. □

Although Proposition 41 holds in general, tighter (and probably more realistic) bounds can be obtained by making assumptions on the form of $\xi(\mathbf{x})$. For instance, in the case where $\xi(\mathbf{x})$ is Gaussian we obtain

$$P\left(\mathbf{1}\left(\xi^{(1)} \leq 0\right) \neq \mathbf{1}\left(\xi^{(2)} \leq 0\right) \,\middle|\, |\eta| \geq \tau\right) \leq 2\Phi(\tau)\Phi(-\tau), \tag{23}$$

where $\Phi(\cdot)$ is the standard normal CDF.

We will make use of $\hat{\eta}_i$ obtained as the UT approximation of η_i. That is, $\hat{\eta}_i$ is in general obtained from the finite-dimensional approximation described in Sect. 4.1, combined with the UT approximation (17) with $\mathbf{Y} = \hat{\xi}_k(\mathbf{x}, \mathbf{E})$.

4.4 Importance sampling estimates with pruning

Let $\{(\mathbf{x}_i, w_i, \hat{\eta}_i) \mid i \in \mathscr{I}\}$, $\mathscr{I} = \{1, \ldots, N_0\}$ be a set of samples generated as described in Sect. 4.3. Given some fixed threshold $\tau > 0$, we define the subset of *pruned samples* as the ones corresponding to the index set $\mathscr{I}_\tau = \{i \in \mathscr{I} \mid \hat{\eta}_i < \tau\}$, and define $\bar{\mathscr{I}}_\tau = \mathscr{I} \backslash \mathscr{I}_\tau$. If $f(\mathbf{x})$ is some $\mathscr{A}$-measurable function where we know a priori the value of $f_i = f(\mathbf{x}_i)$ for all $i \in \bar{\mathscr{I}}_\tau$, then we can immediately compute

$$\bar{h} = \frac{1}{N_0} \sum_{i \in \bar{\mathscr{I}}_\tau} f_i w_i, \tag{24}$$

and the importance sampling estimate of the expectation of $f(\mathbf{X})$ becomes

$$\widehat{E}[f(\mathbf{X})] = \bar{h} + \frac{1}{N_0} \sum_{i \in \mathscr{I}_\tau} f(\mathbf{x}_i) w_i. \tag{25}$$

If we let

$$s_{\bar{h}} = \frac{1}{N_0} \sum_{i \in \bar{\mathscr{I}}_\tau} \left(f_i w_i - \widehat{E}[f(\mathbf{X})]\right)^2, \tag{26}$$

then an unbiased estimate of the sample variance is given as

$$\widehat{\mathrm{var}}(\widehat{E}[f(\mathbf{X})]) = \frac{s_{\bar{h}}}{N_0 - 1} + \frac{1}{N_0(N_0 - 1)} \sum_{i \in \mathscr{I}_\tau} \left(f(\mathbf{x}_i) w_i - \widehat{E}[f(\mathbf{X})]\right)^2, \tag{27}$$

which shows the general idea with this *pruning*, namely that low variance estimates of $E[f(\mathbf{X})]$ can be obtained with a small number of evaluations $f(\mathbf{x}_i)$, assuming that the subset $\mathscr{I}_\tau$ is small compared to $\mathscr{I}$ (and that the assumed values f_i are correct).

One drawback with this procedure is that we do not have control over the number of pruned samples, which still might be very large. In order to set an upper bound on the number of evaluations $f(\mathbf{x}_i)$, we let $\mathscr{I}_\tau^n \subseteq \mathscr{I}_\tau$ contain the first n elements of $\mathscr{I}_\tau$ (or some other subset, as long as the elements of

$\{\mathbf{x}_i \mid i \in \mathscr{I}_\tau^n\}$ remain independent). An importance sampling estimate of $E[f(\mathbf{X})]$ using only samples from $\mathscr{I}_\tau^n$ is given as

$$\widehat{E}[f(\mathbf{X})] = \bar{h} + \bar{r}, \quad \bar{r} = \frac{N_\tau}{nN_0} \sum_{i \in \mathscr{I}_\tau^n} f(\mathbf{x}_i) w_i, \tag{28}$$

where $N_\tau = |\mathscr{I}_\tau|$, and we may estimate the sample variance as

$$\widehat{\mathrm{var}}(\widehat{E}[f(\mathbf{X})]) = \frac{1}{N_0 - 1}(s_{\bar{h}} - \bar{h}^2) + \frac{N_\tau}{nN_0 - N_\tau}\left(-\bar{r}^2 + \frac{N_\tau}{nN_0} \sum_{i \in \mathscr{I}_\tau^n} (f(\mathbf{x}_i) w_i)^2\right). \tag{29}$$

Obtaining consistency results is easy under the ideal assumption that $n(N_0 - N_\tau)/N_\tau$ is an integer, and the formulas in (28)–(29) comes as a consequence of the following result.

Proposition 42 *Assume $n(N_0 - N_\tau)/N_\tau \in \mathbb{N}$. Then (28) is an unbiased estimate of $E[f(\boldsymbol{X})]$ and (29) is an unbiased estimate of the sample variance.*

Proof Let $\bar{\mathscr{I}}_\tau^n$ be a set of $n(N_0 - N_\tau)/N_\tau$ elements selected uniformly random from $\bar{\mathscr{I}}_\tau$ and define $\mathscr{I}^n = \mathscr{I}_\tau^n \cup \bar{\mathscr{I}}_\tau^n$. Then $\{\mathbf{x}_i \mid i \in \mathscr{I}^n\}$ is a set of size $|\mathscr{I}^n| = nN_0/N_\tau$, containing i.i.d. samples from the proposal distribution with density $q(\mathbf{x})$. To show consistency we replace each sample $\mathbf{x}_i$ with i.i.d. random variables $\mathbf{X}_i$ distributed according to q. We then define $\hat{\mu} = \hat{\mu}_1 + \hat{\mu}_2$ where

$$\hat{\mu}_1 = \frac{1}{|\mathscr{I}|} \sum_{i \in \mathscr{I}} \mathbf{1}\left(\eta(\mathbf{X}_i \geq \tau)\right) f(\mathbf{X}_i) w(\mathbf{X}_i),$$

$$\hat{\mu}_2 = \frac{1}{|\mathscr{I}^n|} \sum_{i \in \mathscr{I}^n} \mathbf{1}\left(\eta(\mathbf{X}_i < \tau)\right) f(\mathbf{X}_i) w(\mathbf{X}_i),$$

and where $w(\mathbf{x}) = p(\mathbf{x})/q(\mathbf{x})$, and we can observe that $\hat{\mu} = \widehat{E}[f(\mathbf{X})]$ when $\mathbf{X}_i = \mathbf{x}_i$.

To show that $\widehat{E}[f(\mathbf{X})]$ is unbiased it is enough to observe that $E_q[\hat{\mu}] = E_q[\mathbf{1}\left(\eta(\mathbf{X} \geq \tau)\right) f(\mathbf{X}) w(\mathbf{X})] + E_q[\mathbf{1}\left(\eta(\mathbf{X} < \tau)\right) f(\mathbf{X}) w(\mathbf{X})] = E_q[f(\mathbf{X}) w(\mathbf{X})] = E[f(\mathbf{X})]$.

As for the variance, we first observe that $\mathrm{var}(\hat{\mu}) = \mathrm{var}(\hat{\mu}_1) + \mathrm{var}(\hat{\mu}_2)$ where $\mathrm{var}(\hat{\mu}_1) = \mathrm{var}(\mathbf{1}\left(\eta(\mathbf{X} \geq \tau)\right) f(\mathbf{X}) w(\mathbf{X}))/|\mathscr{I}|$ and $\mathrm{var}(\hat{\mu}_2) = \mathrm{var}(\mathbf{1}\left(\eta(\mathbf{X} < \tau)\right) f(\mathbf{X}) w(\mathbf{X}))/|\mathscr{I}^n|$. Replacing $\mathrm{var}(\hat{\mu}_1)$ and $\mathrm{var}(\hat{\mu}_2)$ with unbiased sample variances using the samples $\mathbf{X}_i = \mathbf{x}_i$ we obtain

$$\begin{aligned}\widehat{\mathrm{var}}(\hat{\mu}_1) &= \frac{1}{|\mathscr{I}|(|\mathscr{I}| - 1)} \sum_{i \in \mathscr{I}} \left(\mathbf{1}\left(\eta(\mathbf{x}_i \geq \tau)\right) f(\mathbf{x}_i) w(\mathbf{x}_i) - \bar{h}\right)^2 \\ &= \frac{1}{|\mathscr{I}| - 1}\left(-\bar{h}^2 + \frac{1}{|\mathscr{I}|} \sum_{i \in \mathscr{I}} \left(\mathbf{1}\left(\eta(\mathbf{x}_i \geq \tau)\right) f(\mathbf{x}_i) w(\mathbf{x}_i)\right)^2\right) \\ &= \frac{1}{|\mathscr{I}| - 1}(-\bar{h}^2 + s_{\bar{h}}),\end{aligned}$$

and similarly

$$\widehat{\mathrm{var}}(\hat{\mu}_2) = \frac{1}{|\mathscr{I}^n| - 1}\left(-\bar{r}^2 + \frac{1}{|\mathscr{I}^n|} \sum_{i \in \mathscr{I}_\tau^n} (f(\mathbf{x}_i) w(\mathbf{x}_i))^2\right),$$

where we have used that $\bar{h}$ and $\bar{r}$ are unbiased estimates of $E_q[\hat{\mu}_1]$ and $E_q[\hat{\mu}_2]$, respectively. The expression in (29) is then obtained as $\widehat{\mathrm{var}}(\hat{\mu}_1) + \widehat{\mathrm{var}}(\hat{\mu}_2)$ using that $|\mathscr{I}| = N_0$ and $|\mathscr{I}^n| = nN_0/N_\tau$. □

4.5 The UT-MCIS approximation of $H_{1,k}$, $H_{2,k}$ and $H_{3,k}$

Using the tools introduced in the preceding subsections, we now present how the measures of residual uncertainty, $H_{1,k}$, $H_{2,k}$ and $H_{3,k}$, can be approximated using Monte Carlo simulation with importance sampling (MCIS) combined with the unscented transform (UT) for epistemic uncertainty propagation.

We first let $\hat{\xi}_k(\mathbf{x}, \mathbf{E})$ be the finite-dimensional approximation introduced in Sect. 4.1, with the corresponding failure probability $\hat{\alpha}_k(\mathbf{E}) = \alpha(\hat{\xi}_k(\mathbf{x}, \mathbf{E}))$. We then let $\{(\mathbf{x}_i, w_i, \hat{\eta}_i) \mid i \in \mathscr{I}\}$, $\mathscr{I} = \{1, \ldots, N_0\}$ be a set of samples generated as described in Sect. 4.3, where $\hat{\eta}_i$ is obtained using the UT approximation of $\hat{\xi}_k(\mathbf{x}_i, \mathbf{E})$. We will make use of importance sampling estimates as introduced in Sect. 4.4, where $\mathscr{I}_\tau = \{i \in \mathscr{I} \mid \hat{\eta}_i < \tau\}$, and estimation is based on a small subset $\{(\mathbf{x}_i, w_i, \hat{\eta}_i) \mid i \in \mathscr{I}_\tau^n\}$ where $\mathscr{I}_\tau^n \subset \mathscr{I}_\tau$ and $|\mathscr{I}_\tau^n| = n < N_\tau = |\mathscr{I}_\tau|$.

4.5.1 Approximating $H_{1,k}$

Let $f_i = \mathbf{1}\left(\hat{\eta}_i \leq 0\right)$ for $i \in \bar{\mathscr{I}}_\tau$ and compute $\bar{h}_1$ as in (24). We will let $\{(v_j, \mathbf{e}_j) \mid j = 1, \ldots, M\}$ denote the set of sigma-points as introduced in Sect. 4.2.

For any fixed $\mathbf{e}_j$, the corresponding importance sampling estimate of the failure probability $\hat{\alpha}_k(\mathbf{e}_j)$ is obtained as

$$\hat{\alpha}_k^j = \bar{h}_1 + \frac{N_\tau}{nN_0} \sum_{i \in \mathscr{I}_\tau^n} \mathbf{1}\left(\hat{\xi}_k(\mathbf{x}_i, \mathbf{e}_j) \leq 0\right) w_i, \tag{30}$$

and we let $\hat{H}_{1,k}$ be given by the UT approximation

$$\begin{aligned}\widehat{E}[\hat{\alpha}_k] &= \sum_{j=1}^M v_j \hat{\alpha}_k^j, \\ \hat{H}_{1,k} &= \widehat{\mathrm{var}}[\hat{\alpha}_k] = \sum_{j=1}^M v_j (\hat{\alpha}_k^j - \widehat{E}[\hat{\alpha}_k])^2.\end{aligned} \tag{31}$$

4.5.2 Approximating $H_{2,k}$ and $H_{3,k}$

Both $H_{2,k}$ and $H_{3,k}$ are defined through the function $\gamma_k(\mathbf{x})$, which represents the uncertainty in the sign of $\xi_k(\mathbf{x})$. We will approximate $\gamma_k(\mathbf{x}_i)$ with the following function

$$\hat{\gamma}_k^i = \Phi(\hat{\eta}_i)\Phi(-\hat{\eta}_i), \tag{32}$$

where $\Phi(\cdot)$ is the standard normal CDF. There are two ways of interpreting this approximation. First of all, $\hat{\gamma}_{k,i}$ corresponds to the case where $\hat{\xi}_k(\mathbf{x}_i, \mathbf{E})$ is Gaussian, which may or may not be an appropriate assumption. Alternatively, we can think of $\gamma_k(\mathbf{x})$ as a measure of uncertainty in $\mathbf{1}(\xi_k(\mathbf{x}) \leq 0)$, and any $\gamma(\mathbf{x}) \propto -|\eta(\mathbf{x})| = -|E[\xi_k(\mathbf{x})]|/\sqrt{\text{var}(\xi_k(\mathbf{x}))}$ is reasonable. In this scenario it is natural to consider $\gamma = s(\eta)s(-\eta)$ for some sigmoid function $s(\cdot)$, and the function $\Phi(\cdot)$ in (32) is one such alternative.

For a single approximation of $H_{2,k}$ or $H_{3,k}$ it is really not necessary to split the importance sampling estimate as in (24)–(28), but we will present it in this form as it will be convenient when we consider strategies for optimization. Given $\hat{\gamma}_k^i$ as in (32), we approximate $H_{2,k}$ and $H_{3,k}$ by

$$\begin{aligned} \hat{H}_{2,k} &= \bar{h}_2 + \frac{N_\tau}{nN_0} \sum_{i \in \mathscr{I}_\tau^n} \hat{\gamma}_k^i w_i, \\ \hat{H}_{3,k} &= \left(\bar{h}_3 + \frac{N_\tau}{nN_0} \sum_{i \in \mathscr{I}_\tau^n} \sqrt{\hat{\gamma}_k^i} w_i \right)^2, \end{aligned} \tag{33}$$

where we let $\bar{h}_2 = \bar{h}_3 = 0$. Alternatively, if the intention is to use $H_{2,k}$ and $H_{3,k}$ as upper bounds on $H_{1,k}$, we could let $\bar{h}_2 = \frac{1}{N_0}\Phi(\tau)\Phi(-\tau)\sum w_i$, $\bar{h}_2 = \frac{1}{N_0}\sqrt{\Phi(\tau)\Phi(-\tau)}\sum w_i$ where the sums are over $i \in \mathscr{I}_\tau$.

5 Numerical procedure for one-step lookahead optimization

In the one-step lookahead case, the optimal decision d_k at each time step k is found by solving the following optimization problem

$$d_k = \arg\min_{d \in \mathbb{D}} J_{i,k}(d) \text{ for } i = 1, 2, \text{ or } 3, \tag{34}$$

where $J_{i,k}(d)$ is the relevant acquisition function as defined in (13). We propose a procedure where we make use of a UT-MCIS approximation of $J_{i,k}(d)$ to find an approximate solution to (34). This will build on the approximation of $H_{i,k}$ introduced in Sect. 4, but where we now also make use of the predictive model δ to approximate expectations with respect to future values of $H_{i,k+1}$.

In Sects. 5.1 and 5.2 we present how the UT-MCIS approximation of $J_{i,k}(d)$ is obtained, and in Sect. 5.3 we propose a criterion for determining when the sequence of experiments should be stopped. The final algorithm is summarized in Sect. 5.4

5.1 The probabilistic model $(\hat{\xi}_k, \hat{\delta}_k)$

Starting with some probabilistic model (ξ_k, δ_k), recall that ξ_k represents uncertainty about the *performance* of the system under consideration, and δ_k represents uncertainty with respect to *outcomes* of certain *decisions*. We have already discussed how to obtain a finite-dimensional approximation of ξ_k, and likewise, this will also be needed for δ_k.

Assuming δ_k is square integrable, we will make use of the same type of finite-dimensional approximation as the one introduced for ξ_k in Sect. 4.1. In this way, we end up with two finite-dimensional $\mathscr{E}_k$-measurable random variables $\mathbf{E}^\xi$ and $\mathbf{E}^\delta$, which in turn determine the approximations $\hat{\xi}_k(\mathbf{x}, \mathbf{E}^\xi)$ and $\hat{\delta}_k(d, \mathbf{E}^\delta)$, where both $\hat{\xi}_k(\mathbf{x}, \mathbf{e})$ and $\hat{\delta}_k(d, \mathbf{e})$ are deterministic functions for $\mathbf{e}$ fixed. Here $\mathbf{E}^\xi$ and $\mathbf{E}^\delta$ are generally not independent.

Remark 8 Note that if $\delta(d)$ is a function of some of the uncertain sub-components of ξ, then we might already have a finite-dimensional approximation of δ available.

Consider for instance the model in Example 4 and the discussion in the end of Sect. 4.1. In this case, $\hat{\xi}$ is obtained as a function of the finite-dimensional approximation $\hat{y}_1(\mathbf{x}, \mathbf{E})$ of a sub-component $\tilde{y}_1(\mathbf{x})$, and $\delta(d)$ is given as $\delta(d(\mathbf{x})) = \tilde{y}_1(\mathbf{x}) + \epsilon(\mathbf{x})$. Hence, all we need is to find a finite-dimensional representation of the noise $\epsilon(\mathbf{x})$. But observational noise such as $\epsilon(\mathbf{x})$ is often described as a function of $\mathbf{x}$ and some 1-dimensional random variable, in which case no additional approximation will be needed.

We will let $(\hat{\xi}_k, \hat{\delta}_k)$ denote the finite-dimensional approximation of (ξ_k, δ_k) corresponding to a finite-dimensional random variable $\mathbf{E} = (\mathbf{E}^\xi, \mathbf{E}^\delta)$, and where $(\hat{\xi}_0, \hat{\delta}_0)$ is the initial model that is used as input for determining the first decision d_1.

Remark 9 In the canonical case where a surrogate $\tilde{y}(\mathbf{x})$ is used to represent some unknown function $y(\mathbf{x})$, an initial set of experiments is often performed to establish $\tilde{y}(\mathbf{x})$ before any sequential strategy is started. For instance, in the case where evaluation of $y(\mathbf{x})$ means running deterministic computer code, it is normal to set up a space-filling initial design using e.g. Latin Hypercube Sampling.

When $\tilde{y}(\mathbf{x})$ is a Gaussian process model as described in Appendix A, specific mean and covariance functions may also be selected based on knowledge or assumptions about

the phenomenon that is being modelled by $y(\mathbf{x})$. For estimation of failure probabilities it is also convenient to make use of conservative prior mean values. That is, prior to any experiment $\tilde{y}(\mathbf{x})$ will correspond to a value associated with poor structural performance (small ξ), such that $\alpha(\xi)$ will be biased towards higher failure probabilities in the absence of experimental evidence. This reasonable from a safety perspective, and also numerically as larger failure probabilities are easier to estimate.

5.2 Acquisition function approximation

To find an approximate solution to the optimization problem (34), we will replace the acquisition function $J_{i,k}(d)$ with an approximation $\hat{J}_{i,k}(d)$. Recall that $J_{i,k}(d)$ as defined in (13) is a function of $E_{k,d}\left[H_{i,k+1}\right]$, where $E_{k,d}$ is the conditional expectation with respect to $\mathscr{E}_k$ with $d_k = d$. In Sect. 4 we introduced an approximation $H_{i,k}$, and we will make use of the same idea to approximate $E_{k,d}\left[H_{i,k+1}\right]$.

Assume k experiments have been performed, giving rise to the model (ξ_k, δ_k) and the approximation $(\hat{\xi}_k, \hat{\delta}_k)$. If we consider the kth decision $d_k = d$, then $H_{i,k+1}$ is a priori a $\delta_k(d)$-measurable random variable. That is, $H_{i,k+1}$ is a function of $\delta_k(d)$, and we are interested in the expectation $E_{k,d}\left[H_{i,k+1}\right] = E\left[H_{i,k+1}(\delta_k(d))\right]$. To approximate this quantity, we can make use of $(\hat{\xi}_k, \hat{\delta}_k)$ in the place of (ξ_k, δ_k), in which case $H_{i,k+1}$ becomes a function of $\mathbf{E}$ and we can approximate its expectation using UT.

The approximate acquisition functions are then given as

$$\hat{J}_{i,k}(d) = \lambda(d)\widehat{E}_{k,d}[\hat{H}_{i,k+1}], \tag{35}$$

where $\widehat{E}_{k,d}[\hat{H}_{i,k+1}]$ is obtained as follows:

5.2.1 Generating samples of $\hat{\xi}_{k+1}$

Let $\{(v_j^{\xi}, \mathbf{e}_j^{\xi}) \mid j = 1, \ldots, M^{\xi}\}$ and $\{(v_m^{\delta}, \mathbf{e}_m^{\delta}) \mid m = 1, \ldots, M^{\delta}\}$ denote sigma-points as introduced in Sect. 4.2 for $\mathbf{E}^{\xi}$ and $\mathbf{E}^{\delta}$, respectively. We then let $\{(\mathbf{x}_i, w_i, \hat{\eta}_i) \mid i \in \mathscr{I}\}$, $\mathscr{I} = \{1, \ldots, N_0\}$ be a set of samples generated as described in Sect. 4.3, where $\hat{\eta}_i$ is obtained using the UT approximation of $\hat{\xi}_k(\mathbf{x}_i, \mathbf{E}^{\xi})$. As for the approximation of $H_{i,k}$ discussed in Sect. 4.5, we let $\mathscr{I}_{\tau} = \{(\mathbf{x}_i, w_i, \hat{\eta}_i) \mid \hat{\eta}_i < \tau\}$ and define the subset $\mathscr{I}_{\tau}^n \subseteq \mathscr{I}_{\tau}$ of size n.

The approximations of $E_{k,d}\left[H_{i,k+1}\right]$ for $i = 1, 2$ and 3 will all be based on samples of $\hat{\xi}_{k+1}$ of the form

$$\hat{\xi}_{k+1}^{m,i,j} = \hat{\xi}_{k+1}(\mathbf{x}, \mathbf{e}_j^{\xi}, d, \mathbf{e}_m^{\delta}), \tag{36}$$

where $\hat{\xi}_{k+1}(\mathbf{x}, \mathbf{e}_j^{\xi}, d, \mathbf{e}_m^{\delta})$ is the finite-dimensional approximation of $\xi_k | d_k = d, o_k = \hat{\delta}(\mathbf{e}_m^{\delta})$ evaluated at $(\mathbf{x}, \mathbf{e}_j^{\xi})$. The scalar $\hat{\xi}_{k+1}^{m,i,j}$ is computed for all $j = 1, \ldots, M^{\xi}$, $m = 1, \ldots, M^{\delta}$ and $i =\in \mathscr{I}_{\tau}^n$. As in Sect. 4.5 we set $\bar{h}_2 = \bar{h}_3 = 0$ and compute $\bar{h}_1$ as in (24) with $f_i = \mathbf{1}\left(\hat{\eta}_i \leq 0\right)$ for $i \notin \mathscr{I}_{\tau}$.

5.2.2 The UT-MCIS approximation of $E_{k,d}\left[H_{1,k+1}\right]$

The approximation $\widehat{E}_{k,d}[\hat{H}_{1,k+1}]$ is just a weighted sum of the terms in (36), but for clarity we present it in the following three steps

MCIS of $\alpha(\hat{\xi}_{k+1})$:

$$\hat{\alpha}_{k+1}^{m,j} = \bar{h}_1 + \frac{N_{\tau}}{nN_0} \sum_{i \in \mathscr{I}_{\tau}^n} \mathbf{1}\left(\hat{\xi}_{k+1}^{m,i,j} \leq 0\right) w_i, \tag{37}$$

UT of $H_{1,k+1}$:

$$\hat{H}_{1,k+1}^{m} = \sum_{j=1}^{M^{\xi}} v_j^{\xi} (\hat{\alpha}_{k+1}^{m,j})^2 - \left(\sum_{j=1}^{M^{\xi}} v_j^{\xi} \hat{\alpha}_{k+1}^{m,j} \right)^2, \tag{38}$$

UT of $E_{k,d}\left[H_{1,k+1}\right]$:

$$\widehat{E}_{k,d}[\hat{H}_{1,k+1}] = \sum_{m=1}^{M^{\delta}} v_m^{\delta} \hat{H}_{1,k+1}^{m}. \tag{39}$$

5.2.3 The UT-MCIS approximation of $E_{k,d}\left[H_{2,k+1}\right]$ and $E_{k,d}\left[H_{3,k+1}\right]$

The weighted sums that gives the approximations of $E_{k,d}\left[H_{2,k+1}\right]$ and $E_{k,d}\left[H_{3,k+1}\right]$ can be obtained as follows

UT of $E[\hat{\xi}_{k+1}(\mathbf{x}_i)]$: $\quad \hat{\mu}_{k+1}^{i,m} = \sum_{j=1}^{M^{\xi}} v_j^{\xi} \hat{\xi}_{k+1}^{m,i,j}$, (40)

UT of $\mathrm{var}[\hat{\xi}_{k+1}(\mathbf{x}_i)]$:

$$(\hat{\sigma}_{k+1}^{i,m})^2 = \sum_{j=1}^{M^{\xi}} v_j^{\xi} (\hat{\xi}_{k+1}^{m,i,j} - \hat{\mu}_{k+1}^{i,m})^2, \tag{41}$$

Using Φ to approximate $\hat{\gamma}_{k+1}(\xi_i)$:

$$\hat{\gamma}_{k+1}^{i,m} = \Phi(\hat{\eta}_{k+1}^{i,m})\Phi(-\hat{\eta}_{k+1}^{i,m}), \quad \hat{\eta}_{k+1}^{i,m} = \hat{\mu}_{k+1}^{i,m} / \hat{\sigma}_{k+1}^{i,m} \tag{42}$$

MCIS of $H_{2,k+1}$:

$$\hat{H}_{2,k+1}^{m} = \bar{h}_2 + \frac{N_{\tau}}{nN_0} \sum_{i \in \mathscr{I}_{\tau}^n} \hat{\gamma}_{k+1}^{i,m} w_i, \tag{43}$$

MCIS of $H_{3,k+1}$:

$$\sqrt{\hat{H}^m_{3,k+1}} = \bar{h}_3 + \frac{N_\tau}{nN_0} \sum_{i \in \mathscr{I}^n_\tau} \sqrt{\hat{\gamma}^{i,m}_{k+1}} w_i, \tag{44}$$

and where $\widehat{E}_{k,d}[\hat{H}_{2,k+1}]$ and $\widehat{E}_{k,d}[\hat{H}_{3,k+1}]$ are obtained with the same formula as for $\widehat{E}_{k,d}[\hat{H}_{1,k+1}]$ in (39).

Remark 10 The number of model updates and function evaluations needed to generate the set $\{\hat{\xi}^{m,i,j}_{k+1}\}$ are M^δ and $nM^\xi M^\delta$. We can view this as a discretization of the system dynamics, where there are only M^δ possible future scenarios corresponding to the decision $d_k = d$, which are given by the model updates $\xi_k \to \xi_{k+1}(\mathbf{e}^\delta_m) = \xi_k | d_k = d, o_k = \hat{\delta}_k(\mathbf{e}^\delta_m)$. The samples in (36) are the ones needed for approximating the measure of residual uncertainty corresponding to $\xi_{k+1}(\mathbf{e}^\delta_m)$ for each $m = 1, \ldots, M^\delta$.

Moreover, although the approximations $\widehat{E}_{k,d}[\hat{H}_{i,k+1}]$ are presented as weighted sums of the $nM^\xi M^\delta$ terms $\hat{\xi}^{m,i,j}_{k+1}$, this can also be obtained from a sequence of nested loops for a more memory efficient implementation. See for instance the schematic illustration in Fig. 3.

5.3 Stopping criterion

For design strategies that make use of heuristic acquisition functions, it can be challenging to determine an appropriate stopping criterion. Here, we have considered the approximation $\hat{H}_{1,k}$ which has a natural interpretation. Hence, even if we make use of a criteria such as $\hat{H}_{2,k}$ or $\hat{H}_{3,k}$ to determine the next optimal decision, it makes sense to use $\hat{H}_{1,k}$ as an indicator of when the potential uncertainty reduction from future experiments is diminishing.

We will let $\widehat{E}[\hat{\alpha}_k]$ and $\hat{H}_{1,k}$ be given as in (31), and define

$$\hat{V}_k = \frac{\sqrt{\hat{H}_{1,k}}}{\widehat{E}[\hat{\alpha}_k]}. \tag{45}$$

Then $\hat{V}_k$ is the UT-MCIS approximation of the *coefficient of variation* of the failure probability α_k with respect to epistemic uncertainty. We will let $\hat{V}_k \leq V_{\max}$ for some threshold $V_{\max}$ serve as a criterion for stopping the iteration procedure, in the case where a predefined maximum number of iterations $K_{\max}$ has not already been reached.

Remark 11 The coefficient of variation is often used as a numerical criterion for convergence in Monte Carlo simulation. In structural reliability analysis, a coefficient of variation below 0.05 is often used as an acceptable level for failure probability estimation.

Note also that the criterion $\hat{V}_k \leq V_{\max}$ for arbitrary $V_{\max} \geq 0$ implicitly assumes that the epistemic uncertainty can be reduced to zero in the limit. If this is not the case, one might instead consider stopping when $\hat{V}_k$ is no longer decreasing. A different stopping criterion is also considered in Sect. 6.4.

5.4 Algorithm

The complete procedure for myopic/one-step lookahead optimization is summarized in Algorithm 1. Note that for simplicity the number of MCIS samples N_0 and n are specified as input, but one may also consider deciding these using (28) and (29). Using a standard technique in Monte Carlo simulation, one could keep increasing N_0 and n until the coefficient of variation (std/mean) of the relevant estimator is sufficiently small.

Algorithm 1: One-step lookahead optimization

input: Model and sigma-points: $(\hat{\xi}_0, \hat{\delta}_0)$ and $\{(v^\xi_j, \mathbf{e}^\xi_j)\}$, $\{(v^\delta_j, \mathbf{e}^\delta_j)\}$.
Number of samples for UT-MCIS and threshold: $N_0, n \in \mathbb{N}$ and $\tau > 0$,
Max number of iterations and convergence criteria: $K_{\max}$ and $V_{\max}$.

for $k = 0$ **to** $K_{max} - 1$ **do**
- (1) Generate samples $\{(\mathbf{x}_i, w_i, \hat{\eta}_i)\}$ as described in Section 4.3 and compute $\bar{h}_1 = \frac{1}{N_0} \sum_{|\hat{\eta}_i| \geq \tau} \mathbf{1}(\hat{\eta}_i \leq 0) w_i$.
- (2) Compute $\hat{V}_k$ as in (45)
- **if** $\hat{V}_k > V_{max}$ **then**
  - (3) Compute the set $\{\hat{\xi}^{m,i,j}_{k+1}\}$ as in (36) and define the function $\hat{J}_{i,k}(d)$ as in (35) (for i = 1, 2, or 3 depending on the acquisition function of choice)
  - (4) Find the optimal decision: $d_k = \arg\min_{d \in \mathbb{D}} \hat{J}_{i,k}(d)$
  - (5) Make decision d_k and obtain (d_k, o_k)
  - (6) Update model $(\hat{\xi}_{k+1}, \hat{\delta}_{k+1}) = (\hat{\xi}_k, \hat{\delta}_k) | (d_k, o_k)$
- **else**
  - Break. Convergence has been reached before $K_{\max}$ iterations.

6 Numerical experiments

Here we present a few numerical experiments using the algorithm for one-step lookahead optimal design presented in Sect. 5.4. Four experiments are presented, each with its own objective:

(1) Section 6.1: A toy example in 1d for conceptual illustration of the sequential design procedure.
(2) Section 6.2: A hierarchical model with multiple 'expensive' sub-components.
(3) Section 6.3: A non-hierarchical benchmark problem for comparison against alternative strategies.

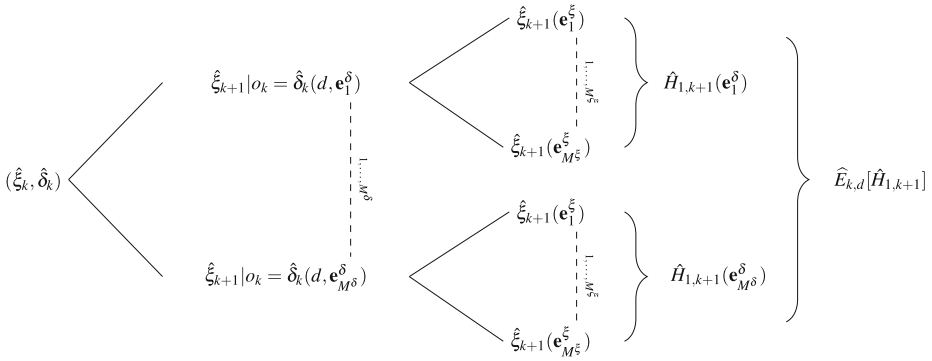


Fig. 3 Illustration of how $\widehat{E}_{k,d}[\hat{H}_{1,k+1}]$ is obtained using UT over epistemic uncertainties. Here $\hat{H}_{1,k+1}(\mathbf{e}_m^\delta)$ for $m = 1, \ldots, M^\xi$ is obtained from the MCIS estimates of $\alpha(\hat{\xi}_{k+1}(\mathbf{e}_j^\xi))$

(4) Section 6.4 A model that is more in resemblance of a realistic application in structural reliability analysis, where we introduce different *types* of decisions by considering both probabilistic function approximation and Bayesian inference of model parameters through measurements with noise.

All numerical experiments have been performed using Algorithm 1 with the parameters $\tau = 3$, $N_0 = 10^4$, $n = 10^3$ and $V_{\max} = 0.05$. This choice of $V_{\max}$ corresponds to a 5% coefficient of variation on the estimated failure probability, and $\tau = 3$ should give a reasonable coverage for importance sampling (from Proposition 41 the probability of misclassification is less than 0.2 in the extreme case (Chebyshev) and less than 0.003 under the Gaussianity assumption). The number of samples, N_0 and n was chosen to make evaluation of the acquisition function reasonably cheap, and the choice $N_0 = 10^4$, $n = 10^3$ worked well in all of our experiments. Note that for final estimates of the failure probability, after an optimal decision has been found, a larger number of samples may be used for increased accuracy. The probabilistic surrogate models used in the examples are all Gaussian process (GP) models with Matérn 5/2 covariance. A short summary of the relevant Gaussian process theory is given in Appendix A.

6.1 Example 1: Illustrative 1d example

To illustrate the one-step lookahead procedure, we present a simple 1d example similar to the one given in (Bect et al. 2012), where we aim to emulate the limit state function

$$g(x) = 1 - \big((0.4x - 0.3)^2 + \exp(-11.534|x|^{1.95}) + \exp(-5(x - 0.8)^2)\big). \tag{46}$$

We assume that $g(x)$ can be evaluated at any $x \in \mathbb{R}$ without error, but that function evaluations are expensive. We will let $\xi(x)$ be the probabilistic surrogate in the form of a Gaussian process, where we use a prior mean $\mu(x) = -0.5$ together with a Matérn 5/2 covariance function with fixed kernel variance $\sigma_c^2 = 0.1$ and length scale $l = 0.5$.

We assume that X follows Normal distribution with mean $\mu_X = -0.5$ and standard deviation $\sigma_X = 0.2$, and our goal is to estimate $\alpha(g) = P(g(X) \le 0)$ using only a small number of evaluations of $g(\cdot)$. The set of decisions is therefore $\mathbb{D} = \cup_x\{\text{evaluate } g(x)\}$ with respective outcomes $o(x) = g(x)$, and a predictive model for outcomes given as $\delta(x) = \xi(x)$.

Using a large number of samples of $g(X)$ we estimate $\alpha(g) \approx 0.0234$, and we will consider this as the 'true' failure probability for comparison.

We initiate ξ by evaluating $g(x)$ at $x = \mu_X$. For subsequent function evaluations, we minimize the expected variance in the failure probability. I.e. we minimize the acquisition function $J_{1,k}$ given in (13) with $\lambda \equiv 1$. For comparison we also evaluate $J_{2,k}$ and $J_{3,k}$, and in this example it seems that all three acquisition functions would perform equally well. Figure 4 shows ξ_k and the corresponding three acquisition functions for the first few experiments, and Fig. 5 shows how $\alpha(\xi_k)$ evolves before converging after $k = 3$ iterations.

