Metamodelling for geotechnical reliability analysis with noisy and incomplete models

van den Eijnden, A. P.; Schweckendiek, T.; Hicks, M. A.

DOI
10.1080/17499518.2021.1952611

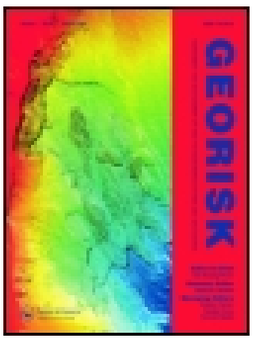
Publication date
2021

Document Version
Final published version

Published in
Georisk

Citation (APA)
van den Eijnden, A. P., Schweckendiek, T., & Hicks, M. A. (2021). Metamodelling for geotechnical reliability analysis with noisy and incomplete models. *Georisk*, *16*(3), 518-535. https://doi.org/10.1080/17499518.2021.1952611

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.



Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ngrk20

Metamodelling for geotechnical reliability analysis with noisy and incomplete models

A. P. van den Eijnden, T. Schweckendiek & M. A. Hicks

To cite this article: A. P. van den Eijnden, T. Schweckendiek & M. A. Hicks (2021): Metamodelling for geotechnical reliability analysis with noisy and incomplete models, Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards, DOI: 10.1080/17499518.2021.1952611

To link to this article: https://doi.org/10.1080/17499518.2021.1952611

Published online: 30 Jul 2021.

GEORISK
https://doi.org/10.1080/17499518.2021.1952611

Metamodelling for geotechnical reliability analysis with noisy and incomplete models

A. P. van den Eijnden [a,b], T. Schweckendiek [a,c] and M. A. Hicks [b]

[a]Deltares, Geo Engineering Unit, Delft, Netherlands; [b]Department of Geoscience and Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, Netherlands; [c]Department of Hydraulic Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Delft, Netherlands

ABSTRACT

A kriging-based metamodelling approach for analysing the structural reliability of a sheetpile wall in a dyke is formulated. This specific problem is characterised by high target reliabilities ($P_f \sim 10^{-7}$) in combination with a noisy and incomplete numerical model response. Starting from the original formulation of active learning kriging-based Monte Carlo simulation (AK-MCS), a robust two-stage metamodel framework is formulated in combination with adaptive multiple importance sampling, Gaussian process classification and kernel enhancements. Learning functions and convergence criteria are revised to maintain consistency with the metamodel enhancements. The developed metamodel is applied in the reliability analysis of a soil-structure interaction problem involving a sheetpile wall in a dyke body, which is representative of a class of problems encountered in engineering practice. Low dimensional example studies demonstrate the workings of the model and give insight into the model response. Full probabilistic analyses are then performed to estimate the probabilities of structural failure in a reliability updating context. The results show that after several necessary enhancements of the classical formulations, metamodelling approaches can be used successfully in combination with noisy and incomplete computational models as are often encountered in geotechnical engineering practice.

ARTICLE HISTORY
Received 1 October 2020
Accepted 2 June 2021

KEYWORDS
Geotechnical reliability; kriging-based metamodelling; noisy and incomplete limit state functions; sheetpile wall in dyke

1. Introduction

Reliability analysis of geotechnical structures enables cost-effective design and is particularly attractive for the assessment of existing structures, as it allows for incorporating performance observations and monitoring data. Despite recent advances in reliability analysis methods, applications in real-life designs and assessments remain scarce and challenging. The approach proposed and demonstrated in this article aims to improve the computational robustness and efficiency for a class of typical geotechnical problems encountered in practice, involving (moderately) complex soil-structure interaction such as sheetpile retaining walls. For such problems:

- numerical modelling (e.g. finite element modelling) is used in evaluating the occurrence of a limit state;
- the number of relevant random variables is approximately 10 to 20;
- the target reliability index is in the range of $3 \lesssim \beta_T \lesssim 6$ (implying target probabilities of failure in the range of $10^{-9} \lesssim P_{f,T} \lesssim 10^{-3}$); take for example the Eurocode reliability classes (EN:1997 2004) or the Dutch safety standards for flood defenses (Schweckendiek et al. 2013).

The low target probabilities of failure, combined with computation times of the order of minutes for numerical models, make analyses with crude Monte Carlo (MC) practically infeasible. Also, importance sampling-based approaches still require long computation times for this combination. Furthermore, numerical modelling implies two additional challenges for reliability analysis: (a) numerical noise in the performance function can lead to convergence problems (e.g. in the case of gradient-based FORM solvers) and (b) certain parameter combinations (or better subregions of the joint probability distribution) may not result in equilibrium and, hence, lead to non-convergence of the computational model of the performance function. The aim of this paper is to propose an approach that is robust and computationally efficient under these conditions. In practical terms, this means that the proposed algorithm should converge to the

CONTACT A. P. van den Eijnden a.p.vandeneijnden@tudelft.nl Department of Geoscience and Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology, Stevinweg 1, 2628 CN Delft, Netherlands

correct solution with a high degree of confidence, with total computation times not exceeding several hours, which seems a reasonable requirement for practical engineering application.

Reliability analysis aims at calculating the probability of failure (not) to occur, with failure defined as not meeting a certain performance criterion. Such a criterion is usually reformulated in a performance function $g(\vec{x})$, such that $g(\vec{x}) \leq 0$ indicates failure. When $\vec{x}$ is a realisation of a stochastic multivariate $\vec{X}$, characterised by a joint probability density function (PDF) $p_{\vec{X}}(\vec{x})$, the probability of failure is defined as

$$P_{\mathrm{f}} = \int_{\vec{x} \in \mathbb{R}^n} I_{\mathrm{f}}(\vec{x}) p_{\vec{X}}(\vec{x}) \, \mathrm{d}\vec{x} \tag{1}$$

where $I_{\mathrm{f}}(\vec{x})$ is the failure indicator function, which is equal to 1 when $g(\vec{x}) \leq 0$ and 0 when $g(\vec{x}) > 0$. Reliability analysis involves solving this integral, for which a closed-form solution is generally not available.

Numerical strategies for solving Equation (1) require evaluating $I_{\mathrm{f}}(\vec{x})$, which can be computationally expensive, and rigorous solution by MC integration is often infeasible. Indeed, due to the computational expense of numerical modelling software, this is also generally true for other sampling-based strategies that use variance reduction methods, such as importance sampling, directional sampling (Melchers and Beck 2018) and subset simulation (Au and Beck 2001). Approximate methods such as FORM (Hasofer and Lind 1974) can be extremely efficient and often yield sufficient accuracy in practical applications but can have great difficulties with noise in the model response, which hinders the (gradient-based) search algorithm needed to find the design point. Moreover, limit state functions with strong non-linearities in multiple parameter-dimensions can be difficult to capture based on a single design point.

Metamodelling (surrogate modelling) strategies have been proposed to tackle both the computational expense and, to some extent, the non-linearity of the computational model response for slope stability problems (see Li et al. (2016) for an overview), tunnel excavation (Mollon, Dias, and Soubra 2009) and foundation footings (Sivakumar Babu and Srivastava 2007). In this type of approach, the model performance function $g(\vec{x})$ is replaced by an approximate model $\hat{g}(\vec{x})$ that provides a model prediction in parameter space based on a concise set of model evaluations. This metamodel can then be used as a proxy for the true model response to make predictions of the reliability of the structure. There are different approaches to formulating the metamodel model, such as response surfaces, polynomial chaos expansion (PCE), support vector machines (SVM), Gaussian processes (GP) and artificial neural networks (ANN). The reader is referred to Teixeira, Nogal, and O'Connor (2021), and the references therein, for an in-depth evaluation and comparison of the different methodologies in the context of reliability analysis for design.

A Gaussian process, in geotechnical engineering better known as a kriging model (Cressie 1993), is one type of model that can be used as a metamodel. The method is particularly well-suited for strong non-linearities (Teixeira, Nogal, and O'Connor 2021), giving it the advantage over classical polynomial response functions and polynomial chaos expansion when dealing with a strongly non-linear and noisy model response. In addition, the method is kernel-based and provides prediction uncertainty in a natural way, which is essential in the formulation of efficient strategies to improve the metamodel. To this end, Echard, Gayton, and Lemaire (2011) linked Gaussian process metamodelling to MC integration for reliability analysis, outlining an iterative scheme for optimal sequential selection of new samples for model evaluations, known as the "Active learning reliability method combining Kriging and Monte Carlo Simulation" (AK-MCS). This scheme forms the blueprint for a series of active learning schemes for reliability analyses, and is used as the starting point of this work.

Many variations of AK-MCS have been proposed, changing the underlying kriging metamodel by, for example, universal or polynomial chaos kriging (Schöbi, Sudret, and Marelli 2016) or a support vector machine-based metamodel (Bourinet 2016). Other variations use alternative learning functions (Hu and Mahadevan 2016), convergence criteria (Schöbi, Sudret, and Marelli 2016) and/or domain integration approaches based on subset simulation (Dubourg, Sudret, and Bourinet 2011), importance sampling (Dubourg, Sudret, and Deheeger 2013; Balesdent, Morio, and Marzat 2013; Cadini, Santos, and Zio 2014), or line sampling (Depina et al. 2016). Applications of kriging-based metamodelling in geotechnical engineering include limit equilibrium slope stability problems (Kang et al. 2015), finite element strip footing problems on spatially variable soils (Al-Bittar, Soubra, and Thajeel 2018; Soubra et al. 2019) and monopile foundations (Depina et al. 2016).

In this work, a complex and strongly non-linear soil-structure interaction model for analysing the reliability of a sheetpile wall in a dyke is studied. The specific challenges for its successful evaluation are the noisy and incomplete structural response of the model, in combination with very small target probabilities of failure $P_{\mathrm{f}} < 10^{-6}$. Although the noisy aspect of numerical methods in geotechnical engineering is usually only

accounted for as part of model uncertainty factors in design (e.g. Phoon and Tang 2019) and has received limited attention in advanced reliability analysis (Teixeira, Nogal, and O'Connor 2021), it is naturally included in kriging-based metamodelling (Rasmussen and Williams 2006). Incomplete model response refers to the computational model not having a solution for some combinations of the input parameters. Not only does this represent scenarios that may be unrealistic and for which distribution and probability updating may be necessary, an incomplete model response may also interfere with the active learning algorithm. The impact of incomplete model response has only received attention in the context of model response missing at random (Forrester, Sobester, and Keane 2008). In this work, an incomplete model response with a physically tractable origin is addressed. Its detrimental effects on the active learning algorithm are alleviated and the resulting conditional probabilities are interpreted in the context of reliability updating.