6.2 Example 2: A 3 layer hierarchical model with 7d input

In this example we consider the structural reliability benchmark problem given as problem RP38 in (Rozsas and Slobbe 2019). Here, $\mathbf{x} = (x_1, \ldots, x_7) \in \mathbb{X} = \mathbb{R}^7$, and the limit state function $g(\mathbf{x})$ can be written in terms of intermediate variables as follows:

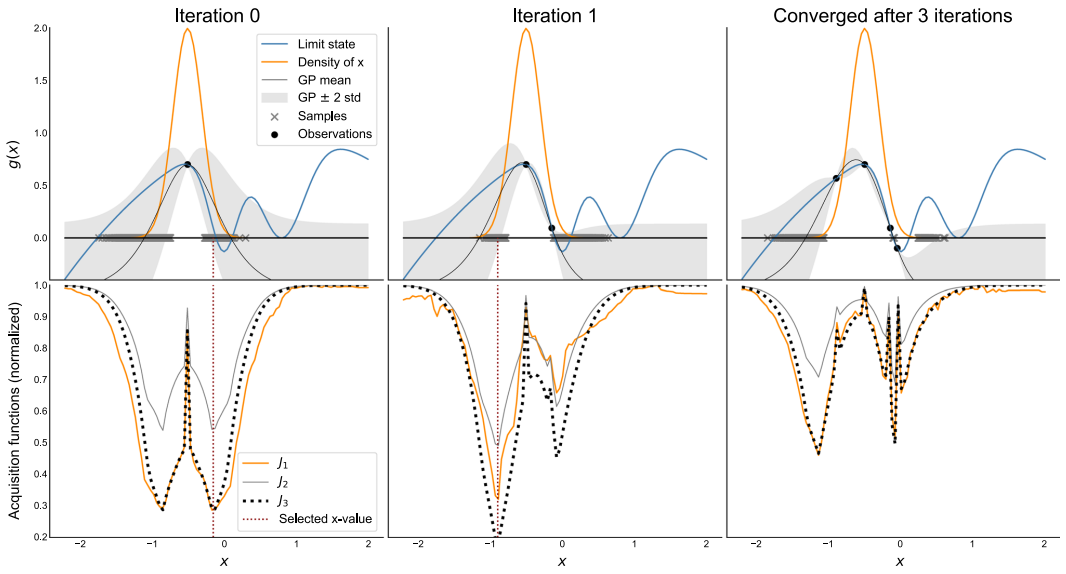


Fig. 4 (Example 1) The top row shows the true limit state function $g(x)$, the probability density of X, and the mean $\pm$ 2 standard deviations of the GP ξ_k for $k = 0$, 1 and 3. The samples indicated with $\times$ on the x-axis are used in the importance sampling estimates of $J_{1,k}$, $J_{2,k}$ and $J_{3,k}$ that are shown in the bottom row

$$y_1(\mathbf{x}) = \frac{x_1 x_2^3}{2c_4 x_3^3}, \quad y_2(\mathbf{x}) = \frac{x_4^2}{c_2},$$
$$y_3(\mathbf{x}) = -4x_5 x_6 x_7^2 + x_4(x_6 + 4x_5 + 2x_6 x_7),$$
$$y_4(\mathbf{x}) = x_4 x_5 (x_4 + x_6 + 2x_6 x_7),$$
$$z_1(\mathbf{y}) = \frac{c_4 y_1 y_2}{c_3}, \quad g(\mathbf{y}, z_1) = 1 - \frac{c_2 c_3 z_1 + c_4 y_1 y_3}{c_1 y_4},$$

where c_1, c_2, c_3 and c_4 are constants: $c_1 = 15.59 \cdot 10^4$, $c_2 = 6 \cdot 10^4$, $c_3 = 2 \cdot 10^5$, $c_4 = 1 \cdot 10^6$.

Figure 6 shows a graphical representation of how $g(\mathbf{x})$ depends on the intermediate variables z_1, y_1, y_3 and y_4. We will assume that the functions $y_2(\mathbf{x})$ and $z_1(\mathbf{y})$ will require probabilistic surrogates, where $y_2(\mathbf{x})$ and $z_1(\mathbf{y})$ can be evaluated without error for any input $\mathbf{x}$ and $\mathbf{y}$. We will also assume that there is no difference in the cost associated with evaluating y_2 or z_1, and our goal is to estimate the failure probability $\alpha(g)$ while keeping the total number of function evaluations of $y_2(\mathbf{x})$ and $z_1(\mathbf{y})$ as small as possible. Note that the effective domain of y_2 is 1-dimensional and the effective domain of z_1 is 2-dimensional. Hence, using surrogates for y_2 and z_1 should be much more efficient than building a single surrogate for g using samples $g(\mathbf{x}_i)$.

As for the random variable $\mathbf{X} = (X_1, \ldots, X_7)$, we assume that all X_i's are independent and normally distributed, $X_i \sim \mathcal{N}(\mu_i, \sigma_i)$, with means $\mu_1 = 350$, $\mu_2 = 50.8$, $\mu_3 = 3.81$, $\mu_4 = 173$, $\mu_5 = 9.38$, $\mu_6 = 33.1$, $\mu_7 = 0.036$, and standard deviation $\sigma_i = 0.1\mu_i$. The 'true' failure probability we aim to estimate is $\alpha(g) \approx 8.1 \cdot 10^{-3}$.

Assuming y_2 and z_1 are expensive to evaluate, we introduce two Matérn 5/2 GP surrogates, $\tilde{y}_2$ and $\tilde{z}_1$. The initial kernel parameters are $(\sigma_c^2 = 0.03, l = 20)$ and $(\sigma_c^2 = 2, l = [0.5, 0.5])$ for $\tilde{y}_2$ and $\tilde{z}_1$, respectively. These parameters may be updated by maximum likelihood estimation, but not until a few observations (resp. 2 and 5) have been made. We know that large values of y_2 or z_1 will result in poor structural performance (small $g(\mathbf{x})$), so we initiate the GP models with conservative prior means of $\mu(\mathbf{x}) = 1$ for $\tilde{y}_2$ and $\mu(\mathbf{y}) = 5$ for $\tilde{z}_1$. Both models are initially updated with one observation each, $\tilde{y}_2(\mu_4) = y_2^0$ and $\tilde{z}_1(y_1^0, y_2^0) = z_1^0$ for $y_1^0 = y_1(\mu_1, \mu_2, \mu_3)$, $y_2^0 = y_2(\mu_4)$ and $z_1^0 = z_1(y_1^0, y_2^0)$.

In this example, we would then define $\xi(\mathbf{x}) = g(y_1, \tilde{z}_1, \tilde{y}_2, y_3, y_4)$. With respect to $\tilde{z}_1$, there is a set of possible decisions for uncertainty reduction, namely $\mathbb{D} = \cup_{y_1,y_2}\{\text{evaluate } z_1(y_1, y_2)\}$, with a corresponding set of observations $\mathbb{O} = \cup_{y_1,y_2}\{z_1(y_1, y_2)\}$, and a predictive model $\delta(y_1, y_2) = \tilde{z}_1(y_1, y_2)$. Similarly, we obtain a set of decisions, outcomes and a predictive model for $\tilde{y}_2$, and we can update $\mathbb{D}$, $\mathbb{O}$ and $\delta(d)$ accordingly.

Convergence was reached at iteration $k = 10$, after 2 additional evaluations of y_2 and 8 additional evaluations of z_1. Figure 7 shows the updated surrogate models, $\tilde{y}_2|I_k$ and

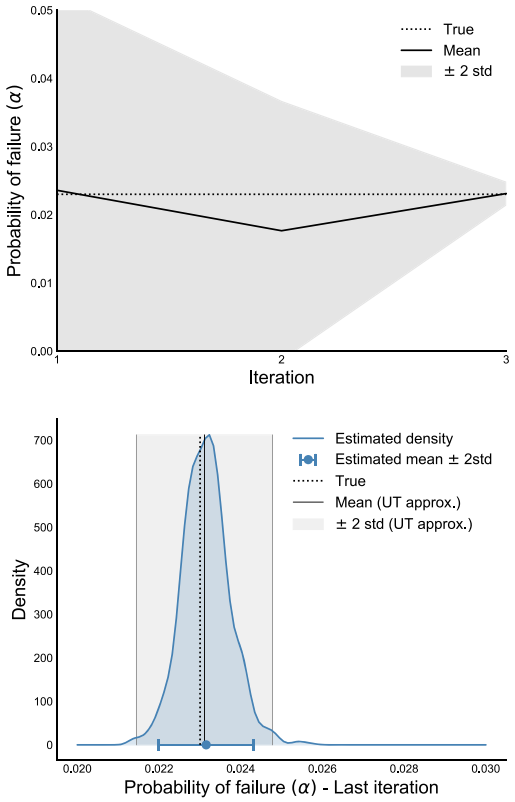


Fig. 5 (Example 1) Top: Mean ± 2 standard deviations of α_k after k iterations, as computed using the approximation described in Sect. 4. Bottom: The distribution of α_k at the final iteration $k = 3$, estimated from a double-loop Monte Carlo (i.e. by sampling from $\alpha(\xi_k)$ without any approximation)

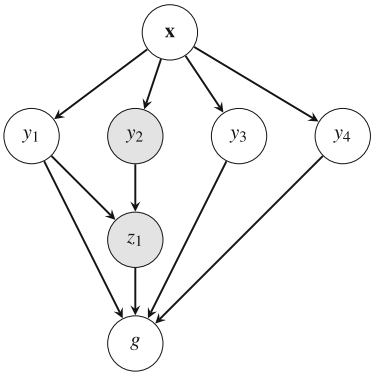


Fig. 6 (Example 2) Hierarchical representation of $g(\mathbf{x})$. We assume that the intermediate variables $y_2(\mathbf{x})$ and $z_1(\mathbf{y})$ are expensive to evaluate

$\tilde{z}_1|I_k$ for $k = 10$, and Fig. 8 shows how $\alpha(\xi_k)$ evolves with each iteration. At each iteration, the next experiment was decided by minimizing the acquisition function $J_{3,k}$ with respect to updating each of the two surrogate models.

6.3 Example 3: The 4 branch system

Here we consider the 'four branch system', a classical 2D benchmark problem given by the limit state

$$g(\mathbf{x}) = \min \begin{Bmatrix} 3 + 0.1(x_1 - x_2)^2 - (x_1 + x_2)\sqrt{2}; \\ 3 + 0.1(x_1 - x_2)^2 + (x_1 + x_2)\sqrt{2}; \\ (x_1 - x_2) + 6\sqrt{2}; \\ (x_2 - x_1) + 6\sqrt{2} \end{Bmatrix}, \quad (47)$$

and where x_1 and x_2 are independent standard normal variables. In this example we will not write (47) as an hierarchical model, in order to compare our method with other alternatives that are tailored to to non-hierarchical setting. We therefore let $\xi(\mathbf{x})$ be a Gaussian process surrogate of $g(\mathbf{x})$, constructed from observations $(\mathbf{x}_i, g(\mathbf{x}_i))$. For the initial 'conservative' Gaussian process we select a prior mean of -1, a Matérn 5/2 kernel with parameters of $(\sigma_c = 1, l = 3)$, and condition on the initial observation $(\mathbf{0}, g(\mathbf{0}))$.

According to Huang et al. (2017), the method called AK-MCS developed by Echard et al. (2011) is considered a typical and mature approach, and should therefore be a suitable candidate for comparison. In addition, Echard et al. (2011) also provide the results from using a number of other alternatives proposed in Schueremans and Gemert (2005). Table 3 gives a summary of the results from Echard et al. (2011), together with the those obtained using the approach presented in this paper.

Our results in Table 3 are obtained using Algorithm 1 with three different stopping criteria, $V_{max} = 0.1$, $V_{max} = 0.05$ and $V_{max} = 0.025$. Instead of point estimates we provide prediction intervals, which in this example contain the 'true' failure probability obtained with Monte Carlo in each scenario. From a practical perspective, even the estimates obtained using only 35 evaluations ($V_{max} = 0.1$) of (47) seems acceptable. If we were to use the mean + 2 standard deviations as a conservative estimate, the relative error with respect to the 'true' failure probability is still less than 3 %. After an additional 30 iterations, this number drops to 0.65 %. Hence, our approach performs well with respect to the alternatives considered in (Echard et al. 2011; Schueremans and Gemert 2005). It should also be noted that the Directional Sampling alternative in Table 3 is a method that is especially suitable for the specific 'radial' type of limit state surfaces as considered here, and a this level of performance is not expected in general.

Optimization was performed using the approximate acquisition function $\hat{J}_{3,k}$, and Fig. 9 shows how the sequence of

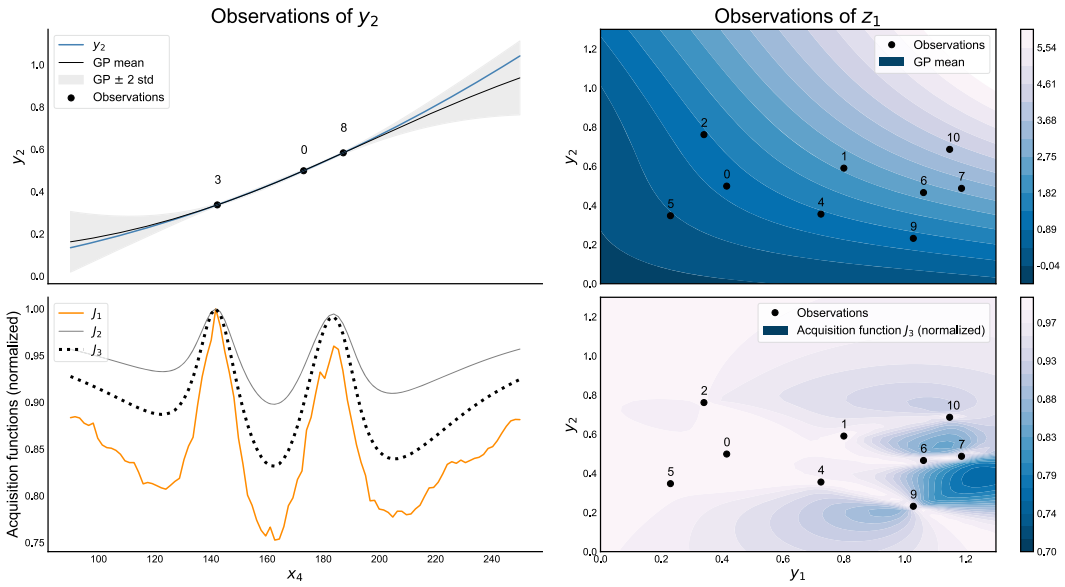


Fig. 7 (Example 2) The top row shows the GP models $\tilde{y}_2$ and $\tilde{z}_1$ with respect to P_k for $k = 10$. The number above each observations is the iteration index k, and convergence was obtained after 2 evaluations of y_2 and 8 evaluations of z_1. The final acquisition functions are shown in the bottom row

Table 3 (Example 3) Table 2 from Echard et al. (2011), where we have appended the method from this paper (UT-MCIS) in the bottom row. The reported failure probabilities ($\widehat{p}_f$) are the estimated mean $\pm$ 2 standard deviations of $\alpha(\xi_k)$ for $k = 35$ (stopped at $\hat{V}_k \leq 0.1$), $k = 48$ (stopped at $\hat{V}_k \leq 0.05$), and $k = 65$ (stopped at $\hat{V}_k \leq 0.025$)

Method	N_{call}	$\widehat{p}_f \times 10^3$
Monte Carlo	10^6	4.416
AK-MCS+U	126	4.416
AK-MCS+EFF	124	4.412
Directional Sampling (DS)	52	4.5
DS + Response Surface	1745	5.0
DS + Spline	145	2.4
DS + Neural Network	165	4.1
Importance Sampling (IS)	1469	4.9
DS + Response Surface	1375	4.5
IS + Spline	428	4.5
IS + Neural Network	52	5.7
UT-MCIS ($V_{max} = 2.5\%$)	65	(4.347−4.444)
UT-MCIS ($V_{max} = 5\%$)	48	(4.288−4.470)
UT-MCIS ($V_{max} = 10\%$)	35	(4.163−4.547)

observations are located with respect to the failure set $g = 0$. The resulting sequence of failure probabilities after each iteration is illustrated in Fig. 10.

6.4 Example 4: Corroded pipeline example

To give an example of a scenario where there are different *types* of experiments, we consider a probabilistic model which is recommended for engineering assessment of offshore pipelines with corrosion (DNV GL 2017). The failure mode under consideration is where a pipeline bursts, when the pipeline's ability to withstand the high internal pressure has been reduced as a consequence of corrosion.

6.4.1 The structural reliability model

Figure 11 shows a graphical representation of the structural reliability model. Here, a steel **pipeline** is characterised by the outer diameter (D [mm]), the wall thickness (t [mm]) and the ultimate tensile strength (s [MPa]). In this example we let $D = 800$, $t \sim \mathcal{N}(\mu = 20, \text{cov} = 0.03)$, and $s \sim \mathcal{N}(\mu = 545, \text{cov} = 0.06)$, where cov is the coefficient of variation (standard deviation / mean).

The pipeline contains a rectangular shaped **defect** with a given depth (d [mm]) and length (l [mm]), where $l \sim \mathcal{N}(\mu = 200, \sigma^2 = 1.49)$ and where d will be inferred from observations. Given a pipeline (D, t, s) with a defect (d, l), we can determine the pipeline's pressure resistance **capacity** (the maximum differential pressure the pipeline can withstand before bursting). We let p_{FE} [MPa] denote the capacity coming from a Finite Element simulation of the physical phe-

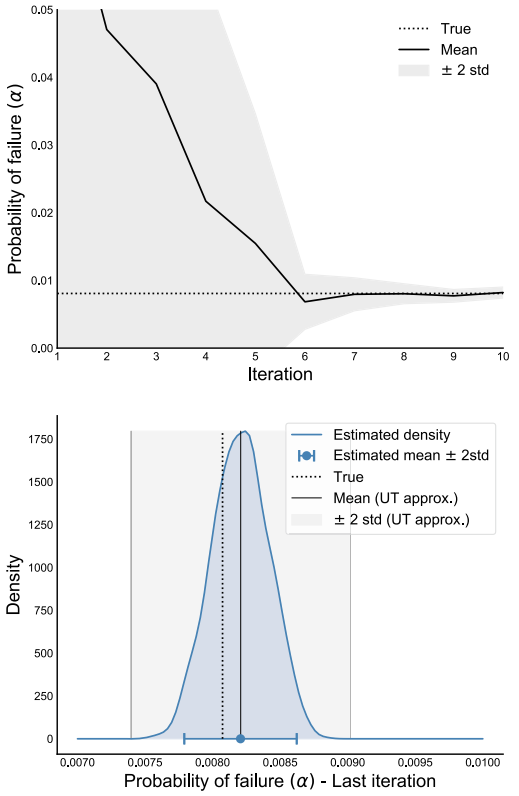


Fig. 8 (Example 2) Top: Mean ± 2 standard deviations of α_k after k iterations, as computed using the approximation described in Sect. 4. Bottom: The distribution of α_k at the final iteration $k = 10$, estimated from a double-loop Monte Carlo (i.e. by sampling from $\alpha(\xi_k)$ without any approximation)

nomenon. From the theoretical capacity p_{FE}, we model the true pipeline capacity as $p_c = X_{\mathrm{m}} \cdot p_{\mathrm{FE}}$, where X_{m} is the **model discrepancy**, $X_{\mathrm{m}} \sim \mathcal{N}(\mu_{\mathrm{m}}, \sigma_{\mathrm{m}}^2)$. For simplicity we have assumed that X_{m} does not depend on the type of pipeline and defect, and we will also assume that $\sigma_{\mathrm{m}} = 0.1$, where only the mean μ_{m} will be inferred from observations of the form p_c/p_{FE}. Finally, the pressure **load** (in MPa) is modelled as a Gumbel distribution with mean 15.75 and standard deviation 0.4725. The limit state representing the transition to failure is then given as $g = p_c - p_d$.

6.4.2 Different types of decisions

We consider the following three types of decisions

1. *Defect measurement* We assume that unbiased measurements of the relative depth d/t can be obtained.

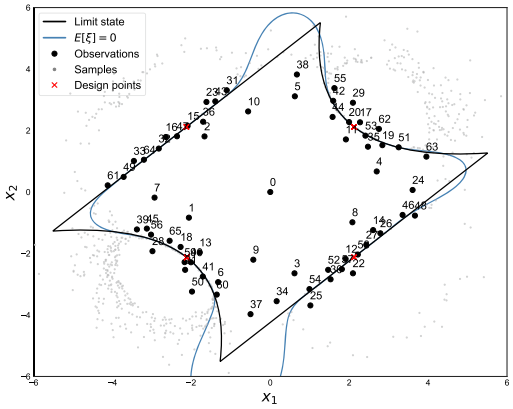


Fig. 9 (Example 3) The limit state (47) together with the expected failure surface $E[\xi_{65}]$ and the 65 observations collected before convergence at $\hat{V}_{65} < 0.025$. The proposal distribution $q_{\mathbf{X}}$ used for importance sampling is a mixture of Gaussian random variables centered at the four design points (×) as described in Appendix B. The pruned samples shown in the figure are mostly located around $E[\xi_{65}] = 0$ and in other regions where the level set $\xi_{65} = 0$ is uncertain

   The measurements come with additive Gaussian noise, $\epsilon \sim \mathcal{N}(0, \sigma_{d/t}^2)$, and we will assume that three types of inspection are available, corresponding to $\sigma_{d/t} = 0.02, 0.04$ and 0.08.
2. *Computer experiment* Evaluate p_{FE} at some deterministic input (D, t, s, d, l).
3. *Lab experiment* Obtain one observation of X_{m}.

In order to generate synthetic data for this experiment, we assume that the true defect depth is $d = 0.3t = 6\,\mathrm{mm}$ and that $\mu_m = 1.0$. Instead of running a full Finite Element simulation to obtain p_{FE}, we will make use of the simplified capacity equation in (DNV GL 2017), in which case

$$p_{\mathrm{FE}} = 1.05 \frac{2ts}{D-t} \frac{1 - d/t}{1 - \frac{d/t}{Q}}, \quad Q = \sqrt{1 + 0.31 \frac{l^2}{Dt}}.$$

6.4.3 Results

To define the initial model ξ_0 we need a prior specification over the epistemic quantities d, μ_{m} and p_{FE}. We let d be a priori normal with mean 0.5 and standard deviation 0.15, and μ_{m} normal with mean 1.0 and standard deviation 0.1. Consequently, the posteriors of d and μ_{m} (and also X_{m}) given any number of observations are all normal. The function p_{FE} is replaced by a GP surrogate with prior mean $\mu = -10$ and $\sigma_c = 10$, $l = [1, 1, 1, 1]$ Matérn 5/2 parameters, which we initiate using a single observations at the expected value of the input.

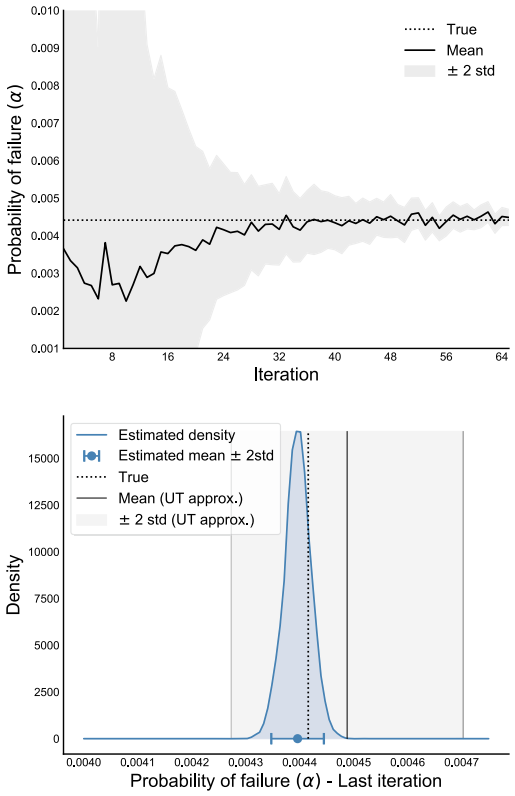


Fig. 10 (Example 3) Top: Mean $\pm$ 2 standard deviations of α_k after k iterations, as computed using the approximation described in Sect. 4. Bottom: The distribution of α_k at the final iteration $k = 65$, estimated from a double-loop Monte Carlo (i.e. by sampling from $\alpha(\xi_k)$ without any approximation)

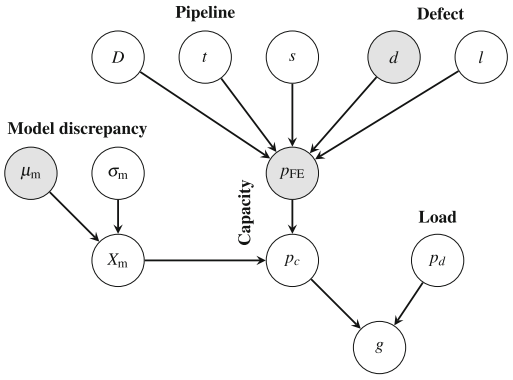


Fig. 11 (Example 4) Graphical representation of the corroded pipeline structural reliability model. The shaded nodes d, p_{FE} and μ_{m} have associated epistemic uncertainty that can be reduced through experiments

We assume that the computer experiments are cheap compared to the lab experiments, and that the direct measurements of d/t is most expensive. To reflect these varying costs, we specify the acquisition function

$$\hat{J}_{i,k}(d) = c(d)\frac{\widehat{E}_{k,d}[\hat{H}_{i,k+1}]}{\hat{H}_{i,k}}, \tag{48}$$

where $c(d)$ is the cost of a given decision. (Note that in (48) the variable d refers to a *decision*, but for the remaining part of this example d will only refer to the *defect depth*). In (48) we have normalized the expected future measure of residual uncertainty with the current, which gives an estimate of the expected improvement given a certain decision. The numerical values representing difference in costs is given by $c = 1$ for computer experiments, $c = 1.1$ for lab experiments, and $c = 1.11, 1.12, 1.13$ for measurements of d/t with accuracy $\sigma_{d/t} = 0.08, 0.04$ and 0.02, respectively.

In structural reliability analysis, the objective is not always to obtain an estimate of the failure probability that is as accurate as possible. A relevant problem in practice is to determine whether a structure satisfies some prescribed target reliability level α_{target}. In this example, we aim to either confirm that the failure probability is less than the target $\alpha_{target} = 10^{-3}$ (in which case we can continue operations as normal), or to detect with confidence that the target is exceeded (and we have to intervene). For this purpose we intend to stop the iterative procedure if the difference between the expected and target failure probability is at least 4 standard deviations. In addition to the standard stopping criterion for convergence (45), we therefore introduce the stopping criterion

$$|\widehat{E}[\hat{\alpha}_k] - \alpha_{target}| < 4\sqrt{\hat{H}_{1,k}}. \tag{49}$$

Figure 12 shows how the UT-MCIS approximation of the failure probability evolves throughout 100 iterations. We have made use of $i = 3$ in (48) as we found the corresponding acquisition surface for p_{FE} smoother than the alternative $i = 1$, and hence easier to minimize numerically. The stopping criterion (49) is reached after $k = 25$ iterations, and Fig. 13 shows the corresponding posteriors of the relative defect depth d/t and the model discrepancy X_{m}.

Throughout the examples in this paper we have initiated GP surrogate models using a single observation at the expected input. A different approach that is often found in practical applications is to initiate the GP surrogate with a space-filling design. A very common alternative is to make use of a Latin Hypercube sample (LHS), of size no more than $10 \times$ the input dimension (although the appropriate number of samples naturally depends on how nonlinear the response is expected to be, see e.g. Loeppky et al. 2009).

Table 4 shows a summary of the results from running this example with and without an initial design consisting of 10

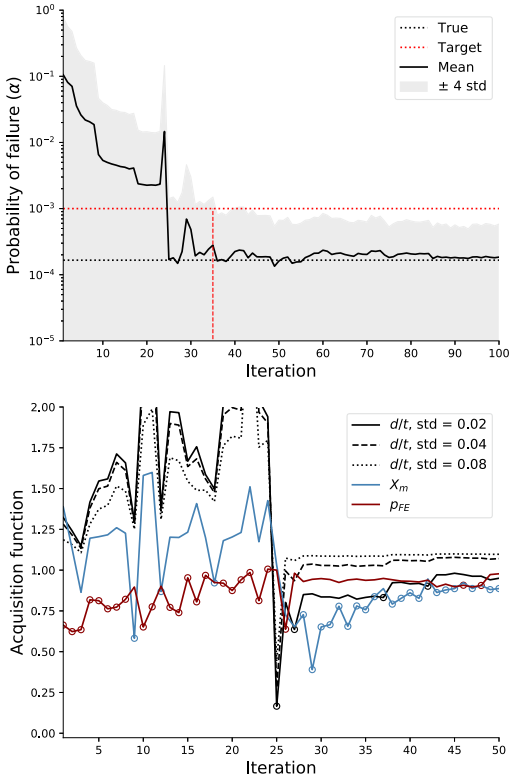


Fig. 12 (Example 4) Top: Mean $\pm$ 4 standard deviations of α_k after k iterations, as computed using the approximation described in Sect. 4. The stopping criterion (49) is reached after 25 iterations. Bottom: The acquisition functions (48) for each type of experiment during the first 50 iterations

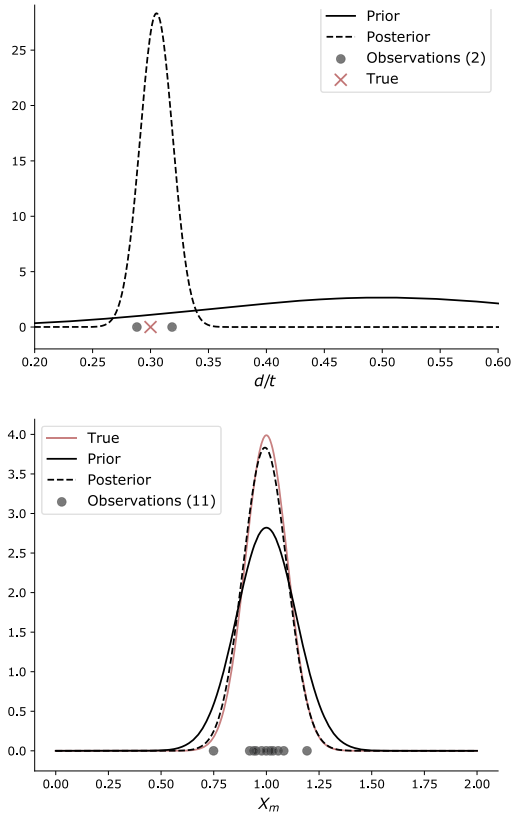


Fig. 13 (Example 4) The posterior distributions of d/t and X_{m} when the stopping criterion (49) is reached at $k = 25$

Table 4 (Example 4) Averages over 100 runs, using 1 versus 10 initial observations of p_{FE}

Initial	Stop at	Cov of α_k	Number of observations		
Design	Target	$(\hat{V}_k)$	p_{FE}	X_{m}	d/t
$E[\mathbf{X}]$	Yes	1.39	23 + 1	10	2
	No	0.63	46 + 1	47	7
LHS 10	Yes	1.37	12 + 10	8	2
	No	0.90	45 + 10	48	7

LHS samples. For this example it does not seem to make any significant difference, but we see why the stopping criterion (49) is useful, as on average we can conclude that the failure probability is below the target value after around 30-40 iterations.

We leave this numerical experiment with an important remark, which is that specifying an appropriate cost in (48) can be difficult. If for instance the cost related to a measurement of d/t is set very high, then the decision to measure d/t will never be taken. In this example, it is not possible to reach the stopping criterion given in (49) without at least one such measurement, and hence, the one-step lookahead strategy will keep requesting measurements of X_{m} and evaluations of p_{FE} indefinitely, accumulating a potentially infinite cost. This is indeed a drawback of the one-step lookahead strategy. Note that a full dynamic programming implementation is typically not feasible in practice as it is too computationally expensive, and this may also be the case for implementations looking only a few more steps ahead. An idea for dealing with this is to study the problem via reinforcement learning instead. This is a work in progress.

7 Concluding remarks

We have presented a general formulation of the Bayesian optimal experimental design problem based on separation of aleatory randomness associated with a physical system, and the epistemic uncertainty that we wish to reduce through experimentation. The effectiveness of a design strategy is evaluated through a *measure of residual uncertainty*, and efficient approximation of this quantity is crucial if we want to apply algorithms that search for an optimal strategy. We make use of a pruned importance sampling scheme for subsequent estimation of (typically small) failure probabilities for a given epistemic realization, combined with the unscented transform epistemic uncertainty propagation. In our numerical experiments, we made use of a rather naive implementation of the unscented transform, in the sense that the number of sigma-points is very low, and that these are determined a priori with a deterministic procedure. Since the alternative by Merwe and Wan (2003) produced satisfactory results in all of our numerical examples, no further consideration was made with respect to alternative methods for sigma-point selection. From applications to Kalman filtering, it has been observed that this version of the unscented transform has a tendency to over-estimate the variance, which is something we notice also in our experiments.

For the application we consider in this paper, we emphasize that the unscented transform is used as a *proxy* for the measure of residual uncertainty to be used in optimization, as a numerically efficient alternative that should be proportional to the true objective. Hence, we view the unscented transform as a tool to find the best decision or strategy, where we get the possibility of exploring many decisions approximately rather than a few exactly. Once an optimal strategy is found, we estimate the corresponding measure of residual uncertainty using a pure Monte Carlo alternative which is exact in the limit. We note that for global optimization of acquisition functions, we have used a combination of random sampling and gradient based local optimization. With this procedure, an optimization objective given by $H_{3,k}$ (and also $H_{2,k}$) is generally more suitable than $H_{1,k}$, as it is less susceptible to noise coming from Monte Carlo estimation (see for instance Fig. 7). On the other hand, $H_{1,k}$ has a natural interpretation (the variance of the failure probability), and is therefore a better measure for evaluating convergence, or for early stopping as discussed in Sect. 6.4.

In Example 4 (Sect. 6.4), we briefly discussed the common alternative of applying a space-filling design, and we observed that starting with an initial LHS design did not make any significant difference, when the remaining design was determined using the one-step lookahead strategy. Similarly, we may compare the one-step lookahead strategy to a naive LHS design, to investigate how useful it is to apply this strategy at all. The number of experiments needed to converge at $V_{\max} = 0.05$ in Examples 1, 2 and 3 in Sect. 6 was 4, 10 and 48. If we instead were to use a (maximin) LHS design over the set of inputs with non-negligible probability density, the expected number of experiments needed to reach $V_{\max} \leq 0.05$ is 30, 150 and 400. Hence, in these examples, the number of experiments is reduced by roughly a factor of 10 by applying the one-step lookahead strategy instead of a space-filling design.

Although we focus on the estimation of *a failure probability* in this paper, many of the main ideas we present should also be applicable for other estimation objectives using models where a hierarchical structure can be utilized. For instance, when α_k is some other quantity of interest depending on the random variable $g(\mathbf{X})$, not necessarily given by an indicator function as in (1). In general, the problem we consider in this paper is estimating the volume of the excursion set $\{\mathbf{x} \in \mathbb{X} | g(\mathbf{x}) \leq 0\}$, under some specified measure on $\mathbb{X} \subseteq \mathbb{R}^m$. For the specific applications considered in this paper, we have assumed that an isoprobabilistic transformation of $\mathbf{X}$ to a standard normal variable is available, which is often the case in structural reliability models. We make use of this assumption only to apply some well known techniques for failure probability estimation, but note that other alternatives, for instance the one presented in Appendix B.3, can be used instead.

There are several ways to improve the methodology presented in this paper. For instance, other alternatives of the unscented transform could be applied, see for instance Menegaz et al. (2015), or the parameters determining the set of sigma-points used in this paper could be optimized as in (Turner and Rasmussen 2010).

As seen in Sect. 6.4, the one-step lookahead/myopic strategy, can make it impossible to reach the stopping criterion of the algorithm. As mentioned, a way to avoid this problem is by looking at the whole dynamic programming formulation (11). However, this formulation suffers from the curse of dimensionality. Since the myopic formulation corresponds to truncating the sum in the dynamic programming formulation (11) to only one term, it is of interest to study methods where more terms of the sum are included (multi-step look ahead). How much better do the estimations get by including an extra term, and how much does the computation time increase? Is it possible to determine an optimal choice of truncation where we weigh accuracy and computation time against one another? Different ways of finding approximate solutions to the complete dynamic programming problem has been the focus of much research within areas such as operations research, optimal control and reinforcement learning, and trying out some of these alternatives is certainly interesting avenue for further research.

Another interesting topic worth investigating is how the numerical examples in this paper compare to the case where we estimate the *buffered failure probability* instead of the

classical failure probability. Buffered failure probabilities were introduced by Rockafellar and Royset (2010) as an alternative to classical failure probabilities in order to take into account the tail distribution of the performance function. See Dahl and Huseby (2019) for an application of this concept to structural reliability analysis.

One may also discuss whether using heuristic optimization objectives chosen to approximate the variance is reasonable. By essentially focusing on minimizing the variance of the failure probability, we say that all deviations from the true value is equally bad. In reality, overestimating the failure probability can be costly, but is not nearly as problematic as underestimating the failure probability. Because of this, the variance may not be the most appropriate measure of risk. It would be interesting to also derive heuristic optimization objectives based on approximating other risk measures.

These questions are of interest, but beyond the scope of the current paper, and the topics are left for future research.

Acknowledgements This work has been supported by Grant 276282 from the Norwegian Research Council and DNV GL Group Technology and Research (Christian Agrell), and by Project 29989 from the Research Council of Norway as part of the SCROLLER project (Kristina Rognlien Dahl). We would also like to thank the reviewers for helpful comments and feedback.

Funding Open access funding provided by University of Oslo (incl Oslo University Hospital).

Appendices

A Gaussian process surrogate models

Here we briefly review the Gaussian process (GP) surrogate model in its canonical form, for Bayesian nonparametric function estimation. For a broader overview of the relevant theory see e.g. Rasmussen and Williams (2006). For applications related to uncertainty quantification (UQ) dealing with deterministic computer simulations, Kennedy and O'Hagan (2001) is a classical reference.

Let $f : \mathbb{X} \to \mathbb{R}$ denote a function that we want to estimate, and assume that a set of k observations $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_k, \mathbf{y}_k)$ have been made. For instance, evaluating $f(\mathbf{x})$ could correspond to running a deterministic (and time consuming) computer simulation, in which case noiseless observations, $\mathbf{y}_i = f(\mathbf{x}_i)$, can be obtained. Alternatively, $f(\mathbf{x}_i)$ could correspond to some physical experiment, resulting in a noise perturbed observation $\mathbf{y}_i$. A GP surrogate model ξ of f is a tool to make inference about the value of $f(\mathbf{x}^*)$ for any new input $\mathbf{x}^* \in \mathbb{X}$, conditioned on the set of observations $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_k, \mathbf{y}_k)$.

A Gaussian process ξ indexed by some set $\mathbb{X}$ is defined by the property that for any finite subset $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of $\mathbb{X}$, $(\xi(\mathbf{x}_1), \ldots \xi(\mathbf{x}_N))$ is an N-dimensional Gaussian random variable. We will view ξ as a Gaussian distribution over real-valued functions defined on $\mathbb{X}$ (such as $f(\mathbf{x})$). Here $\mathbb{X}$ can be arbitrary but typically $\mathbb{X}$ is a subset of $\mathbb{R}^n$. The GP ξ is uniquely defined by its mean function $\mu(\mathbf{x}) = E[\xi(\mathbf{x})]$ and covariance function $c(\mathbf{x}, \mathbf{x}') = E[(\xi(\mathbf{x}) - \mu(\mathbf{x}))(\xi(\mathbf{x}') - \mu(\mathbf{x}'))]$. Hence, any function $\mu : \mathbb{X} \to \mathbb{R}$ paired with a positive semidefinite function $c : \mathbb{X} \times \mathbb{X} \to \mathbb{R}$ defines a GP, which we will denote $\xi \sim \mathcal{GP}(\mu, c)$.

Let $X = (\mathbf{x}_1, \ldots, x_k), Y = (\mathbf{y}_1, \ldots, \mathbf{y}_k)$ denote the observations and assume that $\mathbf{y}_i$ comes with additive Gaussian noise, $\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$ where ϵ_i are i.i.d. zero-mean Gaussian with common variance σ^2. In this scenario, the conditional process $\xi|X, Y$ is still a Gaussian process. In particular, if $X^* = (\mathbf{x}_1^*, \ldots, x_m^*)$ contains m new input locations in $\mathbb{X}$, then the distribution of $\boldsymbol{\xi}^* = \xi(X^*) = (\xi(\mathbf{x}_1^*), \ldots, \xi(\mathbf{x}_m^*))$ given the observations X, Y is Gaussian with the following mean

$$E[\boldsymbol{\xi}^*|X, Y] = \mu(X^*) + c(X^*, X)[c(X, X) + \sigma^2 I_m]^{-1}(Y - \mu(X)), \tag{50}$$

and covariance

$$\mathrm{Cov}(\boldsymbol{\xi}^*|X, Y) = c(X^*, X^*) - c(X^*, X)[c(X, X) + \sigma^2 I_m]^{-1} c(X^*, X)^T. \tag{51}$$

Here $\mu(X^*)$ and $\mu(X)$ are vectors with elements $\mu(\mathbf{x}_i^*)$ and $\mu(\mathbf{x}_i)$ respectively, I_m is the $m \times m$ identity matrix, and $c(X^*, X^*), c(X^*, X)$ and $c(X, X)$ have elements $c(X^*, X^*)_{i,j} = c(\mathbf{x}_i^*, \mathbf{x}_j^*)$, $c(X^*, X)_{i,j} = c(\mathbf{x}_i^*, \mathbf{x}_j)$ and $c(X, X)_{i,j} = c(\mathbf{x}_i, \mathbf{x}_j)$.

For the scenario where observations are noiseless, $\mathbf{y}_i = f(\mathbf{x}_i)$, the distribution of $\boldsymbol{\xi}^*|X, Y$ is obtained with $\sigma = 0$ in (50)–(51).

To define a GP prior $\xi \sim \mathcal{GP}(\mu, c)$ over functions $f : \mathbb{X} \to \mathbb{R}$, we need to specify the mean and covariance function. These are generally given as $\mu(\mathbf{x}|\theta)$ and $c(\mathbf{x}, \mathbf{x}'|\theta)$, conditioned on some parameter θ. An appropriate value for θ is usually found through maximum likelihood estimation or cross validation using the set of observations X, Y. A fully Bayesian approach could also be pursued, where the posterior

calculations typically involve Markov chain Monte Carlo as the formulation in (50)–(51) is not sufficient. In the numerical experiments presented in this paper, we have made use of a constant mean function and a Matérn 5/2 covariance function using plug-in hyperparameters $\theta = (\sigma_c, l_1, \ldots, l_n)$ determined from maximum likelihood estimation. The Matérn 5/2 covariance function for $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$ is defined as

$$c(\mathbf{x}, \mathbf{x}') = \sigma_c^2(1 + \sqrt{5}r + \frac{5}{3}r^2)e^{-\sqrt{5}r},$$
$$r = \sqrt{\sum_{i=1}^{n}\left(\frac{x_i - x_i'}{l_i}\right)^2}. \tag{52}$$

B The sampling distribution $q_{\mathbf{X}}$

Here we present some further details on how the set of samples $\{\mathbf{x}_i, w_i\}$ in Sect. 4.3 can be generated. We start by reviewing some classical techniques from structural reliability analysis that are based on finding 'important' regions in $\mathbb{X}$. The sampling distribution $q_{\mathbf{X}}$ used in this paper is then defined in Section B.2. It is based on the assumption that $\mathbf{X}$ can be transformed to a standard multivariate Gaussian variable $\boldsymbol{U}$, and that $q_{\boldsymbol{U}}$ can be constructed by solving a set of constrained optimization problems in $\boldsymbol{U}$-space. For the scenario where these assumptions do not hold, we present an alternative approach in Section B.3, which is based on a naive exploration of the $\mathbb{X}$-space. Although this will require evaluation of a larger set of samples of $\mathbf{X}$, no optimization is required and numerical implementation is straightforward.

B.1 Local approximations in SRA

In Sect. 4 we briefly discussed the challenges with estimation of the failure probability $\bar{\alpha}(g)$ in (1). A different alternative often used in structural reliability analysis, is to approximate the performance function $g(\mathbf{x})$ with a function $\hat{g}$ where $\bar{\alpha}(\hat{g})$ can be computed analytically. In this scenario, it is convenient to transform $\mathbf{X}$ to a standard normal variable $\boldsymbol{U}$. We will let

$$\mathbf{X} \xrightarrow{\mathscr{T}} \boldsymbol{U} \sim N(\mathbf{0}, I) \tag{53}$$

denote an isoprobabilistic transformation, where $\boldsymbol{U} = \mathscr{T}(\mathbf{X})$ is multivariate standard Gaussian with $\dim(\boldsymbol{U}) = \dim(\mathbf{X})$. Note that for any univariate random variable X with CDF $F(X)$, a transformation of this type available as $\mathscr{T}(X) = \Phi^{-1}(F(X))$. The generalization to multivariate $\mathbf{X}$ is the Rosenblatt transformation, where $\boldsymbol{U}_i = \Phi^{-1}(F_i(\mathbf{X}_i|\mathbf{X}_1, \ldots, \mathbf{X}_{i-1}))$. In structural reliability problems, it is often natural to define $\mathbf{X}$ in terms of the marginal distributions and a copula, in which case the isoprobabilistic transformation (53) can be simplified. A common alternative is to use a Gaussian copula, where (53) can be obtained using the Nataf transformation (Lebrun and Dutfoy 2009).

In the following we let $g(\mathbf{u})$ denote the function $g(\cdot)$ applied to $\mathbf{x} = \mathscr{T}^{-1}(\mathbf{u})$. Methods such as FORM (First Order Reliability Method) and SORM (Second Order Reliability Method) make use of local approximations in the form of a linear or quadratic surface fitted to $g(\mathbf{u}^*)$ at a certain point $\mathbf{u}^* \in \mathbb{R}^n$. This point $\mathbf{u}^*$ is often called the *design point* or *most probable point* (MPP), and it is defined as

$$\mathbf{u}^* = \arg\min_{\mathbf{u}\in\mathbb{R}^n}\{\|\mathbf{u}\| \mid g(\mathbf{u}) \le 0\}. \tag{54}$$

Observe that if $\hat{g}(\mathbf{u})$ is the first-order Taylor approximation of $g(\mathbf{u})$ at $\mathbf{u}^*$, i.e. $\hat{g}(\mathbf{u}) = g(\mathbf{u}^*) + \nabla_{\mathbf{u}} g(\mathbf{u}^*)(\mathbf{u} - \mathbf{u}^*)$, then $\bar{\alpha}(\hat{g}) = \Phi(-\|\mathbf{u}^*\|)$, and this is an upper bound on the failure probability if the failure set is convex in $\boldsymbol{U}$-space.

In Sect. 4.3 we discussed the importance sampling estimate of the failure probability given some proposal distribution q. A natural candidate is to let q be a distribution centered around the design point, $\mathbf{u}^*$ in $\boldsymbol{U}$-space or $\mathbf{x}^* = \mathscr{T}(\mathbf{u}^*)$ in $\mathbf{X}$-space. The alternative where the estimation is performed in $\boldsymbol{U}$-space with $q_{\boldsymbol{U}}(\mathbf{u}) = \phi(\mathbf{u} + \mathbf{u}^*)$ is often used in practice. For a more detailed discussion around this kind of sampling, the local approximations and structural reliability analysis in more general, see for instance (Madsen et al. 2006) or (Huang et al. 2017).

The constrained optimization problem (54) plays an important role in structural reliability analysis. Although any general-purpose algorithm can be used, customized algorithms that take advantage of the special form of the objective function are recommended. Various alternatives have been developed for this purpose, see for instance (Gong and Yi 2011) and the references therein. For the applications in this paper we have made use of the iHL-RF method from (Zhang and Der Kiureghian 1995).

B.2 The design point mixture

We observe first that a solution to (54) is not necessarily unique, and also that multiple *local* minima may exists when the performance function is nonlinear. Most algorithms designed to solve (54) numerically start with some initial guess $\mathbf{u}_0$, and take iterative steps until a minimum is obtained. To reduce the risk of overestimating $\|\mathbf{u}^*\|$, multiple restarts with different (possibly randomized) initial guesses $\mathbf{u}_0$ is often applied.

Given a finite-dimensional approximation of a performance function $\hat{\xi}(\mathbf{x}, \mathbf{E})$, we want to find a proposal distribution q that is appropriate for a range of different realizations $\mathbf{e}$ of $\mathbf{E}$. In particular, if $\{(v_j, \mathbf{e}_j) \mid j = 1, \ldots, M\}$ is the set of sigma-points for $\mathbf{E}$ as introduced in Sect. 4.2, we want a set of samples from q to be applicable for estimation of $\alpha(\hat{\xi}(\mathbf{x}, \mathbf{e}_j))$ for any $1 \le j \le M$.

For any $\mathbf{e}_j$, we will let $\mathbf{u}^*_{1,j}, \ldots, \mathbf{u}^*_{N,j}$ denote N design points in $\boldsymbol{U}$-space corresponding to $\hat{\xi}(\mathbf{x}, \mathbf{e}_j)$, obtained using randomized initialization. (Note that for methods such as iHL-RF, it is also reasonable to use $\mathbf{u}^*_{i,j}$ as an initial guess in the search for $\mathbf{u}^*_{i,j+1}$). We then define $\boldsymbol{Q}$ as the equal-weighted Gaussian mixture of the NM random variables $\boldsymbol{Q}_{i,j} = \boldsymbol{U}_{i,j} + \mathbf{u}^*_{i,j}$, where $\boldsymbol{U}_{i,j}$ are i.i.d. standard multivariate Gaussian. Sampling from $\boldsymbol{Q}$ is then straightforward, and importance sampling estimates can be obtained in the $\boldsymbol{U}$-space using $p_{\boldsymbol{U}}(\mathbf{u}) = \phi(\mathbf{u})$ and $q_{\boldsymbol{U}}(\mathbf{u}) = \frac{1}{NM}\sum_{i,j}\phi(\mathbf{u} - \mathbf{u}^*_{i,j})$, where ϕ is the multivariate standard normal density.

B.3 A simple alternative

The sampling strategy presented in Sect. 4.3 is based on (1) generating a set of samples that should "cover relevant locations" in the input space $\mathbb{X}$, and (2) *prune* the set of samples using a threshold on the measure of insignificance (21).

The "relevant locations" in the first step is typically somewhere in the "tail" of the distribution of $\mathbf{X}$, where also the (uncertain) performance function $\hat{\xi}_k(\mathbf{x})$ may be close to zero. In Section B.2 we made use of importance sampling around design points, which is a common technique in structural reliability analysis. As a simple alternative, we can let q be any distribution from which it is easy to generate samples covering the effective support of $p_{\mathbf{X}}$ (i.e. a bounded domain where $\mathbf{X}$ lies with probability ≈ 1). For instance, assuming $\boldsymbol{U}$ is n-dimensional standard normal (e.g. $\boldsymbol{U} = \mathscr{T}(\mathbf{X})$ if the isoprobabilistic transformation is still applicable), we could let q be a uniform density on the hypercube $[-b, b]$ where $b = \Phi^{-1}(1 - p_{min})$ for some absolute lower bound on the failure probability p_{min}.

Because the initial set of N samples from q will be reduced to a fixed number of n samples after the pruning step, this is a viable alternative. However, in order to obtain similar importance sampling variances [see (29)] as with the method in Section B.2, the initial number of samples N (and hence the number of evaluations of the pruning criterion $\eta(\mathbf{x})$) will have to be larger.

C Selecting sigma-points for the unscented transform

Here we briefly review the method for sigma-point selection by Merwe (2004) and present the sigma-points used for the numerical experiments in Sect. 6.

According to Labbe (2014), research and industry have mostly settled on the version published in (Merwe 2004). Here, the sigma-points are given as a function of the mean and covariance matrix of the input variable, together with three real-valued parameters α, β and κ. In the case where $\boldsymbol{U}$ is a standardized n-dimensional random variable with $E[\boldsymbol{U}] = \mathbf{0}$ and $E[\boldsymbol{U}^2] = I$, we obtain $2n + 1$ points $\mathbf{u}_i$ are as follows

$$\begin{aligned}\mathbf{u}_0 &= \mathbf{0},\\ \mathbf{u}_i &= \alpha\sqrt{n+\kappa}\,\boldsymbol{v}_i,\\ \mathbf{u}_{i+n} &= -\mathbf{u}_i,\end{aligned}$$

for $i = 1, \ldots, n$ where $\boldsymbol{v}_i = (0, \ldots, 1, \ldots, 0)$ is the standard unit vector in $\mathbb{R}^n$. Two different sets of weights are used with this procedure, one for the mean and one for the covariance in (17). We denote these v_i^m and v_i^c respectively, and they are given as

$$\begin{aligned}v_0^m &= 1 - \frac{n}{\alpha^2(n+\kappa)}, \quad v_0^c = v_0^m + 1 - \alpha^2 + \beta,\\ v_i^m &= v_i^c = \frac{1}{2\alpha^2(n+\kappa)} \; \textit{for } i = 1, \ldots, 2n.\end{aligned}$$

For Gaussian distributions, it is often recommended to set $\beta = 2$, $\kappa = 3 - n$ and let $\alpha \in (0, 1]$. In the numerical examples presented in this paper we have used this set of parameters with $\alpha = 0.9$.

References

Bect, J., Ginsbourger, D., Li, L., Picheny, V., Vazquez, E.: Sequential design of computer experiments for the estimation of a probability of failure. Stat. Comput. **22**(3), 773–793 (2012)

Bect, J., Bachoc, F., Ginsbourger, D.: A supermartingale approach to Gaussian process based sequential design of experiments. Bernoulli (2019)

Bichon, B., Eldred, M., Swiler, L., Mahadevan, S., McFarland, J.: Efficient global reliability analysis for nonlinear implicit performance functions. AIAA J. **46**, 2459–2468 (2008)

Dahl, K., Mohammed, S.E., Øksendal, B., Røse, E.E.: Optimal control of systems with noisy memory and BSDEs with Malliavin derivatives. J. Funct. Anal. **271**(2), 289–329 (2016)

Dahl, K.R., Huseby, A.B.: Buffered environmental contours. Safety and Reliability - Safe Societies in a Changing World Proceedings of ESREL 2018 (2019)

Defourny, B., Ernst, D., Wehenkel, L.: Multistage stochastic programming: A scenario tree based approach to planning under uncertainty. Decis. Theory Model. Appl. Artif. Intell.: Concepts Solut.(2011)

Der Kiureghian, A., Ditlevsen, O.: Aleatory or epistemic? does it matter? Struct. Saf. **31**(2), 105–112 (2009)

DNV, G.L.: Recommended Practice: Corroded Pipelines DNVGL-RP-F101. DNV GL, Høvik, Norway (2017)

Echard, B., Gayton, N., Lemaire, M.: AK-MCS: An active learning reliability method combining Kriging and Monte Carlo simulation. Struct. Saf. **33**(2), 145–154 (2011)

Fernandez, G., Park, C., Kim, N., Haftka, R.: Review of multi-fidelity models. arXiv:1609.07196v3 (2017)

Goldstein, M., Wooff, D.: Bayes Linear Statistics: Theory and Methods. John Wiley & Sons, New York (2007)

Gong, J.X., Yi, P.: A robust iterative algorithm for structural reliability analysis. Struct. Multidiscip. Optim. **43**, 519–527 (2011)

Huan, X., Marzouk, Y.: Sequential bayesian optimal experimental design via approximate dynamic programming. arXiv:1604.08320v1 (2016)
Huang, C., ELHami, A., Radi, B.: Overview of Structural Reliability Analysis Methods-Part I, II, p. 17. III, Incertitudes et fiabilité des systémes multiphysiques (2017)
Jian, W., Zhili, S., Qiang, Y., Rui, L.: Two accuracy measures of the Kriging model for structural reliability analysis. Reliab. Eng. Syst. Safety **167**, 494–505 (2017)
Jones, M., Goldstein, M., Jonathan, P., Randell, D.: Bayes linear analysis of risks in sequential optimal design problems. Electron. J. Stat. **12**, 4002–4031 (2018)
Julier, S., Uhlmann, J.: Unscented filtering and nonlinear estimation. Proc. IEEE **92**, 401–422 (2004)
Kennedy, M.C., O'Hagan, A.: Bayesian calibration of computer models. J. Royal Stat. Soc.: Ser. B (Stat. Methodol.) **63**(3), 425–464 (2001)
Kyzyurova, K.N., Berger, J.O., Wolpert, R.L.: Coupling computer models through linking their statistical emulators. SIAM/ASA J Uncertain Quantif. **6**(3), 1151–1171 (2018)
Labbe, R.: Kalman and Bayesian Filters in Python. In: https://github.com/rlabbe/Kalman-and-Bayesian-Filters-in-Python, GitHub eBook (2014)
Lebrun, R., Dutfoy, A.: A generalization of the Nataf transformation to distributions with elliptical copula. Probab. Eng. Mech. **24**(2), 172–178 (2009)
Loeppky, J., Sacks, J., Welch, W.: Choosing the sample size of a computer experiment: A practical guide. Technometrics **51**, 366–376 (2009)
Madsen, H., Krenk, S., Lind, N.: Methods of Structural Safety. Dover Civil and Mechanical Engineering Series. Dover Publications, New York (2006)
Menegaz, H.M.T., Ishihara, J.Y., Borges, G.A., Vargas, A.N.: A systematization of the unscented kalman filter theory. IEEE Trans. Autom. Control **60**(10), 2583–2598 (2015)
Merwe, R.: Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models. PhD thesis, OGI School of Science and Engineering (2004)
Merwe, R., Wan, E.: Sigma-point kalman filters for probabilistic inference in dynamic state-space models. Proceedings of the Workshop on Advances in Machine Learning (2003)
Perrin, G.: Active learning surrogate models for the conception of systems with multiple failure modes. Reliab. Eng. Syst.Safety **149**, 130–136 (2016)
Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press, Cambridge (2006)
Rockafellar, R.T., Royset, J.O.: On buffered failure probability in design and optimization of structures. Reliab. Eng. Syst. Safety **95**(5), 499–510 (2010)
Rozsas, A., Slobbe, A.: Repository and Black-box Reliability Challenge 2019. https://gitlab.com/rozsasarpi/rprepo/ (2019)
Schueremans, L., Gemert, D.V.: Benefit of splines and neural networks in simulation based structural reliability analysis. Struct. Saf. **27**(3), 246–261 (2005)
Sun, Z., Wang, J., Li, R., Tong, C.: LIF: A new Kriging based learning function and its application to structural reliability analysis. Reliab. Eng. Syst. Safety **157**, 152–165 (2017)
Turner, R., Rasmussen, CE.: Model based learning of sigma points in unscented kalman filtering. In: 2010 IEEE International Workshop on Machine Learning for Signal Processing, pp 178–183 (2010)
Uhlmann, J.: Dynamic map building and localization : New theoretical foundations. PhD thesis, University of Oxford (1995)
Wang, L.: Karhunen-Loéve expansions and their applications. PhD thesis, London School of Economics and Political Science (2008)
Zhang, Y., Der Kiureghian, A.: Two Improved Algorithms for Reliability Analysis, pp. 297–304. Springer, US, Boston, MA (1995)

Paper IV

Optimal sequential decision making with probabilistic digital twins

Christian Agrell, Kristina Rognlien Dahl, Andreas Hafver

Submitted for publication. arXiv: 2103.07405

IV

Paper V

Risk-based functional black-box optimization – Contribution to the NASA Langley UQ challenge on optimization under uncertainty

Christian Agrell, Simen Eldevik, Odin Gramstad, Andreas Hafver

Mechanical Systems and Signal Processing (2021) Vol. 164 108266.

V

Mechanical Systems and Signal Processing 164 (2022) 108266

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Risk-based functional black-box optimization
Contribution to the NASA Langley UQ challenge on optimization under uncertainty

Christian Agrell [a,b,*,1], Simen Eldevik [a,1], Odin Gramstad [a,1], Andreas Hafver [a,1]

[a] DNV Group Research and Development, Norway
[b] Department of Mathematics, University of Oslo, Norway

ARTICLE INFO

Communicated by J.E. Mottershead

Keywords:
Model calibration
Functional data
Epistemic uncertainty
Robust optimization

ABSTRACT

This paper presents an approach to solve the 2019/2020 NASA Langley UQ challenge problem on optimization under uncertainty. We define an uncertainty model (UM) as a pair $\langle f_{a|e}, E\rangle$, where $f_{a|e}$ is a probability density over a for each $e \in E$, and proceed to infer $f_{a|e}$ in a Bayesian fashion. Special attention is given to dimensionality reduction of the functional (time-series) data, to obtain a finite dimensional representation suitable for robust Bayesian inversion. Reliability analysis is performed using $f_{a|e}$, whereas for design optimization we approximate $f_{a|e}$ using truncated Gaussians and a Gaussian copula. We apply an unscented transform (UT) in the standard normal space to estimate moments of the limit state, which is numerically very efficient. Design optimization is performed with this procedure to obtain negligible failure probability in g_1 and g_3 and acceptable failure probability and severity in g_2.

1. Introduction and notation

This paper[2] presents an approach to solve the NASA Langley UQ challenge problem on optimization under uncertainty [2]. We recall just the main problem setup and notation herein, and refer to [2] for a complete description.

In the first part of the problem, the goal is to establish a probability distribution for an aleatory random variable $a \in A = [0,2]^5$, that will depend on another variable $e \in E_0 = [0,2]^4$ with epistemic uncertainty. Inference is based on a set of time series, $D = \{y_i(t)\}_{i=1}^N$, that are the result of a functional mapping $Y : A \times E_0 \to y(t)$ where $y(t) : [0,5] \to \mathbb{R}$. The time series data D correspond to $y_i(t) = Y(a_i, e_{true})(t)$ for some fixed (but unknown) $e_{true} \in E_0$ and $N = 100$ i.i.d. samples a_i of $a \in A$.

We assume that the random variable a can be represented by a joint density f_a, that we wish to infer given the set of observations D. A numerical model of the data generating process, $\hat{Y}(a,e) \approx Y(a,e)$, is provided, and will be the basis for inference on the aleatory random variable a, as well as finding plausible values of the epistemic variable e. As we do not know the true value e_{true} used to generate the data, we will find a set $E \subseteq E_0$ of possible candidates for e, together with a family of probability distributions $\{f_{a|e}\}$, parametrized by $e \in E$ (see Tables 1–3).

* Corresponding author at: Department of Mathematics, University of Oslo, Norway.
E-mail address: chrisagr@math.uio.no (C. Agrell).

[1] All authors contributed equally.

[2] This is an updated and extended version of the paper [1] which was prepared for the 30th European Safety and Reliability (ESREL) 2020 Conference.

https://doi.org/10.1016/j.ymssp.2021.108266
Received 21 January 2021; Received in revised form 28 June 2021; Accepted 21 July 2021
Available online 31 July 2021

Table 1
Variables.

Variable	Domain	Description
a	$A = [0,2]^5$	Aleatory input variable
u	$\mathbb{R}^5$	Standard normal variable
e	$E_0 = [0,2]^4$	Epistemic input variable
θ	$\mathbb{R}^9$	Control variable

Table 2
Physical system.

$y(t) = Y(a,e)(t)$	Subsystem
$\hat{y}(t) = \hat{Y}(a,e)(t)$	Numerical model
$z(t) = (z_1, z_2)(t) = Z(a,e,\theta)(t)$	Integrated system
$\hat{z}(t) = \hat{Z}(a,e,\theta)(t)$	Numerical model
$g(a,e,\theta) = (g_1, g_2, g_3)(a,e,\theta)$	Limit-state
$w(a,e,\theta) = \max_{i=1,2,3} g_i(a,e,\theta)$	Limit-state

Table 3
Uncertainty model.

$\langle f_{a\|e}, E \rangle$	Uncertainty model
$f_{a\|e}$	Probability density of a given (fixed) e
$f^{\alpha}_{a\|e}$	Parametric approximation of $f_{a\|e}$ with parameter α
E	Hyper-rectangular set
$\zeta(a,e)$	Parameter mapping
f_{ζ}	Probability density of ζ

The pair $\langle f_{a|e}, E \rangle$ is referred to as the uncertainty model (UM) of (a,e). Once $\langle f_{a|e}, E \rangle$ has been established, the remaining tasks are related to structural reliability analysis (SRA) of a larger physical system, where the mechanics given by $Y(a,e)$ is a subcomponent. This includes estimation of failure probabilities for a set of provided limit-state functions, sensitivity analysis and design optimization. We leave further details on the specific tasks to the relevant subsections below.

Our notation is aligned with the problem description, with some few additions. In particular, we may write $y(a,e,t)$ as $Y(a,e)(t)$ to clarify when we are working with the functional mapping Y vs a given function of time y. We also write $\hat{Y}$ and $\hat{y}$ to emphasize when Y and y are computed using the provided numerical model, and similarly for the integrated system $z(a,e,\theta,t)$. An overview of the notation used is given below, and the following sections, Section 2 to Section 7, correspond to respective subproblems in the UQ challenge. The final results are collected in Section 8, and we end with some concluding remarks in Section 9.

2. (A) model calibration & uncertainty quantification of a subsystem

Given a numerical model of the physical subsystem, $\hat{Y}(a,e) \approx Y(a,e)$, we seek to characterize the parameters (a,e) from a limited set of observations $\{y_i(t)\}$. An observation y_i corresponds to $Y(a_i, e_{true})$ where e_{true} is fixed (but unknown) and a_i are i.i.d. samples from some (unknown) distribution f_a. Our approach here is based on fitting a distribution to the observations, $y_1(t), \dots, y_{100}(t)$, from which the conditional distribution $f_{a|e}$ can be determined.

2.1. Dimensionality reduction

As we wish to fit a density to the observations, we first need to compress the functional data to a finite-dimensional (preferably low-dimensional) representation. Various alternatives were considered, from the naive approach of evaluating $y(t)$ at a finite set of times $\{t_1, \dots, t_N\}$, to more sophisticated function approximation techniques. The Karhunen–Loéve transform as used in functional principal component analysis [3] was first considered. This approach is based on finding an orthonormal eigenbasis of L^2 (all square integrable functions) from the estimated covariance function corresponding to the observations. By projecting each observation $y_i(t)$ onto the subspace spanned by the first 10 eigenfunctions, we observed that the residual (the projection onto the complement space), had negligible L^2 norm.

As both observations $y_i(t)$ and samples from $\hat{Y}(a,e)(t)$ consistently showed two or three distinct frequency components when performing a Fast Fourier Transform (FFT), it was also deemed appropriate to consider damped complex exponentials as the function basis. With this approach we write

$$y_{exp}(t) = \sum_{i=1}^{k} B_i \cos(\omega_i t + \phi_i) e^{d_i t}, \tag{1}$$

where the coefficients $(B_i, \omega_i, \phi_i, d_i)$ can be estimated efficiently using Prony's method (see for instance [4]). Unlike the eigenfunctions used in the Karhunen–Loéve transform, the damped complex exponentials do not form an orthonormal basis. But we noticed

that any of the observations $y_i(t)$, as well as any function $\hat{y}(t)$ computed using the numerical model $\hat{Y}(a,e)$ for some $(a,e) \in A \times E_0$, could be represented as the sum of $k = 3$ damped complex exponentials, up to machine precision.

Given some parametric function approximation, we define the parameter mapping

$$\zeta(a,e) : A \times E_0 \to \mathbb{R}^{n_\zeta}. \tag{2}$$

We will write $\zeta(a,e)$ or $\zeta(y)$ interchangeably, depending on whether we consider a specific function $y(t)$ or variables (a,e), for which $\zeta(a,e) = \zeta(Y(a,e))$. If $\zeta(a,e)$ is the parameter vector corresponding to the damped complex exponentials, then $\zeta(a,e)$ is given by (some of) the $4k$ parameters needed in Eq. (1). In the case where the Karhunen–Loéve transform is used to represent $Y(a,e)(t)$, we could let $\zeta(a,e)$ correspond to the projection coefficients (FPCA scores) of the first n_ζ eigenfunctions with the largest eigenvalues.

The reason for introducing $\zeta(a,e)$ is that, if we can fit a probability density $f_\zeta(\zeta)$ to ζ, then we can establish a distribution $f_{a|e}(a)$ as

$$f_{a|e}(a) = \frac{1}{C} f_\zeta(\zeta(a,e)), \tag{3}$$

where C is a normalizing constant that assures that $f_{a|e}$ integrates to 1. Eq. (3) comes from a Bayesian formulation, assuming that $e = e_{true}$ and a uniform prior on a (with constant density that goes into C), and where for a fixed e, the density f_ζ defines a (improper) likelihood for any $\zeta(a,e)$ which we assign to a.

The reason why we define $f_{a|e}(a)$ as in (3), by first fitting a probability density $f_\zeta(\zeta)$ to the (transformed) observations, is because we find it easier to determine the appropriate amount of regularization needed to avoid overfitting. Note that this approach to inversion is not the same as computing the distribution corresponding to a transformation of f_ζ through ζ^{-1} (if ζ was bijective), which would include the Jacobian of ζ in the right hand side of (3).

Because of this, together with the fact that the number of observations is limited, we need some means of ensuring that the distribution $f_{a|e}(a)$ is conservative, in the sense that it does not assign negligible probability to values of (a,e) that could be plausible. For this, some qualitative judgment is usually needed. With our approach, we address this by fitting a high-entropy distribution to f_ζ.

We observed that both the eigenfunction approximation and the damped complex exponentials could provide a reasonable dimensionality reduction, from the relevant space of functions $y(t)$ to a set of 10-12 parameters. We chose to go with the damped complex exponentials, as it turned out to provide lossless compression of the functional data. We also found that it was more straightforward to fit a density f_ζ to $\zeta(a,e)$ using this approach.

2.2. Model discrepancy

The problem of assessing model discrepancy without controlled experiments, i.e. when only observed output of the physical system is available, is generally ill-posed. This is due to problems with identifiability, and additional assumptions on the accuracy of the numerical model, as well as other sources of uncertainty in the true physical process, are generally needed. See for instance [5] and the discussions therein.

We will assume that there are no other sources of uncertainty in the physical subsystem besides input uncertainty. That is, variability in the observed $y_i's$ is due to variability in aleatory input a alone. Similarly, we would like to assume zero model discrepancy as well.[3] i.e. $\hat{Y}(a,e) = Y(a,e)$ for all $(a,e) \in A \times E_0$. However, we found that we needed to assume some model discrepancy in order for inference on (a,e) to be possible.

If we want to fit a probability density to ζ, and use this to create a distribution on the input a assuming zero error in $\hat{Y}(a,e)$, then we must first verify that the parameter vector of the observations, $\zeta(y_i)$, are within the range of $\zeta(a,e)$ (i.e. each y_i can be reproduced from $\hat{Y}$). Otherwise, the resulting distribution $f_{a|e}$ could assign zero probability to almost all $a \in A$.

In practice, we will not make use of all components of ζ when fitting a distribution, in order to impose some regularization. But before we do this, it is useful to use the complete ζ to investigate if there is any model discrepancy. As noted in Section 2.1, if we let the parameter mapping be defined as $\zeta(a,e) = [\zeta_1, \zeta_2, \zeta_3]$, where $\zeta_i = [B_i, \omega_i, \phi_i, d_i]$ are the parameters of the ith wave component in Eq. (1), then ζ provides a bijection between the range of the numerical model, $\hat{Y}(A, E_0)$, and a subset of $\mathbb{R}^{12}$. We can therefore investigate whether the observations y_i are within the range of $\hat{Y}$ through ζ. We found that this was not the case, as it turns out that $\zeta(y_i)$ falls outside $\zeta(A, E_0)$ in the subspace spanned by (d_1, ω_1), see Fig. 1. If we exclude the parameter d_1 from ζ, the set $\zeta(A, E_0)$ will include all transformed observations $\zeta(y_i)$. In practice, the exclusion of d_1 corresponds to the assumption that the response from the true physical subsystem, $y = Y(a,e)$, is a bit less "damped", compared to what would be expected were the response to agree with the numerical model $\hat{Y}(a,e)$. Of course, this may be an assumption that is not appropriate, and generally one would assess such an assumption based on knowledge related to the physical phenomenon and information regarding what kind of model discrepancy (or observational noise/error) to expect.

Fig. 2 shows effectively what the assumed model discrepancy looks like under this assumption. Here, to give one example, one of the observations (# 71) was fitted using (1), and we vary d_1 within the relevant range from Fig. 1. Any amount of damping that keeps the time series within the blue shaded area is considered negligible in terms of model discrepancy. In Fig. 3 we see some examples of functions $\hat{y}(t)$ that are *not* equivalent with this observation, which illustrates that the assumed model discrepancy is rather small. We note that the assumed model discrepancy is probably negligible for all practical purposes, but it is necessary for inference on the model input (a,e) to be possible.

[3] In reality it would be natural to make use of some model uncertainty, either estimated or assumed, when the UM we aim to establish will be used for SRA of a safety-critical system.

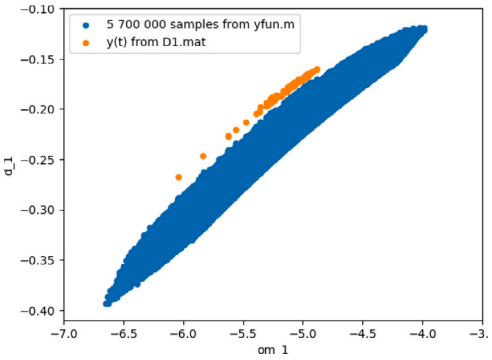


Fig. 1. Frequency (ω_1) vs damping (d_1) coefficients of the first wave component in Eq. (1), corresponding to observations (y_i) and output from the numerical model ($\hat{y}$).

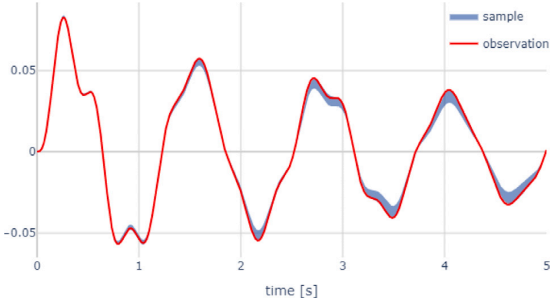


Fig. 2. Example of assumed model discrepancy. Any function $\hat{y}(t)$ is interpreted as equivalent to the observation $y_{i=71}$ (red) if it is more damped (compressed in the y-direction) than $y_{i=71}$, as long as $\hat{y}(t)$ falls within the blue shaded area.

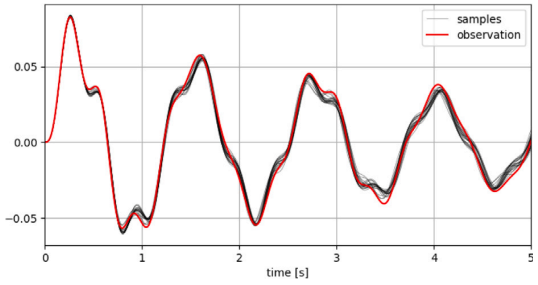


Fig. 3. Examples of functions $\hat{y}(t)$ that do *not* agree with the observation $y_{i=71}$ (red), with respect to the assumed model discrepancy. There are relevant features of the time series that make the samples and the observation significantly different.

Remark 2.1. The results presented in this paper may be sensitive to the assumed model discrepancy. If the criterion illustrated in Figs. 2 and 3 is too strict, i.e. larger deviations between the data generating process $Y(a, e)$ and the computer model $\hat{Y}(a, e)$ is expected, the resulting UM may be overly optimistic. That is, as inputs (a, e) are deemed less plausible, the volume of E and the entropy of $f_{a|e}$ are reduced.

This assumed model discrepancy is based on: (1) no model error has been specified for $\hat{Y}(a, e)$, and (2) with the relaxed assumption on damping of the time series data, it is for each observation y_i possible to find inputs (a, e) such that $\hat{Y}(a, e)$ agrees with y_i.

2.3. The uncertainty model $\langle f_{a|e}, E \rangle$

We will establish an uncertainty model $\langle f_{a|e}, E \rangle$ based on the parameter mapping (ζ) given by the damped complex exponentials with $k = 3$. As discussed in Section 2.2, the component d_1 is excluded to account for some model discrepancy, and 3 other parameters

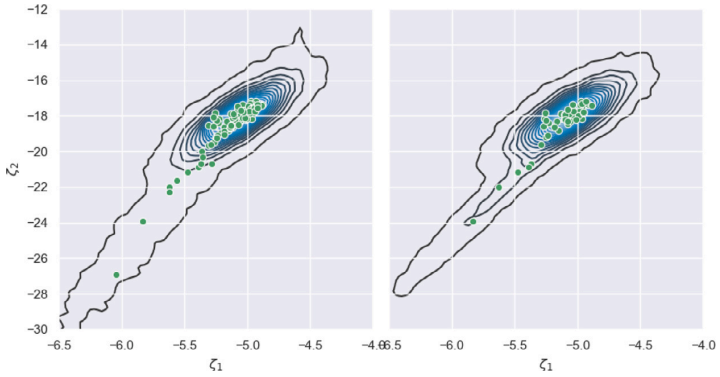


Fig. 4. Marginal of two components of ζ, where f_ζ is obtained from the 100 observations y_i (left), and a subset of 50 observations (right).