To tackle the challenges of small probabilities in combination with noisy and incomplete model response, several enhancements of the classical formulation of AK-MCS are combined in this paper. In Section 2, the theoretical basis of kriging-based metamodelling is presented as a summary of the relevant state of art, after which the specific enhancements of AK-MCS for dealing with noisy and incomplete geotechnical models are discussed in Section 3. These enhancements are novel contributions in the form of problem-specific variations and improvements on the existing metamodelling techniques. Specifically, these novel contributions are a two-stage metamodelling approach, an improved learning function accounting for random noise, and a procedure to account for multiple design points for multiple importance sampling. In Section 4, the method is applied on the sheet pile wall problem in a series of simulations. These simulations serve to (1) demonstrate the workings of the adapted two-stage AK-MCS scheme, (2) perform the reliability analysis of the sheetpile problem at hand, and (3) extend the reliability analysis with reliability updating by implicit distribution updating. The latter can be seen as a way of incorporating the information of past survival in the assessment of an existing structure.

2. Gaussian process regression for metamodelling and reliability analysis

2.1. Gaussian process modelling

The response of a physical or computational model $\mathcal{F}(\vec{x})$ can be approximated by a numerical or analytical metamodel $y(\vec{x}) \approx \mathcal{F}(\vec{x})$. Such a metamodel $y(\vec{x})$ is defined as a Gaussian process (GP) model if (Santner et al. 2003) "for any $L \geq 1$ and any choice of $\vec{x}_1, \ldots, \vec{x}_L$, [...] the vector $(y(\vec{x}_1), \ldots, y(\vec{x}_L))$ has a multivariate normal distribution". Accordingly, $y(\vec{x})$ is here written as a GP, defined by its mean function, $m(\vec{x}) = \mathbb{E}[y(\vec{x})]$ and its covariance function, or kernel, $k(\vec{x}, \vec{x'} \mid \vec{\theta}) = \mathbb{E}[(y(\vec{x}) - m(\vec{x}))(y(\vec{x'}) - m(\vec{x'}))]$, such that the GP at a finite number of locations $\mathbf{x}$ is given by:

$$y(\mathbf{x}) = m(\mathbf{x}) + \mathbf{K}^{1/2}\vec{\xi} \tag{2}$$

with kernel matrix $\mathbf{K} = k(\mathbf{x}, \mathbf{x}|\vec{\theta})$ and standard normal multivariate $\vec{\xi} \sim \mathcal{N}(\vec{0}, \mathbf{I})$. Hyperparameters $\vec{\theta}$ are internal parameters that define the shape of the kernel.

In the context of reliability analysis, variables $\vec{x}$ are realisations of stochastic variables $\vec{X}$ with a joint probability distribution. For convenience, the GP is formulated in standard normal space with uncorrelated variables $\vec{U} = T(\vec{X})$, where T is a transformation mapping from parameter space to standard normal space. Moreover, the model response is expressed in terms of a performance function $g(\vec{u})$ and is described by the GP metamodel such that $y(\vec{u}) \approx g(\vec{u})$.

Although the trend of the performance function can be included in the general formulation of universal kriging (Cressie 1993; Sacks et al. 1989; Santner et al. 2003), simple kriging is used here, such that $m(\vec{u}) = 0$. This fits well with the application of metamodelling on the performance function, in which the final goal is a classification into failing and non-failing realisations (i.e. the sign of the performance function). The bias of simple kriging towards the prior mean (i.e. 0) thereby introduces a conservative estimate of the prediction uncertainty, with the metamodel regressing to g=0 in regions with no training data. As such, the prediction of the performance function by the metamodel $\hat{g}(\vec{u})$ is used to solve the integral of Equation (1) by MC integration over a sample set $\mathbf{u}_{\mathrm{MC}}$. The performance function metamodel is thus formulated as a Gaussian process with Gaussian prior $\vec{g} = g(\mathbf{u}) \sim \mathcal{N}(\vec{0}, k(\mathbf{u}, \mathbf{u}|\vec{\theta}))$. When a certain number of data on $g(\vec{u})$ are known, splitting the vector $\vec{g}$ into known (training) data $\vec{g}_{\mathrm{t}}$ and unknown (predicted) data $\vec{g}_{\mathrm{p}}$ gives:

$$\begin{bmatrix} \vec{g}_{\mathrm{p}} \\ \vec{g}_{\mathrm{t}} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}_{\mathrm{pp}} & \mathbf{K}_{\mathrm{tp}}^{\top} \\ \mathbf{K}_{\mathrm{tp}} & \mathbf{K}_{\mathrm{tt}} \end{bmatrix}\right) \tag{3}$$

Rewriting leads to $\vec{g}_{\mathrm{p}} \sim \mathcal{N}(\hat{\vec{g}}, \boldsymbol{\sigma}_{\hat{g}}^2)$, with the best estimate $\hat{\vec{g}}$ and variance $\sigma_{\hat{g}}^2$ defined as:

$$\hat{\vec{g}} = \mathbf{K}_{\mathrm{tp}}^{\top}\mathbf{K}_{\mathrm{tt}}^{-1}\vec{g}_{\mathrm{t}} \tag{4}$$

$$\boldsymbol{\sigma}_{\hat{g}}^2 = \mathbf{K}_{\mathrm{pp}} - \mathbf{K}_{\mathrm{tp}}^{\top}\mathbf{K}_{\mathrm{tt}}^{-1}\mathbf{K}_{\mathrm{tp}} \tag{5}$$

The choice of kernel $k(\vec{u}, \vec{u'}|\vec{\theta})$ can be based on the expected behaviour of the approximated function and is generally expressed in terms of an a-priori variance and the Matèrn correlation function $\rho_{\mathrm{M}}(\vec{u} - \vec{u'}|\vec{\theta}, \nu)$, with $\vec{\theta}$ representing the internal parameters (hyperparameters) and ν a shape function controlling the GP smoothness. The Matèrn correlation function is used instead of the classical radial basis function as it is better able to approximate models with non-differentiable mean functions (Rasmussen and Williams 2006). The variance term is here considered as one of the kernel hyperparameters, facilitating a straightforward extension in Section 2.4.

The hyperparameters $\vec{\theta}$ are optimised using the quasi-Newton optimisation scheme L-BFGS-B (Zhu et al. 1997) to maximise the log-likelihood of the training data. Once the GP is trained, the GP forms a predictor or metamodel of the performance function.

2.2. GP metamodelling for reliability analysis

The metamodelling approach to reliability analysis involves replacing the computational model with an approximative metamodel when performing the time-consuming (Monte Carlo) integration in evaluating the probability of failure. To this end, the metamodel prediction $\hat{g}(\vec{u})$ is used in the MC integration in Equation (1) to approximate P_{f}. As a result, any uncertainty in $\hat{g}(\vec{u})$ propagates through the MC integration and leads to uncertainty in the prediction of the probability of failure P_{f}. Conservative upper and lower bounds $P_{\mathrm{f}}^{\pm}$ related to metamodel prediction uncertainty are given by Schöbi, Sudret, and Marelli (2016) as:

$$\hat{P}_{\mathrm{f}}^{\pm} = \mathbb{P}\left[\hat{g}(\vec{u}) \mp 1.96\sigma_{\hat{\mathrm{g}}}(\vec{u}) \leq 0\right] \tag{6}$$

These bounds are used here to assess the metamodel-related uncertainty in the predictions, without having to evaluate the propagation of the uncertainty explicitly (see e.g. Hu and Mahadevan 2016), which is computationally demanding.

Although the MC sample set $\mathbf{u}_{\mathrm{MC}}$ typically contains $N_{\mathrm{MC}} > 10^5$ samples, uncertainty in the MC integration itself remains. This is quantified in terms of the coefficient of variation:

$$\delta_{\mathrm{P}_f,\mathrm{MC}} = \frac{\sigma_{\hat{\mathrm{P}}_f}}{\hat{P}_{\mathrm{f}}} = \sqrt{\frac{1 - \hat{P}_{\mathrm{f}}}{N_{\mathrm{MC}}\hat{P}_{\mathrm{f}}}} \tag{7}$$

These measures can be used to formulate convergence criteria for the active learning scheme of the metamodel as

$$\frac{\hat{P}_{\mathrm{f}}^{+} - \hat{P}_{\mathrm{f}}^{-}}{\hat{P}_{\mathrm{f}}} = \varepsilon_{\mathrm{P}_f,\mathcal{M}} < r_{\mathcal{M}} \tag{8}$$

and for the MC integration as

$$\delta_{\mathrm{P}_f,\mathrm{MC}} = \varepsilon_{\mathrm{P}_f,\mathrm{MC}} < 0.05 \tag{9}$$

Equation (9) is a well-established convergence criterion generally accepted in a probabilistic context and, when aiming for the same level of accuracy in the metamodel prediction itself, can be used to define $r_{\mathcal{M}}$. Considering Equation (8) to represent a confidence range of a normal distribution leads to:

$$\varepsilon_{\mathrm{P}_f,\mathcal{M}} = 2k\delta_{\mathrm{P}_f,\mathcal{M}} \tag{10}$$

where $k = -\Phi^{-1}(0.025) = 1.96$ is the standard score when considering the 95% range centred on the mean. Taking $\delta_{\mathrm{P}_f,\mathcal{M}} = 0.05$ then leads to the metamodel convergence criterion parameter $r_{\mathcal{M}} = 0.196$. This is less strict than $r_{\mathcal{M}} = 0.05$ as proposed in Schöbi, Sudret, and Marelli (2016), but generally more strict than the global sensitivity convergence criterion of Hu and Mahadevan (2016).

For the case when $\varepsilon_{\mathrm{P}_f,\mathcal{M}} > r_{\mathcal{M}}$, the metamodel has not converged and the training data need to be expanded with additional evaluations of the computational model on learning samples $\vec{u}_{\mathrm{learn}}$. A learning function $L_{\mathrm{p}}(\mathbf{u}_{\mathrm{MC}})$ is used, which aims at selecting the most informative next realisation $\vec{u}_{\mathrm{learn}}$ to be evaluated. The most common learning function is the U-learning function, which selects the realisation for which the performance (i.e. I_{f}) is the most likely to be predicted incorrectly (Echard, Gayton, and Lemaire 2011):

$$\vec{u}_{\mathrm{learn}} = L_{\mathrm{p}}(\mathbf{u}_{\mathrm{MC}}) = \underset{\vec{u}\in\mathbf{u}_{\mathrm{MC}}}{\arg\min}\left(\frac{|\hat{g}(\vec{u})|}{\sigma_{\hat{g}}(\vec{u})}\right) \tag{11}$$

The performance function is evaluated for the selected learning sample and added to the training data. The enhanced training data are used to train the metamodel which, now being based on more training data, tends to yield a better prediction. This iterative procedure of metamodel prediction, learning and training is known as active learning and forms the basis of AK-MCS.