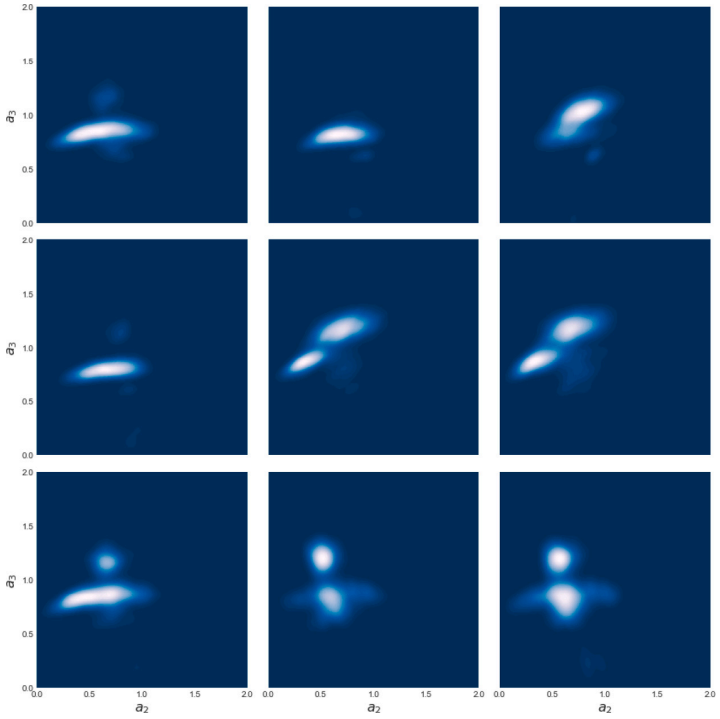


Fig. 5. Marginal of $f_{a|e}$ in the (a_2, a_3)-subspace for 9 different values of e.

are excluded after principal component analysis (PCA) where we found that 99% of the variance could be described by the final 8 parameters. From the final parameter vector $\zeta : A \times E_0 \to \mathbb{R}^8$, we fit a mixture of two Gaussians to the 100 observations y_i under the transformation $\zeta(y_i)$.

Fig. 4 shows the marginal distribution corresponding to two of the components of ζ. Some conservatism was included by increasing the variance of the fitted distribution, in order to account for the limited number of observations. Starting from a maximum likelihood fit, the variance was increased by a constant to produce a more conservative (higher entropy) distribution. Hence, the corresponding distribution $f_{a|e}$ will likely underfit the data, which is intentional. The method used to fit a distribution to ζ is fairly robust to the number of observations. By sampling subsets of size 50 out of the total 100 observations, we find that the resulting distributions fitted to $\{\zeta(y_i)\}$ are fairly consistent as illustrated in Fig. 4.

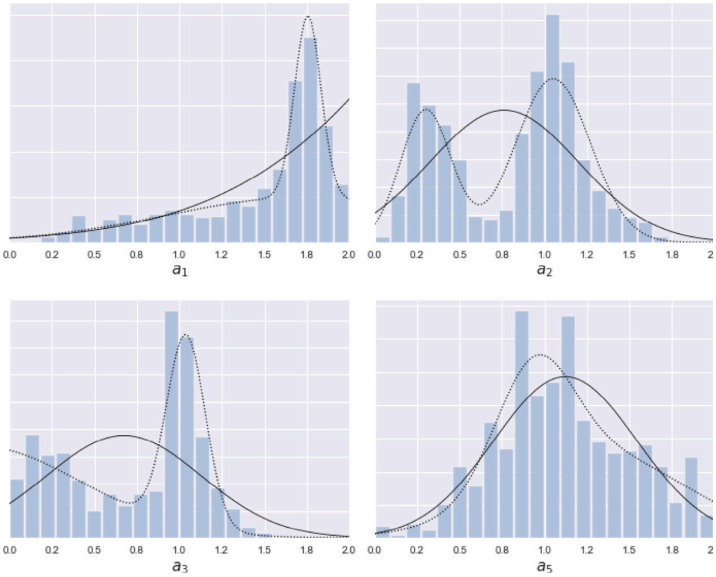


Fig. 6. Parametric distributions fitted to marginals of $f_{a|e}$. The dotted curve shows a mixture of two truncated normal distributions versus the one single truncated normal.

Using the distribution f_ζ, we can assess the likelihood $f_\zeta(\zeta(a,e))$ for any $(a,e) \in A \times E_0$. For a point $e \in E_0$ to be plausible, it should be possible to find some $a \in A$ such that $f_\zeta(\zeta(a,e))$ is large. That is, there must exist some $a \in A$ such that $\zeta(a,e)$ lies within the main bulk of the distribution shown in Fig. 4. In fact, there must exist some $a_i \in A$ such that $\zeta(a_i,e)$ corresponds with $\zeta(y_i)$, for all observations y_i, in order for e to be plausible. However, such a strategy for finding the set E would require computing $\zeta(a,e)$ for all $a \in A$. As a practical alternative, we generate a large set of samples $(a,e)_i \in A \times E_0$, and filter out all samples with likelihood ≈ 0. We then estimate the set $\hat{E}$ of plausible e-values from the remaining samples, and determine E as the smallest hyper-rectangular set containing $\hat{E}$. This is the initial strategy we use when E is determined for the first time. For further uncertainty reduction we will use a refined strategy discussed in Section 3.1.

From the fitted distribution f_ζ, we obtain the non-parametric distribution $f_{a|e}$ as in Eq. (3). An illustration is given in Fig. 5 where some 2d marginals of $f_{a|e}$ are plotted for a few different values of e. It will also be useful to establish a parametric approximation to $f_{a|e}$, and for this we use a multivariate truncated Gaussian over (a_1, a_2, a_3, a_5), with a_4 uniform on the interval $[0,2]$. We write the parametric approximation as

$$\hat{f}^{\alpha}_{a|e} \approx f_{a|e}, \tag{4}$$

with distribution parameter α. We define the parametric approximation in terms of the marginals

$$\begin{aligned} &a_i \sim TN(\mu_i, \sigma_i, 0, 2) \text{ for } i = 1,2,3,5, \\ &a_4 \sim U([0,2]), \end{aligned} \tag{5}$$

and a Gaussian copula specified by a 5×5 correlation matrix $R = [\rho_{i,j}]$, where $\rho_{i,j}$ is the Spearman rank correlation coefficient between a_i and a_j. Here $a_i \sim TN(\cdot)$ denotes that a_i has a univariate normal distribution, conditioned on the event $a_i \in [0,2]$, and α is the vector of all parameters, μ_i, σ_i and $\rho_{i,j}$. Note that all of these parameters depend on e, i.e. $\alpha = \alpha(e)$, and we will estimate α based on samples (MCMC) from $f_{a|e}$.

Fig. 6 shows an example of the parametric distribution for a given value of e, using a maximum likelihood estimate of α. It turns out that for many values of $e \in E$, a mixture of two truncated Gaussians would provide a better fit. Other alternatives for the marginals could also be considered. However, we only intend to use the parametric distribution for initial estimates of failure probabilities, to help with importance sampling, and to approximate moments of $g(a,e,\theta)$ (in particular, as a crude approximation of how the variability of g changes with e and θ). And for this purpose, this simple model seems sufficient.

From the correlation matrix R we observe that there is usually some correlation between a_1, a_2 and a_3. Fig. 7 shows 2D marginals of some of these pairs (a_i, a_j). Note that the value of e used to generate the plots in Figs. 6 and 7 may be far from the correct one, but cannot be ruled out by the observations $\{y_i\}$ alone.

In reliability analysis, it is often useful to work with the random variables in the standard normal space. From the selected parametric distribution, we can perform the Nataf-type of transformation

$$a \xrightarrow{T_e} u \sim N(0, I),$$

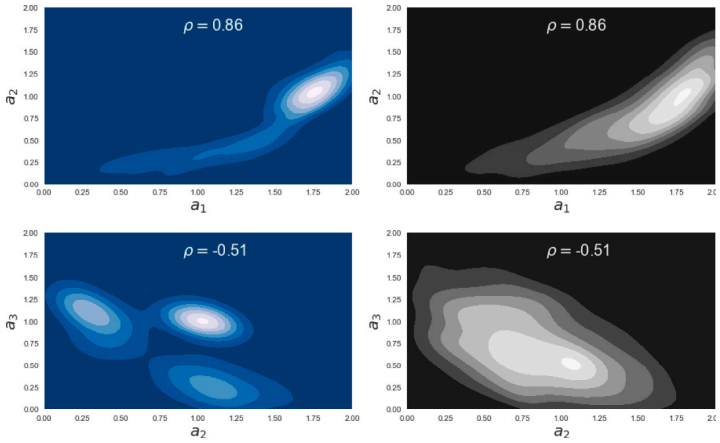


Fig. 7. 2D marginal distributions of (a_1, a_2) and (a_1, a_3), from the non-parametric distribution $f_{a|e}$ (left) and parametric approximation (right), for a fixed value of e.

which has the property that, for any $e \in E_0$ and $a \sim \hat{f}^{\alpha}_{a|e}$, $u = T_e(a)$ is n^a-dimensional standard normal. The transformation is obtained by first letting $z_i = \Phi^{-1}(F_i(a_i))$ where Φ is the standard normal CDF and F_i is the CDF of a_i. Then $z = (z_1, \dots, z_{n_a}) \sim N(0, R_0)$ with covariance matrix

$$[R_0]_{i,j} = 2 \sin\left(\frac{\pi \rho_{i,j}}{6}\right).$$

See for instance [6] for details. The standard normal variable is then obtained by $u = L_0^{-1} z$ where $L_0 L_0^T = R_0$.

Remark 2.2. Different values of e have been used to generate the plots in Fig. 5, Fig. 6 and Fig. 7, in order to illustrate some different versions of $f_{a|e}$ and $\hat{f}^{\alpha}_{a|e}$. These are not necessarily plausible values of e.

3. (B) uncertainty reduction

3.1. Ranking of epistemic parameters

To rank the epistemic parameters according to their ability to improve the predictive ability of $\hat{Y}$, we study how $f_{a|e}$ changes with respect to $e \in E$. To measure the effect of one component of $e = (e_1, e_2, e_3, e_4)$, say e_1, we estimate the expected Kullback–Leibler divergence

$$\mathbb{E}[\mathcal{KL}(f_{a|e} \parallel f_{a|e'})] = \mathbb{E}\left[\int_A f_{a|e}(a) \log \frac{f_{a|e}(a)}{f_{a|e'}(a)}\, da\right].$$

Here e' represents a small perturbation of e_1, $e_1' = e_1 + 0.1$, and the expectation is taken over the other parameters (e_2, e_3, e_4) that are assumed uniform within the bounds set by E.

From Fig. 8 we conclude that e_3 is more influential than e_2 and e_4, in the sense that small perturbations of e_3 has a larger effect on the related distribution $f_{a|e}$. The same is true for e_1, if e_1 is large in the first place. When deciding on the uncertainty reductions to make, we must also take into account that we may only request to increase or decrease the bounds of the initial set $E_0 = [0, 2]^4$. We observed that $e \in E \Rightarrow e_1 \ll 2$, and so decreasing the upper bound on e_1 might not provide any new information. Hence, we combine the information from Fig. 8 with the initial uncertainty reduction described in Section 2.3, where we assess the plausability of e. When also considering which e_i is close to the border of E_0, we decided to request uncertainty reduction on the lower bound on e_3 and e_4 ($\{e_3^-, e_4^-\}$).

3.2. First UM update

Our initial method for defining a set E of plausible e-values was based on the simple procedure described in Section 2.3, where we generate samples in $A \times E_0$ and filter out those with negligible likelihood. Now, when the bounding intervals have been reduced by this method, together with additional refinement provided by the request $\{e_3^-, e_4^-\}$, we switch to a more detailed method for further refinement of $E \to E_1$. The first updated UM is then $\langle f_{a|e}, E_1 \rangle$.

We may assess whether any $e \in E$ is plausible by generating samples $\{a^{(j)}\}$ from $f_{a|e}$, computing $\hat{y}_j = \hat{Y}(a^{(j)}, e)$, and comparing the set $\{\hat{y}_j\}$ with the provided observations $\{y_i\}$. In practice we will work in the reduced space, comparing $\zeta(a^{(j)}, e)$ against $\{\zeta_i\} = \{\zeta(y_i)\}$.

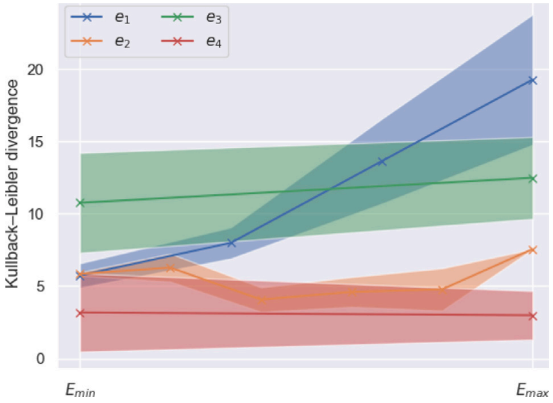


Fig. 8. Kullback–Leibler divergence between respective distributions $f_{a|e}$ for $e \in E$, when component e_i is perturbed by $e_i \to e_i + 0.1$. The figure shows mean and mean $\pm$ 1 standard deviation, when the remaining components, e_j for $j \neq i$, are uniform within E.

There are different ways of defining *plausability* of e, and we decided to go with a simple approach, where we compare the median of the probability density of 10^3 samples of $\zeta(a^{(j)}, e)$, with the density of the observations $\{\zeta_i\}$ given by (3). We use this approach, because of its simplicity, and because it was easy to find a suitable threshold that could be used to determine when a value of e was implausible. This procedure is more refined than the one presented in Section 2.3, where we filtered out the values of e with likelihood ≈ 0. But we note that the initial filtering was necessary, as there would be numerical issues with sampling from $f_{a|e}$ for the extremely unlikely values of e.

Our strategy for reducing the size of E is based on this procedure, where we check whether $e^{(k)}$ is plausible for a representative set $\{e^{(k)}\} \subset E$, and let E_1 be the smallest hyper-rectangular set containing all plausible $e^{(k)}$. The representative set $\{e^{(k)}\}$ is given as the union of the following two sets:

- a Latin Hypercube (LHS) sample within E,
- samples gathered form a Bayesian Optimization (BO) targeted at finding the most plausible $e \in E$.

Bayesian optimization (see for instance [7]) is a method for finding the maximum of a function using a small number of function evaluations, and we use it here as a technique to locate the most plausible candidates in E. We use this procedure as evaluating the plausability of a single value of e is based on sampling from $f_{a|e}$ and many computations of $\zeta(\cdot)$, which takes a bit of time.

Remark 3.1. The updated set E is determined from a threshold on the likelihood of the epistemic variable e. Initially we consider a threshold that is not overly conservative when we perform the model calibration. This makes it easier to find a design θ which is both acceptable with respect to improbable values of e, and closer to optimal with respect to the more realistic values of e. In Section 7 we discuss this in more detail, and we will also verify that the final design is acceptable even for a more conservative (larger) set E.

4. (C) reliability analysis of baseline design

The reliability analysis is performed for multiple designs, θ_{baseline}, θ_{new}, θ_{final} and $\theta_{\tilde{r}\%\text{risk}}$, and the final results are collected in Section 8.

4.1. Numerical procedure for estimation of failure probabilities

To estimate failure probabilities, we will work with the density $f_{a|e}$ given in Eq. (3). For some of the subproblems we address, it is not necessary to compute the normalizing constant C, and we define the proportional density

$$\dot{f}_{a|e}(a) = f_\zeta(\zeta(a, e)), \tag{6}$$

such that $f_{a|e}(a) = \dot{f}_{a|e}(a)/C$. In particular, from $\dot{f}_{a|e}(a)$ we may estimate moments of $f_{a|e}(a)$ using importance sampling, or generate samples using Markov chain Monte Carlo (MCMC).

Given a limit state function $g_i(a, e, \theta)$, the simplest way to estimate the failure probability p_f is by crude Monte Carlo (MC) sampling,

$$\hat{p}_{f,MC} = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}_{\{g_i(a_j, e, \theta) \geq 0\}}, \tag{7}$$

where $a_1, \dots, a_n$ are i.i.d. samples from $f_{a|e}$. The unbiased sample variance can be estimated by

$$\widehat{var}(\hat{p}_{f,MC}) = \frac{\hat{p}_{f,MC}(1-\hat{p}_{f,MC})}{n}. \tag{8}$$

A common technique to reduce the sample variance is by importance sampling, where we make use of some importance distribution $q(a)$, with the property that $q(a) \neq 0$ whenever $f_{a|e}(a) \neq 0$. The importance sampling estimate and the estimated sample variance are given by

$$\begin{aligned} \hat{p}_{f,IS} &= \frac{1}{Cn}\sum_{j=1}^{n} \mathbb{1}_{\{g_i(a_j,e,\theta)\geq 0\}} \frac{\dot{f}_{a|e}(a_j)}{q(a_j)}, \\ \widehat{var}(\hat{p}_{f,IS}) &= \frac{1}{C^2 n}\cdot \sum_{j=1}^{n}\left(\mathbb{1}_{\{g_i(a_j,e,\theta)\geq 0\}} \frac{\dot{f}_{a|e}(a_j)}{q(a_j)} - C\hat{p}_{f,IS}\right)^2, \end{aligned} \tag{9}$$

where a_j are sampled from $q(a)$.

We will use two different strategies for failure probability estimation, depending on whether the failure probability is large (> 0.01) or small (< 0.01). For large failure probabilities, we will rely on the MC estimate Eq. (7), where n is chosen such that the relative error is acceptable. To generate the samples a_j we make use of the Affine Invariant Markov chain Monte Carlo (MCMC) Ensemble sampler provided by [8].

For smaller failure probabilities, sampling directly from $f_{a|e}$ will be inefficient. Here we use the importance sampling estimate Eq. (9), where we select $q(a)$ from the design point of a FORM analysis. This is achieved by first running a FORM analysis using the parametric approximation $\hat{f}^{\alpha}_{a|e}$. FORM is based on obtaining a linear approximation to the limit-state at a point a^* called the *design point*. The design point is the point on the limit state, $g_i = 0$, with largest probability density. See for instance [9] for further details. We define $q(a)$ as a modified version of $\hat{f}^{\alpha}_{a|e}$, where the distribution is shifted such that $\mathbb{E}[q] = a^*$ (it may also be useful to increase the variance slightly). The distribution $q(a)$ obtained in this way should then be able to produce samples a_j where both $g_i(a_j, e, \theta) \geq 0$ and $f_{a|e}(a_j)$ is large, which is needed to reduce the variance of the estimated failure probability. When the failure probability for g_1, g_2 and g_3 have been estimated, the samples used for each individual limit state can also be used for w in order to reduce the total number of function evaluations of $g(\cdot)$.

To estimate the range of the failure probability for e in some set E, we compute the failure probability for a finite subset $\{e^{(k)}\} \subset E$ as described in Section 3.2. In order to capture the relevant ranges of failure probabilities, we found that around 100 e-values, $|\{e^{(k)}\}| \approx 100$, seemed sufficient.

4.2. Ranking of epistemic uncertainties

To rank the epistemic uncertainties according to the contraction of the failure probability $p_{f,w}(e,\theta) = \mathbb{P}(w(a,e,\theta) \geq 0)$, we estimate the change in the minimum and maximum of $\{p_{f,w}(e,\theta)\}_{e\in E}$ when we replace E with a reduced set E'. Again we make use of the finite subset $\{e^{(k)}\} \subset E$ for which $p_{f,w}(e,\theta)$ is already available. The results are given in Section 8, where we let E' be the set given by increasing the lower bounds and decreasing the upper bounds of E with 25%.

5. (D) reliability-based design

5.1. Optimality criterion

As we do not have any information on the criticality of each failure mode, i.e. whether some limit state g_i should be seen as more important than the others, we will seek a design θ where $g_i(a,e,\theta)$ is as "small as possible" for all $i = 1,2,3$. Our optimality criterion will be based on the characteristic values

$$c_i(e,\theta) = \mathbb{E}\left[g_i(a,e,\theta)\right] + 2\ \mathrm{Std}(g_i(a,e,\theta)), \tag{10}$$

where the expectation and standard deviation (Std) are taken with respect to $a \sim f_{a|e}$. From the characteristic values c_i, we define the following loss function:

$$L(e,\theta) = \sum_{i=1}^{3} \exp[\gamma_i c_i(e,\theta)]. \tag{11}$$

Here $\gamma_i > 0$ are constants needed to bring each c_i to the same scale, as the output of each g_i are of different orders of magnitude. In our implementation we have used $\gamma_1 = 10$, $\gamma_2 = 200$ and $\gamma_3 = 1$.

Given some set E of plausible e-values, we define the reliability-optimal design as a solution θ^* of the optimization problem

$$\theta^* \in \arg\min_{\theta}\{\max_{e\in E} L(e,\theta)\}. \tag{12}$$

We will make use of an approximation to Eq. (11) described below, and the final results are collected in Section 8.

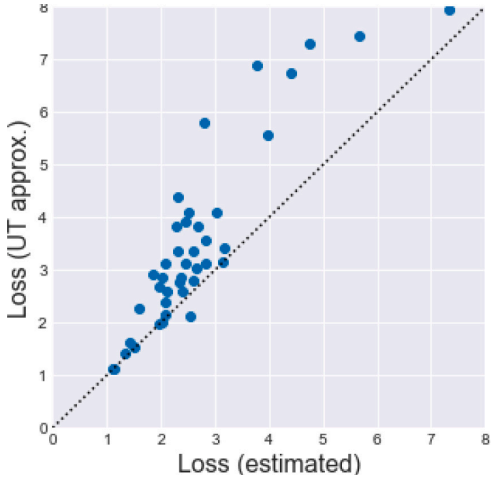


Fig. 9. UT approximation of the loss function Eq. (11) vs estimated (MCMC) loss.

5.2. Numerical UT approximation

For numerical efficiency, we introduce an approximation of the loss function Eq. (11) based on the Unscented Transform (UT). UT is an efficient method for estimating the mean and covariance of a random variable after nonlinear transformation. In short, given some random variable x we define a set of *weighted sigma-points* ($\{w_i, x_i\}$), such that if ($\{w_i, x_i\}$) were considered as a discrete probability distribution, then its mean and covariance would coincide with x. For any nonlinear transformation $y = h(x)$, if x is discrete we may compute the mean and covariance of y exactly. The UT approximation is the result of such computation, when we approximate x with ($\{w_i, x_i\}$). For details see e.g. [10].

To select the set of sigma-points in UT, we make use of the method developed by [11], which produces a set of $2n_a + 1$ sigma-points and weights in $\mathbb{R}^{n_a}$. These points are generated under the assumption that x follows a n_a-dimensional standard normal distribution. The sigma-points corresponding to the parametric distribution $\hat{f}^{\alpha}_{a|e}$ can then be obtained by the transformation discussed in Section 2.3. From here, the characteristic values c_i in Eq. (10) can then be estimated by just 11 evaluations of the limit state.

Fig. 9 shows the UT approximation of the loss $L(e, \theta)$ for some different values of θ and e. With the goal of minimizing the loss estimated using samples from $f_{a|e}$, the UT approximation seems like a viable proxy, and the optimization problem in Eq. (12) can then be solved by standard tools for numerical optimization.

6. (E) model update and design tuning

Here we rely on the same procedure for dimensionality reduction as discussed in Section 2.1, in order to refine the UM based on samples from the integrated system $\{z^{(i)}\}$, together with the numerical model $\hat{Z}(a, e, \theta)$. Hence, we obtain an updated density ζ fitted to two sets of parameter vectors, corresponding to the complex exponential representation of the two datasets $\{y^{(i)}\}$ and $\{z^{(i)}\}$. We found it difficult to assess potential model discrepancy in $\hat{Z}(a, e, \theta)$. We suspect that there is more discrepancy in $\hat{Z}(a, e, \theta)$ than in $\hat{Y}(a, e)$, but chose to update the UM under the assumption that $\hat{Z}(a, e, \theta)$ still provides an accurate model of the data generating process. However, if this is not the case then our second refined UM may be too optimistic (see Remark 2.1). Based on the same type of assessment as in Section 3.1, we decided to again request the uncertainty reduction $\{e_3^-, e_4^-\}$.

We refer to the second updated UM as $\langle f_{a|e}, E_2 \rangle$, where $E_2 \subset E_1$ and $f_{a|e}$ is given by Eq. (3) where f_ζ is the updated based on $\{z^{(i)}\}$.

7. (F) risk-based design

As discussed in Remark 3.1, the set E_2 used to find θ_{final} is based on a threshold on the likelihood of the epistemic variable e which is not overly conservative. Now, we will introduce a set $E_{0\%\text{risk}}$ where $e \notin E_{0\%\text{risk}}$ is assumed impossible. We say that $e \notin E_{0\%\text{risk}}$ if there is at least one observation y_i which cannot be explained by $\hat{Y}(a, e)$, for any $a \in A$. (But note that the comments in Remark 2.1 still apply).

When we define smaller sets $E_{r\%\text{risk}} \subset E_{0\%\text{risk}}$, corresponding to neglecting a portion of "r% risk", we let $E_{r\%\text{risk}}$ be the set containing the $(1 - r)$% values of e with largest likelihood. In practice we use a finite sets of points $\{e_i\}$ to represent these sets, and so the shape of $E_{r\%\text{risk}}$ is not necessarily rectangular. But we can still associate to each set $E_{r\%\text{risk}}$ the smallest hyper-rectangular set containing it, as illustrated in Fig. 10. The set E_2 defined in Section 6, which is used in the optimization of θ_{final}, is the set

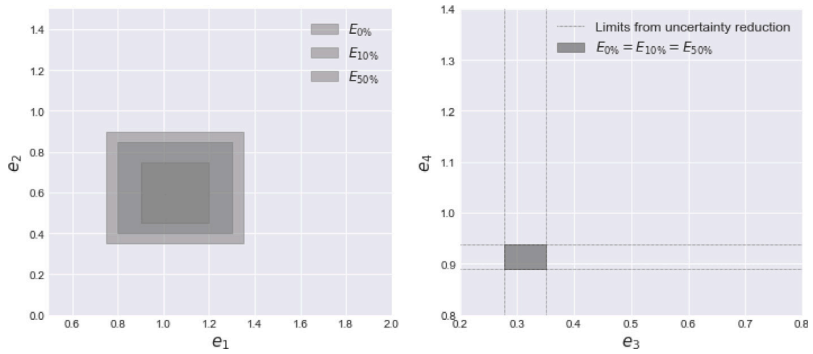


Fig. 10. Illustration of the risk-based sets $E_{r\%\text{risk}}$. The set E_2 used in the optimization of θ_{final} is $E_2 = E_{50\%\text{risk}}$. The projection onto e_3–e_4 is the same for all three sets.

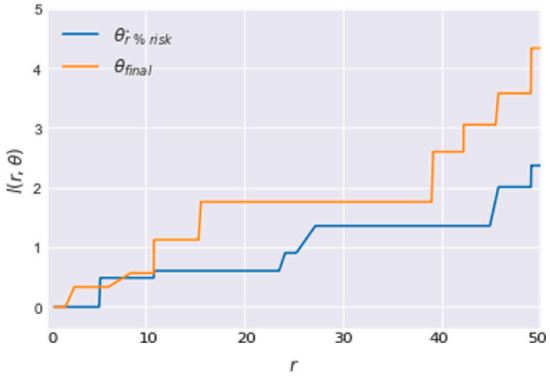


Fig. 11. The gain (13) for θ_{final} and $\theta_{\hat{r}\%\text{risk}}$.

corresponding to $r = 50\%$ risk. Now we will also find an optimal design $\theta_{\hat{r}\%\text{risk}}$ for $\hat{r} = 5\%$, corresponding to $E_{5\%\text{risk}}$. Both θ_{final} and $\theta_{\hat{r}\%\text{risk}}$ will then be evaluated with respect to the conservative scenario $E_{0\%\text{risk}}$ and the more optimistic scenario $E_{50\%\text{risk}}$ in Section 8.

To quantify the gain $l(r,\theta)$ resulting from taking the risk r, we let

$$l(r,\theta) = \max_{e\in E_{0\%}} L(e,\theta) - \max_{e\in E_{r\%}} L(e,\theta), \tag{13}$$

where $L(e,\theta)$ is the loss function defined in (11). The evaluation of the design $\theta_{\hat{r}\%\text{risk}}$ that maximizes $l(\hat{r},\theta)$ for $\hat{r} = 5\%$ is presented in Section 8, Table 6, and the gain $l(r,\theta)$ for both θ_{final} and $\theta_{\hat{r}\%\text{risk}}$ is shown for a range of values r in Fig. 11.

Here we note that a 5% reduction of the epistemic space has very little effect, and optimization over $E_{5\%\text{risk}}$ is practically the same as optimization over $E_{0\%\text{risk}}$. We may therefore consider $\theta_{\hat{r}\%\text{risk}}$ as the design optimized for the worst-case scenario, which we can compare against the more optimistic θ_{final} corresponding to $r = 50\%$. In Fig. 11 we would expect that the curve for θ_{final} had the steepest slope, i.e. that there is more to be gained by reducing the epistemic set E for θ_{final} than $\theta_{\hat{r}\%\text{risk}}$, but in practice it is difficult to determine if there is any significant difference in the two designs at all. This is also reflected in the evaluation of failure probabilities in Section 8.

8. Final results

8.1. Failure probability and severity

We planned to make use of crude MC for initial computation of failure probabilities that are not too small, and switch to importance sampling after UM refinement and design optimization where more accurate estimation is needed. However, after the optimization and UM refinement, the failure probabilities are no longer computable. This happens when it is not possible to find any $(a,e) \in A \times E$ where $g_i \geq 0$. And this seems to be the case for the second refined UM and θ_{final}, as we were not able to find any $(a,e) \in A \times E_2$ where $g_i(a,e) \geq 0$ for $i = 1$ and $i = 3$ through numerical global optimization (maximization). We could of course tune

Table 4
Failure probability — First refined UM.

θ	Limit-state	p_f min	p_f max	Severity
θ_{baseline}	g_1	–	0.145	0.030
	g_2	0.010	0.715	0.003
	g_3	–	0.507	0.246
	w	0.048	0.727	NA
θ_{new}	g_1	–	0.045	$1.8 \cdot 10^{-4}$
	g_2	–	0.522	$1.1 \cdot 10^{-3}$
	g_3	–	–	–
	w	–	0.522	NA

*MC estimates. Missing p_f values (–) can be assumed $< 10^{-3}$.

Table 5
Failure probability — Second refined UM.

θ	Limit-state	p_f min	p_f max	Severity
θ_{baseline}	g_1	–	0.224	$2.1 \cdot 10^{-2}$
	g_2	0.019	0.162	$4.5 \cdot 10^{-4}$
	g_3	–	0.021	$1.1 \cdot 10^{-2}$
	w	0.026	0.341	NA
θ_{new}	g_1	–	0.001	$3.9 \cdot 10^{-6}$
	g_2	0.006	0.030	$2.9 \cdot 10^{-5}$
	g_3	–	–	–
	w	0.006	0.030	NA
θ_{final}	g_1	**0**	**0**	**0**
	g_2	–	0.008	$1.6 \cdot 10^{-5}$
	g_3	**0**	**0**	**0**
	w	–	0.008	NA

*MC estimates. Missing p_f values (–) can be assumed $< 10^{-3}$, and **0** indicates that $\max_{a\in A, e\in E_2} g_i < 0$.

Table 6
Failure probability — Risk-based evaluation.

θ	Limit-state	Full E		Reduced E	
		p_f max	Severity	p_f max	Severity
θ_{risk}	g_1	–	–	–	–
	g_2	0.055	$1.2 \cdot 10^{-4}$	0.021	$3.8 \cdot 10^{-5}$
	g_3	–	–	–	–
	w	0.055	NA	0.021	NA
θ_{final}	g_1	–	–	**0**	**0**
	g_2	0.061	$1.6 \cdot 10^{-4}$	0.008	$1.6 \cdot 10^{-5}$
	g_3	–	–	**0**	**0**
	w	0.061	NA	0.008	NA

*MC estimates. Missing p_f values (–) can be assumed $< 10^{-3}$, and **0** indicates that $\max_{a\in A, e\in E} g_i < 0$. Full E and reduced E correspond to $E_{0\%}$ and $E_{50\%}$ in Fig. 10.

the optimization of θ_{final} further, putting less wight on g_1 and g_3, but chose to with the current alternative under the assumption that g_1 and g_3 are the most critical failure modes.

The range of the failure probability, $p_f(e, \theta)$, for each of the limit-states g_1, g_2, g_3 and w, is shown in Tables 4 and 5 for the first and second refined UM's respectively. The tables also include the severity of each individual requirement violation.

8.2. Ranking of epistemic uncertainties

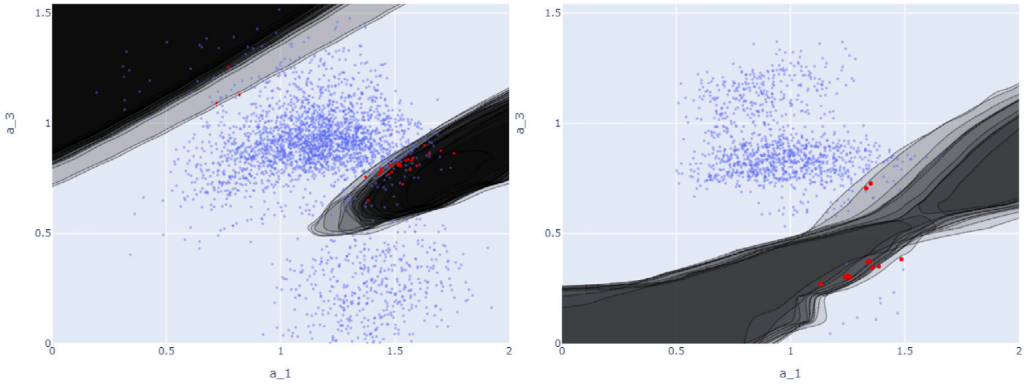
The resulting ranking as described in Section 4.2 is given in Table 7.

8.3. Transition to failure

Based on a set of plausible $e \in E_1$, the design point a^* (most probable point on $g_i = 0$) has been estimated based on MCMC. Fig. 12(a) shows an example of the failure regions for a_1, a_3. The black lines represent the transition boundary $g_1 = 0$ for plausible e's, while the shaded area represent the failure regions where $g_1 > 0$. a_1 and a_3 has been varied across the entire possible range $[0, 2]$ while a_2, a_4, a_5 has been kept fixed at each e's design point a^* plotted as red dots. The blue dots are samples from $f_{a|e}$ for different values of e. Note that, even though some of the blue dots are located inside the failure domains, this does not necessarily

Table 7
e-ranking — First refined UM.

θ	#1	#2	#3	#4
	First refined UM			
θ_{baseline}	e_3	e_4	e_1	e_2
θ_{new}	e_3	e_1	e_2	e_4
	Second refined UM			
θ_{baseline}	e_1	e_2	e_4	e_3
θ_{new}	e_3	e_4	e_1	e_2
θ_{final}	e_4	e_3	e_1	e_2



(a) $g_1(a_1, a_3, \theta_{\text{baseline}}|a^*, e)$ for $e \in E_1$ (b) $g_2(a_1, a_3, \theta_{\text{final}}|a^*, e)$ for $e \in E_2$

Fig. 12. Transition to failure.

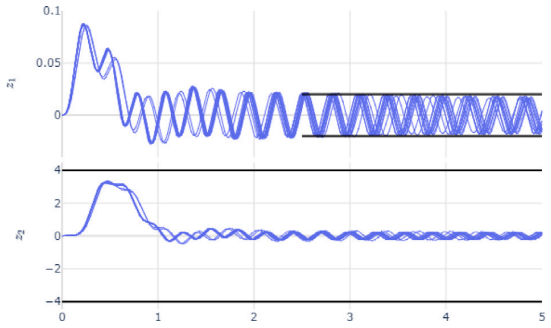


Fig. 13. $\hat{Z}(a^*, e, \theta_{\text{final}})$ for $e \in E_2$.

mean that these are failure points, it is just their projection down in a_1, a_3. But it gives an indication of where the main mass of the distribution lies with respect to the failure regions.

The failure boundaries in Fig. 12(a) are quite co-located in the a_1, a_3 space, which shows that for the e-box E_1 the failure region is reasonably stable in this parameter space. Other parameter spaces do not have a similar clear failure region.

Fig. 12(b) shows a similar plot of the g_2 failure region and transition boundary for plausible $e \in E_2$ in the a_1, a_3 parameter space, and Fig. 13 shows the corresponding time responses of the integrated system.

9. Concluding remarks

We conclude by summing up some of the key lessons learned through the challenge. A central part of the problem is dealing with functional (time-series) data, which makes inference challenging, especially without underlying knowledge about what the data represents. In particular, the nonparametric route can give challenges with MCMC, and ensuring that this works as correctly can

be time consuming. With some assumptions regarding the input distribution (if correct), we could make use of a more numerically stable alternative.

The combination of epistemic and aleatory uncertainty also makes the Bayesian inference challenging. Especially with a set of observations that is relatively small, combined with a nonlinear mapping that makes the true input distribution unidentifiable. That is, there could be different values of the epistemic variable e, combined with quite different distributions over the aleatory variable a, that could explain the data equally well. But for the objectives posed in this challenge, inferring the correct distribution of a is not really important, as long as we can find designs that are robust to any of the plausible distributions $f_{a|e}$.

Our initial attempt was to try an entropy-based approach based on information geometry. Here we would let the aleatory distribution be the distribution with maximum entropy that satisfies some constraints set by the data. This is possible, but numerically challenging, and we did not find a way to develop a numerically stable method that could handle time-series data. As discussed in Section 2.1, we did FPCA (Functional Principal Component Analysis) on the time-series data, which gives a set of uncorrelated features that can be used for dimensionality reduction. Interestingly, these features still had a very complicated dependency structure, and the theoretical motivation for using FPCA was then not so relevant.