2.3. Importance sampling

Although the metamodelling approach to reliability analysis provides a computationally inexpensive way of performing MC integration over the sampling space, very small probabilities (e.g. $< 10^{-6}$) still cannot be evaluated efficiently by brute-force MC integration alone. A common variance reduction method to tackle this problem is importance sampling (IS) (Melchers 1989), in which the samples are drawn from an importance distribution $q(\vec{u})$. A common choice for the selection of $q(\vec{u})$ is a standard normal distribution centred at the design

point (Melchers 1989). The importance distribution defines sample importance weights $w_{\text{imp}}(\vec{u}) = p/q$, in which $p = p(\vec{u})$ is the standard normal probability density function. The importance weights are accounted for in calculating the probability of failure:

$$\hat{P}_{\text{f}}^{\text{IS}} = \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} w_{\text{imp},i} \hat{I}_{\text{f},i} \tag{12}$$

The convergence of $\hat{P}_{\text{f}}^{\text{IS}}$ is expressed in terms of the coefficient of variation:

$$\delta_{\text{P}_f}^{\text{IS}} = \frac{\hat{\sigma}_{\text{P}_f}^{\text{IS}}}{\hat{P}_{\text{f}}^{\text{IS}}} \tag{13}$$

$$\sigma_{\hat{\text{P}}_f}^{\text{IS}\,2} = \frac{1}{N_{\text{MC}} - 1} \left(\frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} w_{\text{imp}}^2 \hat{I}_{\text{f},i} - \hat{P}_{\text{f}}^2 \right) \tag{14}$$

The combination of IS and AK-MCS is known as AK-IS, introduced by Echard et al. (2013). They centred the distribution of the sampling pool $\mathbf{u}_{\text{MC}}$ at the design point determined by an initial FORM analysis on the performance function. To account for multiple design points and strong non-linearity of the limit state surface, Cadini, Santos, and Zio (2014) proposed importance sampling based on multiple quasi-design points determined through clustering of the initial failure domain as characterised by the MC sampling set. Parts of this concept are used in this work to formulate a new adaptive importance sampling strategy with multiple importance regions, to come to a converged MC integration. This adaptive multiple importance sampling procedure is outlined in Section 3.3. In addition to providing a more efficient convergence of the MC integration, importance sampling can also improve the effectiveness of the active learning procedure, by providing a sampling pool with more relevant candidate points located near the relevant parts of the limit state.

Note that the terms "MC integration" and "MC sampling pool" are used here for random sample-based integration in general, including importance sampling.

2.4. Noisy data metamodel

Although observations from computer experiments such as finite element analyses were initially considered noise-free (e.g. Sacks et al. 1989), the precision of complex numerical models is finite. The resulting numerical errors may appear as random noise in the computational result of the analysis. Even though these numerical errors are generally small compared to the computed result, they can have a strong impact on the further use of the model response in, for example, optimisation or probabilistic interpretation.

A discussion on the problem of noisy data in the context of global optimisation with GP metamodels is given in Forrester, Sobester, and Keane (2008), demonstrating that noise in the data prevents convergence (of the optimisation) and hinders the training algorithm. The same mechanisms work in active learning for reliability analysis, where the overfitting of a Gaussian process to noisy model response data leads to an increase in uncertainty with added data.

The computational model of the sheetpile wall introduced below shows a level of noise in the calculated response up to approximately 1% of the total variation in system response. This noise can be attributed to both the required convergence tolerance and the stability of the numerical procedures of the finite element software. To account for this numerical noise and to avoid overfitting by the metamodel, a noise term is added to the kernel (Rasmussen and Williams 2006):

$$k_{\text{wn}}(\vec{u}, \vec{u'}) = \sigma_{\text{wn}}^2 \delta_{\vec{u}\vec{u'}} + k(\vec{u}, \vec{u'} \mid \sigma^2, \vec{\theta}) \tag{15}$$

Noise variance σ_{wn}^2 now becomes an additional hyperparameter, which is included in the hyperparameter optimisation scheme discussed above.

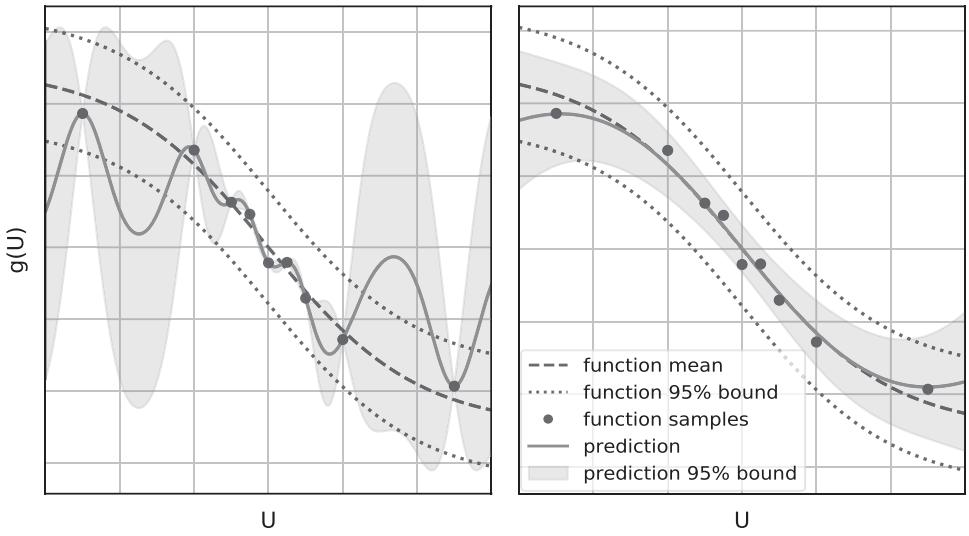
The noise term is only added to the kernel when it is applied on training data (i.e. in matrix $\mathbf{K}_{\text{tt}}$ in Equations (4) and (5)). As a result, the metamodel predictions are smooth, although the prediction uncertainty by Equation (5) at training data locations is non-zero due to the training data uncertainty.

The effect of including training data uncertainty σ_{wn}^2 on the metamodel is demonstrated in Figure 1, where a series of samples is taken from a smooth mean function with an additional noise term. In Figure 1(a), a metamodel with a noise-free kernel is trained on the samples, resulting in overfitting. In Figure 1(b), a metamodel with a noise component in the kernel is trained on the same data. This leads to a better representation of the underlying function and a metamodel prediction that includes the uncertainty in the training data $\vec{g}_{\text{t}}$. The prediction uncertainty at the training datapoint locations is non-zero and locally reduces when more local training data are available as an effect of spatio-statistical averaging.

3. Two-stage metamodelling of noisy and incomplete models

3.1. UNIS learning function

The U-learning function of Equation (11) was formulated for noise-free training data and MC integration with



(a) overfitting (b) noise term in kernel

Figure 1. Example of training a metamodel on a noisy model response with and without a noise term in the kernel. (a) Overfitting and (b) noise term in kernel.

equal sample weights. Under these conditions, Equation (11) is equivalent to selecting the sample which contributes most to reducing the prediction uncertainty when adding that sample to the training data, since this will reduce its probability of mis-classification to zero. In the case of importance sampling, with unequal weights over the samples, this equivalence no longer holds and the sample weight should be included. In the case of noisy metamodelling, the probability of mis-classification of the sample added to the training data does not reduce to zero. To account for both effects, a novel learning function is formulated here. It is based on the concept of utility maximisation, in which the utility is formulated as the most likely reduction in the expected prediction uncertainty when adding the sample to the DoE. The sample selection for learning then comes down to selecting the sample that maximises utility. The utility function is here formulated per sample. A global utility (e.g. Hu and Mahadevan 2016; El Haj, Soubra, and Al-Bittar 2019) would require a global sensitivity analysis, which is computationally impractical due to excessive computational costs.

The utility function is based on the most likely difference in the probability of mis-classification $\Phi(-|\hat{g}|/\sigma_{\hat{g}})$ before and after adding a learning candidate sample $\vec{u}_i$ to the training data:

$$U_{\mathrm{NIS}}(\vec{u}_i) = \left(\Phi\left(-\frac{|\hat{g}(\vec{u}_i)|}{\sigma_{\hat{g}}}\right) - \Phi\left(-\frac{|\hat{g}(\vec{u}_i)|}{\sigma_{\hat{g}+1}}\right)\right) w_{\mathrm{imp},i} \tag{16}$$

The left-hand probability is the current probability of mis-classification based on the current prediction variance $\sigma^2_{\hat{g}}(\vec{u}_i)$ given by Equation (5). The right-hand probability is based on the model response expectation $\mathbb{E}[g(\vec{u}_i)] = \hat{g}(\vec{u}_i)$ and the prediction variance $\sigma^2_{\hat{g}+1}$ after adding candidate $\vec{u}_i$ to the training data. Under the assumption of unchanged hyperparameters in the kernel during training, this term is given by (see Appendix 2 for a derivation):

$$\sigma^2_{\hat{g}+1} = \sigma^2_{\hat{g}} \frac{\sigma^2_{\mathrm{wn}}}{\sigma^2_{\hat{g}} + \sigma^2_{\mathrm{wn}}} \tag{17}$$

Maximising the utility leads to the learning function being defined as:

$$\vec{u}_{\mathrm{learn,P}} = \mathcal{L}_{\mathrm{P}}(\mathbf{u}_{\mathrm{MC}}) = \underset{\vec{u}\in\mathbf{u}_{\mathrm{MC}}}{\arg\max}\,(U_{\mathrm{NIS}}(\vec{u})) \tag{18}$$

Note that it is the difference in classification uncertainty which is important here, since the local uncertainty does not reduce to zero upon adding a point to the training data due to the inherent noise in the training data. In the case where the noise is negligible, the difference in uncertainty equals the initial total uncertainty (since all uncertainty would be reduced upon adding the point to the training data) and the UNIS-learning function reduces to the classical U-learning function with an importance weight correction.

3.2. Classification metamodelling: Gaussian process classification

Computational models do not always provide a response, which can lead to incomplete training data. This is sometimes due to numerical issues such as non-convergence

of the computational model, but also as a result of the inconsistent formulation of the problem. In these cases where numerical or software issues lead to data missing at random, while the true model response does exist, the missing data can be imputed (i.e. temporarily filled-in) based on the metamodel prediction (Little and Rubin 2002; Forrester, Sobester, and Keane 2008). This allows the active learning algorithm to continue learning based on realisations that do provide a model response.

In other cases, a solution may be non-existent because the prior formulation of the input parameters is partly inconsistent with respect to the computational model. This may lead to regions in parameter space for which no solution to the problem exists, referred to here as incompatible domains. Incompatible realisations sampled from such a domain thus lead to missing response data. In the case of the sheet pile bending moment analysed in Section 4, incompatible realisations exist when the combination of input parameters leads to slope failure in the initialisation phase, before installation or loading of the sheet pile wall, preventing the evaluation of the bending moments.

The incompatible domains in parameter space have a significant impact on the application of active learning, as they form pitfalls for the learning algorithm. Without proper measures, the learning algorithm is likely to get stuck on these missing data and an enhancement is clearly needed to deal with incompatible domains. This is done here by combining the GP predictive metamodel $\mathcal{M}_\mathrm{p}$ with a GP classification metamodel $\mathcal{M}_\mathrm{c}$, which classifies the parameter space into an incompatible domain (with missing data) and a feasible domain (where solutions can be found). The predictive metamodel then predicts the performance function in the feasible domain, based on which the reliability analysis is performed and the learning function is applied.

Because no response data exist in the incompatible domain, a classification model is used, based on binary data I_c for feasible ($I_\mathrm{c} = 0$) and incompatible ($I_\mathrm{c} = 1$) realisations. Many approaches can be applied for binary classification, ranging from simple nearest neighbour to kernel-based support vector classification. Here, Gaussian process classification (GPC) (Rasmussen and Williams 2006) is used, with training data $\{\mathbf{u}_\mathrm{t}, \vec{I}_{\mathrm{c,t}}\}$. GPC is formulated as a probabilistic classification, using a Gaussian process $f(\vec{x})$ as a so-called latent function, which is scaled down to a class probability $\pi(\vec{u})$ using a sigmoid function:

$$\pi(\vec{u}) = \frac{1}{1 + \exp\left(-f(\vec{u})\right)} \tag{19}$$

An optimisation scheme is applied to calibrate the latent function hyperparameters to maximise the correct classification of the binary training data by $\pi(\vec{u})$, similar to the optimisation scheme for the prediction metamodel, following Algorithms 3.1 and 3.2 in Rasmussen and Williams (2006).

After training of the GP classifier, the classification of the sample pool $\mathbf{u}_\mathrm{MC}$ is done by means of the class probability:

$$\hat{I}_\mathrm{c}(\vec{u}) = \begin{cases} 1 & \text{if } \pi(\vec{u}) > 0.5 \\ 0 & \text{if } \pi(\vec{u}) \leq 0.5 \end{cases} \tag{20}$$

This provides the subset of samples $\mathbf{u}_\mathrm{MC}(\hat{\boldsymbol{I}}_\mathrm{c} = 1)$ that is classified as incompatible. When incompatible realisations are observed, the probability of incompatibility $P_1 = \mathbb{P}[I_\mathrm{c} = 1]$ can be estimated from the class prediction:

$$\hat{P}_1 = \frac{1}{N_\mathrm{MC}} \sum_{i=1}^{N_\mathrm{MC}} \hat{I}_{\mathrm{c},i} w_{\mathrm{imp},i} \tag{21}$$

Using GPC for the classification metamodel provides a probability of (in)correct classification $\bar{\pi}(\vec{u})$ when integrating out the Gaussian posterior distribution of latent function $f(\vec{u})$ in the expression of $\pi(\vec{u})$ (see Rasmussen and Williams 2006 for details). This probability is used here to define a learning function for the classification metamodel, selecting the highest probability of mis-classification. This is conceptually identical to Equation (11), which now becomes:

$$\vec{u}_{\mathrm{learn,I_c}} = \mathcal{L}_\mathrm{c}(\mathbf{u}_\mathrm{MC}) = \underset{\vec{u} \in \mathbf{u}_\mathrm{MC}}{\arg\min}(|0.5 - \bar{\pi}(\vec{u})|) \tag{22}$$

Two criteria are used for the convergence of the classification model: the first looking at the convergence of $\hat{P}_1$ itself, by comparing the maximum relative difference in $\hat{P}_1$ between the latest (N_c) and the 4 preceding ($N_\mathrm{c} - i, \forall\, i \in \{1, 2, 3, 4\}$) learning iterations related to the classification model; the second based on the relevance of the missing data in the determination of $\hat{P}_\mathrm{f}$:

$$\max_{i \in \{1,2,3,4\}} \left(\frac{|\hat{P}_1^{(N_\mathrm{c}-i)} - \hat{P}_1^{(N_\mathrm{c})}|}{\hat{P}_1^{(N_\mathrm{c})}} \right) < \varepsilon_{P_1} = 5\% \quad \text{or} \quad \frac{\hat{P}_1}{\hat{P}_\mathrm{f}} < 0.01 \tag{23}$$

with $^{(.)}$ containing the iteration counter for the N_c learning iterations for which classification learning (Equation (22)) was used.

3.3. Adaptive multiple importance sampling

The evaluation of the reliability uses Monte Carlo integration of the failure indicator function I_f, over (the feasible domain of) the parameter space, using the sampling

pool $\mathbf{u}_{\mathrm{MC}}$ containing N_{MC} samples. At the beginning of the analysis, $\mathbf{u}_{\mathrm{MC}}$ is sampled with increased variance from a normal distribution q with zero mean and covariance $\sigma^2\mathbf{I}$. To guarantee sufficient candidate points for learning in the sampling pool, σ needs to be sufficiently large in order to cover the potential failure domain. At the same time, σ needs to be small enough to have sufficient sampling points close to the limit state. Here, the initial value for σ was set at $\beta_0/2$, with β_0 being the initial estimation of the reliability prior to the analysis. This value has proven effective in starting the adaptive importance sampling scheme and has led to an acceptable compromise between MC integration convergence and failure domain coverage. Nevertheless, the initial importance distribution could easily be improved based on more prior knowledge of the performance function (e.g. sensitivity, location of limit state). The increased variance sampling pool is used for the active learning algorithm of the metamodel until metamodel convergence.

If, after convergence of the metamodel, increased variance sampling over N_{MC} samples is insufficient to obtain MC convergence (i.e. if $\delta^{\mathrm{IS}}_{\mathrm{P}_f} > 0.05$), the sampling pool $\mathbf{u}_{\mathrm{MC}}$ is updated by resampling from a distribution $q_{\mathrm{mult}}(\vec{u})$. This distribution is defined as a weighted average of k isotropic normal distributions $q_i(\vec{u})$ with unit variance, and means $\vec{\mu}_i$ located at a series of k quasi-design points $\vec{u}^{\mathrm{des}}_i$:

$$q_{\mathrm{mult}}(\vec{u} \mid \vec{u}^{\mathrm{des}}_{i=1..k}) = \sum_{i=1}^{k} \lambda_i q_i(\vec{u}) \tag{24}$$

The quasi-design points are estimated from the current estimates of the limit state and the incompatible domain. The quasi-design points are distributed such that they span all directions in U-space in which a failure or incompatible domain is predicted by the current metamodel. In this way, multiple design points are naturally accounted for in the multiple importance sampling procedure. Details of the selection of the k quasi-design points and weights λ_i are given in Appendix 1.

The distribution $q_{\mathrm{mult}}(\vec{u})$ is in essence a Gaussian mixture model with isotropic distributions. It is used to resample the new sampling pool $\mathbf{u}_{\mathrm{MC}}$, with corresponding importance weights $w_{\mathrm{imp}}(\vec{u}) = p(\vec{u})/q_{\mathrm{mult}}(\vec{u})$.

With an updated sampling set, the active learning of the metamodel continues until metamodel convergence, after which the integration convergence is evaluated again (see Figure 2).

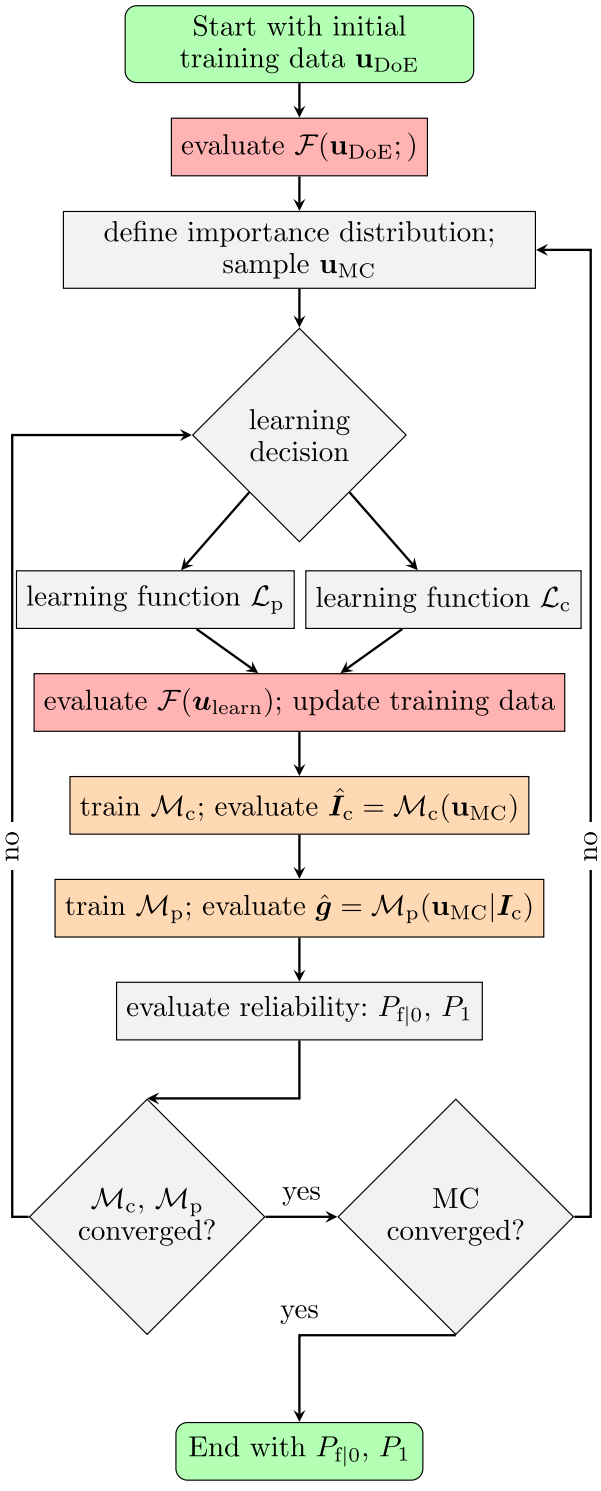


Figure 2. Flowchart of the two-stage metamodel learning algorithm. The evaluations of $\mathcal{F}(u)$ are the parts of high computational cost, the training blocks are the parts of moderate computational cost.

3.4. Combined two-stage metamodel learning algorithm

The classification metamodel $\mathcal{M}_{\mathrm{c}}$ and the prediction metamodel $\mathcal{M}_{\mathrm{p}}$ are combined into one two-stage

metamodel $\{\hat{I}_c, \hat{g}\} \leftarrow \mathcal{M}(\vec{u})$, which subsequently classifies into incompatible/feasible data $\hat{I}_c = \mathcal{M}_c(\vec{u})$ and predicts the performance function in the case of feasibility $\hat{g} = \mathcal{M}_p(\vec{u} \mid \hat{I}_c = 0)$. This means that the two-stage metamodel needs two stages of training, and the training data are expanded with class data I_c into $\{\mathbf{u}_t, \vec{g}_t, \vec{I}_{c,t}\}$. Active learning is based on the learning functions for both metamodels as outlined above. The only difference is that in the presence of an incompatible domain, only the feasible data are considered for the learning function.