We define the different epistemic sets E using a threshold on the likelihood (plausibility) of the epistemic variable e, and this threshold value is something that we set manually. We chose a non-conservative set ($\theta_{50\%\text{risk}}$ in Section 7) to use in the initial design optimization, followed by a verification of the design using a larger conservative set ($\theta_{0\%\text{risk}}$) of epistemic values. The idea behind this, is that there may be many designs that are acceptable with respect to a large set E, and out of these we want to select one that has good performance with respect to the most likely values of e. We considered a more generic methodology, by establishing an criterion that gives a suitable balance between acceptable performance in the worst-case scenario and optimality in the high-likelihood scenarios. Alternatively by considering Pareto optimality with respect to a criterion for acceptable worst-case performance. But we found it difficult to come up with a meaningful criterion, especially without knowledge about the true physical system.

As discussed in Section 2.2, it is important that model discrepancy is handled appropriately. For inference to be possible, we need to decide when two time series are "basically the same". The choice of metric will have to be made based on some assumptions, which could have a large effect on subsequent analysis. Since we assume noiseless observations, we did not want to impose too much regularization, as this might "wash out" the information given in the relatively small set of observations. As a result, we had to spend some time investigating model discrepancy. In the end, in order to assume zero observational error, we needed to assume a small model discrepancy in order for any form of inference to be possible. As we comment in Section 6 and Remark 2.1 in Section 2.2, our final results may be sensitive to this assumption. But in a real-world scenario, involving either observational noise, model discrepancy, or both, this part of the challenge would of course be treated quite differently.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been supported by grant 276282 from the Norwegian Research Council and DNV Group Research and Development, Norway. Finally we would like to thank Luis G. Crespo and Sean P. Kenny from the NASA Langley Research Center for providing an interesting and relevant problem to the research community and for hosting the UQ challenge.

References

[1] C. Agrell, S. Eldevik, O. Gramstad, A. Hafver, Contribution to the NASA langley UQ challenge on optimization under uncertainty, in: ESREL 2020, 2020.
[2] L.G. Crespo, S.P. Kenny, The NASA langley challenge on optimization under uncertainty, Mech. Syst. Signal Process. 152 (2021) 107405.
[3] J.-L. Wang, J.-M. Chiou, H.-G. Müller, Functional data analysis, Annu. Rev. Stat. Appl. 3 (1) (2016) 257–295.
[4] J.F. Hauer, C.J. Demeure, L.L. Scharf, Initial results in Prony analysis of power system response signals, IEEE Trans. Power Syst. 5 (1) (1990) 80–89.
[5] M.C. Kennedy, A. O'Hagan, Bayesian Calibration of computer models, J. R. Stat. Soc. Ser. B Stat. Methodol. 63 (3) (2001) 425–464.
[6] R. Lebrun, A. Dutfoy, An innovating analysis of the Nataf transformation from the copula viewpoint, Probab. Eng. Mech. 24 (2009) 312–320.
[7] E. Brochu, V.M. Cora, N. de Freitas, A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning, 2010, arXiv:1012.2599.
[8] D. Foreman-Mackey, D.W. Hogg, D. Lang, J. Goodman, Emcee: The MCMC hammer, PASP 125 (2013) 306–312.
[9] H. Madsen, S. Krenk, N. Lind, Methods of Structural Safety, in: Dover Civil and Mechanical Engineering Series, Dover Publications, 2006.
[10] S. Julier, J. Uhlmann, Unscented filtering and nonlinear estimation, Proc. IEEE 92 (2004) 401–422.
[11] R. van der Merwe, E. Wan, Sigma-point Kalman filters for probabilistic inference in dynamic state-space models, in: Proceedings of the Workshop on Advances in Machine Learning, 2003.

Paper VI

Environmental contours as Voronoi cells

Andreas Hafver, Christian Agrell, Erik Vanem

Submitted for publication. arXiv: 2008.13480

Environmental contours as Voronoi cells

Andreas Hafver · Christian Agrell · Erik Vanem

Abstract Environmental contours are widely used as basis for design of structures exposed to environmental loads. The basic idea of the method is to decouple the environmental description from the structural response. This is done by establishing an envelope of joint extreme values representing critical environmental conditions, such that any structure tolerating loads on this envelope will have a failure probability smaller than a prescribed value.

Specifically, given an n-dimensional random variable $\mathbf{X}$ and a target probability of failure p_e, an *environmental contour* is the boundary of a set $\mathcal{B} \subset \mathbb{R}^n$ with the following property: For any *failure set* $\mathcal{F} \subset \mathbb{R}^n$, if $\mathcal{F}$ does not intersect the interior of $\mathcal{B}$, then the probability of failure, $P(\mathbf{X} \in \mathcal{F})$, is bounded above by p_e. As is common for many real-world applications, we work under the assumption that failure sets are convex.

In this paper, we show that such environmental contours may be regarded as boundaries of Voronoi cells. This geometric interpretation leads to new theoretical insights and suggests a simple novel construction algorithm that guarantees the desired probabilistic properties. The method is illustrated with examples in two and three dimensions, but the results extend to environmental contours in arbitrary dimensions. Inspired by the Voronoi-Delaunay duality in the numerical discrete scenario, we are also able to derive an analytical representation where the environmental contour is considered as a differentiable manifold, and a criterion for its existence is established.

Keywords Multivariate extremes · Convexity · Computational geometry · Differential geometry

1 Introduction and background

1.1 A brief review of environmental contours

The use of environmental contours is a well-established practice in design of marine structures, and helps the designer identify design sea states corresponding to extreme environmental loads associated with a certain return period. The concept of environmental contours is an efficient method for estimating multivariate extreme conditions, and it is an alternative to full long-term response analysis in situations where this is not feasible. An environmental contour is a set constructed based on a joint probability distribution for the relevant input parameters, for example

Andreas Hafver
DNV Group Research and Development
E-mail: andreas.hafver@dnv.com

Christian Agrell
Department of Mathematics, University of Oslo, Norway
E-mail: chrisagr@math.uio.no
DNV Group Research and Development
E-mail: christian.agrell@dnv.com

Erik Vanem
Department of Mathematics, University of Oslo, Norway
E-mail: erikvan@math.uio.no
DNV Group Research and Development
E-mail: erik.vanem@dnv.com

significant wave height and wave period. The environmental contour method is recommended in standards and recommended practices such as [14,45].

The concept of environmental contours was first introduced by [19,20] as a means to study the joint distribution of significant wave height and wave period of ocean waves. These early environmental contours were based on constant densities, but the concept of environmental contours was developed further by [61] by using the Inverse First Order Reliability Method (IFORM) and considering exceedance probabilities in the transformed standard normal space [22]. The IFORM method avoids unnecessary conservatism in the equi-density contours [30], and has since then become the most applied contour method. Several applications of the environmental contour method in marine engineering and design are reported in the literature [44,62,5,51,6,3,4,29,21,42]. A comparison study presented in [1] investigated the influence of the choice of contour method on some vessel responses.

Environmental contours continues to be an active area of research, and several modified approaches have been suggested in recent years, e.g., a dynamical IFORM method [34], a modified approach to account for non-monotonic behaviour of the responses [32], an approach including preprocessing and principal component analysis prior to estimating IFORM contours [15], contours for sub-populations such as directional sectors or seasonality [56,24], contours for a combination of circular and linear variables [17], contours for copula-based joint distributions [53,39] and contours based on a direct IFORM approach [13]. Contours for buffered failure probabilities were proposed in [12] and contours based on a particular version of the inverse second order reliability method (ISORM) were derived in [11]. Recently, the initial equi-density method was revisited in [18]. The uncertainties associated with environmental contours due to uncertainties in the underlying joint distribution model and due to sampling variability are investigated in [41] and [59], respectively, and weighted environmental contours based on combining data from different datasets were explored in [57]. Reviews of various contour methods are presented in e.g. [36,49].

An alternative approach to constructing environmental contours that avoids the transformation into standard normal space, but rather defines exceedance probabilities in the original parameter space, was proposed in [26,28]. This is based on Monte Carlo simulations from the joint distribution of environmental parameters, and initial inaccuracies due to insufficient number of Monte Carlo samples were overcome by a scheme for tail sampling as outlined in [27]. It is argued that the contours obtained in this way have more well defined probabilistic properties, and an evaluation of the properties of the IFORM-based environmental contours is presented in [25]. However, in some situations it is found that the direct sampling contours may contain irregularities in the form of small loops, as discussed in [28]. One reason for this is related to the Monte Carlo variance and the fact that the contours are estimated based on a finite sample from the joint distribution, and the issue may be resolved by increasing the number of Monte Carlo samples. However, the reason may also be genuine features of the underlying joint distribution, i.e. that the joint distribution does not admit a proper convex environmental contour. A comparison study on the IFORM and the Monte Carlo-based approach to environmental contours was presented in [58], which demonstrated that in certain cases, notably different contours are obtained. The comparison study was extended to consider various simple structural problems in [54] and to compare contour-based methods to response-based methods in [60].

Even though many structural problems depend on more than two environmental variables, most applications of environmental contours are restricted to two-dimensional contours. For example, in the multivariate problem addressed in [43], environmental contours were only calculated for pairs of variables. However, some examples of three-dimensional contours based on the IFORM approach, are shown in [33,50,47,40]. An extension of the direct sampling approach to three-dimensional problems was outlined in [55], and this method was applied to the tension in a mooring line of a semi-submersible in [48]. However, even though extensions of the direct sampling approach to environmental contour to higher dimensional problems is indeed possible, calculating the contours becomes increasingly cumbersome in higher dimensions.

1.2 Contribution of this paper

In this paper, an alternative way of constructing environmental contours is proposed, that easily generalises to arbitrary dimensions. With this method, environmental contours can be described as boundaries of Voronoi cells, which may easily be found from standard software packages at reasonable computational costs. The method makes use of Monte Carlo samples from the underlying

distribution, but overcomes the common loop-problem of direct sampling methods, and can be used to produce convex contours with the desired probabilistic properties.

In Section 2 we briefly review the mathematical definition of environmental contours. In Section 3 we give a general introduction to Voronoi cells, before showing in Section 4 that environmental contours may be interpreted as boundaries of Voronoi cells. In Section 5 we generalise results from Section 4 to the continuous limit, deriving additional theoretical insights, including an analytic formula for environmental contours in terms of a given percentile function. Section 6 details the practical application of the proposed algorithm, and examples in two and three dimensions are provided in Section 7. Some concluding remarks are provided in section 8. For brevity, proofs are contained in appendices.

2 Definition of environmental contours

We consider a structure or component exposed to some environmental loads. The environmental loads can be represented by a vector of variables $\mathbf{X} \in \mathcal{X} \subseteq \mathbb{R}^n$, distributed according to some multivariate probability distribution $f_{\mathbf{X}}(\mathbf{x})$. We further define a performance function $g(\mathbf{x})$, where $\mathbf{x}$ is a specific environmental state, such that the structure or component remains intact/functioning as long as $g(\mathbf{x}) \geq 0$, and fails if $g(\mathbf{x}) < 0$.

The failure region $\mathcal{F} = \{\mathbf{x} \in \mathcal{X} : g(\mathbf{x}) < 0\}$ and the corresponding failure probability $p_f = P(\mathbf{X} \in \mathcal{F}) = \int_{\mathcal{F}} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}$ are generally unknown. However, in many cases, one may argue based on physics that $\mathcal{F}$ must be convex. Therefore, if we can find another convex set $\mathcal{B}$ such that $g(\mathbf{x}) \geq 0$ $\forall \mathbf{x} \in \mathcal{B}$, it follows from convexity theory that there exist a *supporting hyperplane* Π that separates $\mathcal{B}$ and $\mathcal{F}$ (i.e. $\mathcal{B} \subseteq \Pi^-$ and $\mathcal{F} \subseteq \Pi^+$, where Π^- and Π^- are the two half spaces separated by Π), and $p_f \leq P(\mathbf{X} \in \Pi^+) = \int_{\Pi^+} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}$.

In particular, we may construct the set

$$\mathcal{B}_{p_e} = \bigcap_{\mathbf{u} \in \mathcal{U}} \Pi^-_{p_e}(\mathbf{u}), \tag{1}$$

where $\mathcal{U}$ denotes the set of all unit vectors in $\mathbb{R}^n$, i.e.

$$\mathcal{U} = \{\mathbf{u} \in \mathbb{R}^n \mid \|\mathbf{u}\| = 1\}, \tag{2}$$

and $\Pi^-_{p_e}(\mathbf{u})$ is the half-space normal to $\mathbf{u}$ with the property that $P(\mathbf{X} \in \Pi^-(\mathbf{u})) = 1 - p_e$. More precisely,

$$\Pi^-_{p_e}(\mathbf{u}) = \{\mathbf{x} : \mathbf{u} \cdot \mathbf{x} \leq C_{p_e}(\mathbf{u})\}, \tag{3}$$

where C_{p_e} denotes the p_e-level percentile function, defined by

$$C_{p_e}(\mathbf{u}) = \inf\{c : P(\mathbf{u} \cdot \mathbf{X} > c) \leq p_e\}. \tag{4}$$

We will assume that the distribution of $\mathbf{X}$ is absolutely continuous with respect to the Lebesgue measure on $\mathbb{R}^n$, so the function $C_{p_e}(\mathbf{u})$ in (4) is well defined. We note also that (1) uniquely defines a convex set, as all half-spaces $\Pi^-_{p_e}(\mathbf{u})$ are convex.

Depending on the distribution of $\mathbf{X}$, the definition of $\mathcal{B}_{p_e}$ in (1) does not imply that all hyperplanes $\Pi_{p_e}(\mathbf{u})$ intersect $\mathcal{B}_{p_e}$. (See for instance the discussion in Section 4 or the example given in Figure 7.) In the case where all hyperplanes $\Pi_{p_e}(\mathbf{u})$ intersect $\mathcal{B}_{p_e}$, the authors in [28] state that $\mathbf{X}$ *admits* a p_e-contour. We will make use of the equivalent definition below.

Definition 1 Let $\mathcal{B}_{p_e}$ be a nonempty convex set in $\mathbb{R}^n$ and $p_e \in (0, 0.5)$. If

$$P(\mathbf{X} \in \Pi^+) \leq p_e \tag{5}$$

for any supporting half-space Π^+ of $\mathcal{B}_{p_e}$, we say that $\partial\mathcal{B}_{p_e}$ is a valid environmental contour of $\mathbf{X}$ with respect to the target probability p_e. If (5) holds with equality for all the supporting half-spaces Π^+, then $\partial\mathcal{B}_{p_e}$ is also a proper environmental contour.

In the case where a proper convex environmental contour exists, it is necessarily given by the representation in (1). This follows from the fact that any closed convex subset $\mathcal{B} \subset \mathbb{R}^n$ is the intersection of all supporting half-spaces that contain $\mathcal{B}$ (see e.g. Theorem 3.6.18 in [31]). If all those half-spaces satisfy (5) with equality, then the representation in (1) follows. For reference we state this in a separate proposition.

Proposition 1 *Assume that the random variable* $\mathbf{X}$ *admits a proper convex environmental contour* $\partial\mathcal{B}_{p_e}$ *with respect to a target probability* $p_e \in (0, 0.5)$*. Then the closure of* $\mathcal{B}_{p_e}$ *is uniquely defined by* (1)*.*

In the following we will start by assuming that $\mathbf{X}$ admits a proper convex environmental contour, and also that the probabilities $P(\mathbf{X} \in \Pi^+)$ can be computed without error. After introducing the connection with Voronoi cells and an algorithm for constructing $\mathcal{B}_{p_e}$, we present an approach that can be used when these assumptions are relaxed.

3 Voronoi cells

The Voronoi diagram is a fundamental data structure in computational geometry that has found applications in a variety of fields, including physics, biology, cartography, crystallography, ecology, geology, anthropology, and meteorology to mention some [46]. Given a set of points $p_1, \dots p_k$ in a metric space $\mathcal{X}$, the Voronoi diagram is defined as the partitioning of $\mathcal{X}$ into regions $R_1, \dots R_k$, such that R_i contains all points in $\mathcal{X}$ whose distance to p_i is not greater than their distance to any other p_j for $j \neq i$. The region R_i is often referred to as the *Voronoi cell* of p_i (with respect to the remaining points p_j, $j \neq i$).

In its canonical form, a Voronoi diagram is constructed from a set of points in $\mathbb{R}^n$ endowed with the Euclidean metric, and other alternatives are usually referred to as *Generalised Voronoi diagrams* [35,2]. In this paper, we will consider the Voronoi cell of a point $\mathbf{o} \in \mathbb{R}^n$ with respect to a set $S \subset \mathbb{R}^n$. We denote the Voronoi cell by $\mathrm{Vor}(\mathbf{o}, S)$, and it is the set containing all points that are at least as close to $\mathbf{o}$ as any point in S, measured by the Euclidean distance in $\mathbb{R}^n$.

$$\mathrm{Vor}(\mathbf{o}, S) = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{o}\| \leq \inf_{\mathbf{s} \in S} \|\mathbf{x} - \mathbf{s}\| \right\}. \tag{6}$$

The distance function used to define $\mathrm{Vor}(\mathbf{o}, S)$ could also be interpreted as the Hausdorff distance between the singleton set $\{\mathbf{o}\}$ and S, but we will not make use of this property in this paper. To motivate the algorithm presented in this paper we will make use of the rather trivial property that if the set S is finite, then it is equivalent to the canonical definition of (point) Voronoi cells as illustrated in Figure 1. In the following section we show that an environmental contour can be represented as a Voronoi cell of the form (6). A numerical approximation is then achieved by replacing the set S in (6) with a finite subset, where available algorithms developed for canonical (point) Voronoi diagrams can be used. In this case we will also make use of the Delaunay triangulation of the finite point set, that correspond to the dual graph of the Voronoi diagram. This is illustrated for points in the plane in Figure 1, and we refer to [46] for further details.

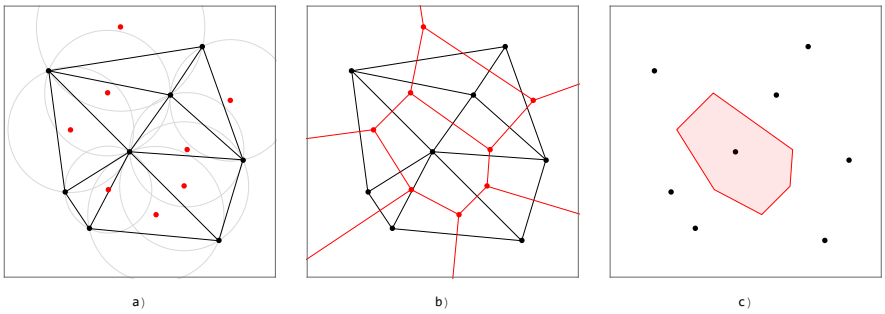


Fig. 1: Illustration of Delaunay triangulation and Voronoi diagram of a set of points. a) A Delaunay triangulation of the 8 **black points** is defined as a triangulation such that no point lies inside the circumcircle of any triangle. The **red points** are the centers of each circumcircle. b) The Voronoi diagram (**red lines**) corresponds to the graph with the circumcenters as edges. c) The Voronoi cell of one of the points.

4 Environmental contours as boundaries of Voronoi cells

In this section we give a representation of the environmental contours described in Section 2 using Voronoi cells of the form (6). We start by introducing the general construction and present some theoretical properties, in anticipation of a practical procedure for approximation of environmental contours that will follow in Section 6.

In Section 2 we defined the environmental contours in terms of half-spaces that were parametrized by their perpendicular distance to the origin. However, a half-space may equivalently be parametrized in terms of perpendicular distance to any other point $\mathbf{o} \in \mathbb{R}^n$, i.e.

$$\Pi^-_{p_e}(\mathbf{u}) = \{\mathbf{x} : \mathbf{u} \cdot (\mathbf{x} - \mathbf{o}) \leq C^{\mathbf{o}}_{p_e}(\mathbf{u})\}, \tag{7}$$

with

$$C^{\mathbf{o}}_{p_e}(\mathbf{u}) = \inf\{c : P(\mathbf{u} \cdot (\mathbf{X} - \mathbf{o}) > c) \leq p_e\}. \tag{8}$$

By comparing (1) and (8) it is evident that

$$C^{\mathbf{o}}_{p_e}(\mathbf{u}) = C_{p_e}(\mathbf{u}) - \mathbf{u} \cdot \mathbf{o}, \tag{9}$$

and that the two definitions of $\Pi^-_{p_e}(\mathbf{u})$ given in (3) and (7) are equivalent.

Using this alternative parametrization for $\Pi^-_{p_e}(\mathbf{u})$, we define the set $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$ as

$$\mathcal{S}^{\mathbf{o}}_{p_e}(U) = \{\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} = \mathbf{o} + 2C^{\mathbf{o}}_{p_e}(\mathbf{u})\mathbf{u}\}_{\mathbf{u}\in U}, \tag{10}$$

where U is a subset of the unit vectors in $\mathbb{R}^n$.

A point $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} \in \mathcal{S}^{\mathbf{o}}_{p_e}(U)$ represents the reflection of the point $\mathbf{o} \in \mathbb{R}^n$ with respect to the boundary of the half-space $\Pi^-_{p_e}(\mathbf{u})$ (i.e. with respect to $\Pi_{p_e}(\mathbf{u})$). Stated differently, the half-space $\Pi^-_{p_e}(\mathbf{u})$ contains all points that are closer to $\mathbf{o}$ than to $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$. Intuitively, if $\mathbf{o}$ is in the interior of $\mathcal{B}_{p_e}$, then all points in the convex set $\mathcal{B}_{p_e}$ should be closer to $\mathbf{o}$ than to any point in $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$. This means that $\mathcal{B}_{p_e}$ is the Voronoi cell of $\mathbf{o}$ with respect to the set of points $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$. The latter insight is stated formally as a lemma below.

Lemma 1 *Let $\mathcal{B}_{p_e}$ be defined as in* (1)*. Then*

$$\mathbf{o} \in \mathcal{B}_{p_e} \iff C^{\mathbf{o}}_{p_e}(\mathbf{u}) \geq 0 \ \forall \ \mathbf{u} \in \mathcal{U},$$
$$\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e} \iff C^{\mathbf{o}}_{p_e}(\mathbf{u}) > 0 \ \forall \ \mathbf{u} \in \mathcal{U}.$$

Furthermore, if $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$ we have for any subset $U \subseteq \mathcal{U}$ that

$$\mathit{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U)) = \bigcap_{\mathbf{u}\in U} \Pi^-_{p_e}(\mathbf{u}),$$

where $\mathit{Vor}(\cdot,\cdot)$ is the Voronoi cell as defined in (6)*.*

The proof is given in Appendix A. Using this result we arrive at the following proposition that motivates the algorithm presented in this paper.

Proposition 2 *Let $\mathcal{B}_{p_e}$ be defined as in* (1)*, and let U_1 and U_2 be sets of unit vectors in $\mathbb{R}^n$, such that $U_1 \subseteq U_2 \subseteq \mathcal{U}$. If $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$ then the following holds:*

$$\mathcal{B}_{p_e} = \mathit{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})) \subseteq \mathit{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U_2)) \subseteq \mathit{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U_1)).$$

This proposition follows directly from Lemma 1 (see Appendix B for details). The first interesting observation is that the environmental contour, $\partial\mathcal{B}_{p_e}$, can be represented as the boundary of the Voronoi cell $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U}))$. This insight immediately suggests a new algorithm for numerical approximation of environmental contours, by replacing the set of unit vectors $\mathcal{U}$ with a finite subset $U = \{\mathbf{u}_i \mid \mathbf{u}_i \in \mathcal{U}, i = 1, \ldots, k\}$, as illustrated in Figure 2. The proposition also states that any such approximation of a *proper* convex environmental contour will be conservative, in the sense that the resulting Voronoi cell is guaranteed to contain $\mathcal{B}_{p_e}$. Accordingly, any approximation will be a *valid* convex environmental contour. Moreover, including more unit vectors in the set U improves the approximation (or at least does not make it worse). Intuitively, the error in the approximation can be made arbitrarily small, although this naturally will depend on the sampling strategy used.

A natural procedure for approximating $\mathcal{B}_{p_e}$ could therefore be as follows:

Step 1 Select a set of unit vectors $U = \{\mathbf{u}_j\}_{j=1}^M$.
Step 2 Compute $C_{p_e}(\mathbf{u}_1), \ldots, C_{p_e}(\mathbf{u}_M)$.
Step 3 Compute $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$ for some $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$.
Step 4 Compute the Voronoi cell of $\mathbf{o}$ with respect to $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$.

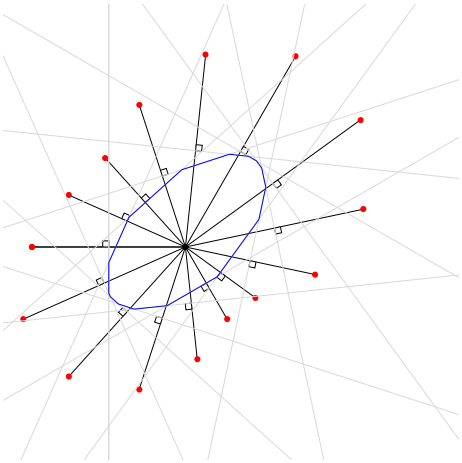


Fig. 2: Construction of environmental contour using the Voronoi method. The black point is the chosen origin $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$. The red points correspond to the finite set $\mathcal{S}^{\mathbf{o}}_{p_e}$. The boundaries of the half planes $\Pi^{-}_{p_e}(\mathbf{u})$ half way between $\mathbf{o}$ and the respective points $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} \in \mathcal{S}^{\mathbf{o}}_{p_e}$ are drawn as light grey lines, and their perpendicularity on the black lines from $\mathbf{o}$ to $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$ is indicated with small squares. The boundary of the Voronoi cell of $\mathbf{o}$ with respect to $\mathcal{S}^{\mathbf{o}}_{p_e}$ is drawn in blue, and it can be seen that the grey lines are tangential on the Voronoi cell.

Under the assumption that a proper convex environmental contour exists (for the given random variable $\mathbf{X}$ and target probability p_e), the set $\widehat{\mathcal{B}}_{p_e} = \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ is guaranteed to contain $\mathcal{B}_{p_e}$, and the difference can be made arbitrarily small by including sufficiently many unit vectors in U. For practical application, however, it is not reasonable to assume that the function $C_{p_e}(\mathbf{u})$ can be computed exactly, and we might not have *a priori* a point $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$. We will postpone these questions to Section 6. For now, we will assume that a point $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$ is given and that the function $C_{p_e}(\mathbf{u})$ can be evaluated without error, in order to study the final major assumption. Namely, that the random variable of interest $\mathbf{X}$ admits a proper convex environmental contour for the target probability p_e.

In practice, it might not be possible to determine *a priori* whether a proper convex environmental contour exists. To see how we might account for this issue, we first study what will happen if $\mathbf{X}$ *does not* admit a proper convex environmental contour. In Figure 3 we reproduce the example given in [28], illustrating the scenario where a supporting half-space can have exceedance probability larger than p_e. That is, one of the hyperplanes $\Pi^{-}_{p_e}(\mathbf{u})$ in (1) does not intersect $\mathcal{B}_{p_e}$. Hence, if a scenario such as the one in Figure 3 a) occur, this means that a proper environmental contour cannot exist (for the selected target probability p_e). As we illustrate in the figure, there is an interesting connection with the dual representation of the Voronoi cell, the Delaunay triangulation, that can be exploited when studying this problem. We recall that every edge on a Voronoi cell corresponds to the circumcenter of a Delaunay triangle (in general a Delaunay simplex for higher dimensions), and we say that a Delaunay triangulation connects two points $\mathbf{a}, \mathbf{b} \in \mathcal{X}$ if both $\mathbf{a}$ and $\mathbf{b}$ are part of the same triangle (simplex) in the triangulation. With this terminology, we may state the observation made in Figure 3 formally as follows.

Proposition 3 *Assume $\partial\mathcal{B}_{p_e}$ is a proper convex environmental contour with $\mathcal{B}_{p_e}$ defined as in* (1). *Let $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$ be defined as in* (10) *for some finite set $U \subset \mathcal{U}$, and $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$.*

Then, for all $\mathbf{s} \in \mathcal{S}^{\mathbf{o}}_{p_e}(U)$, there exists a Delaunay triangulation of the point set $\{\mathbf{o}\} \cup S^{\mathbf{o}}_{p_e}(U)$ that connects $\mathbf{s}$ and $\mathbf{o}$.

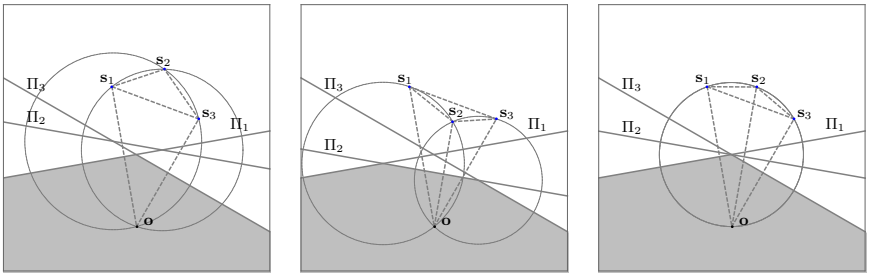


Fig. 3: Three points from $\mathcal{S}^{\mathbf{o}}_{p_e}(U)$ with corresponding hyperplanes, $\mathbf{s}_i = \mathbf{s}^{\mathbf{o},\mathbf{u}_i}_{p_e}$ and $\mathit{\Pi}_i = \mathit{\Pi}_{p_e}(\mathbf{u}_i)$ for three unit vectors $U = \{\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3\}$. The Voronoi cell $\text{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ corresponds to the shaded area in each figure, and the dual Delaunay triangulation is indicated with dashed lines. a) $\mathit{\Pi}_2$ is not a supporting hyperplane of $\text{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ since $\mathbf{s}_2$ is not connected to $\mathbf{o}$ by any Delaunay edge. b) All planes $\mathit{\Pi}_i$ intersect $\text{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ as $\mathbf{s}_i$ is connected to $\mathbf{o}$ by a Delaunay edge for all i. c) The Delaunay triangulation is not unique, and $\mathit{\Pi}_2$ only intersects a vertex of $\text{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$.

A proof of Proposition 3 is given in Appendix C, where we refer to [46] for results regarding the Voronoi-Delaunay duality. We may also make use of the fact that a Delaunay triangulation of a point set is unique if the points are in *general position*. In the general n-dimensional case, a set $\mathbf{P}$ of points is in *general position* if the affine hull of $\mathbf{P}$ is n-dimensional, and there is no subset of $n+2$ points in $\mathbf{P}$ that lie on the boundary of a ball whose interior does not intersect $\mathbf{P}$. Figure 3 c) shows a scenario where this condition is violated. Here, the affine hull of the set $\mathbf{P} = \{\mathbf{o}, \mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3\}$ is clearly 2-dimensional, but the four points in $\mathbf{P}$ all lie on a circle (whose interior does not contain any points in $\mathbf{P}$). Hence, the Delaunay triangulation is not unique. There are in fact two possible Delaunay triangulations as illustrated in Figure 3 c), $\{\{\mathbf{o}, \mathbf{s}_1, \mathbf{s}_3\}, \{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3\}\}$ and $\{\{\mathbf{o}, \mathbf{s}_1, \mathbf{s}_2\}, \{\mathbf{o}, \mathbf{s}_2, \mathbf{s}_3\}\}$. Using this condition for uniqueness together with Proposition 3, we immediately achieve the following convenient result.

Corollary 1 *Under the assumptions of Proposition 3, if also the points in* $\{\mathbf{o}\} \cup S^{\mathbf{o}}_{p_e}(U)$ *are in general position, then the Delaunay triangulation is unique and connects all points* $\mathbf{s} \in \mathcal{S}^{\mathbf{o}}_{p_e}(U)$ *with* $\mathbf{o}$.

Corollary 1 is useful as it gives a criterion for checking whether a proper convex environmental contour exists, and for identification of directions (for which unit vector $\mathbf{u}$) there might be problems. The general idea is also illustrated in Figure 4, where we can conclude that no proper convex environmental contour exists, for the given distribution of $\mathbf{X}$ and target probability p_e, as the grey shaded triangle contains a point $\mathbf{s} \in S^{\mathbf{o}}_{p_e}(U)$ which is not connected with $\mathbf{o}$.

5 Voronoi contours in the continuous limit

From the illustrations in Figure 3 and Figure 4, we could also imagine what happens as more points are added, moving to the limit as $S^{\mathbf{o}}_{p_e}(U) \to S^{\mathbf{o}}_{p_e}(\mathcal{U})$. Consider the Delaunay triangle $\{\mathbf{o}, \mathbf{s}_2, \mathbf{s}_3\}$ in Figure 3 b). This triangle has the property that its circumcircle contains no other points from $S^{\mathbf{o}}_{p_e}(U)$ in its interior. As the points $\mathbf{s}_2$ and $\mathbf{s}_3$ move arbitrarily close together, the circumcircle of this "triangle" is the circle that contain $\mathbf{o}$ and is tangential to $\mathbf{s}_2 \approx \mathbf{s}_3$. Moreover, the center of this circle is a point on $\partial\mathcal{B}_{p_e}$. From this intuition we arrive at the geometric property of proper convex environmental contours, which is illustrated in Figure 5. We state this formally in Proposition 4, with a proof given in Appendix D.

Proposition 4 *Assume* $\partial\mathcal{B}_{p_e}$ *is a proper convex environmental contour with* $\mathcal{B}_{p_e}$ *defined as in* (1). *Let* $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ *be as in* (10) *and define, for any* $\mathbf{b} \in \partial\mathcal{B}_{p_e}$ *and* $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$, *the* n*-dimensional ball* $\mathcal{W}^{\mathbf{o}}(\mathbf{b}) = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x} - \mathbf{b}\| \leq \|\mathbf{b} - \mathbf{o}\|\}$.

Then for any $\mathbf{u} \in \mathcal{U}$, *there exists some* $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ *where* $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U}) \cap \mathcal{W}^{\mathbf{o}}(\mathbf{b}) \subseteq \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$, *and* $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} \in \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$.

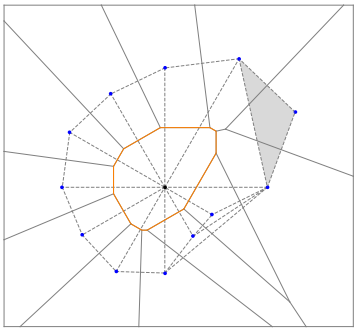


Fig. 4: Illustration of the idea behind Proposition 3 and Corollary 1 in 2D. The dashed lines shows the Delaunay triangulation of the points $\{\mathbf{o}\} \cup S^{\mathbf{o}}_{p_e}(U)$, which are in general position. The grey triangle contains a point $\mathbf{s} \in S^{\mathbf{o}}_{p_e}(U)$ that is not connected to $\mathbf{o}$. Hence, no proper convex environmental contour exists for the selected probability p_e and the random variable $\mathbf{X}$ used to generate $S^{\mathbf{o}}_{p_e}(U)$.

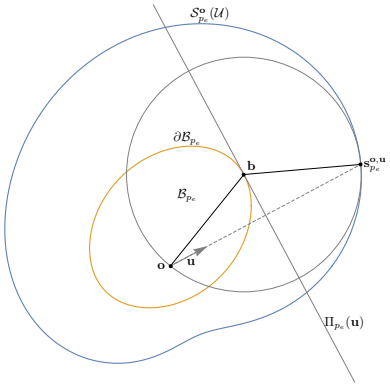


Fig. 5: Geometric illustration of Proposition 4 in 2D. For any $\mathbf{u} \in \mathcal{U}$ there exists some $\mathbf{b} \in \Pi_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$, such that the circle centered at $\mathbf{b}$ that also contains $\mathbf{o}$ is tangent to $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$, and contains no points from $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ in its interior.

A consequence of the geometric property stated in Proposition 4 is that, given a parametrization of unit vectors in $\mathbb{R}^n$, we will be able to derive a parametric characterization of $\partial\mathcal{B}_{p_e}$. The key insight from Figure 5 is that, given certain regularity assumptions, the vectors tangential to the set $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ and the ball $\mathcal{W}^{\mathbf{o}}(\mathbf{b})$ coincide at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$. This will eventually let us derive a parametric representation of the set $\partial\mathcal{B}_{p_e}$ as a $(n-1)$-dimensional manifold. So now, motivated by the properties derived in the discrete scenario using tools from computational geometry, i.e. the Voronoi and Delaunay tessellations, we will move to the continuous limit and study environmental contours in the context of differential geometry.

We will start by assuming that the set $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$, viewed as a $(n-1)$-dimensional manifold embedded in $\mathbb{R}^n$, is differentiable. We recall that a m-dimensional manifold $\mathcal{S}$ in $\mathbb{R}^n$, for $m \leq n$, can be represented by a set of *charts* $\sigma_i : V_i \to \mathcal{S}$, where V_i are open non-empty subsets of $\mathbb{R}^m$. Any set of charts $\{\sigma_i, V_i\}_i$ that cover $\mathcal{S}$, i.e. $\mathcal{S} = \cup_i \sigma_i(V_i)$, is called an *atlas* of $\mathcal{S}$. We will in particular consider a regular parametrization of the unit $(n-1)$-sphere $\mathcal{U}$, by which we mean a set of charts $\{\sigma_i, V_i\}_i$ covering $\mathcal{U}$ where each σ_i is smooth and where the Jacobi matrix of σ_i has rank $n-1$ at any point in V_i. With the canonical alternative of spherical coordinates in mind, we will let $\{\mathbf{u}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_i$ denote an atlas of $\mathcal{U}$ with these properties. With some abuse of terminology, we will also refer to $\{\mathbf{u}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_i$ as a *regular parametrization* of $\mathcal{U}$. Given such a regular parametrization of $\mathcal{U}$, we will continue to construct corresponding parametrizations of $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ and eventually $\partial\mathcal{B}_{p_e}$. But first we will need a preliminary result given in Lemma 2 below.

Lemma 2 *Assume $\partial\mathcal{B}_{p_e}$ is a proper convex environmental contour with $\mathcal{B}_{p_e}$ defined as in* (1)*, let $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$ and assume $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ is a differentiable manifold.*

If the pair $(\mathbf{a}, \mathbf{u})$, for some $\mathbf{a} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathcal{U}$, satisfies the following

1. $\|\mathbf{a} - \mathbf{o}\| = \left\|\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{a}\right\|$*, and*
2. $(\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{a})$ *is orthogonal to $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$,*

then $\{\mathbf{a}\} = \Pi_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$.