The combination of the established methods discussed in Section 2, together with the enhancements and extensions proposed in Section 3.1–3.4, leads to the following algorithmic description of the metamodelling approach for reliability analysis with complex noisy and incomplete numerical models:

(1) Define an initial training data set (DoE), and evaluate the corresponding model response $\{\mathbf{u}_t, \vec{I}_{c,t}, \vec{g}_t\}$.
(2) Define an importance distribution using the criterion in Section 3.2 and generate a Monte Carlo sampling pool $\mathbf{u}_{MC}$ (Section 3.2).
(3) Choose between active learning for $\mathcal{M}_p$ or for $\mathcal{M}_c$. In the case where only $\mathcal{M}_c$ has converged, the learning function $\mathcal{L}_p$ is selected; in the case where only $\mathcal{M}_p$ has converged, the learning function $\mathcal{L}_c$ is selected. In the case where neither model has converged, learning function $\mathcal{L}_p$ is selected if $\hat{P}_{f\,|\,0}/(\hat{P}_{f|0} + \hat{P}_1) > p_{rnd}$, and learning function $\mathcal{L}_c$ is selected otherwise, with p_{rnd} being a random number sampled from a standard uniform distribution in each iteration, and $\hat{P}_{f|0}$ and P_1 are as defined in Section 3.5.
(4) Select the next training point $\vec{u}_{learn}$ from $\mathbf{u}_{MC}$ using the selected learning function (Equation (18) or (22)).
(5) Evaluate the computational model $\mathcal{F}(\vec{u}_{learn})$ and use the result $\{\vec{u}_{learn}, I_c, g\}$ to augment the training data $\{\mathbf{u}_t, \vec{I}_{c,t}, \vec{g}_t\}$.
(6) Train the classification metamodel $\mathcal{M}_c$ on $\{\mathbf{u}_t, \vec{I}_{c,t}\}$ and evaluate the classification index $\hat{\vec{I}}_c = \mathcal{M}_c(\mathbf{u}_{MC})$ (Section 3.2).
(7) Train the prediction metamodel $\mathcal{M}_p$ on $\{\mathbf{u}_t, \vec{g}_t\}|_{\vec{I}_{c,t}=0}$ and evaluate the performance function prediction $\hat{\vec{g}} = \mathcal{M}_p(\mathbf{u}_{MC} \mid \vec{I}_c)$ (Section 2.1).
(8) Evaluate the updated probabilities $\hat{P}_{f|0}$ and P_1 (Equations (12) and (21)).
(9) Evaluate the convergence of the classification model (Equation (23)) and prediction model (Equation (8)) in terms of the prediction of P_1 and $P_{f|0}$. Continue if both converged, otherwise return to (3).
(10) Evaluate the convergence of the Monte Carlo integration. Continue if converged, otherwise return to (2).
(11) End with results $P_{f|0}$, P_1, etc.

This algorithmic two-stage metamodel approach is summarised in the flowchart of Figure 2.

3.5. Two-stage model interpretation

The presence of the incompatible domain in combination with the performance of the sheet pile wall expressed in the performance function $\hat{g}$ results in different probability definitions:

- $P_1 := \mathrm{P}(I_c = 1)$: Probability of data incompatibility. This is the probability of failure in the initial conditions prior to the final loading stage, such that the final stage (i.e. performance function $g(\vec{u})$) cannot be evaluated.
- $P_f^* := \mathrm{P}(g < 0)$: This is the imputed unconditional probability of failure where any potential incompatibility (either observed or unobserved) is imputed by $\mathcal{M}_p$). This is the prior probability of failure when no information is available on survived initial states.
- $P_{f|0} := \mathrm{P}(g < 0 \mid I_c = 0)$: This is the probability of failure conditional on compatible data. This is the probability of failure conditional on the survival of the initial conditions prior to the final loading stage.

Note that, with complete knowledge on data incompatibility (for example because of evidence of survival of the initial stages of the structure), P_f^* forms a prior estimation and becomes obsolete since the updated, conditional probability $P_{f\,|\,0}$ should be used. This is because P_f^* corresponds to the case where information on incompatible domains or survived stages is not or only partly available, as would for example be the case in FORM, point-estimate or semi-probabilistic analyses. This aspect of the imputed probability of failure P_f^* is discussed further in Section 5 to highlight the importance of the two-stage model in the evaluation of reliability in the context of reliability updating. Aside from this specific context for which P_f^* is used, $P_{f\,|\,0}$ is the probability of primary interest in the analysis.

4. Application and evaluation

The computational model of a Dutch river dyke cross-section with a sheet pile wall reinforcement is evaluated herein. This model is based on Noordam (2019) and

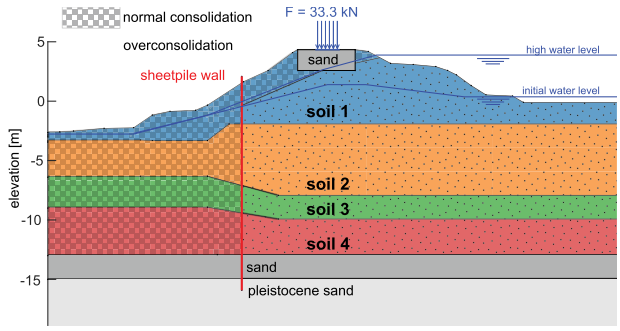


Figure 3. Model geometry with soil stratification and active loading conditions (water levels, crest load), as defined in Noordam (2019). Far-field boundary conditions (left, right, below) were chosen sufficiently far away from the zone of interest and are not depicted.

represents an existing dyke with representative distributions for typically available site data and design requirements. The dyke body rests on a foundation of soft soil blanket layers, underlain by Pleistocene sand (see Figure 3). During flood conditions, the increase of pore water pressures in the sand and in the soft soil blankets can lead to instability of the landside slope; hence the insertion of the sheetpile wall as a reinforcement measure.

The Soft Soil constitutive model (Vermeer and Neher 1999; Brinkgreve et al. 2018) is used for the soft soil layers, with the friction ratio $\tan\phi'$, yield stress σ_y, and inverse of the modified compression index λ^{-1} as stochastic parameters, following Noordam (2019). The yield stress indirectly represents the isotropic preconsolidation stress P_p in the Soft Soil model, from which the pre-overburden pressure (POP) is calculated via equivalent isotropic stress states per layer, as input in PLAXIS (Brinkgreve et al. 2018). The swelling index κ is related to the compression index by $\kappa = \lambda/10$. All other parameters are deterministic, and have no direct influence on this work. The Hardening Soil model (Schanz, Vermeer, and Bonnier 1999) with deterministic soil parameters is used for the sand layers. The sheetpile wall is modelled as a zero-thickness elastic plate.

Starting from an initial situation, the model evaluates the sheetpile wall performance under high water conditions in combination with a traffic load. Three modelling phases are identified, each consisting of two model stages:

- *Phase 1.* An initialisation stage based on a K_0 procedure followed by a stress redistribution stage to attain initial equilibrium conditions;
- *Phase 2.* A sheetpile wall installation stage, followed by a long-term settlement stage;
- *Phase 3.* A first loading stage in which the water level is raised to the high water condition, followed by a second loading stage in which the crest traffic load is applied.

Two types of failure are considered in the analysis: Geotechnical failure is defined as slope failure in the initialisation phase, when the dyke fails under normal conditions. Structural failure is defined as the exceedance of the bending moment capacity $T_{M_{max}}$ in the final stage. In this model, geotechnical failure in the initialisation phase is the only type of failure that prevents the evaluation of the maximum bending moment in the sheetpile wall in the final stage. Other modes of failure, such as geotechnical failure without interaction with the sheetpile wall in phases 2 or 3, were not observed.

The model is formulated as the computational model $M_{max} = \mathcal{F}(\vec{X}) = \mathcal{F}_U(\vec{U})$, in which $\vec{X}$ is the vector of stochastic input parameters, $\vec{U}$ is the standard normal transform of $\vec{X}$ and M_{max} is the calculated maximum bending moment in the final stage. In the case of geotechnical failure, M_{max} is undetermined. An indicator function I_c is defined such that $I_c = 1$ when M_{max} is undetermined and $I_c = 0$ when a solution of $\mathcal{F}(\vec{X})$ exists.

The stochastic soil parameters given by Noordam (2019) are summarised in Table 1. Transformation into standard normal equivalent parameters U_i allows the problem to be reformulated based on 10 standard normal variables U_1–U_{10}. Due to the full correlation between the degrees of overconsolidation of the

Table 1. Soil property distribution parameters and their independent stochastic standard normal variables U_i.

	$\tan(\phi')$		λ^{-1}		σ_y^{NC} (kPa)		σ_y^{OC} (kPa)	
	$\sim LN(\mu, \delta = 0.10)$		$\sim LN(\mu, \delta = 0.10)$		$\sim \mathcal{N}(\mu, \delta = 0.21)$		$\sim \mathcal{N}(\mu, \delta = 0.21)$	
Layer	μ	U_i	μ	U_i	μ	U_i	μ	U_i
Soil 1 (dyke clay)	0.562	U_1	21.9	U_5	34.4	U_9	99.3	U_9
Soil 2 (sandy clay)	0.500	U_2	35.4	U_6	34.4	U_{10}	191.0	U_9
Soil 3 (peat)	0.918	U_3	6.4	U_7	91.7	U_{10}	236.8	U_9
Soil 4 (heavy clay)	0.419	U_4	15.8	U_8	183.3	U_{10}	275.0	U_9

Distribution type ($\mathcal{N}$ for normal, *LN* for log-normal); coefficient of variation δ is constant per parameter type; mean value μ is different for each parameter. Yield stresses are considered fully correlated due to the loading history and are effectively controlled by only two independent stochastic variables related to the overconsolidated (U_9) and normally consolidated (U_{10}) regions (Noordam 2019).

different layers, variables U_9 and U_{10} are considered to control the degree of overconsolidation of all four layers for, respectively, the overconsolidated and normally consolidated domains.

A semi-probabilistic calculation is performed first, using design values derived from the lower 5 percentile of the parameters and application of a partial factor of 1.18 to $\tan\phi'_i$, based on a target reliability index $\beta_T = -\Phi^{-1}(P_{f,T}) = 5.11$ (Noordam 2019; Kanning et al. 2017). The resulting semi-probabilistic standard normal design values u_i^{des} are -3.304 for $\tan\phi'_i$ and -1.645 for the other stochastic variables.

Averaging out the numerical noise over 100 closely sampled realisations, the semi-probabilistic maximum bending moment is estimated at $M_{max} = 279.2$ kNm/m', and is used in the following as the threshold $T_{M_{max}}$ as the basis for a probabilistic analysis using metamodelling.

The performance function is formulated relative to the computational model response for mean variables $\mathcal{F}_U(\vec{u} = \vec{0}) \approx 90\,\text{kNm/m}'$ and the performance threshold $T_{M_{max}}$, such that failure is associated with $g(\vec{u}) < 0$. Normalisation is applied for convenience:

$$g(\vec{u}) = \frac{T_{M_{max}} - \mathcal{F}_U(\vec{u})}{T_{M_{max}} - \mathcal{F}_U(\vec{u} = \vec{0})} \tag{25}$$

In the following, the MC integration sampling pool $\mathbf{u}_{MC}$ contains $N = 500,000$ points, all of which are used in the Monte Carlo integration. In addition, these points serve as potential candidates for training data, which is iteratively extended in the active learning scheme. A maximum of 100 learning iterations is performed before updating the MC sampling set. The Matèrn correlation function is used with a constant shape parameter $\nu = 2.5$.