In the proof of Lemma 2, given in Appendix E, we also show that for any $\mathbf{u} \in \mathcal{U}$, $\Pi_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ is a singleton set, as $\Pi_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ is nonempty when $\partial\mathcal{B}_{p_e}$ is a proper convex environmental contour and the pair $(\mathbf{b}, \mathbf{u})$ satisfies the conditions in Lemma 2 for any $\mathbf{b} \in \Pi_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$. This means that the set $\mathcal{B}_{p_e}$ has no "flat parts", and that $\mathcal{B}_{p_e}$ is in fact strictly convex. But besides this, the conditions in Lemma 2 will also serve as a more practical criterion to verify that a given mapping (soon to be given explicitly) gives a representation of the environmental contour $\partial\mathcal{B}_{p_e}$. This result is summarised in Proposition 5 below, with a proof given in Appendix F.

Proposition 5 *Let $F : \mathcal{U} \to \mathbb{R}^n$ be a mapping such that the assumptions and conditions of Lemma 2 hold for any pair $(F(\mathbf{u}), \mathbf{u})$. Then $F(\mathcal{U}) = \partial\mathcal{B}_{p_e}$.*

Now, the next step is to introduce a specific parametrization of $\partial\mathcal{B}_{p_e}$ that we will use Proposition 5 to verify. We will achieve this by mapping a parametrization of the unit $(n-1)$-sphere $\mathcal{U}$ to a parametrization of $\partial\mathcal{B}_{p_e}$. This idea has been explored in [28,23] for the 2-dimensional case using the parametrization $\mathbf{u}(\theta) = (\cos(\theta), \sin(\theta))$, where also the existence of a proper convex environmental contour is determined from properties related to the parametrized percentile function $C_{p_e}(\theta) = C_{p_e}(\mathbf{u}(\theta))$. In the following we will extend this to the n-dimensional case.

Let $\{\mathbf{u}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_i$ be the regular parametrization of $\mathcal{U}$ introduced previously. Suppressing the index i, for any chart $\mathbf{u}(\boldsymbol{\theta}) : \Theta \to \mathcal{U}$ we define the functions $C^{\mathbf{o}}_{p_e}(\boldsymbol{\theta})$ and $\mathbf{s}^{\mathbf{o}}_{p_e}(\boldsymbol{\theta})$ accordingly,

$$\begin{aligned} C^{\mathbf{o}}_{p_e}(\boldsymbol{\theta}) &= C^{\mathbf{o}}_{p_e}(\mathbf{u}(\boldsymbol{\theta})) : \Theta \to \mathbb{R}, \\ \mathbf{s}^{\mathbf{o}}_{p_e}(\boldsymbol{\theta}) &= \mathbf{o} + 2C^{\mathbf{o}}_{p_e}(\boldsymbol{\theta})\mathbf{u}(\boldsymbol{\theta}) : \Theta \to \mathbb{R}^n, \end{aligned} \tag{11}$$

where we will assume that both $\mathbf{u}(\boldsymbol{\theta})$ and $C^{\mathbf{o}}_{p_e}(\boldsymbol{\theta})$ are continuously differentiable as functions of $\boldsymbol{\theta}$, and let $\nabla_{\boldsymbol{\theta}}$ denote the Jacobian. That is, for functions $\mathbf{f} : \Theta \to \mathbb{R}^m$, $\nabla_{\boldsymbol{\theta}}\mathbf{f}$ is the $m \times (n-1)$ matrix with entries $[\nabla_{\boldsymbol{\theta}}\mathbf{f}]_{i,j} = \partial\mathbf{f}_i/\partial\boldsymbol{\theta}_j$. The assumption that $\mathbf{u}(\boldsymbol{\theta})$ is a *regular* parametrization means that we also assume that $\nabla_{\boldsymbol{\theta}}\mathbf{u}(\boldsymbol{\theta})$ has rank $n-1$ for any $\boldsymbol{\theta} \in \Theta$.

Theorem 1 (Representation of proper convex environmental contours) *Assume the n-dimensional random variable $\mathbf{X}$ admits a proper convex environmental contour $\partial\mathcal{B}_{p_e}$ with respect to a target probability $p_e \in (0, 0.5)$, and assume that the p_e-level percentile function $C_{p_e}(\boldsymbol{u})$ is k-times continuously differentiable on the unit $(n-1)$-sphere for $k \geq 1$.*

Then $\mathcal{B}_{p_e}$ is strictly convex, and $\partial\mathcal{B}_{p_e}$ is a $(k-1)$-times differentiable manifold. Furthermore, if $\{\mathbf{u}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_{i=1}^m$ is a regular parametrization of the unit $(n-1)$-sphere, then an atlas of $\partial\mathcal{B}_{p_e}$ is obtained by $\{\mathbf{b}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_{i=1}^m$, where $\mathbf{b}_i(\boldsymbol{\theta})$ is obtained from $\mathbf{u}_i(\boldsymbol{\theta})$ using the following relation:

$$\mathbf{b}(\boldsymbol{\theta}) = C_{p_e}(\boldsymbol{\theta})\mathbf{u}(\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}}\mathbf{u}(\boldsymbol{\theta})g^{-1}(\boldsymbol{\theta})(\nabla_{\boldsymbol{\theta}}C_{p_e}(\boldsymbol{\theta}))^T, \tag{12}$$

and where $g(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}}\mathbf{u}(\boldsymbol{\theta})^T\nabla_{\boldsymbol{\theta}}\mathbf{u}(\boldsymbol{\theta})$ is the metric tensor of the $(n-1)$-sphere induced by the parametrization $\mathbf{u}(\boldsymbol{\theta})$.

The proof of Theorem 1 is given in Appendix G. Note that Theorem 1 gives an analytic expression for the environmental contour (i.e. $\mathbf{b}_i(\boldsymbol{\theta})$) in terms of the p_e-level percentile function $C_{p_e}(\boldsymbol{\theta})$. Thus, given a specific parametrization and a differentiable approximation of $C_{p_e}(\boldsymbol{\theta})$ it is possible to compute $\mathbf{b}(\boldsymbol{\theta})$ directly, as an alternative to explicitly constructing a Voronoi cell as described in section 4. One common parametrization in the n-dimensional case is given by $\mathbf{u}(\boldsymbol{\theta}) = (u_0, u_1, \ldots, u_{n-1})$ with $u_i = \cos\theta_i \prod_{j=0}^{i-1} \sin\theta_j$ for $i = 0, 1, \ldots, n-2$ and $u_{n-1} = \prod_{j=0}^{n-2} \sin\theta_j$, where $\theta_i \in [0, \pi)$ for $i = 1, 2, \ldots, n-2$ and $\theta_{n-2} \in [0, 2\pi)$. The corresponding induced metric tensor has entries $g_{0,0} = 1$, $g_{i,i} = \prod_{j=0}^{i-1} \sin\theta_j{}^2$ for $i = 0, 1, \ldots, n-2$ and $g_{i,j} = 0$ if $i \neq j$.

It would be desirable to have a criterion for $C_{p_e}(\boldsymbol{\theta})$ that guarantees that $\mathbf{b}_i(\boldsymbol{\theta})$ represent a proper environmental contour. To obtain such a criterion, we will need a couple of intermediate results given in the following to Lemmas.

Lemma 3 *The random variable* $\mathbf{X}$ *admits a proper convex environmental contour with respect to* $p_e \in (0, 0.5)$ *if and only if the following holds:*

For any $\mathbf{u}' \in \mathcal{U}$*, there exists some* $\mathbf{o} \in \Pi_{p_e}(\mathbf{u}')$ *such that* $C^{\mathbf{o}}_{p_e}(\mathbf{u}) \geq 0$ *for all* $\mathbf{u} \in \mathcal{U}$*.*

Lemma 4 *Assume the percentile function* $C_{p_e}(\boldsymbol{\theta})$ *is twice differentiable and that* $\mathbf{u}(\boldsymbol{\theta}) : \Theta \to \mathcal{U}$ *is regular (*$\nabla \mathbf{u}(\boldsymbol{\theta})$ *exists and has full rank for all* $\boldsymbol{\theta}$*). Let* $\mathbf{b}(\boldsymbol{\theta})$ *be defined as in* (12)*. Then*

$$\mathbf{u}(\boldsymbol{\theta})^T \mathbf{b}(\boldsymbol{\theta}) = C_{p_e}(\boldsymbol{\theta}) \text{ and } \mathbf{u}(\boldsymbol{\theta})^T \nabla \mathbf{b}(\boldsymbol{\theta}) = \mathbf{0}$$

for all $\boldsymbol{\theta} \in \Theta$*. This means that* $\Pi_{p_e}(\boldsymbol{\theta})$ *is tangential to* $\mathbf{b}(\Theta)$ *at the point* $\mathbf{b}(\boldsymbol{\theta})$*.*

Lemma 3 comes as a consequence of Lemma 1, and the proof is given in Appendix H. In Appendix I we present the proof of Lemma 4, which states that for any $\boldsymbol{\theta}$, the hyperplane $\Pi_{p_e}(\boldsymbol{\theta})$ is tangential to $\mathbf{b}(\Theta)$ at the point $\mathbf{b}(\boldsymbol{\theta})$.

Armed with these results we can prove the following criteria for existence.

Theorem 2 (Existence of proper convex environmental contours) *Let* $\mathbf{X}$ *be any* n*-dimensional random variable where the percentile function* $C_{p_e}(\cdot)$ *is differentiable on the unit* $(n-1)$*-sphere. Let* $\{\mathbf{u}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_{i=1}^m$ *be a regular parametrization of the unit* $(n-1)$*-sphere, and define for any* $\mathbf{u}(\boldsymbol{\theta}) = \mathbf{u}_i(\boldsymbol{\theta})$ *the function*

$$\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}') = C^{\mathbf{b}(\boldsymbol{\theta}')}_{p_e}(\boldsymbol{\theta}) = C_{p_e}(\boldsymbol{\theta}) - \mathbf{u}(\boldsymbol{\theta}) \cdot \mathbf{b}(\boldsymbol{\theta}'), \tag{13}$$

where $C_{p_e}(\boldsymbol{\theta}) = C_{p_e}(\mathbf{u}(\boldsymbol{\theta}))$ *and* $\mathbf{b}(\boldsymbol{\theta}')$ *is given by* (12) *with* $\boldsymbol{\theta} = \boldsymbol{\theta}'$*.*

Then the following are equivalent:

1. $\mathbf{X}$ *admits a proper convex environmental contour.*
2. *The hypersurface given by the parametrization* $b(\boldsymbol{\theta})$ *in* (12) *is the boundary of a closed convex set.*
3. $\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}') \geq 0$ *for all* $\mathbf{u}(\boldsymbol{\theta}) = \mathbf{u}_i(\boldsymbol{\theta}), \boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta_i$*, and* $i = 1, \ldots, m$*.*
4. $\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}')$ *attains its global minimum at* $\boldsymbol{\theta} = \boldsymbol{\theta}'$ *for all* $\mathbf{u}(\boldsymbol{\theta}) = \mathbf{u}_i(\boldsymbol{\theta}), i = 1, \ldots, m$*.*

The proof of Theorem 2 is provided in Appendix J. In the 2-dimensional case with polar coordinates, one can also show that existence is equivalent to the criterion that $C_{p_e}(\theta) + C''_{p_e}(\theta) > 0$ for all $\theta \in [0, 2\pi)$ (see Theorem 3.13 in [23]). As a consequence of Theorem 2, we can obtain the following similar result stated in Corollary 2 below.

Corollary 2 *Assume the* n*-dimensional random variable* $\mathbf{X}$ *admits a proper convex environmental contour, and that* $C_{p_e}(\boldsymbol{\theta})$ *is two times differentiable. Then* $Hess(C_{p_e}(\boldsymbol{\theta})) + g(\boldsymbol{\theta}) C_{p_e}(\boldsymbol{\theta})$ *is positive semi-definite for all* $\boldsymbol{\theta} \in \Theta$*, where* $Hess(\cdot)$ *is the Hessian operator on the* $(n-1)$*-sphere and* $g(\boldsymbol{\theta})$ *is the* $(n-1)$*-sphere metric tensor.*

The proof of Corollary 2 is given in Appendix K. Note that the metric tensor on the unit circle is simply $g = 1$, so the 2-dimensional version of Corollary 2 states that $C_{p_e}(\theta) + C''_{p_e}(\theta) \geq 0$. As a stronger version of the statement holds in the 2-dimensional case, we might conjecture that the criterion in Corollary 2 with strict positive definiteness could hold as a necessary condition, but we have currently not explored this further in any detail.

6 Practical application of the Voronoi method for environmental contour approximation

In Section 4 we outlined a potential procedure for approximating environmental contours using the Voronoi-representation. Based on this idea, we present the steps involved in Algorithm 3 below, followed up by a discussion on how each step may be implemented in practice.

Algorithm 3 *Approximating* $\mathcal{B}_{p_e}$ *using the Voronoi method*

1. *Select a set of unit vectors* $U = \{\mathbf{u}_j\}_{j=1}^M$*.*
2. *Estimate* $\widehat{C}_{p_e}(\mathbf{u}_j) \approx C_{p_e}(\mathbf{u}_j)$ *for each* $j = 1, \ldots, M$*.*
3. *Compute* $\widehat{\mathcal{S}}^{\mathbf{o}}_{p_e}(U)$*, using* $\widehat{C}^{\mathbf{o}}_{p_e}(\mathbf{u}_j)$ *in* (10)*, for some* $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial \mathcal{B}_{p_e}$*.*

4. *Compute the approximation* $\widehat{\mathcal{B}}_{p_e} = Vor(\mathbf{o}, \widehat{\mathcal{S}}^{\mathbf{o}}_{p_e}(U))$.
5. *Check that each point in* $\widehat{\mathcal{S}}^{\mathbf{o}}_{p_e}(U)$ *is connected with* $\mathbf{o}$ *in the Delaunay triangulation of the point set* $\{\mathbf{o}\} \cup S^{\mathbf{o}}_{p_e}(U)$.

Step 1: The algorithm will produce finer approximations as more unit vectors are included. However, the main computational burden is usually related to the estimation of $C_{p_e}(\mathbf{u}_j)$ for each unit vector, so the number of unit vectors is often decided by the desired run-time of the entire algorithm. In applications such as design of marine structures, there might be knowledge related to which directions that are the most informative, and the set U might be chosen on this basis. Alternatively, a uniform selection may be applied. One way to generate uniform random samples from the unit $(n-1)$-sphere is to let $\mathbf{u}_j = \mathbf{v}_j / \|\mathbf{v}_j\|$ where $\mathbf{v}_j = (v_{1,j}, \ldots, v_{n,j})$ and all $v_{1,j}$ are i.i.d. Gaussian [37].

Step 2: In practice, we might not be able to compute $C_{p_e}(\mathbf{u}_j)$ exactly. However, this can be estimated based on a finite number of Monte Carlo samples from the joint distribution, in the same way as outlined in [26,28]. The estimation error will depend on the sample size and may in principle be reduced to an acceptable level by increasing the number of samples, or for example using the importance sampling scheme proposed in [27]. Moreover, if one were to apply conservative estimates, i.e. $\widehat{C}_{p_e}(\mathbf{u}_j) \geq C_{p_e}(\mathbf{u}_j)$, this would produce a conservative (larger) environmental contour approximation as well.

Step 3: In order to compute $\widehat{\mathcal{S}}^{\mathbf{o}}_{p_e}(U)$, we first need some point of reference $\mathbf{o}$ from the interior of $\mathcal{B}_{p_e}$. The criterion that $C^{\mathbf{o}}_{p_e}(\mathbf{u}) > 0$ for any $\mathbf{u} \in \mathcal{U}$ (see Lemma 1) can be used to identify if the selected origin $\mathbf{o}$ is not in the interior of $\mathcal{B}_{p_e}$. We can then also observe that, in the case where we want to replace the origin $\mathbf{o}$ with some new point $\mathbf{o}^*$, the new set $\mathcal{S}^{\mathbf{o}^*}_{p_e}$ can be computed using that $C^{\mathbf{o}^*}_{p_e}(\mathbf{u}) = C^{\mathbf{o}}_{p_e}(\mathbf{u}) + \mathbf{u} \cdot (\mathbf{o} - \mathbf{o}^*)$, and hence

$$\mathbf{s}^{\mathbf{o}^*,\mathbf{u}}_{p_e} = \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} + 2\mathbf{u} \cdot (\mathbf{o} - \mathbf{o}^*)\mathbf{u} - (\mathbf{o} - \mathbf{o}^*). \tag{14}$$

This means that the estimates $\widehat{C}_{p_e}(\mathbf{u}_j)$ can be reused, as going from $\widehat{\mathcal{S}}^{\mathbf{o}}_{p_e}(U)$ to $\widehat{\mathcal{S}}^{\mathbf{o}^*}_{p_e}(U)$ is a simple linear transformation. We may also note the geometric interpretation, by observing that the added term $2\mathbf{u} \cdot (\mathbf{o} - \mathbf{o}^*)\mathbf{u} - (\mathbf{o} - \mathbf{o}^*)$ is the reflection of the point $(\mathbf{o} - \mathbf{o}^*)$ with respect to the unit vector $\mathbf{u}$. As both checking whether $C^{\mathbf{o}}_{p_e}(\mathbf{u}) > 0$ and moving the origin $C^{\mathbf{o}}_{p_e}(\mathbf{u}) \to C^{\mathbf{o}^*}_{p_e}(\mathbf{u})$ are cheap computationally, one could derive an iterative procedure to determine $\mathbf{o}$. Alternatively, finding the point $\mathbf{o}$ with maximal distance to all hyperplanes under the restriction that $C^{\mathbf{o}}_{p_e}(\mathbf{u}_j) > 0$, which is equivalent to $C_{p_e}(\mathbf{u}_j) > \mathbf{u}_j \cdot \mathbf{o}$, for each $j = 1, \ldots, M$ can be solved by linear programming. In our implementation, the geometric median of a set of samples from the joint distribution of $\mathbf{X}$ (the ones used to estimate $C_{p_e}(\mathbf{u}_j)$ in Step 2) was selected as the origin $\mathbf{o}$. This choice of $\mathbf{o}$ will with high probability lie inside $\mathcal{B}_{p_e}$ for any $p_e > 0.5$, and in our experiments we did not find the need to iterate further beyond this initial guess.

Step 4: Some of the motivation for this paper comes from the fact that the Voronoi tessellation is a well studied object. As a result, a wide range of software and programming languages come with efficient procedures for computing Voronoi cells, including Python/Scipy, R, Wolfram Language/Mathematica, Matlab and Octave. Moreover, Voronoi algorithms work in arbitrary dimensions, which is what makes the proposed algorithm agnostic to the dimensionality of $\mathbf{X}$.

Step 5: This check comes as a consequence of Proposition 3 and Corollary 1. There are two scenarios that may cause this check to fail. 1) When the selected probability distribution does not admit a proper convex environmental contour with respect to the chosen target probability, and 2) when the percentile function $C_{p_e}(\mathbf{u})$ is estimated with error. In the case where the check fails due to noise in the estimates $\widehat{C}_{p_e}(\mathbf{u}_j)$, we can make refinements based on the relevant unit vectors. For instance, if it is found that the point $\hat{\mathbf{s}}_k \in \widehat{\mathcal{S}}^{\mathbf{o}}_{p_e}(U)$ corresponding to unit vector $\mathbf{u}_k$ is not connected with $\mathbf{o}$, the estimates $\widehat{C}_{p_e}(\mathbf{u}_j)$ can be refined for relevant indices j. The relevant indices here, besides $j = k$, are the ones corresponding to points $\hat{\mathbf{s}}_j$ affecting the Delaunay triangulation in the vicinity of $\hat{\mathbf{s}}_k$, which are the points connected with $\hat{\mathbf{s}}_k$ and the neighbouring Delaunay simplices. With reference to the previous step, we also note that the task of obtaining the Delaunay triangulation usually "comes for free", in the sense that available algorithms used to obtain the Voronoi tessellation do this by computing the Delaunay triangulation and taking the dual.

The goal of this numerical procedure presented in Algorithm 3 is to provide a good approximation in the case where a proper convex environmental contour exists. In the case where a proper

convex environmental contour *does not* exist, one might still be interested in finding a *valid* convex environmental contour that is "as small as possible". That is, a convex set where the exceedance probability of each supporting half-space is less than or equal to p_e (where it cannot be equal to p_e for all supporting half-spaces as no *proper* convex environmental contour exists). We will end this section with a modified version of the algorithm to accommodate this scenario.

The contour $\partial\mathcal{B}_{p_e}$ corresponding to the boundary of a Voronoi cell $Vor(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}) = \bigcap_{\mathbf{u}\in U} \Pi^-_{p_e}(\mathbf{u})$ is only a valid and proper environmental contour if $\partial\mathcal{B}_{p_e} \cap \Pi^-_{p_e}(\mathbf{u}) \neq \emptyset\ \forall \mathbf{u} \in U$. Otherwise, it is invalid. We may however use an *invalid* Voronoi contour to create a *valid improper* contour by the following algorithm:

Algorithm 4 *Let V be a Voronoi contour computed by Algorithm 3 based on a set of unit vectors U.*

1. *Initialise $Z = V$.*
2. *For each direction $\mathbf{u} \in U$:*
   (a) *Find the point $\mathbf{v}' \in V$ that is furthest out in direction $\mathbf{u}$, i.e. $\mathbf{v}' = \underset{\mathbf{v}\in V}{\operatorname{argmax}}\{\mathbf{v}\cdot\mathbf{u}\}$.*
   (b) *Compute the projection of $\mathbf{v}'$ onto the plane $\Pi_{p_e}(\mathbf{u})$, i.e. $\mathbf{z} = \mathbf{v}' + (C_{p_e}(\mathbf{u}) - \mathbf{v}'\cdot\mathbf{u})\mathbf{u}$.*
   (c) *Update $Z \to Z \cup \{\mathbf{z}\}$.*
3. *Compute the convex hull of Z. This is the corrected Voronoi contour.*

The algorithm above guarantees a valid environmental contour with respect to U, because it intersect all the hyperplanes $\Pi^-_{p_e}(\mathbf{u})\ \forall \mathbf{u} \in U$ by construction. The projection algorithm is illustrated in figure 6.

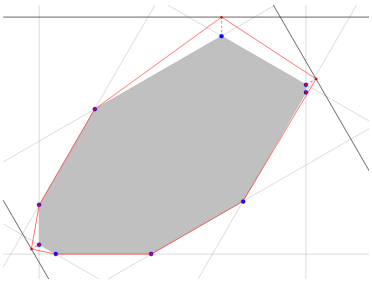


Fig. 6: Illustration of algorithm 4 to construct a valid environmental contour (red).

Figure 7 shows two examples using the above algorithms and also the direct method presented in [26]. First, a scenario where a proper convex environmental contour exists, and then a scenario where a proper environmental contour does not exist. The top row corresponds to a centered bivariate normal distribution with covariance $0.16 \cdot [1\ 0.5; 0.5\ 1]$, and the bottom row represents a Gaussian mixture; $\mathbf{X} = 0.8\mathbf{X}_1 + 0.1\mathbf{X}_2 + 0.1\mathbf{X}_3$ where $\mathbf{X}_1 \sim \mathcal{N}([0\ 0]^T, 0.16I)$, $\mathbf{X}_2 \sim \mathcal{N}([0.5\ 1]^T, 0.04I)$ and $\mathbf{X}_3 \sim \mathcal{N}([-0.5\ 1]^T, 0.04I)$. The contours are computed with $p_e = 0.15$.

7 Examples

7.1 2D example

To illustrate the Voronoi approach in two dimensions, we use the same example as [26]. The environmental variables of interest are the significant wave height, H_S, and the zero-upcrossing wave period, T_Z. Their joint distribution is modelled using a conditional modelling approach [7, 10], and can be expressed as

$$f_{H,T}(h,t) = f_H(h) f_T(t|h). \tag{15}$$

Here, $f_H(h)$ is a 3-parameter Weibull distribution for significant wave height, with scale parameter α, shape parameter β, and location parameter γ. $f_T(t|h)$ is a conditional log-normal distribution

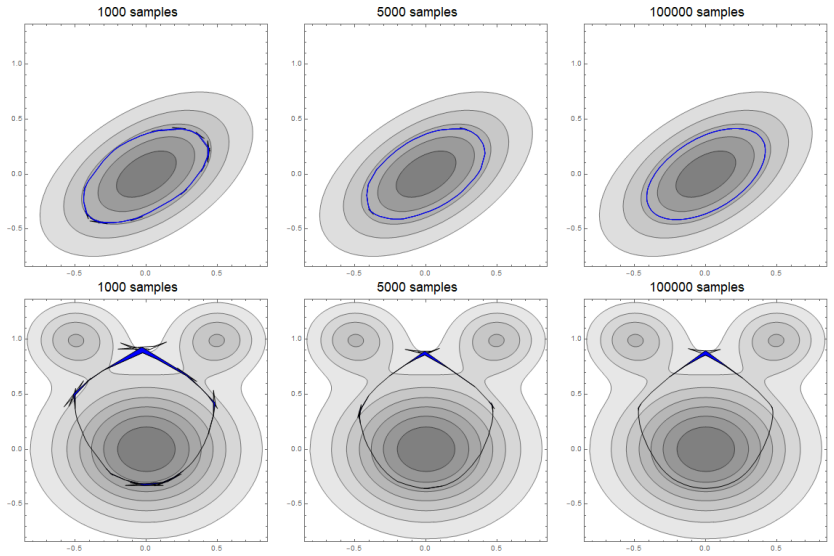


Fig. 7: Top: Contours for a multinormal distribution, constructed using the direct method of [26]. The loops disappear as the number of samples increased, indicating that the loops is a sampling issue. Bottom: Contours for a multimodal distribution constructed using the direct method of [26]. The top loops do not disappear as the number of samples increase, indicating that the loops is a feature of the underlying distribution (i.e. the distribution does not admit a proper convex contour for the selected target probability).

for wave period, where the model parameters are functions of significant wave height, as outlined in e.g. [14,58], i.e.

$$\begin{aligned} \mu_T(h) &= E\left(\ln T_Z | H_S = h\right) = a_1 + a_2 h^{a_3} \\ \sigma_T(h) &= sd\left(\ln T_Z | H_S = h\right) = b_1 + b_2 e^{b_3 h}. \end{aligned} \tag{16}$$

The parameter values used are listed in Table 1.

Table 1: Parameters assumed for the bivariate distribution of H_S and T_S.

3-p Weibull (H_S)	α	β	γ
	2.776	1.471	0.8888
Conditional log-normal (T_Z)	i = 1	i = 2	i = 3
a_i	0.1000	1.4890	0.1901
b_i	0.0400	0.1748	-0.2243

Figure 8 shows comparisons of results for different methods. The number of samples that the contours are based on is varied in the rows, but the samples are identical within each row. The number of unit vectors used to compute the contours is varied in the columns.

The direct sampling method of [26] is drawn in black. This method does not guarantee convex contours, but sometimes produce loops. Keeping the samples fixed, the loops tend to be larger as the number of unit vectors increase, which is undesirable. However, the loops tend to get smaller with increased number of samples. The convex hull of the black contours are drawn in red. Note that for the same number of samples, these red contours tend to get larger when the number of directions is increased, due to the larger loops.

Contours based on the Voronoi method are shown in blue. More precisely, blue regions are plotted, where the inner boundary correspond to the simple Voronoi method (i.e. Algorithm 3), and the outer boundary correspond to the corrected Voronoi method (i.e. Algorithm 4). Note that, unlike the other methods, the contours produced by the Voronoi methods do not diverge as the number of directions is increased. We also see that the shaded region is generally thin, indicating

that the simple Voronoi method is a good approximation to the 'true' environmental contour. The inset shows the error, i.e. the difference between the two Voronoi methods in the various directions. The directions with high error corresponds to directions where the direct method of [26] produces loops, i.e. the Voronoi method provides a warning for directions where more sampling may be needed.

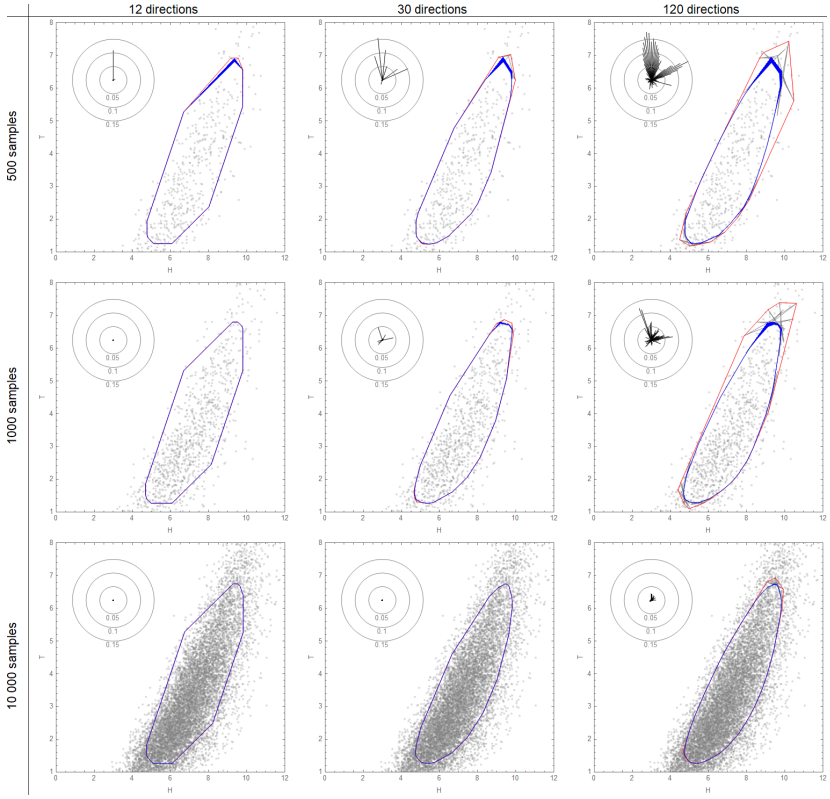


Fig. 8: Comparison of results, for $p_e = 0.05$. The samples that contours are computed from are shown in grey. The grey curves represent the direct sampling method of [26] (visible only in the third column). The red curves represent the convex hull of the grey curves. The blue regions represent the Voronoi methods; the inner boundary correspond to the simple Voronoi method, and the outer boundary correspond to the corrected Voronoi method. The insets show the error in different directions, i.e. the difference between the simple and corrected Voronoi methods.

7.2 3D example

To illustrate the Voronoi approach in three dimensions, we include an example from [55]. The environmental variables of interest are the significant wave height, H_S, the zero-upcrossing wave period, T_Z, and the 10-minute mean wind speed at a particular height, U_{10}. Their joint distribution is modelled using a conditional modelling approach [7,10], and can be expressed as

$$f_{H,T,U}(h,t,u) = f_H(h) f_T(t|h) f_U(u|h). \tag{17}$$

$f_H(h)$ is a 3-parameter Weibull distribution for significant wave height, with scale parameter α, shape parameter β, and location parameter γ.

$f_T(t|h)$ is a conditional log-normal distribution for wave period, where the model parameters are a function of significant wave height as outlined in e.g. [14,58], i.e.

$$\begin{aligned}\mu_T(h) &= E\left(\ln T_Z | H_S = h\right) = a_1 + a_2 h^{a_3} \\ \sigma_T(h) &= sd\left(\ln T_Z | H_S = h\right) = b_1 + b_2 e^{b_3 h}.\end{aligned} \tag{18}$$

The parameters a_i, b_i, $i = 1, 2, 3$ are estimated from data.

$f_U(u|h)$ is a conditional 2-parameter Weibull distribution with parameters modelled as functions of significant wave height as suggested by [14,9,8]. The scale parameter, λ_U, and shape parameter, κ_U, are modelled as

$$\begin{aligned}\lambda_U(h) &= c_1 + c_2 h^{c_3} \\ \kappa_U(h) &= d_1 + d_2 h^{d_3}.\end{aligned} \tag{19}$$

For the significant wave height and wave period, parameters corresponding to average world wide operations of ships according to appendix C of [14] are assumed, as summarised in Table 2. For the conditional distribution of wind speed, the average sectoral parameters reported in [9,8] will be assumed, as summarised in Table 2. It is noted that the parameter d_3 is omitted in [9], so this is simply set to 1 in this study.

Figure 9 shows the result of applying the Voronoi methods (simple and corrected) to the example described above. As can be seen, the simple method and corrected method are very similar, indicating that the simple Voronoi method is a good approximation for the 'true' environmental contour.

Table 2: Parameters assumed for the trivariate distribution of H_S, T_S and U_{10}.

3-p Weibull (H_S)		α	β	γ
average World wide trade		1.798	1.214	0.856
Conditional log-normal (T_Z)		i = 1	i = 2	i = 3
average World wide trade	a_i	-1.010	2.847	0.075
	b_i	0.161	0.146	-0.683
Conditional 2-p Weibull (U_{10})		i = 1	i = 2	i = 3
average directional sector	c_i	2.58	0.12	1.60
	d_i	4.6	2.05	1

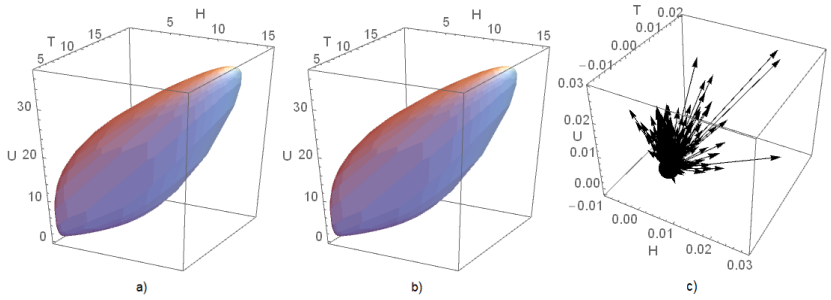


Fig. 9: a) Approximate (invalid) environmental contour for 3D example, computed using the simple Voronoi method (i.e. Algorithm 3). b) Valid (improper) environmental contour for 3D example, computed using the corrected Voronoi method (i.e. Algorithm 4). c) Difference between the corrected and simple Voronoi methods, showing that the simple method gives good approximation to a valid environmental contour.

8 Concluding remarks

In this paper, a novel algorithm for constructing environmental contours has been presented, based on a geometric interpretation of environmental contours as Voronoi cells. One advantage of this approach is that many software libraries exist for Voronoi cell computation, making the algorithm simple to implement. Another advantage is that the Voronoi method also makes it easy to compute environmental contours in higher than two dimensions. The Voronoi environmental contours are not guaranteed to be proper, but with a simple modification to the algorithm, valid environmental contours can always be constructed from improper Voronoi environmental contours.

The Voronoi geometric interpretation also has given new intuition and theoretical insights about environmental contours, including representation and existence theorems for proper convex environmental contours. The presented analytical formula provides another alternative algorithm to compute environmental contours. Interestingly, this formula has an analogy in shadow systems and can be interpreted as an inverse Gauss map [52,16,38]. Further exploration of this correspondence between environmental contours and shadow functions could potentially reveal new insights in both domains, and potentially provide some information on the class of random variables for which proper environmental contours exist.

Acknowledgements This work has been supported by grant 276282 from the Research Council of Norway (RCN) and DNV Group Research and Development. Parts of the work has also been carried out within the research project ECSADES, with support from RCN under the MARTEC II ERA-NET initiative; project no. 249261/O80.

Data availability The examples in this article are based on simulated data. All parameters needed to reproduce the examples are provided in the article.

Appendices

A Proof of Lemma 1

Proving the first statement is trivial, as $\mathbf{x} \in \mathcal{B}_{p_e}$ by definition means that $\mathbf{u} \cdot (\mathbf{x} - \mathbf{o}) \leq C^{\mathbf{o}}_{p_e}(\mathbf{u})$ for any $\mathbf{u} \in \mathcal{U}$. So, in particular, we have that $\mathbf{o} \in \mathcal{B}_{p_e} \Leftrightarrow 0 = \mathbf{u} \cdot (\mathbf{o} - \mathbf{o}) \leq C^{\mathbf{o}}_{p_e}(\mathbf{u})$.

To prove the second statement we use that

$$\mathbf{x} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e} \Rightarrow \mathbf{x} \in \bigcap_{\mathbf{u}\in\mathcal{U}} \left(\Pi^-_{p_e}(\mathbf{u}) \setminus \partial\Pi^-_{p_e}(\mathbf{u}) \right).$$

That is, a point $\mathbf{x}$ in the interior of $\mathcal{B}_{p_e}$ is also in the intersection of all interior half-spaces. Hence, $\mathbf{x} \in \{\mathbf{x} : \mathbf{u} \cdot (\mathbf{x} - \mathbf{o}) < C^{\mathbf{o}}_{p_e}(\mathbf{u})\}$ for all $\mathbf{u} \in \mathcal{U}$. And so by the same argument as above we have that $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e} \Rightarrow 0 = \mathbf{u} \cdot (\mathbf{o} - \mathbf{o}) < C^{\mathbf{o}}_{p_e}(\mathbf{u}) \ \forall \mathbf{u} \in \mathcal{U}$.

To prove the converse, we first observe that if $\mathbf{o} \in \partial\mathcal{B}_{p_e}$, then then there exists some $\mathbf{u}^* \in \mathcal{U}$ where $\mathbf{o} \in \partial\Pi^-_{p_e}(\mathbf{u}^*)$ (by the supporting hyperplane theorem) which means that $C^{\mathbf{o}}_{p_e}(\mathbf{u}^*) = 0$, and if $\mathbf{o} \notin \mathcal{B}_{p_e}$ then we have already shown that $C^{\mathbf{o}}_{p_e}(\mathbf{u}^*) < 0$ for some $\mathbf{u}^*$. Putting this together we get that $\mathbf{o} \notin \mathcal{B}_{p_e} \setminus \mathcal{B}_{p_e} \Rightarrow \exists \mathbf{u}^* \in \mathcal{U}$ s.t. $C^{\mathbf{o}}_{p_e}(\mathbf{u}^*) \leq 0$, and hence $C^{\mathbf{o}}_{p_e}(\mathbf{u}) > 0 \ \forall \mathbf{u} \in \mathcal{U} \Rightarrow \mathbf{o} \in \mathcal{B}_{p_e} \setminus \mathcal{B}_{p_e}$.

As for the final statement, we first recall that a point $\mathbf{x}$ is in $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ if and only if $\|\mathbf{x} - \mathbf{o}\| \leq \left\|\mathbf{x} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}\right\|$, or equivalently $\|\mathbf{x} - \mathbf{o}\|^2 \leq \left\|\mathbf{x} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}\right\|^2$, for any $\mathbf{u} \in U$. We first observe that

$$\left\|\mathbf{x} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}\right\|^2 = \left\|\mathbf{x} - \mathbf{o} - 2C^{\mathbf{o}}_{p_e}(\mathbf{u})\mathbf{u}\right\|^2 = \|\mathbf{x} - \mathbf{o}\|^2 + 4(C^{\mathbf{o}}_{p_e}(\mathbf{u}))^2 - 4C^{\mathbf{o}}_{p_e}(\mathbf{u})(\mathbf{x} - \mathbf{o}) \cdot \mathbf{u}, \tag{20}$$

and so,

$$\|\mathbf{x} - \mathbf{o}\|^2 \leq \left\|\mathbf{x} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}\right\|^2 \Leftrightarrow C^{\mathbf{o}}_{p_e}(\mathbf{u})(\mathbf{x} - \mathbf{o}) \cdot \mathbf{u} \leq (C^{\mathbf{o}}_{p_e}(\mathbf{u}))^2.$$

Hence, using the second statement of the Lemma, we have that if $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$ then $C^{\mathbf{o}}_{p_e}(\mathbf{u}) > 0$, and so $\|\mathbf{x} - \mathbf{o}\|^2 \leq \left\|\mathbf{x} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}\right\|^2 \Leftrightarrow (\mathbf{x} - \mathbf{o}) \cdot \mathbf{u} \leq C^{\mathbf{o}}_{p_e}(\mathbf{u})$ for any $\mathbf{u} \in U$ which completes the proof. □

B Proof of Proposition 2

First we recall that by definition $\mathcal{B}_{p_e} = \bigcap_{\mathbf{u}\in\mathcal{U}} \Pi^-_{p_e}(\mathbf{u})$. Using Lemma 1 we then have $\mathcal{B}_{p_e} = \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U}))$, and also $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U_i)) = \bigcap_{\mathbf{u}\in U_i} \Pi^-_{p_e}(\mathbf{u})$ for $i = 1, 2$.

Since $U_1 \subseteq U_2 \subseteq \mathcal{U}$ the proof is completed by observing that

$$\bigcap_{\mathbf{u}\in\mathcal{U}} \Pi^-_{p_e}(\mathbf{u}) \subseteq \bigcap_{\mathbf{u}\in U_2} \Pi^-_{p_e}(\mathbf{u}) \subseteq \bigcap_{\mathbf{u}\in U_1} \Pi^-_{p_e}(\mathbf{u}).$$

□

C Proof of Proposition 3

The proof will follow from the Voronoi-Delaunay duality, which tell us that the Voronoi cells are convex polytopes with vertices corresponding to circumcenters of the Delaunay simplices. In particular, the vertices of $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ are the circumcenters of the simplices in $\{\tau \in \mathcal{D} \mid \mathbf{o} \in \tau\}$, where $\mathcal{D}$ is any Delaunay triangulation of the point set $\{\mathbf{o}\} \cup S^{\mathbf{o}}_{p_e}(U)$.

Assume that $\mathcal{D}$ is such a Delaunay triangulation, and that there exists a point $\mathbf{s}^* \in \mathcal{S}^{\mathbf{o}}_{p_e}(U)$ such that $\mathbf{s}^*$ and $\mathbf{o}$ are not connected by $\mathcal{D}$. This means (by definition) that any simplex in $\mathcal{D}$ containing $\mathbf{o}$ does not contain $\mathbf{s}^*$, and vice versa. Hence,

$$\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U)) = \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U) \setminus \{\mathbf{s}^*\}).$$

We now let $\mathbf{u}^* \in U$ denote the unit vector corresponding to $\mathbf{s}^*$, i.e. $\mathbf{s}^* = \mathbf{s}^{\mathbf{o},\mathbf{u}^*}_{p_e}$. Making use of Lemma 1 we then observe that

$$\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U)) = \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U) \setminus \{\mathbf{s}^*\}) \Rightarrow \bigcap_{\mathbf{u}\in U} \Pi^-_{p_e}(\mathbf{u}) = \bigcap_{\mathbf{u}\in U\setminus\{\mathbf{u}^*\}} \Pi^-_{p_e}(\mathbf{u}). \tag{21}$$

This means that, either 1) $\Pi_{p_e}(\mathbf{u}^*) \cap \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U)) = \emptyset$, or 2) that there exists some vertex $\mathbf{v}^*$ of $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ such that $\mathbf{v}^* \in \Pi_{p_e}(\mathbf{u}^*) \cap \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$. From Proposition 2 we have that $\mathcal{B}_{p_e} \subseteq \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$. Since we assume that $\partial\mathcal{B}_{p_e}$ is a proper convex environmental contour, $\Pi_{p_e}(\mathbf{u}^*) \cap \mathcal{B}_{p_e} \neq \emptyset$, and so

$$\Pi_{p_e}(\mathbf{u}^*) \cap \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U)) \neq \emptyset. \tag{22}$$

From (21) and (22) we can therefore conclude that there exists some vertex $\mathbf{v}^*$ of $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$ such that $\mathbf{v}^* \in \Pi_{p_e}(\mathbf{u}^*) \cap \mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$.

We then observe that

$$\mathbf{v}^* \in \Pi_{p_e}(\mathbf{u}^*) \Rightarrow \|\mathbf{v}^* - \mathbf{o}\| = \|\mathbf{s}^* - \mathbf{o}\|. \tag{23}$$

This follows from the definition of $\Pi_{p_e}(\cdot)$ and the set $S^{\mathbf{o}}_{p_e}(U)$, which says that $\mathbf{s}^*$ is the reflection of $\mathbf{o}$ with respect to the hyperplane $\Pi_{p_e}(\mathbf{u}^*)$. Now, since $\mathbf{v}^*$ is also a vertex of $\mathrm{Vor}(\mathbf{o}, \mathcal{S}^{\mathbf{o}}_{p_e}(U))$, then $\mathbf{v}^*$ is the circumcenter of a Delaunay simplex τ, with $\mathbf{o} \in \tau$. From (23) we see that $\mathbf{s}^*$ also lies on this circum-hypersphere, together with $\mathbf{o}$. Hence, if the Delaunay triangulation $\mathcal{D}$ was unique, we could conclude that $\{\mathbf{s}^*, \mathbf{o}\} \subset \tau \in \mathcal{D}$, which contradicts the initial assumption that $\mathbf{s}^*$ and $\mathbf{o}$ are not connected in $\mathcal{D}$.

In the case where there is no *unique* Delaunay triangulation of the point set $\{\mathbf{o}\} \cup S^{\mathbf{o}}_{p_e}(U)$, the fact that $\mathbf{s}^*$ and $\mathbf{o}$ lie on the same circum-hypersphere of *some* Delaunay simplex τ lets us conclude that there exists *some* Delaunay triangulation $\mathcal{D}'$ where $\mathbf{s}^*$ and $\mathbf{o}$ are part of the same simplex. We can therefore conclude that, if there exists a Delaunay triangulation $\mathcal{D}$ that does not connect $\mathbf{s}^*$ and $\mathbf{o}$, then there must exist a different Delaunay triangulation $\mathcal{D}'$ that connects $\mathbf{s}^*$ and $\mathbf{o}$.

□

D Proof of Proposition 4

For any $\mathbf{u} \in \mathcal{U}$ we first recall that the existence of some $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ follows from the definition of proper convex environmental contours. We then note that, as any element of $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ is of the form $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} = \mathbf{o} + 2C^{\mathbf{o}}_{p_e}(\mathbf{u})\mathbf{u}$, we have that

$$\begin{aligned}\left\|\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b}\right\|^2 &= \left\|\mathbf{o} - \mathbf{b} + 2C^{\mathbf{o}}_{p_e}(\mathbf{u})\mathbf{u}\right\|^2 = \|\mathbf{o} - \mathbf{b}\|^2 + 4(C^{\mathbf{o}}_{p_e}(\mathbf{u}))^2 + 4C^{\mathbf{o}}_{p_e}(\mathbf{u})(\mathbf{o} - \mathbf{b}) \cdot \mathbf{u}. \\ &= \|\mathbf{o} - \mathbf{b}\|^2 + 4C^{\mathbf{o}}_{p_e}(\mathbf{u}) \left(C^{\mathbf{o}}_{p_e}(\mathbf{u}) - (\mathbf{b} - \mathbf{o}) \cdot \mathbf{u}\right).\end{aligned} \tag{24}$$

Now if $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u})$ we have that $(\mathbf{b} - \mathbf{o}) \cdot \mathbf{u} = C^{\mathbf{o}}_{p_e}(\mathbf{u})$ (by definition), and hence $\left\|\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b}\right\|^2 = \|\mathbf{o} - \mathbf{b}\|^2$, which means that $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} \in \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$.

The statement that $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U}) \cap \mathcal{W}^{\mathbf{o}}(\mathbf{b}) \subseteq \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$ means that there are no $\mathbf{u}' \in \mathcal{U}$ such that $\mathbf{s}^{\mathbf{o},\mathbf{u}'}_{p_e}$ lies in the interior of the ball $\mathcal{W}^{\mathbf{o}}(\mathbf{b})$. Assume, on the contrary, that there exists some $\mathbf{s}^{\mathbf{o},\mathbf{u}'}_{p_e} \in \mathcal{W}^{\mathbf{o}}(\mathbf{b}) \setminus \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$. Then $\left\|\mathbf{s}^{\mathbf{o},\mathbf{u}'}_{p_e} - \mathbf{b}\right\| < \|\mathbf{o} - \mathbf{b}\|$ by definition. From (24) we then have that $4C^{\mathbf{o}}_{p_e}(\mathbf{u}')\left(C^{\mathbf{o}}_{p_e}(\mathbf{u}') - (\mathbf{b} - \mathbf{o}) \cdot \mathbf{u}'\right) < 0$. We have assumed that $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$, and so by Lemma 1 $C^{\mathbf{o}}_{p_e}(\mathbf{u}') > 0$. Hence,

$$\mathbf{s}^{\mathbf{o},\mathbf{u}'}_{p_e} \in \mathcal{W}^{\mathbf{o}}(\mathbf{b}) \setminus \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b}) \Rightarrow C^{\mathbf{o}}_{p_e}(\mathbf{u}') - (\mathbf{b} - \mathbf{o}) \cdot \mathbf{u}' < 0.$$

But this means that $\mathbf{b} \in \mathit{\Pi}^{+}_{p_e}(\mathbf{u}')$, which is impossible when $\mathbf{b} \in \partial\mathcal{B}_{p_e}$.

□

E Proof of Lemma 2

We first observe that the condition 1) is just a different way of stating that a point is on the hyperplane $\mathit{\Pi}_{p_e}(\mathbf{u})$ (alternatively, compute the norms as in (24) and note that $C^{\mathbf{o}}_{p_e}(\mathbf{u}) > 0$). That is, for any $\mathbf{x} \in \mathbb{R}^n$, we have $\mathbf{x} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \Leftrightarrow \|\mathbf{x} - \mathbf{o}\| = \left\|\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{x}\right\|$.

Hence, $\mathbf{a} \in \mathit{\Pi}_{p_e}(\mathbf{u})$ by condition 1). Then, by Proposition 4 there exists some $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ where $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U}) \cap \mathcal{W}^{\mathbf{o}}(\mathbf{b}) \subseteq \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$, and $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} \in \partial\mathcal{W}^{\mathbf{o}}(\mathbf{b})$. This means that the n-dimensional closed ball $\mathcal{W}^{\mathbf{o}}(\mathbf{b})$, centered at $\mathbf{b}$ with radius $\|\mathbf{b} - \mathbf{o}\|$ is tangent to $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ at the point $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$. As both $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ and $\mathcal{W}^{\mathbf{o}}(\mathbf{b})$ are differentiable $(n-1)$-dimensional manifolds, they share the same $(n-1)$-dimensional tangent space at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$. We let $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_{n-1}\} \subset \mathbb{R}^n$ denote a basis for this tangent space.

From the above argument, it is clear that also $\mathbf{b}$ satisfies both of the criteria in the Lemma, as 1) $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u})$ and 2) $(\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b})$ is orthogonal to $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$ since $(\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b})$ is orthogonal to $\mathcal{W}^{\mathbf{o}}(\mathbf{b})$ at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$.

Hence, starting with a pair $(\mathbf{a}, \mathbf{u})$ that satisfies the two conditions of the Lemma, we have identified a point $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ such that $(\mathbf{b}, \mathbf{u})$ satisfies the same conditions. Using that $(\mathbf{a}, \mathbf{u})$ and $(\mathbf{b}, \mathbf{u})$ satisfy these conditions simultaneously, we obtain

1. $\Rightarrow \mathbf{a}, \mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \Rightarrow \mathbf{a} \cdot \mathbf{u} = \mathbf{b} \cdot \mathbf{u}$,
2. $\Rightarrow (\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{a}) \cdot \mathbf{v} = (\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b}) \cdot \mathbf{v} = 0$ for any $\mathbf{v} \in V$.

From these conditions we see that $(\mathbf{a} - \mathbf{b}) \cdot \mathbf{u} = 0$ and $(\mathbf{a} - \mathbf{b}) \cdot \mathbf{v} = 0$ for any $\mathbf{v} \in V$. Hence, if $\mathbf{u}$ is linearly independent of V, we can conclude that $\mathbf{a} = \mathbf{b}$.

Assume $\mathbf{u} = \sum_{i=1}^{n-1} \alpha_i \mathbf{v}_i$ for some $\alpha_1, \ldots, \alpha_{n-1} \in \mathbb{R}$. Then $(\mathbf{b} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}) \cdot \mathbf{u} = \sum_{i=1}^{n-1} \alpha_i (\mathbf{b} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}) \cdot \mathbf{v}_i = 0$. Then, by definition of the hyperplane $\mathit{\Pi}_{p_e}(\mathbf{u})$, $C^{\mathbf{o}}_{p_e}(\mathbf{u}) = (\mathbf{b} - \mathbf{o}) \cdot \mathbf{u} = (\mathbf{b} - \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} + \mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{o}) \cdot \mathbf{u} = (\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{o}) \cdot \mathbf{u}$. But this means that $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} \in \mathit{\Pi}_{p_e}(\mathbf{u})$, which is impossible.

We may therefore conclude that $\mathbf{a} = \mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$. By the same argument as above, if $\mathbf{b}_1$ and $\mathbf{b}_2$ are two elements of $\mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$, then since $(\mathbf{b}_1, \mathbf{u})$ and $(\mathbf{b}_2, \mathbf{u})$ both satisfy the conditions of the Lemma, we must have $\mathbf{b}_1 = \mathbf{b}_2$. $\mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ is therefore a singleton set, and we can conclude that $\{\mathbf{a}\} = \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$.

□

F Proof of Proposition 5

We first recall that if $\partial\mathcal{B}_{p_e}$ is a proper convex environmental contour, then for any $\mathbf{b} \in \partial\mathcal{B}_{p_e}$ there exists some $\mathbf{u} \in \mathcal{U}$ such that $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u})$, and so $\partial\mathcal{B}_{p_e} \subset \cup_{\mathbf{u}\in\mathcal{U}} \mathit{\Pi}_{p_e}(\mathbf{u})$.

Then, if $F : \mathcal{U} \to \mathbb{R}^n$ is a mapping such that the assumptions and conditions of Lemma 2 hold for any pair $(F(\mathbf{u}), \mathbf{u})$, Lemma 2 lets us conclude that $\{F(\mathbf{u})\} = \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ for any $\mathbf{u} \in \mathcal{U}$.

Hence, $F(\mathcal{U}) = \cup_{\mathbf{u}\in\mathcal{U}}(\mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}) = \partial\mathcal{B}_{p_e} \cap (\cup_{\mathbf{u}\in\mathcal{U}} \mathit{\Pi}_{p_e}(\mathbf{u})) = \partial\mathcal{B}_{p_e}$.

□

G Proof of Theorem 1

If the the p_e-level percentile function $C_{p_e}(\mathbf{u})$ is continuously differentiable on the unit $(n-1)$-sphere, then as $\mathbf{s}^{\mathbf{o}}_{p_e} = \mathbf{o} + 2(C_{p_e}(\mathbf{u}) - \mathbf{u}\cdot\mathbf{o})\mathbf{u}$, the set $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U}) = \{s^{\mathbf{o}}_{p_e}\}$ is a differentiable manifold. Hence, the assumptions of Lemma 2 are satisfied.

We first note that, as a consequence of Lemma 2, any supporting hyperplane intersects $\partial\mathcal{B}_{p_e}$ at a single point, which means that $\mathcal{B}_{p_e}$ is strictly convex. For details we refer to the proof of Lemma 2 in Appendix E, where we observe that $\mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$ is a singleton set for any $\mathbf{u} \in \mathcal{U}$, as the pair $(\mathbf{b}, \mathbf{u})$ satisfies the conditions in Lemma 2 for any $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u}) \cap \partial\mathcal{B}_{p_e}$. (And for any $\mathbf{b} \in \partial\mathcal{B}_{p_e}$ we have $\mathbf{b} \in \mathit{\Pi}_{p_e}(\mathbf{u})$ for some $\mathbf{u}$ as $\partial\mathcal{B}_{p_e}$ is proper).

We will show that the proposed parametrization in the theorem is valid using Lemma 2 and Proposition 5. That is, for any $\mathbf{u} = \mathbf{u}(\boldsymbol{\theta}) \in \mathcal{U}$, we must show that

1. $\|\mathbf{b}(\boldsymbol{\theta}) - \mathbf{o}\| = \left\|\mathbf{s}^{\mathbf{o},\mathbf{u}(\boldsymbol{\theta})}_{p_e} - \mathbf{b}(\boldsymbol{\theta})\right\|$, and
2. $(\mathbf{s}^{\mathbf{o},\mathbf{u}(\boldsymbol{\theta})}_{p_e} - \mathbf{b}(\boldsymbol{\theta}))$ is orthogonal to $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ at $\mathbf{s}^{\mathbf{o},\mathbf{u}(\boldsymbol{\theta})}_{p_e}$,

for $\mathbf{o} \in \mathcal{B}_{p_e} \setminus \partial\mathcal{B}_{p_e}$. To simplify the notation we will suppress writing out the dependency on $\boldsymbol{\theta}$, and write

$$\mathbf{b} = C_{p_e}\mathbf{u} + \nabla\mathbf{u}\, g^{-1}(\nabla C_{p_e})^T.$$

Using (9) we can express $\mathbf{b}$ in terms of $C^{\mathbf{o}}_{p_e}$:

$$\begin{aligned}
\mathbf{b} &= C_{p_e}\mathbf{u} + \nabla\mathbf{u}\, g^{-1}(\nabla C_{p_e})^T \\
&= C_{p_e}\mathbf{u} + \mathbf{u}\mathbf{u}^T\mathbf{o} + \nabla\mathbf{u}\, g^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T + \nabla\mathbf{u}\, g^{-1}(\nabla\mathbf{u})^T\mathbf{o} \\
&= \mathbf{o} + C^{\mathbf{o}}_{p_e}\mathbf{u} + \nabla\mathbf{u}\, g^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T,
\end{aligned}$$

where we made use of the property that $\mathbf{u}\mathbf{u}^T + \nabla\mathbf{u}\, g^{-1}(\nabla\mathbf{u})^T = I$ (i.e. the identity operator). Note that the metric tensor $g = (\nabla\mathbf{u})^T\nabla\mathbf{u}$ is invertible because we have assumed a regular parametrization (and so $\nabla\mathbf{u}$ has full rank).

To show condition (1) above, we can just compute the norms

$$\begin{aligned}
&\|\mathbf{b} - \mathbf{o}\|^2 - \left\|\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b}\right\|^2 \\
&= \left\|C^{\mathbf{o}}_{p_e}\mathbf{u} + \nabla\mathbf{u}\, g^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T\right\|^2 - \left\|C^{\mathbf{o}}_{p_e}\mathbf{u} - \nabla\mathbf{u}\, g^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T\right\|^2 \\
&= 4C^{\mathbf{o}}_{p_e}\mathbf{u}\cdot\nabla\mathbf{u}\, g^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T \\
&= 0.
\end{aligned}$$

Here we have used the fact that $\mathbf{u}\cdot\nabla\mathbf{u} = \mathbf{u}^T\nabla\mathbf{u} = \frac{1}{2}\nabla(\mathbf{u}^T\mathbf{u}) = \nabla(1) = 0$.

To show condition (2) we will use that the columns of $\nabla\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$ form a basis for the tangent space of $\mathcal{S}^{\mathbf{o}}_{p_e}(\mathcal{U})$ at $\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e}$. The orthogonality condition (2) is therefore equivalent to saying that $\nabla(s^{\mathbf{o},\mathbf{u}}_{p_e})^T(s^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b}) = \mathbf{0}$. But this follows from the definition of $s^{\mathbf{o},\mathbf{u}}_{p_e}$, as $\nabla\mathbf{s}^{\mathbf{o},\mathbf{u}}_{p_e} = \nabla(\mathbf{o} + 2C^{\mathbf{o}}_{p_e}\mathbf{u}) = 2(C^{\mathbf{o}}_{p_e}\nabla\mathbf{u} + \mathbf{u}\nabla C^{\mathbf{o}}_{p_e})$, and hence

$$\begin{aligned}
\frac{1}{2}\nabla(s^{\mathbf{o},\mathbf{u}}_{p_e})^T(s^{\mathbf{o},\mathbf{u}}_{p_e} - \mathbf{b}) &= \left(C^{\mathbf{o}}_{p_e}\nabla\mathbf{u}^T + (\nabla C^{\mathbf{o}}_{p_e})^T\mathbf{u}^T\right)\left(C^{\mathbf{o}}_{p_e}\mathbf{u} - \nabla\mathbf{u}A^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T\right) \\
&= (C^{\mathbf{o}}_{p_e})^2\underbrace{\nabla\mathbf{u}^T\mathbf{u}}_{\mathbf{0}} - C^{\mathbf{o}}_{p_e}\underbrace{\nabla\mathbf{u}^T\nabla\mathbf{u}A^{-1}}_{I}(\nabla C^{\mathbf{o}}_{p_e})^T \\
&\quad + C^{\mathbf{o}}_{p_e}(\nabla C^{\mathbf{o}}_{p_e})^T\underbrace{\mathbf{u}^T\mathbf{u}}_{1} - (\nabla C^{\mathbf{o}}_{p_e})^T\underbrace{\mathbf{u}^T\nabla\mathbf{u}}_{\mathbf{0}}A^{-1}(\nabla C^{\mathbf{o}}_{p_e})^T \\
&= -C^{\mathbf{o}}_{p_e}(\nabla C^{\mathbf{o}}_{p_e})^T + C^{\mathbf{o}}_{p_e}(\nabla C^{\mathbf{o}}_{p_e})^T \\
&= \mathbf{0}.
\end{aligned}$$

Using Proposition 5 we may then conclude that, given an atlas $\{\mathbf{u}_i(\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta_i\}_i$ on $\mathcal{U}$ where each $(\mathbf{u}_i, \Theta_i)$ is a regular parametrization, the corresponding charts $(\mathbf{b}_i, \Theta_i)$ is an atlas on $\partial\mathcal{B}_{p_e}$. Finally, differentiability of $\partial\mathcal{B}_{p_e}$ then follows from the given expression for $\mathbf{b}_i$ as a function of $\boldsymbol{\theta}$.

□

H Proof of Lemma 3

We first observe that, as a direct consequence of Definition 1, $\mathbf{X}$ admits a proper convex environmental contour if and only if every hyperplane $\Pi_{p_e}(\mathbf{u})$ is a supporting hyperplane of $\mathcal{B}_{p_e}$. That is, if and only if $\mathcal{B}_{p_e} \cap \Pi_{p_e}(\mathbf{u}) \neq \emptyset$ for all $\mathbf{u} \in \mathcal{U}$.

Hence, if $\mathbf{X}$ admits a proper convex environmental contour, we can select $\mathbf{o} \in \Pi_{p_e}(\mathbf{u}') \cap \partial\mathcal{B}_{p_e}$ which (by Lemma 1) satisfies the condition.

If $\mathbf{X}$ does not admit a proper convex environmental contour, then there is some hyperplane $\Pi_{p_e}(\mathbf{u}')$ that does not intersect B_{p_e}. Hence, for any $\mathbf{o} \in \Pi_{p_e}(\mathbf{u}')$ we have $\mathbf{o} \notin B_{p_e}$, and by Lemma 1 there must exist some $\mathbf{u}^*$ where $C^{\mathbf{o}}_{p_e}(\mathbf{u}^*) < 0$.

□

I Proof of Lemma 4

Dropping the dependency on $\boldsymbol{\theta}$ and p_e for simpler notation, we may write

$$\mathbf{u}^T\mathbf{b} = \mathbf{u}^T(C\mathbf{u} + \nabla\mathbf{u}g^{-1}\nabla C^T) = C\mathbf{u}^T\mathbf{u} + \mathbf{u}^T\nabla\mathbf{u}g^{-1}\nabla C^T = C,$$

as $\mathbf{u}^T\mathbf{u} = 1$ and $\mathbf{u}^T\nabla\mathbf{u} = \frac{1}{2}\nabla(\mathbf{u}^T\mathbf{u}) = \nabla(1) = 0$. This means that $\mathbf{b}(\boldsymbol{\theta}) \in \Pi(\boldsymbol{\theta})$. Similarly, we observe that

$$\nabla\mathbf{u}^T\mathbf{b} = C\nabla\mathbf{u}^T\mathbf{u} + \nabla\mathbf{u}^T\nabla\mathbf{u}g^{-1}\nabla C^T = \nabla C^T,$$

as $\nabla\mathbf{u}^T\nabla\mathbf{u} = g$ by definition. From the chain rule we then get $\mathbf{u}^T\nabla\mathbf{b} = \nabla(\mathbf{u}^T\mathbf{b}) - (\nabla\mathbf{u}^T\mathbf{b})^T = \nabla C - \nabla C = \mathbf{0}$. Since the hyperplane $\Pi(\boldsymbol{\theta})$ has normal vector $\mathbf{u}(\boldsymbol{\theta})$, we can conclude that $\Pi(\boldsymbol{\theta})$ is tangential to $\mathbf{b}(\Theta)$ at $\mathbf{b}(\boldsymbol{\theta})$.

□

J Proof of Theorem 2

To simplify notation, we drop the dependency p_e and the index i of the parametrization.

Assume (2) is true and let $\mathcal{B}$ denote the closed convex set. Then Lemma 4 implies that all hyperplanes $\Pi(\boldsymbol{\theta})$ are supporting hyperplanes of $\mathcal{B}$, and so $\partial\mathcal{B}$ is a proper convex environmental contour. The fact that (1) $\Rightarrow$ (2) comes as a direct consequence of Theorem 1, so we have that (1) $\Leftrightarrow$ (2).

To show that (1) $\Rightarrow$ (3), we first note that when $\mathbf{X}$ admits a proper convex environmental contour, then since $\boldsymbol{b}(\boldsymbol{\theta}) \in \Pi(\boldsymbol{\theta})$ (see Lemma 4) it follows from Lemma 1 that $\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}') \geq 0$ for all $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$. For the converse, assume that $\mathbf{X}$ does not admit a proper convex environmental contour. Then from Lemma 3 there exists some $\mathbf{u}'$ such that for any $\mathbf{o} \in \Pi(\mathbf{u}')$ we can find some $\mathbf{u}$ where $C^{\mathbf{o}}(\mathbf{u}) < 0$. In forms of the given parametrization, this means that we can find some $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ where $C^{\mathbf{o}}(\boldsymbol{\theta}) < 0$ for any $\mathbf{o} \in \Pi(\boldsymbol{\theta}')$. As $\mathbf{b}(\boldsymbol{\theta}') \in \Pi(\boldsymbol{\theta}')$ we have that $\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}') = C^{\mathbf{b}(\boldsymbol{\theta}')}(\boldsymbol{\theta}) < 0$. Hence (1) $\Leftrightarrow$ (3).

Finally, (3) $\Leftrightarrow$ (4) follows from the fact that $\mathbf{b}(\boldsymbol{\theta}') \in \Pi(\boldsymbol{\theta}')$ which means that $\kappa(\boldsymbol{\theta}'|\boldsymbol{\theta}') = 0$.

□

K Proof of Corollary 2

From statement (4) in Theorem 2, $\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}')$ attains a local minimum at $\boldsymbol{\theta} = \boldsymbol{\theta}'$, which means that the matrix $A(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}}\nabla_{\boldsymbol{\theta}}\kappa(\boldsymbol{\theta}|\boldsymbol{\theta}')|_{\boldsymbol{\theta}=\boldsymbol{\theta}'}$ is positive semi-definite $\forall\boldsymbol{\theta} \in \Theta$. Suppressing the notation $\boldsymbol{\theta}$ and p_e we can write

$$\begin{aligned} A &= \nabla\nabla C + \mathbf{b}^T\nabla\nabla\mathbf{u} \\ &= \nabla\nabla C + \left(C\mathbf{u} + \nabla\mathbf{u}g^{-1}\nabla C^T\right)^T\nabla\nabla\mathbf{u} \\ &= \nabla\nabla C + C\mathbf{u}^T\nabla\nabla\mathbf{u} + (\nabla\mathbf{u}g^{-1}\nabla C^T)^T\nabla\nabla\mathbf{u}. \end{aligned} \tag{25}$$

The second term in the last line of (25) above can be rewritten in terms of the metric tensor g:

$$C\mathbf{u}^T\nabla\nabla\mathbf{u} = C\nabla\left(\mathbf{u}^T\nabla\mathbf{u}\right) - C(\nabla\mathbf{u})^T\nabla\mathbf{u} = -gC,$$

because $\mathbf{u}^T\nabla\mathbf{u} = \mathbf{0}$ and $(\nabla\mathbf{u})^T\nabla\mathbf{u} = g$.

The third term in the last line of (25) can be expressed as $\nabla C g^{-T}(\nabla\mathbf{u})^T\nabla\nabla\mathbf{u}$. In index form (using Einstein summation convention) we may write the matrix elements of this term as $c_{,m}g^{lm}u_{k,l}u_{k,ij} = c_{,m}g^{lm}\Gamma_{lij} = c_{,m}\Gamma^m_{ij}$, where we have recognised the Christoffel symbols of the first and second kind, i.e. $\Gamma_{lij} = u_{k,l}u_{k,ij}$ and $\Gamma^m_{ij} = g^{lm}\Gamma_{lij}$. Therefore we may write

$$A_{ij}(\boldsymbol{\theta}) = \left(\frac{\partial C(\boldsymbol{\theta})}{\partial\theta_i\partial\theta_j} - \Gamma^m_{ij}\frac{C(\boldsymbol{\theta})}{\partial\theta_m}\right) + g_{ij}(\boldsymbol{\theta})C(\boldsymbol{\theta}). \tag{26}$$

The term in brackets correspond to the Hessian on a Riemann manifold, and we may therefor write

$$A(\boldsymbol{\theta}) = Hess(C(\boldsymbol{\theta})) + g(\boldsymbol{\theta})C(\boldsymbol{\theta}). \tag{27}$$

□

References

1. Armstrong, C., Chin, C., Penesis, I., Drobyshevski, Y.: Sensitivity of vessel response to environmental contours of extreme sea states. In: Proc. 34th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2015). American Society of Mechanical Engineers (ASME) (2015)
2. Aurenhammer, F.: Voronoi diagrams - a survey of a fundamental geometric data structure. ACM Comput. Surv. **23**(3), 345–405 (1991)
3. Baarholm, G.S., Haver, S.: Application of environmental contour lines - a summary of a number of case studies. In: Proc. International Conference on Floating Structures for Deepwater Operations. ASRANet (2009)
4. Baarholm, G.S., Haver, S., Økland, O.D.: Combining contours of significant wave height and peak period with platform response distributions for predicting design response. Marine Structures **23**, 147–163 (2010)
5. Baarholm, G.S., Moan, T.: Application of contour line method to estimate extreme ship hull loads considering operational restrictions. Journal of Ship Research **45**, 228–240 (2001)
6. Baarholm, G.S., Sverre, H., Larsen, C.M.: Wave sector dependent contour lines. In: Proc. 26th International Conference on Offshore Mechanics and Arctic Engineering (OMAE 2007). American Society of Mechanical Engineers (ASME) (2007)
7. Bitner-Gregersen, E.: Joint long term models of met-ocean parameters. In: C. Guedes Soares (ed.) Marine Technology and Engineering: CENTEC Anniversary Book. CRC Press (2012)
8. Bitner-Gregersen, E., Haver, S.: Joint long term description of environmental parameters for structural response calculation. In: Proc. 2nd International Workshop on Wave Hindcasting and Forecasting (1989)
9. Bitner-Gregersen, E., Haver, S.: Joint environmental model for reliability calculations. In: Proc. 1st International Offshore and Polar Engineering conference (ISOPE 1991). The International Society of Offshore and Polar Engineering (ISOPE) (1991)
10. Bitner-Gregersen, E.M.: Joint met-ocean description for design and operation of marine structures. Applied Ocean Research **51**, 279–292 (2015)
11. Chai, W., Leira, B.J.: Environmental contours based on inverse SORM. Marine Structures **60**, 34–51 (2018)
12. Dahl, K.R., Huseby, A.B.: Buffered environmental contours. In: Proc. ESREL 2018. European Safety and Reliability Association(ESRA) (2018)
13. Derbanne, Q., da Hauteclocque, G.: A new approach for environmental contour and multivariate de-clustering. In: Proc. 38th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2019). American Society of Mechanical Engineers (ASME) (2019)
14. DNV GL: Environmental Conditions and Environmental Loads. DNV GL, september 2019 edn. (2019). DNVGL-RP-C205
15. Eckert-Gallup, A.C., Sallaberry, C.J., Dallman, A.R., Neary, V.S.: Application of principal component analysis (PCA) and imrpoved joint probability distributions to the inverse first-order reliability method (I-FORM) for predicting extreme sea states. Ocean Engineering **112**, 307–319 (2016)
16. l. Epstein, C.: Convex regions, shadows, and the Gauss map. URL `https://www.math.upenn.edu/~cle/papers/slatgm.pdf`
17. Haghayeghi, Z.S., Ketabdari, M.J.: Development of environmental contours for circular and linear metocean variables. International Journal of Renewable Energy Research **7**, 682–693 (2017)
18. Haselsteiner, A.F., Ohlendorf, J.H., Wosniok, W., Thoben, K.D.: Deriving environmental contours from highest density regions. Coastal Engineering **123**, 42–51 (2017)
19. Haver, S.: Analysis of uncertainties related to the stochastic modelling of ocean waves. Tech. Rep. UR-80-09, Norges tekniske høgskole (1980)
20. Haver, S.: On the joint distribution of heights and periods of sea waves. Ocean Engineering **14**, 359–376 (1987)
21. Haver, S., Bruserud, K.: Environmental contour method: An approximate method for obtaining characteristic response extremes for design purposes. In: Proc. 13th International Workshop on Wave Hindcasting and Forecasting & 4th Coastal Hazard Symposium (2013)
22. Haver, S., Winterstein, S.: Environmental contour lines: A method for estimating long term extremes by a short term analysis. Transactions of the Society of Naval Architects and Marine Engineers **116**, 116–127 (2009)

23. Huseby, A.B., Vanem, E., Agrell, C., Hafver, A.: Convex environmental contours. Submitted
24. Huseby, A.B., Vanem, E., Barbosa, M.H.: Environmental contours for mixtures of distributions. In: Proc. ESREL 2019. European Safety and Reliability Association(ESRA) (2019)
25. Huseby, A.B., Vanem, E., Eskeland, K.: Evaluating properties of environmental contours. In: Proc. ESREL 2017. European Safety and Reliability Association(ESRA) (2017)
26. Huseby, A.B., Vanem, E., Natvig, B.: A new approach to environmental contours for ocean engineering applications based on direct Monte Carlo simulations. Ocean Engineering **60**, 124–135 (2013)
27. Huseby, A.B., Vanem, E., Natvig, B.: A new Monte Carlo method for environmental contour estimation. In: Proc. ESREL 2014. European Safety and Reliability Association(ESRA) (2014)
28. Huseby, A.B., Vanem, E., Natvig, B.: Alternative environmental contours for structural reliability analysis. Structural Safety **54**, 32–45 (2015)
29. Jonathan, P., Ewans, K., Flynn, J.: On the estimation of ocean engineering design contours. In: Proc. 30th International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2011). American Society of Mechanical Engineers (ASME) (2011)
30. Leira, B.J.: A comparison of stochastic process models for definition of design contours. Structural Safety **30**, 493–505 (2008)
31. Leonard, I., Lewis, J.: Geometry of Convex Sets. Wiley (2015)
32. Li, Q., Gao, Z., Moan, T.: Modified environmental contour method for predicting long-term extreme responses of bottom-fixed offshore wind turbines. Marine Structures **48**, 15–32 (2016)
33. van de Lindt, J., Niedzwecki, J.: Environmental contour analysis in earthquake engineering. Engineering Structures **22**, 1661–1676 (2000)
34. Lutes, L.D., Winterstein, S.R.: A dynamic inverse FORM method: Design contours for load combination problems. Probabilistic Engineering Mechanics **44**, 118–127 (2016)
35. M. Schaller, F., Kapfer, S., Evans, M., J.F. Hoffmann, M., Aste, T., Saadatfar, M., Mecke, K., W. Delaney, G., Schröder-Turk, G.: Set Voronoi diagrams of 3D assemblies of aspherical particles. Philosophical Magazine **93** (2013)
36. Manuel, L., Nguyen, P.T., Canning, J., Coe, R.G., Eckert-Gallup, A.C., Martin, N.: Alternative approaches to develop environmental contours from metocean data. Journal of Ocean Engineering and Marine Energy **4**, 293–310 (2018)
37. Marsaglia, G.: Choosing a point from the surface of a sphere. Ann. Math. Statist. **43**(2), 645–646 (1972)
38. Martini, H., Montejano, L., Oliveros, D.: Bodies of constant width. Springer (2019)
39. Montes-Iturrizaga, R., Heredia-Zavoni, E.: Environmental contours using copulas. Applied Ocean Research **52**, 125–139 (2015)
40. Montes-Iturrizaga, R., Heredia-Zavoni, E.: Multivariate environmental contours using C-vine copulas. Ocean Engineering **118**, 68–82 (2016)
41. Montes-Iturrizaga, R., Heredia-Zavoni, E.: Assessment of uncertainty in environmental contours due to parametric uncertainty in models of the dependence structure between metocean variables. Applied Ocean Research **64**, 86–104 (2017)
42. Muliawan, M.J., Gao, Z., Moan, T.: Application of the contour line method for estimating extreme responses in the mooring lines of a two-body floating wave energy converter. Journal of Offshore Mechanics and Arctic Engineering **135**, 031301:1–10 (2013)
43. Nerzic, R., Frelin, C., Prevesto, M., Quiniou-Ramus, V.: Joint distribution of wind/waves/current in West Africa and derivation of multivariate extreme I-FORM contours. In: Proc. 17th International Offshore and Polar Engineering Conference (ISOPE 2007). The International Society of Offshore and Polar Engineering (ISOPE) (2007)
44. Niedzwwecki, J.M., van de Lindt, J., Yao, J.: Estimating extreme tendon response using environmental contours. Engineering Structures **20**, 601–607 (1998)
45. NORSOK: NORSOK Standard N-003:2017. Action and action effects (2017). Edition 3
46. Okabe, A., Boots, B., Sugihara, K., Chiu, S.N.: Spatial Tessellations: Concepts and Applications of Voronoi Diagrams, 2nd ed. edn. Series in Probability and Statistics. John Wiley and Sons, Inc. (2000)
47. Orsero, P., Fontaine, E., Quiniou, V.: Reliability and response based design of a moored FPSO in West Africa using multivariate environmental contours and response surfaces. In: Proc. 17th International Offshore and Polar Engineering Conference (ISOPE 2007). The International Society of Offshore and Polar Engineering (ISOPE) (2007)
48. Raillard, N., Prevesto, M., Pineau, H.: 3-d environmental extreme value models for the tension in a mooring line of a semi-submersible. Ocean Engineering **184**, 23–31 (2019)
49. Ross, E., Astrup, O.C., Bitner-Gregersen, E., Bunn, N., Feld, G., Gouldby, B., Huseby, A., Liu, Y., Randell, D., Vanem, E., Jonathan, P.: On environmental contours for marine and coastal design. Ocean Engineering **195**, 106194 (2020)
50. Saranyasoontorn, K., Manuel, L.: Efficient models for wind turbine extreme loads using inverse reliability. Journal of Wind Engineering and Industrial Aerodynamics **92**, 789–804 (2004)
51. Saranyasoontorn, K., Manuel, L.: Design loads for wind turbines using the environmental contour method. In: 44th AIAA Aerospace Sciences Meeting and Exhibit, pp. AIAA 2006–1365. American Institute of Aeronautics and Astronautics (AIAA) (2006)
52. Shephard, G.C.: Shadow systems of convex sets. Israel Journal of Mathematics **2**(4), 229–236 (1964)
53. Silva-González, F., Heredia-Zavoni, E., Montes-Iturrizaga, R.: Development of environmental contours using Nataf distribution model. Ocean Engineering **58**, 27–34 (2013)
54. Vanem, E.: A comparison study on the estimation of extreme structural response from different environmental contour methods. Marine Structures **56**, 137–162 (2017)
55. Vanem, E.: 3-dimensional environmental contours based on a direct sampling method for structural reliability analysis of ships and offshore structures. Ships and Offshore Structures **14**, 74–85 (2018)
56. Vanem, E.: A simple approach to account for seasonality in the description of extreme ocean environments. Marine Systems & Ocean Technology **13**, 63–73 (2018)

57. Vanem, E.: Environmental contours for describing extreme ocean wave conditions based on combined datasets. Stochastic Environmental Research and Risk Assessment **33**, 957–971 (2019)
58. Vanem, E., Bitner-Gregersen, E.M.: Alternative environmental contours for marine structural design - a comparison study. Journal of Offshore Mechanics and Arctic Engineering **137**, 051601:1–8 (2015)
59. Vanem, E., Gramstad, O., Bitner-Gregersen, E.M.: A simulation study on the uncertainty of environmental contours due to sampling variability for different estimation methods. Applied Ocean Research **91**, 101870 (2019)
60. Vanem, E., Guo, B., Ross, E., Jonathan, P.: Comparing different contour methods with response-based methods for extreme ship response analysis. Marine Structures **69**, 102680 (2919)
61. Winterstein, S., Ude, T., Cornell, C., Bjerager, P., Haver, S.: Environmental parameters for extreme response: Inverse FORM with omission factors. In: Proc. 6th International Conference on Structural Safety and Reliability (1993)
62. Winterstein, S.R., Jha, A.K., Kumar, S.: Reliability of floating structures: Extreme response and load factor design. Journal of Waterway, Port, Coastal and Ocean Engineering **125**, 163–169 (1999)

Appendices

Appendix A

Pitfalls of machine learning for tail events in high risk environments

Christian Agrell, Simen Eldevik, Andreas Hafver, Frank Børre Pedersen, Erik Stensrud, Arne Huseby

In Stein Haugen, Anne Barros, Coen van Gulijk, Trond Kongsvik, and Jan Erik Vinnem, editors, Safety and Reliability – Safe Societies in a Changing World. Proceedings of ESREL 2018. CRC Press, June 2018.