4.1. Metamodel simulations with 1 stochastic variable

To demonstrate the model behaviour and the kriging-based metamodelling, a series of simulations is performed with a single stochastic variable, while all other variables are set to the semi-probabilistic design values. Figure 4 shows the metamodel response surfaces from 10 simulations (corresponding to U_1 to U_{10} in Table 1), as well as the respective training data and the 95% confidence bounds. The response surfaces correctly reach the limit state $g = 0$ at the semi-probabilistic design points (indicated by arrows) and the uncertainty bounds are narrow at these limit state crossings, indicating the correct metamodel convergence.

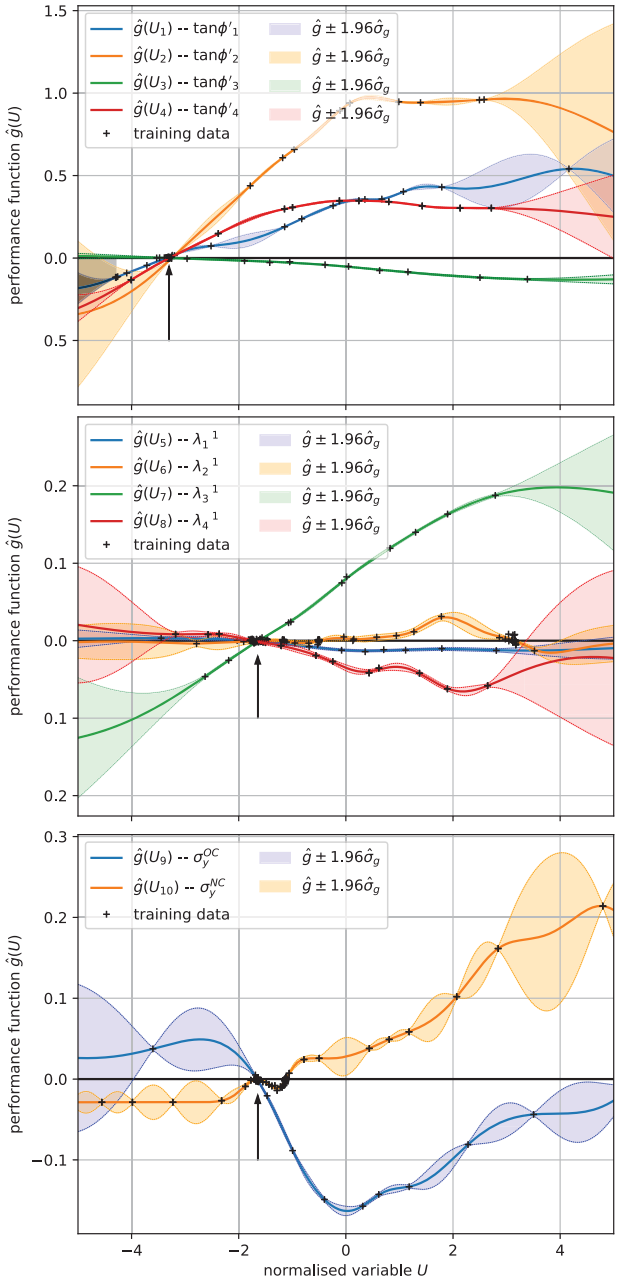


Figure 4. 1D metamodels through semi-probabilistic design point for sensitivity analysis. Arrows indicate the location of the semi-probabilistic design point for reference.

The 1-D response surfaces also show the sensitivity of the model response to the individual variables, locally via the response surface slope around the design point and more globally by the range of model responses captured by the 1-D surfaces. The model response appears most sensitive to $\tan\phi'_2$, followed by $\tan\phi'_1$ and $\tan\phi'_4$ and, to some extent, to $\tan\phi'_3$, λ_3^{-1}, σ_y^{OC} and σ_y^{NC}. The other variables seem to have a marginal influence on the variation of the (local) model response. Another important observation is that some parameters ($\tan\phi'_3$, λ_4^{-1} and σ_y^{OC}) have an inverse effect on the

performance (i.e. increasing strength and stiffness leading to decreasing performance). Furthermore, Figure 4 demonstrates the (sometimes strongly) non-linear behaviour of the computational model, combined with a modest noise level. This is most evident in the response surface of σ_y^{NC}, between $u_{10} = -2$ and $u_{10} = -1$, where many model evaluations are performed in order to reduce the prediction uncertainty stemming from numerical noise.

Finally, incompatible realisations were encountered only in the metamodel learning for variable $\tan\phi_1'$ for values $u_1 < -4.4$, with the resulting missing data domain indicated by the grey shaded area. Domains with missing data might also exist for other parameters, but, without incompatible realisations in the training data, no evidence for this domain is available and no missing data are predicted as a result.

4.2. Metamodel simulation with 2 stochastic variables

Figure 5 shows the results of a simulation with $\tan\phi_1'$ and $\tan\phi_4'$ as stochastic variables, while all other variables are set at the design values. For $T_{M_{max}} = 279.2$ kNm/m', the predicted response surface $\hat{g}(\vec{u})$ is given as a contour plot, together with the training data and the incompatible domain as predicted by $\mathcal{M}_c$. Including the 10 evaluations of the initial training data, a total of 46 function calls were required for convergence of both parts of the metamodel. Increasing the variance of the initial sampling pool $\mathbf{u}_{MC}$ was sufficient to reach convergence of the MC integration, and no importance sampling refinement was required.

The resulting limit state function correctly passes through the semi-probabilistic design point, demonstrating the consistency between the predictions in Figures 4 and 5. Figure 5(b) shows the bending moment profiles along the sheetpile wall for different points on the limit state surface. The profile for Point A, corresponding to a low value for $\tan\phi_1'$ and a high value for $\tan\phi_4'$, shows a maximum bending moment at the bottom of layer 2, indicating active loading on the sheetpile wall by layers 1 and 2 and the initiation of a failure mechanism through layer 2. In contrast, the profile for Point B shows the maximum bending moment at the bottom of layer 4, indicating active loading on the sheetpile wall from all four layers. The incompatible realisation at Point C leads to slope failure in the initialisation phase, as illustrated in Figure 5(c). Overall, Figure 5 demonstrates that the contemplated case clearly includes aspects of non-linearity in the limit state, a relevant incompatible domain, and a modest level of noise in the system response.

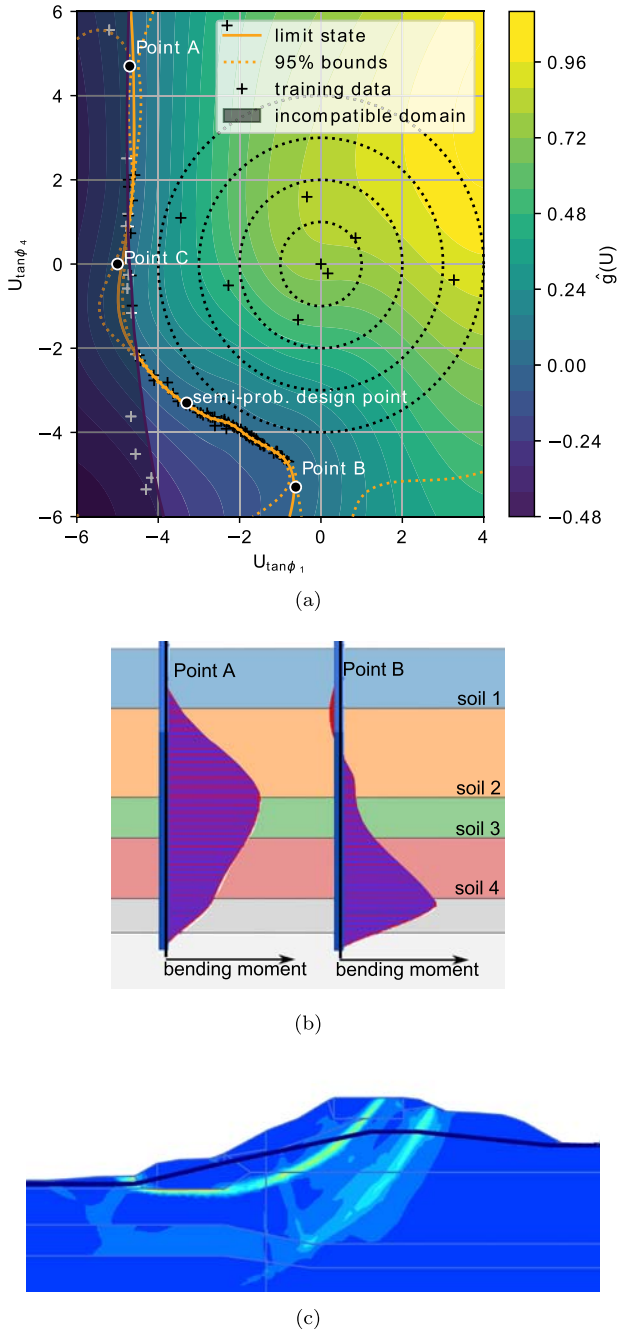


Figure 5. (a) 2D response surface with estimated limit state, limit state 95% confidence bounds, incompatible domain and training datapoints. (b) Bending moment profiles corresponding to Points A and B, with maximum bending moment $M_{max} \approx 279.2$ kNm/m' at both A and B. (c) Deviatoric strain contours indicating the failure mechanism for an incompatible realisation at Point C, in the initialisation phase before sheetpile wall installation.

4.3. Metamodel simulations with 4 and 10 stochastic variables

A series of reliability analyses is performed considering $\tan\phi'$ of the four layers as stochastic variables, while all other parameters are kept at the design values. The

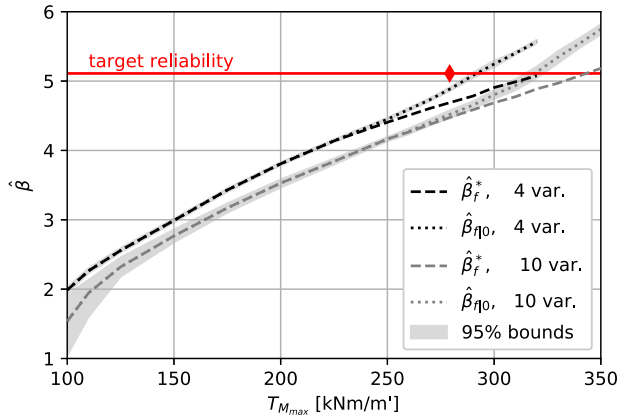


Figure 6. Reliability curves for the 4-variable and 10-variable simulations. The semi-probabilistic design point is indicated by the diamond marker.

reliability at different limit state thresholds is evaluated in 15 independent analyses, leading to the fragility curve $\hat{\beta} - T_{M_{max}}$ shown in Figure 6, which shows the estimates of the imputed and conditional probabilities. Metamodel uncertainty bounds based on $P_{f|0}^-$ and $P_{f|0}^+$ (Equation (6)) are indicated by shaded uncertainty bands, showing a well-converged model prediction for all 4-variable simulations.