Safety and Reliability – Safe Societies in a Changing World – Haugen et al. (Eds)
© 2018 Taylor & Francis Group, London, ISBN 978-0-8153-8682-7

Pitfalls of machine learning for tail events in high risk environments

C. Agrell, S. Eldevik, A. Hafver, F.B. Pedersen & E. Stensrud
DNV GL AS, Norway

A. Huseby
University of Oslo, Norway

ABSTRACT: Most of today's Machine Learning (ML) methods and implementations are based on correlations, in the sense of a statistical relationship between a set of inputs and the output(s) under investigation. The relationship might be obscure to the human mind, but through the use of ML, mathematics and statistics makes it seemingly apparent. However, to base safety critical decisions on such methods suffer from the same pitfalls as decisions based on any other correlation metric that disregards causality. Causality is key to ensure that applied mitigation tactics will actually affect the outcome in the desired way. This paper reviews the current situation and challenges of applying ML in high risk environments. It further outlines how phenomenological knowledge, together with an uncertainty-based risk perspective can be incorporated to alleviate the missing causality considerations in current practice.

1 INTRODUCTION

In this paper we highlight some pitfalls to avoid in the design of Machine Learning (ML) applications for high risk engineering applications. We also propose some recommendations to consider in model development, and emphasize the introduction of causality constraints based on phenomenological knowledge.

1.1 *Background*

ML is recognized as one of the key enablers for the fourth industrial revolution[1]. In this setting it is often communicated as a tool that, together with increased access to data and computational power, can unlock a huge potential for increased efficiency, new insights and ultimately new revenue streams. Success stories of businesses that have disrupted entire industries are often shared to inspire investment in similar technologies. However, for operation of complex engineering systems in high risk environments, the new challenges that appear are often not clearly expressed.

Concerns have been raised regarding the reliability and trustworthiness of systems relying on Artificial Intelligence (AI) in general, and specifically related to the current main strategy of implementation. Knight (2017) emphasizes that "not knowing how the most advanced algorithms do what they do might become a serious problem as computers become more responsible for making important decisions". The problem of accidents in ML systems is discussed in detail in a paper led by Google Brain researchers Amodei, Olah, Steinhardt, Christiano, Schulman, & Mané (2016). Here the authors motivate the increasing need to address these safety problems by some general trends, one of which relates to the increasing autonomy in AI systems. "Systems that simply output a recommendation to human users, such as speech systems, typically have relatively limited potential to cause harm. By contrast, systems that exert direct control over the world, such as machines controlling industrial processes, can cause harms in a way that humans cannot necessarily correct or oversee."

In this paper we turn our attention towards the introduction of ML in design and operation of complex engineering systems in high risk environments. These are systems where today's methods for assessing risk relies heavily on understanding the underlying physical processes and our ability to model these. This is in contrast to e.g. mass production of components where more data-driven, statistically founded methods can be applied to estimate rates of failure.

We acknowledge the need for ML technologies to address increasing complexity of engineering systems, and the challenges that follow in quantifying uncertainty and risk based on detailed numerical simulation. However, this type of application reduces the tolerance for erroneous model behavior. Most of the methods applied in ML are based

[1]For a broader discussion see e.g. Lu (2017) and Dopico, Gomez, De la Fuente, García, Rosillo, & Puche (2016).

on historic statistical models, enhanced by recent breakthroughs in computer science. The assumptions, limitations and practical challenges of these statistical models still remain, and serve as recurring pitfalls in the digital era.

For a given application both the statistical and ML mindsets may have its advantages and drawbacks, but we argue that for the high risk engineering problems discussed in this paper, uncritical application of ML is unwarranted. We believe that experienced ML practitioners know this, and the main problem lies in how ML is communicated in the current digitalization boom, and how it is perceived by the increasing number of new practitioners in the field that may also be strongly incentivized to develop solutions for cutting costs through automation.

1.2 *Correlation and causation*

Most of today's ML methods and implementations are based on correlations, which we define in the general sense as *any statistical relationship, whether causal or not, between random variables*, i.e. the degree of which two or more variables tend to vary together.

"Correlation does not imply causation" is a well used phrase within statistics, and describes what still remains as one of the major pitfalls in the analysis of data. The importance of this distinction depends on the degree to which one intends to intervene, and the consequence of erroneous intervention. See e.g (Pearl 2010). For many ML applications this may not be significant. But for the use cases considered in this paper it plays an important role, and we will argue that causality constraints from phenomenological knowledge should be incorporated in ML models—to strengthen model performance for tail events, to increase model transparency, and to make it easier to falsify models that do not comply with observations.

1.3 *Structure of this paper*

Section 2 is included for readers that are unfamiliar or new to the field of ML, and gives a brief introduction to some core concepts and their relation to classical statistics. The experienced ML practitioner may jump directly to Section 3, where we propose a set of model properties that should be accommodated for ML applications in high risk, low probability scenarios. Going beyond model selection, some general pitfalls are highlighted in Section 4 which gives an example illustrating the use of ML for anomaly detection and space exploration in Structural Reliability Analysis (SRA). Finally, our conclusions and some final remarks are summarized in Section 5.

2 A BIT OF HISTORY

Often times discussions regarding ML and AI can become fairly opaque, colored by data science jargon, cognitive analogies and marketing buzzwords. This section will clarify the relation between ML and statistical methods by comparing the classes of ML problems with their statistical counterparts, and tracing their origins. This section will only briefly present the statistical background that underpin the main classes of ML methods. See e.g. (Hastie, Tibshirani, & Friedman 2001) for a thorough introduction, or (Domingos 2012) for a more informal description on how they are used in practice.

2.1 *A formal definition of machine learning*

Although the field of ML is often defined in cognitive terms, as giving computers the ability to learn without being programmed, a more formal definition often cited is the one by Mitchell (1997). "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." Further, ML is generally classified into two categories called supervised and unsupervised learning.

Supervised learning: For some unknown relationship between variables $\mathbf{x} \rightarrow \mathbf{y}(\mathbf{x})$ where N pairs of data are observed $\left\{\left(\mathbf{x}_i, \mathbf{y}_i\right)\right\}_{i=1}^{N}$, the goal is to estimate $\mathbf{y}^*$ for unobserved $\mathbf{x}^*$. Here, the experience E is the observed data, the task T is to predict $\mathbf{y}^*$ and P is usually a measure of the difference between the predicted $\mathbf{y}^*$ and the true value $\mathbf{y}(\mathbf{x}^*)$.

Unsupervised learning: The goal is to discover patterns in unlabeled data. For instance, based only on the x-values in the above example, $\left\{\mathbf{x}_i\right\}_{i=1}^{N}$, estimate the distribution over the data or identify clusters, etc. With the task T of clustering in mind, the experience E is still the observed data and P might be a measure on cluster compactness or separability.

2.2 *Machine learning and statistical modelling*

The increasing popularity of ML today may be credited to recent advancements within computer science, although most of supervised and unsupervised learning have roots in traditional statistical methods. An overview is illustrated in Figure 1. *Supervised learning* is based on *prior* knowledge, and covers regression and classification (discriminant analysis).

Regression considers a continuous dependent variable represented by a model function fitted to data. Assumptions are generally made about the data generation process, e.g. homoscedasticity (equal finite

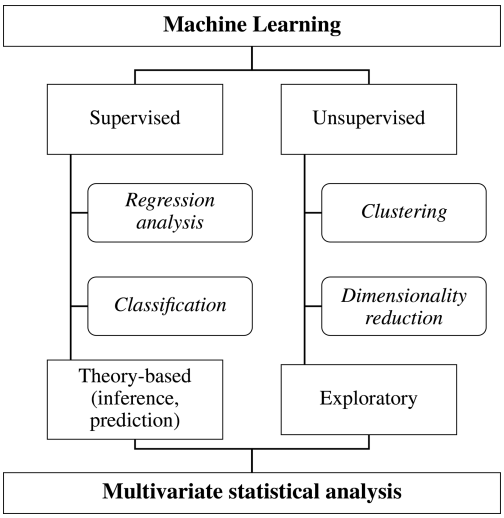


Figure 1. Machine learning vs. multivariate statistics.

variance), independence and normality. The earliest form of regression was the method of least squares, published by Legendre (1805) and Gauss (1809).

Classification or discriminant analysis has the same objective as regression, but where the dependent variable is discrete, and typically comes in a categorical form as labels from a finite set. Some of the earliest work was by Fisher (1936) leading to Fisher's linear discriminant function. Other popular methods are Logistic regression that dates back to Verhulst (1838), decision trees (Morgan & A. Sonquist 1963) and k-nearest neighbors (Cover & Hart 1967).

Note that due to the close link between these two categories of supervised learning problems, regression models may be altered to work for discriminant analysis and vice versa.

Unsupervised learning is related to data exploration problems. The main exploratory methods are often classified as clustering or dimensionality reduction.

Clustering analysis considers the task of grouping objects into sets (clusters) such that objects in the same set are more similar to each other than to those in other sets. No precise definition of a cluster exists, and this is one of the reasons why there are so many different clustering algorithms (Estivill-Castro 2002). Some popular alternatives representing different approaches to the problem of clustering are k-means (MacQueen 1967), hierarchical clustering (Sibson 1973), Gaussian mixture models using expectation-maximization (Dempster, Laird, & Rubin 1977) and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester, Kriegel, Sander, & Xu 1996).

Dimensionality reduction is the procedure of reducing the number of input variables to a smaller set of principal variables. One fundamental approaches is principal component analysis (Pearson 1901), developed by Karl Pearson, considered the father of modern statistics.

The methods mentioned above is by no means a complete list, and the references cited are meant to give indications on early work on the different subjects as true origins are often debatable and outside the scope of this paper. A lot of research has gone into extending or improving on these methods since first invented, which has spawned the large variety of models and algorithms used in ML today. There are also other popular methods and widely used techniques within ML that can be placed in more than one category in Figure 1, e.g. artificial neural networks dating back to Hebb (1950) and using the backpropagation algorithm developed by Werbos in 1974.

2.3 *Model design vs. trial-and-error*

Many of the similarities between statistics and ML are just hidden behind different terminology. For instance, where we in ML refer to *features* in a model and that model *weights* are *learned* from data, a statistician would refer to *variables* in a model where data is used to *estimate* model *parameters*. But there are also some important differences in how the (same) models are used, and whether the main emphasis is on designing a good model or obtaining good prediction by trial-and-error.

In classical statistics, the focus is often on testing hypothesis of causes and effects and interpretability of the models applied. A common aphorism in statistics is that "All models are wrong, but some are useful". The analyst should know when the model will break, how it breaks, and if one can still use it anyways. The main goal is understanding the underlying mechanisms that drive the things we observe.

On the contrary, ML has more focus on predictive accuracy of models, with less attention towards model interpretation. Model selection is often based on trial-and-error through cross validation to evaluate goodness-of-fit criteria, where prediction accuracy is evaluated on data that was excluded in the learning process. Although the main goal is to obtain a good prediction, special considerations are made based on what the prediction is used for in a given application. For high risk applications for instance, false positives may be far worse than false negatives (or vice versa), and the model optimization can be weighted accordingly. Still, it is based on observing the desired behavior in future or excluded data.

Although science always has been concerned with both prediction and explanation, the different

philosophies have resulted in much debate between the statistics and machine learning communities. See for instance the discussion in Breiman (2001).

3 ML FOR HIGH RISK ENGINEERING APPLICATIONS

This section highlights some of the main challenges when using ML for high-risk and low probability scenarios. We continue by proposing recommendations to consider in ML model development to address these challenges.

3.1 *Tail events in high risk environments*

Three of the main challenges with ML applied to high risk and low probability scenarios are related to the following:

- **High risk reduces the tolerance for wrong predictions**. The consequence might be catastrophic.
- **Critical consequences often relate to tail events—for which data is naturally scarce**. This increases uncertainty and reduces the accuracy of predictions.
- **The ML models that are able to fit the data well are often opaque.** This makes the model less falsifiable, increasing uncertainty and reducing decision makers' ability to trust the model.

The first point relates to the decisions that are made based on model predictions. When the high risk is associated with a catastrophic consequence, the tolerance for a wrong prediction is clearly low. Moreover, severe consequences are often related to a rare event. This introduces additional uncertainty as data is scarce[2]. These first two points are in direct contrast to the typical ML applications today (e.g. non-consequential recommendation engines). In this respect, assurance of safety critical systems relying on ML is also recieving increasing attention, see e.g. Brandsæter & Knutsen (In press).

The last point on model opaqueness (lack of transparency) relates to quantification of model discrepancy and the ability to falsify models that do not comply with observations. The added uncertainty from model opaqueness may at first seem purely subjective. However, as we will see in the following section, addressing this by increasing model transparency may relax requirements on accuracy and vice versa.

[2]Scarce in the sense that the size of data is small compared to the number of relevant dimensions. This is because the event under consideration, e.g. structural failure, is rare and expensive to approximate by experiments.

3.2 *Recommendations on model development*

The challenges stated above will make any ML or statistics based model prone to erroneous, and potentially dangerous, use. However, many of the most common pitfalls in using such correlation based or data-driven methods may be avoided by proper model selection and design. To increase confidence in the safe use of ML models we propose some recommendations in this section.

For supervised learning, and when we want to use ML for prediction in general engineering applications, one should seek to adopt or develop models with the following attributes:

1. **Flexibility:** The models ability to represent a large class of functions.
2. **Constrainability:** The ability to impose model constraints based on phenomenological knowledge.
3. **Probabilistic inference:** Probabilistic representation of model output defined on the entire model range (i.e. the model output is represented by a distribution).

The first two concepts relate to the problem of underfitting and overfitting respectively. The third is important when the model is used in assessment of risk and reliability.

From an engineering perspective, **flexibility** is needed to capture the behavior of complex physical systems and their response. Most non-parametric models fulfill this requirement. The definition of "a large class of functions" in this context may not be precise, as it certainly depends on the problem at hand. A typical example may be all continuous or differentiable functions with a finite number of discontinuities.

Constrainability is beneficial for two main reasons. First, it reduces the possibility of overfitting the data, thus increasing the robustness and the performance for application on future data. This is particularly important for tail behavior problems where data is scarce. The second reason is that imposing constraints based on phenomenological knowledge reduces model opaqueness, i.e. increases transparency.

By a *transparent model* we mean that the relationship between model inputs and outputs can be meaningfully understood by humans, and that model characteristic properties and limitations of the model can be understood without explicit numerical computation. The most straightforward example is simple linear regression with linear basis, i.e. fitting a line. In contrast, for an opaque model, the model behavior can only be investigated through computation. These models are often referred to as black boxes, where the only way to fully characterize the model's properties is through

exhaustive computation; evaluating the model for *all* possible inputs.

The scientific method is based on the principle that any model, or hypothesis, should be falsifiable. All models, including ML models, are based on a set of assumptions. For a model or an assumption to be falsifiable, it must, in principle, be possible to make an observation that would show the assumption to be false. Thus, model transparency is important as it enables one way of falsification. ML models are typically falsified by observing poor accuracy. That is, observations are made that differ significantly from model predictions. However, understanding why such discrepancies occur is often difficult in an opaque model. If we assume that the observation is not erroneous (observed discrepancy is not related to noise), then the root cause might either be lack of relevant data, i.e. the model is built on too few datapoints close to the observed discrepancy—in which case a larger degree of uncertainty is expected. Or the root cause is related to violation of the underlying model assumptions—and the model must be changed.

In order to develop useful ML models for high risk applications, a compromise usually has to be made between model transparency and flexibility. To counteract the negative tradeoff of an initially opaque, but flexible model, we might impose constraints based on phenomenological knowledge related to causality. This can be thought of as "putting the black box inside a white box", i.e. enabling deduction of bounding model properties through the imposed constraints.

Figure 2 illustrates constraints in the form of boundedness, monotonicity and convexity. Three datapoints have been observed, and for simplicity we assume the data does not contain noise. This means that we are looking for an interpolation model, a function passing through all three points. The shaded areas show where the function cannot enter due to the imposed constraints. Hence, starting with the space of all functions passing through the three points, the constraints reduce the space of possible interpolation functions. Assuming that the constraints are based on phenomenological knowledge that hold in reality, the constrained models are less prone to overfitting (more robust). In the case where an observation is made within the shaded area, the model is falsified immediately as the assumptions behind the constraints do not conform with an observed outcome. Hence, by imposing constraints based on phenomenological knowledge, either a) performance is increased, or b) the model is falsified and the modeler learn something fundamentally new about the phenomenon studied, which can be applied in future modelling.

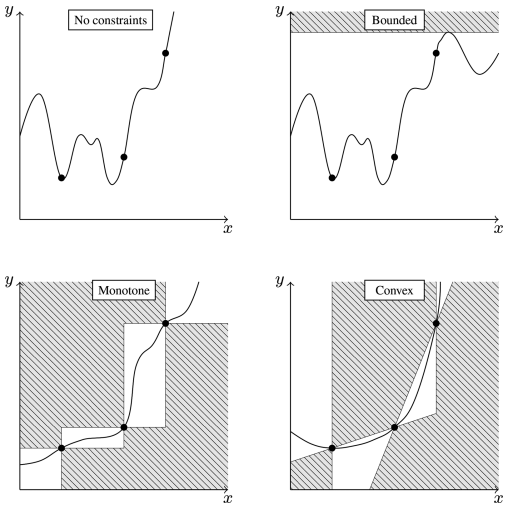


Figure 2. Effect of some different constraints for interpolation in data without noise. The interpolation function cannot enter the shaded areas.

The constrained models shown in Figure 2 may be restricted further by imposing multiple constraints, e.g boundedness and monotonicity or monotonicity and convexity. Note also that for noisy data, this means interpreting constraints in terms of probabilities using the assumed distribution of noise. The example is motivated by the more general class of constraints in the form of partial differential inequalities, for which phenomenological knowledge related to causal effects in various physical phenomena may often be available.

Practically, imposing constraints such as the ones illustrated in Figure 2 means translating the phenomenological constraints to constraints in the ML optimization algorithm. Many techniques exist for including constraints in ML through optimization, usually in order to obtain regularization effects, but we emphasize that developing the necessary links between these constraints and phenomenological knowledge will be highly beneficial. See for instance (Yu 2007) or (Maatouk & Bay 2017) for some examples and further discussion.

Probabilistic inference on the ML model output is needed for risk and reliability analysis applications. This means that the model output should optimally be in the form of a distribution. Model predictions in the form of fixed values and best estimates are not applicable. The dangers of expressing risk through expected values is well known, and modern definitions of risk usually relate to the distribution over possible outcomes. Hence, some quantification of prediction uncertainty is essential. It should be noted that this goes beyond the probabilities often used to report model accuracy for ML classifiers. The fact that an object is correctly classified 99% of the time might be insignificant if the outcomes of the remaining 1% is associated with severe consequences.

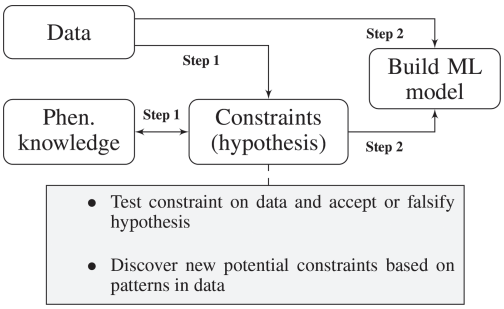


Figure 3. ML workflow with emphasis on constraints.

There is a traditional approach to this problem, where ML has played a role in mathematical models for calculation of risk and reliability. First, the models are scrutinized through human quality control to ensure that the model accuracy is sufficient or at least on the conservative side. This quickly becomes infeasible for higher dimensional models. Further, a single distribution representing the model uncertainty is often established from statistical analysis of prediction accuracy alone, assuming uniformly distributed data. This is no longer feasible for higher dimensional models or when the input data is far from uniformly distributed.

3.3 *Working with constraints*

Figure 3 illustrates of the workflow for building ML models with emphasis on constraints. For simplicity we ignore work on data cleaning and feature selection that naturally comes prior to model development.

Any hypothesis on model constraints coming from assumed causalities in the phenomenon under consideration must be tested to identify to what degree they hold in the observed data. Hence, the task of hypothesis testing is emphasized. There might also be valid constraints that are not immediately identifiable from phenomenological knowledge. It could therefore be valuable to search for possible constraints by unsupervised learning, to serve as hints to the modeler, and help identifying additional constraints before further testing and possible inclusion in the ML model.

This type of workflow will move the typical approach for building ML models today closer to traditional statistical modelling.

4 RELEVANT AREA OF APPLICATION – SRA

In this section we give a concrete example of an application area where ML is linked with engineering risk analysis – Structural Reliability Analysis (SRA), where the recommendations given Section 3 are highly relevant. In addition, we highlight two general pitfalls related to a common, but possibly misconceived, idea on how ML may be used in this context.

4.1 *Structural reliability analysis*

Structural reliability analysis, or SRA for short, is the fundamental building block of modern risk-based engineering methodologies. For a thorough introduction reference is made to Madsen, Krenk, & Lind (2006). The underlying theory combines structural analysis with statistics and probabilistic modelling to assess uncertainties of information that contributes to the probability of structural failure.

SRA may generally be described as the problem of establishing the probability

$$P\big(G(\mathbf{x}) \leq 0\big) \tag{1}$$

where $\mathbf{x}$ is a vector of stochastic variables, e.g. structural dimensions, material properties, loads and model uncertainties. The function $G(\mathbf{x})$ is referred to as the limit state, and is defined such that $G(\mathbf{x}) \leq 0$ if and only if the scenario represented by $\mathbf{x}$ results in structural failure, see Figure 4. In the literature the limit state is often presented as

$$G(\mathbf{x}) = R(\mathbf{x}) - L(\mathbf{x}) \tag{2}$$

where $R(\mathbf{x})$ and $L(\mathbf{x})$ represent the structural resistance, or capacity, and load effect respectively, although these are often not separable in practice.

The main tasks in a structural reliability analysis is to establish a suitable limit state function and distributions of all the input parameters, so that one may estimate the failure probability given by Eq. 1 and analyse the sensitivity of parameters and

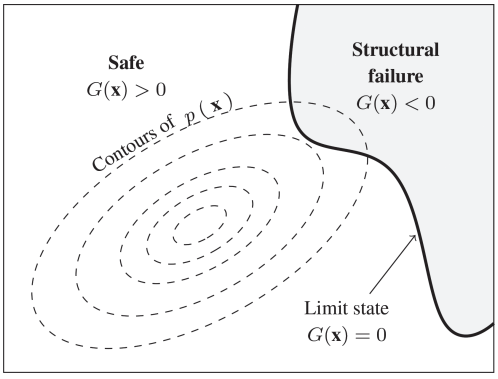


Figure 4. Illustration of SRA in two dimensions.

decisions that will affect the system. Informally, one might say that we use data and domain knowledge to establish the input distributions describing the current state of the system, and extrapolate to states where the system has failed using very limited data related to system failure in combination with advanced phenomenological simulations based on first principle physics.

4.2 *Machine learning in SRA*

Some of the key challenges in SRA today relate to the rapid increase in structural complexity of engineering systems, including more automation and software intensive control systems, together with a demand for higher system utilization and the need for more accurate and reliable models to support decision making under uncertainty. At the same time, ubiquitous sensor data provides new information that could potentially reduce the uncertainty if the information could be incorporated into the models.

In practice, this means that the function $G(\mathbf{x})$ in Eq. 1 and the distribution over the input space $\mathbf{x}$ will take a more complicated form. To cope with this, the technologies we now label ML can be useful, e.g. to address the following problems:

- Find $\mathbf{x}$: Establish distribution over $\mathbf{x}$ using all relevant data.
- Find G: Through data related to structural behavior, combined with data from past experience, experiments and simulations of structural failure, establish the limit state function that classifies all $\mathbf{x}$'s as *Safe* or *Failure*.

Note that in practice these two problems are generally intertwined, in the sense that inference about a model parameter (the $\mathbf{x}$'es) may only be observable through its effect on the system response. E.g. some $y(\mathbf{x})$ is observed, where the mapping $y(\cdot)$ is in our representation baked into the general function G.

The use of ML in SRA has traditionally been confined to smaller subcomponents where human quality control is possible, but for more complex systems this quickly becomes infeasible. The more general task of approximating functions like the limit state $G(\mathbf{x})$ using ML together with a limited number of realisations (experiments or numerical simulations) has received increasing attention over the last decades, within the field of Uncertainty Quantification (UQ). See for instance (Sullivan 2015). UQ aim to quantify the ML uncertainty introduced through approximation, as well as how uncertainty in input parameters propagates through such models.

4.3 *General ML pitfalls in SRA*

As for all ML applications, there are some general pitfalls to look out for. This section highlights some of the challenges related to how introduction of ML in reliability analysis is often depicted. This relates to a growing appetite for ideas like the following

- Due to the increasing instrumentation of systems, more data is available about the loads and structural behavior of systems at any time. By combining this with historical data from many other similar systems where the structural integrity is known (we know whether or not they have failed, and how), we could detect any abnormal behavior. With this information we could create warnings before potential failure occurs, and possibly also help the system back into normal operation.

Anomaly detection will probably play a larger role in risk assessment in the future, but there are some pitfalls that the industry needs to be aware of. The first relates to the quantification of the safety margin, which is often represented as a probability of failure or through some other metric relating the current physical condition with conditions corresponding to structural failure.

- For complex engineering systems, quantifying the margin of safety based on operational data alone is unlikely.

This statement might be obvious, from the many different ways a system may fail in practice and the assumption that these systems are designed not to fail. The next argument however is a bit more subtle

- From a data exploration perspective, when observing system states outside normal operation one might unknowingly have transitioned away from the default system behavior, leaving all previous observations biased, and possibly irrelevant.

This statement impacts the basic assumption in ML that future data will come from the same distribution as the data the model was trained on. This is illustrated by an example in Figure 5.

Following the SRA setting illustrated in Figure 4, we assume that the limit state is defined in terms of material over-utilization. Often the criteria for when ultimate failure occurs is difficult to express mathematically, and conservative approaches are applied by defining failure as some identifiable prior event. One such limit from material science is the yielding criteria of ductile materials such as steel. The stress-strain relationship of a material under some loading is linear up until the yielding point, and the material behaves elastic in this region. I.e. unloading the material will bring it back to its original unharmed state[3]. For continued loading beyond the yield point, the material will exhibit plastic behavior until rupture.

[3]Ignoring other failure modes such as fatigue.

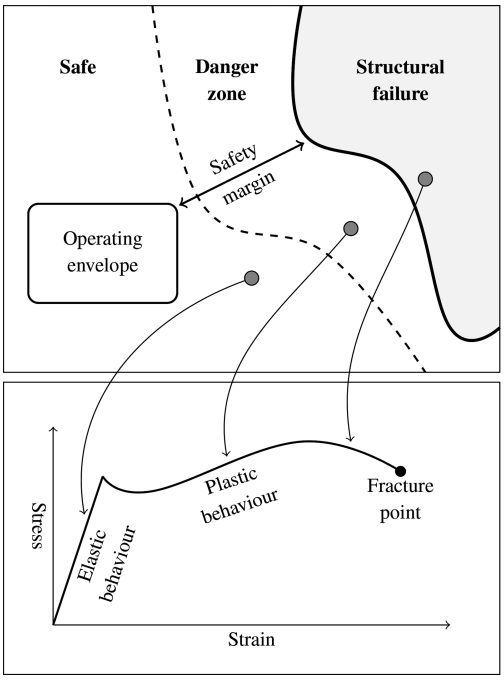


Figure 5. Illustration of structural response in different operating environments.

A limit state defined from the yield criterion can be illustrated as the dashed line in Figure 5, whereas the boundary of the structural failure set (black line) represents material fracture[4]. Imagine an anomaly detection agent that warns whenever the system behaves outside the normal operating envelope, and estimates procedures for moving the system back into normal operation. For elastic material response (leftmost point), the data used to train the model is still valid. But when materials are pushed closer to their limits, some properties are fundamentally changed. The elastic limit will change due to work hardening, and the stress-strain curve is no longer valid. Furthermore, when the material is over-utilized over a certain threshold, the reduction of load may initiate failure modes previously non-relevant, and uninformed decisions may be catastrophic.

This example is an oversimplification, but illustrates some challenges with introducing purely data driven agents. Due to the increased computational capacities and scientific models available today there is an increased push to utilise systems closer to their limits. In the above example this means allowing operation closer to the true failure limit, and compensating by increased control, uncertainty reduction, and more detailed understanding of the failure modes.

[4]For load controlled scenarios the top of the stress-strain curve represents maximum capacity. Unless the load is decreased the material will eventually fracture

5 CONCLUDING REMARKS

The field of ML is largely based on statistical methods, but with a focus that is shifted more towards predictive accuracy and with limited attention towards model interpretation and testing hypotheses on causes and effects.

For tail events in high risk environments the modeler is faced with additional challenges, as the tolerance for error is reduced and accuracy is needed in distribution tails rather than where the main bulk of data is. Because of this, opaque black-box type models are difficult to work with as the means for falsification may be limited to observations of future performance.

Therefore, research and development of ML models for such applications should be guided towards enabling incorporation of causality constraints reflecting the modeler's phenomenological knowledge.

REFERENCES

Amodei, D., C. Olah, J. Steinhardt, P. Christiano, J. Schulman, & D. Mané (2016). Concrete problems in AI safety. *CoRR abs/1606.06565*.

Brandsæter, A. & K.E. Knutsen (in press). Towards a framework for assurance of autonomous navigation systems in the maritime industry. European Safety and Reliability Conference, ESREL 2018.

Breiman, L. (2001, 08). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *16*.

Cover, T. & P. Hart (1967, January). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory 13*(1), 21–27.

Dempster, A.P., N.M. Laird, & D.B. Rubin (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological) 39*(1), 1–38.

Domingos, P. (2012, October). A few useful things to know about machine learning. *Commun. ACM 55*(10), 78–87.

Dopico, M., A. Gomez, D. De la Fuente, N. García, R. Rosillo, & J. Puche (2016). A vision of industry 4.0 from an artificial intelligence point of view. In *Proceedings on the International Conference on Artificial Intelligence (ICAI)*, pp. 407. WorldComp.

Ester, M., H.-P. Kriegel, J. Sander, & X. Xu (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. pp. 226–231. AAAI Press.

Estivill-Castro, V. (2002, June). Why so many clustering algorithms: A position paper. *SIGKDD Explor. Newsl. 4*(1), 65–75.

Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics 7*(2), 179–188.

Gauss, C. (1809). Theoria motus corporum coelestium in sectionibus conicis solem ambientum.

Hastie, T., R. Tibshirani, & J. Friedman (2001). *The Elements of Statistical Learning*. Springer.

Hebb, D.O. (1950). The organization of behavior: A neuropsychological theory. *Science Education 34*(5), 336–337.

Knight, W. (2017). The dark secret at the heart of AI. *MIT Technology Review 120*(3), 55–63.

Legendre, A. (1805). *Nouvelles méthodes pour la détermination des orbites des comètes*. Nineteenth Century Collections Online (NCCO): Science, Technology, and Medicine: 1780–1925. F. Didot.

Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. *Journal of Industrial Information Integration 6*(Supplement C), 1–10.

Maatouk, H. & X. Bay (2017, Jul). Gaussian process emulators for computer experiments with inequality constraints. *Mathematical Geosciences 49*(5), 557–582.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, Berkeley, Calif., pp. 281–297. University of California Press.

Madsen, H., S. Krenk, & N. Lind (2006). *Methods of Structural Safety*. Dover Civil and Mechanical Engineering Series. Dover Publications.

Mitchell, T.M. (1997). *Machine Learning* (1 ed.). New York, NY, USA: McGraw-Hill, Inc.

Morgan, J. & J.A. Sonquist (1963, 06). Problems in the analysis of survey data and a proposal. *58*, 415–434.

Pearl, J. (2010, Sep). The mathematics of causal relations. *In P.E. Shrout, K. Keyes, and K. Ornstein (Eds.), Causality and Psychopathology: Finding the Determinants of Disorders and their Cures*, 47–65.

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2*(11), 559–572.

Sibson, R. (1973, 01). Slink: An optimally efficient algorithm for the single-link cluster method. *16*.

Sullivan, T. (2015). *Introduction to Uncertainty Quantification*. Texts in Applied Mathematics. Springer International Publishing.

Verhulst, P.F. (1838). Notice sur la loi que la population suit dans son accroissement. *Curr. Math. Phys 10*, 113–120.

Werbos, P. (1974, 01). Beyond regression: new tools for prediction and analysis in the behavioral sciences.

Yu, T. (2007). *Incorporating prior domain knowledge into inductive machine learning: its implementation in contemporary capital markets*. Ph. D. thesis, University of Tech., Sydney. Faculty of Information Technology.

Appendix B

Source code

Two open source python packages have been developed in the project:

GPconstr Python package for Gaussian process regression with constraints used in Paper I. It is available at https://github.com/cagrell/gp_constr.

HAL The HAL (Hierarchical Active Learning) python package contains various modules needed for structural reliability analysis and design of experiments, and was used for the numerical examples in Paper III and Paper IV. It is available at https://github.com/cagrell/HAL.