The 4-variable metamodel gives an estimation of the probability of failure, conditional on the design values of the remaining six parameters. The 10-variable predicted probabilities, also shown in Figure 6, are significantly higher than the 4-variable predictions, demonstrating that including variables 5 to 10 as stochastic variables does not lower the probability of failure and thereby that the chosen design values for variables 5 to 10 in the 4-variable simulations are not conservative.

The prediction for the 10-variable analyses shows a significantly wider uncertainty range than for the 4-variable analyses. In fact, the metamodel learning algorithm is unable to reduce the prediction uncertainty caused by the noise in the response sufficiently to reach the requested convergence criterion of $\varepsilon_{P_f,\mathcal{M}} < 0.196$.

Figure 7(a) shows the evolution of the active learning prediction and its 95% metamodel confidence bounds as a function of model evaluations for the 4-variable and 10-variable simulations for $T_{M_{max}} = 270$ kNm/m', estimating a probability of failure of $P_{f|0} = 1.1 \times 10^{-6}$ and $P_{f|0} = 5.3 \times 10^{-6}$, respectively. Both analyses are based on initial training data from 30 model evaluations, after which the active learning scheme is performed until convergence of both the metamodels and MC integration, or until a maximum of 400 model evaluations is reached. In Figure 7(b), the uncertainties in the metamodel prediction and the MC integration are expressed in terms of back-calculated coefficients of

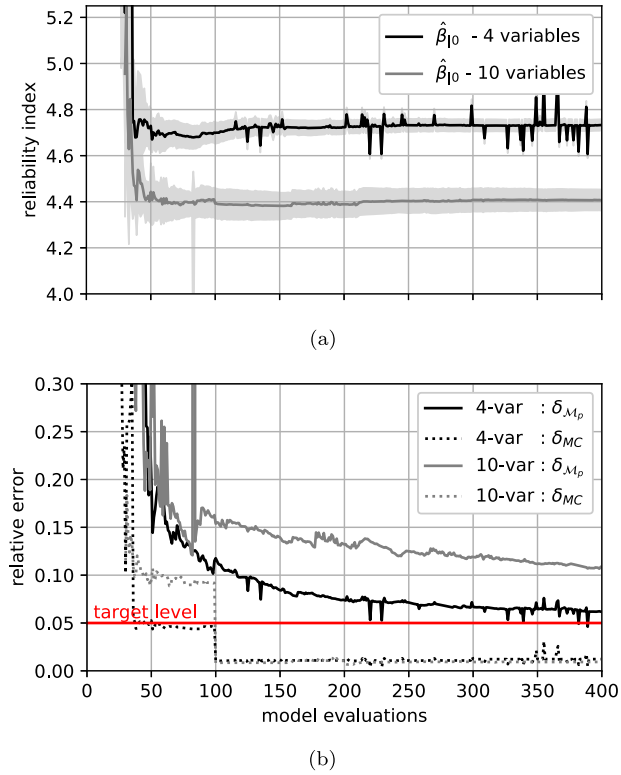


Figure 7. Model convergence for 4- and 10-variable analyses with $T_{M_{max}} = 270$ kNm/m'. (a) Reliability predictions and (b) prediction uncertainty expressed as equivalent coefficient of variation δ.

variation. In both cases shown, convergence of the prediction metamodel was not reached after 400 simulations. This non-convergence is due to the noise in the system response, modelled by the white noise term in the kernel, and the remaining prediction uncertainty can only be reduced by statistical averaging over a large number of training data. Despite this remaining uncertainty, the predictions stabilise after 100 to 150 model evaluations. The spikes in the 4-variable predictions are caused by instabilities in the training of the classification model, occasionally leading to for example underfitting and overfitting. These issues can generally be solved by more careful training of the metamodel (e.g. by setting constraints on the development of the hyperparameters, and increasing the number of learning iterations). The spikes represent artefacts of instabilities in the training of the classification model and are not representative for the classification uncertainty of a well-trained classification model. Note that the uncertainty bounds in Figure 7(a) do not account for uncertainty in the classification model.

Importance sampling refinement is performed, regardless of the metamodel convergence, after each set of 100 model evaluations, which leads to the jump in integration uncertainty visible at 100 model

evaluations. Only in a limited number of simulations has a second update been required to reach convergence of the integration accuracy.

4.4. Model verification

For computational models without incompatible domains and noise-free behaviour, the presented two-scale model behaves like the classical AK-MCS and AK-IS models, for which methodological verification can be found in literature (e.g. Echard, Gayton, and Lemaire 2011; Echard et al. 2013; Cadini, Santos, and Zio 2014). In the absence of established computational models incorporating noise and incompatible domains, the accuracy of the presented approach is here verified directly for the 10-variable computational model of the sheet pile wall problem using importance sampling. An importance distribution $q \sim \mathcal{N}(\vec{\mu}_{250}, \mathbf{I})$ was defined as a normal distribution around an approximate design point $\vec{\mu}_{250}$ for $T_{\mathrm{M}_{max}} = 250$ kNm/m'. A total of 10,000 importance sampling realisations were evaluated to estimate the relationship between $M_{\max}$ and $\beta_{\mathrm{f}\,|\,0}$, as well as separate solutions for β_1, to compare with the results of the two-stage metamodel.

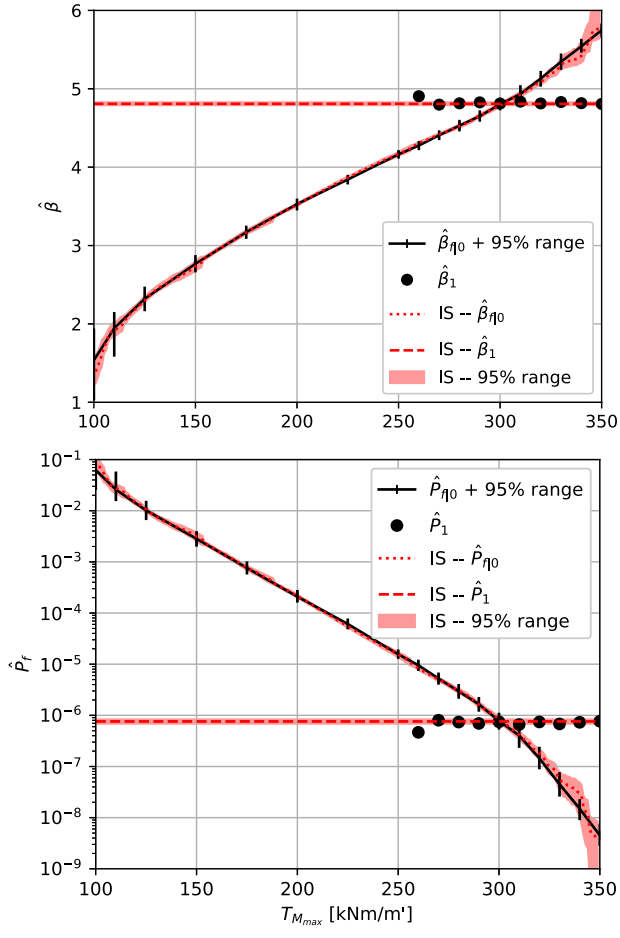


Figure 8. Verification of two-stage metamodelling approach for 10-variable problem against importance sampling solution, in terms of estimated reliability index $\hat{\beta}$ and estimated probability of failure $\hat{P}_{\mathrm{f}}$. Vertical line markers indicate the 95% confidence range of the results.

Figure 8 compares the results for the different metamodel analyses with those based on importance sampling. From the consistency between the results, it can be concluded that the proposed method leads to accurate and consistent estimations of the probability of failure. Although the potential bias introduced in the estimation by replacing the computational model with the metamodel has not been evaluated explicitly, the good consistency between the results obtained using metamodelling and importance sampling, respectively, indicate that the potential bias in the metamodel prediction is insignificant in comparison with the inherent metamodel prediction uncertainty expressed in the (conservative) uncertainty bounds based on Equation (6). For a more complete quantification of the potential bias in the estimation, without the need for many additional model evaluations, the evaluation of a bias correction term by means of for example leave-one-out verification could be considered (as for example suggested by Dubourg, Sudret, and Deheeger 2013). Such an additional verification could also be included in a stopping criterion, to reduce the potential for stopping the active learning algorithm too early as a result of for example underfitting of the limit state response.

4.5. Convergence and efficiency

The efficiency of the proposed method is assessed in a comparison against established methods for the reliability at $T_{\mathrm{M}_{max}} = 250\,\mathrm{kNm/m'}$, corresponding to $P_{\mathrm{f}\,|\,0} \approx 5.2 \times 10^{-6}$. The efficiency is expressed by the normalised coefficient of variation $\delta_n = \delta\sqrt{N}$, where N is the number of function evaluations required to reach the coefficient of variation δ. For Monte Carlo analysis, δ_n is based on Equation (7). For subset simulation (SS), a lower-bound solution for δ_n is given, under the assumption of fully independent samples and subset probability $p_0 = 0.10$ (Au and Beck 2001; van den Eijnden and Hicks 2017). For importance sampling, the analysis presented in Figure 8 is used, with δ_n based on $\delta = 0.062$ as evaluated by Equation (13) over N=10,000 realisations. For the two-stage metamodel presented in this work, the coefficient of variation of the estimation of $P_{\mathrm{f}\,|\,0}$ is defined based on Equation (13) at each active learning iteration, normalised against the number of function evaluations thus-far. Over the final 150 out of 200 iterations, δ_n varied around 1.2,

Table 2. Efficiency comparison for the evaluation of $P_{f|0}$ at $T_{M_{max}} = 250$ kNm/m′ on the basis of the normalised coefficient of variation δ_n and the relative number of realisations N_{relative}.

Method	δ_n	N_{relative}
Monte Carlo	447	138,880
Subset simulation (theoretical lower bound)	15.9	176
Importance sampling (using prior knowledge)	6.2	26.7
This study	~ 1.2	1

with values ranging from 0.9 to 1.4, for the 10-variable case.

Table 2 summarises the normalised coefficients of variation for comparison, from which it is clear that the metamodelling approach is substantially more efficient than the purely sampling-based methods for the presented problem. It must be noted that the value for metamodelling is a conservative error bound as discussed in Section 2.2, while for subset simulation the theoretical lower (i.e. unconservative) bound of δ_n is shown and the importance sampling analysis was started based on prior information on the location of the limit state. Despite this cautious approach towards quantitative comparison, it can be concluded that the metamodelling approach is orders of magnitude faster than the established methods. This is, however, under the condition that the computational load scales linearly with δ_n^2, which means that training the metamodel is relatively fast compared to the computational model. Also, the metamodelling approach cannot be combined effectively with distributed parallel computation strategies for massive parallel computations, since active learning is inherently sequential.

5. Reliability updating

Although the primary focus of the classification metamodel is to facilitate the learning of the prediction metamodel, the results presented in Figure 6 are consistent enough to formulate conditional probabilities. The classification metamodel is thereby used for conditioning to the feasible domain. The distinction between the feasible and the infeasible domain is not merely an artefact of the analysis, but can also be attributed a physical meaning. The infeasible domain represents realisations of the random variables implying loss of equilibrium of the investigated structure. If this loss of equilibrium occurs in the analysis of an existing structure, i.e. in a construction stage which was carried out with instability of the structure, we can regard the respective subset of the parameter space as implausible. In other words, we can update the probability of failure of the existing structure by taking into account the survival of observed load conditions during the execution of the construction works. This is similar to reliability updating with monitoring data, which typically involves equality-type information, whereas survival implies inequality-type information (Straub 2011). The concept is similar to the reliability updating for internal erosion of flood defences described in Schweckendiek, Vrouwenvelder, and Calle (2014).

In other words, this past performance information is evidence of feasible data ($I_c = 0$) and allows updating of the prediction of structural failure from $\hat{P}_f^*$ to $\hat{P}_{f|0}$. Figure 9 shows details of the reliability curves for the relevant range of bending moment thresholds, with the difference between $\hat{P}_f^*$ and $\hat{P}_{f|0}$ indicating the range for updating. This difference is up to a factor 20 in the calculated probability in the case of the 10-variable analysis, as indicated by the arrow. If no past performance information can be used, as for example with an entirely new structure, $\hat{P}_f^*$ reflects the reliability given the current state of knowledge, and possible failure in intermediate calculation stages can give information on the probability of failure during the various stages of construction. However, after successful construction (i.e. survival of all construction stages) the reliability corresponds again to the updated probability of failure $P_{f|0}$, with the analysis possibly enriched by monitoring data or performance observations obtained during construction.

6. Concluding remarks

The noisy and incomplete system response of a sheetpile wall has been modelled by means of a two-stage metamodel in the framework of kriging-based active learning for reliability analysis. With the successful application of the method in a reliability analysis that is representative

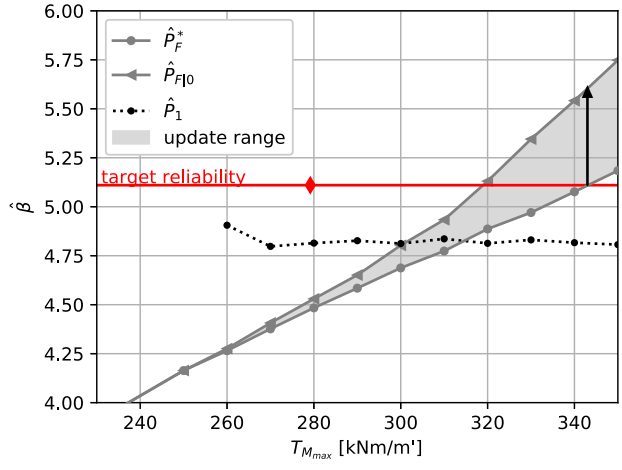


Figure 9. Details of the reliability curves for the 10-variable simulations. The arrow indicates the potential for reliability updating based on knowledge of incompatible modes.

for a class of problems in geotechnical engineering practice, the following can be concluded:

- Although the general concept of AK-MCS is conveniently extended to deal with small probabilities and moderate noise levels through combining with importance sampling and kernel enhancements, a consistent updating of its learning functions and convergence criteria is needed. A variation of the U-learning function has therefore been proposed for better consistency with the case of importance sampling and models with noise components. This variation does not introduce additional computational complexity.
- The obstructive effects of missing model response data in kriging-based active learning algorithms can effectively be mitigated by the introduction of an additional classification model. While the primary objective of this additional classification is to predict domains of missing data that are to be avoided in the active learning of the predicting metamodel, the additional information is also valuable in the model interpretation.
- When missing data are linked to incompatibility between input parameters and model predictions of observed data, the information of the classification metamodel can be used in the context of reliability updating. If incompatible realisations are interpreted as failures in the computational model, even though the structure in question has actually survived the simulated load conditions, the reliability can be based on the compatible domain only, which is equivalent to Bayesian updating with survival information. Similar approaches can be considered for including other types of monitoring data or performance observations. Depending on the type of data or evidence, a classification or a predictive model can be more appropriate, for example in the case of equality or inequality information (Straub 2011).
- Although the white noise component in the numerical model prevents complete convergence of the predictive model uncertainty, the consistency between results in different (independent) simulations demonstrates the accuracy of the proposed method.

Disclosure statement

No potential conflict of interest was reported by the author(s).

ORCID

A. P. van den Eijnden http://orcid.org/0000-0003-0941-9521
T. Schweckendiek http://orcid.org/0000-0002-8292-595X
M. A. Hicks http://orcid.org/0000-0002-7603-7040

References

Al-Bittar, T., A. H. Soubra, and J. Thajeel. 2018. "Kriging-based Reliability Analysis of Strip Footings Resting on Spatially Varying Soils." *Journal of Geotechnical and Geoenvironmental Engineering* 144: 04018071.

Au, S. K., and J. L. Beck. 2001. "Estimation of Small Failure Probabilities in High Dimensions by Subset Simulation." *Probabilistic Engineering Mechanics* 16: 263–277.

Balesdent, M., J. Morio, and J. Marzat. 2013. "Kriging-based Adaptive Importance Sampling Algorithms for Rare Event Estimation." *Structural Safety* 44: 1–10.

Bourinet, J. M. 2016. "Rare-event Probability Estimation with Adaptive Support Vector Regression Surrogates." *Reliability Engineering & System Safety* 150: 210–221.

Brinkgreve, R. B. J., S. Kumarswamy, W. M. Swolfs, and F. Foria. 2018. *PLAXIS 2018 (Users Manual)*. Delft: Plaxis bv.

Cadini, F., F. Santos, and E. Zio. 2014. "An Improved Adaptive Kriging-based Importance Technique for Sampling Multiple Failure Regions of Low Probability." *Reliability Engineering and System Safety* 131: 109–117.

Cressie, N. A. C. 1993. *Statistics for Spatial Data*. New York: John Wiley & Sons, INC.

Depina, I., T. M. H. Le, G. Fenton, and G. Eiksund. 2016. "Reliability Analysis with Metamodel Line Sampling." *Structural Safety* 60: 1–15.

Dubourg, V., B. Sudret, and J. M. Bourinet. 2011. "Reliability-based Design Optimization Using Kriging Surrogates and Subset Simulation." *Structural and Multidisciplinary Optimization* 44: 673–690.

Dubourg, V., B. Sudret, and F. Deheeger. 2013. "Metamodel-based Importance Sampling for Structural Reliability Analysis." *Probabilistic Engineering Mechanics* 33: 47–57.

Echard, B., N. Gayton, and M. Lemaire. 2011. "AK-MCS: An Active Learning Reliability Method Combining Kriging and Monte Carlo Simulation." *Structural Safety* 33: 145–154.

Echard, B., N. Gayton, M. Lemaire, and N. Relun. 2013. "A Combined Importance Sampling and Kriging Reliability Method for Small Failure Probabilities with Time-demanding Numerical Models." *Reliability Engineering & System Safety* 111: 232–240.

El Haj, A. K., A. H. Soubra, and T. Al-Bittar. 2019. "Probabilistic Analysis of Strip Footings Based on Enhanced Kriging Metamodeling." *International Journal for Numerical and Analytical Methods in Geomechanics* 43 (17): 1–20.

EN:1997. 2004. Eurocode 7: Geotechnical Design – Part 1: General rules. CEN.

Forrester, A., A. Sobester, and A. Keane. 2008. *Engineering Design Via Surrogate Modelling: A Practical Guide*. Chichester: John Wiley & Sons.

Hasofer, A. M., and M. D. Lind. 1974. "An Exact and Invarient First Order Reliability Format." *Journal of Engineering Mechanics* 100: 111–121.

Hu, Z., and S. Mahadevan. 2016. "Global Sensitivity Analysis-enhanced Surrogate (GSAS) Modeling for Reliability Analysis." *Structural and Multidisciplinary Optimization* 53: 501–521.

Kang, F., S. Han, R. Salgado, and J. Li. 2015. "System Probabilistic Stability Analysis of Soil Slopes Using Gaussian Process Regression with Latin Hypercube Sampling." *Computers and Geotechnics* 63: 13–25.

Kanning, W., A. Teixeira, M. van der Krogt, K. Rippi, T. Schweckendiek, and B. Hardeman. 2017. Calibration of Factors of Safety for Slope Stability of Dikes." In *Geo-Risk*. Part V, edited by J. Huang, G. A. Fenton, Limin Zhang, and D. V. Griffiths, Denver, Colorado.

Li, D. Q., D. Zheng, Z. J. Cao, X. S. Tang, and K. K. Phoon. 2016. "Response Surface Methods for Slope Reliability Analysis: Review and Comparison." *Engineering Geology* 203: 3–14.

Little, R. J. A., and D. B. Rubin (2002). *Statistical Analysis with Missing Data*. Wiley Series in Probability and Statistics. New York, NY: Wiley.

Melchers, R. E. 1989. "Importance Sampling in Structural Systems." *Structural Safety* 6: 3–10.

Melchers, R. E., and A. T. Beck. 2018. *Structural Reliability Analysis and Prediction*. Chichester: John Wiley & Sons.

Mollon, G., D. Dias, and A. H. Soubra. 2009. "Probabilistic Analysis of Circular Tunnels in Homogeneous Soil Using Response Surface Methodology." *Journal of Geotechnical and Geoenvironmental Engineering*135: 1314–1325.

Noordam, A. F. 2019. *Verkennende Faalkansanalyse Stabiliteitswanden – POVM Actuele Sterkte – Activiteit 9*. Technical Report. Deltares, The Netherlands. https://www.povmacrostabiliteit.nl/rapporten. In Dutch.

Phoon, K. K., and C. Tang. 2019. "Characterisation of Geotechnical Model Uncertainty." *Georisk: Assessment and Management of Risk for Engineered Systems and Geohazards* 13: 101–130.

Rasmussen, C. E., and C. K. I. Williams. 2006. *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press.

Sacks, J., W. J. Welch, T. J. Mitchell, and H. P. Wynn. 1989. "Design and Analysis of Computer Experiments." *Statistical Science* 4: 409–423.

Santner, T. J., B. J. Williams, W. Notz, and B. J. Williams. 2003. *The Design and Analysis of Computer Experiments*. Vol. 1. New York: Springer.

Schanz, T., P. A. Vermeer, and P. G. Bonnier. 1999. The Hardening Soil Model: Formulation and Verification." In Beyond 2000 in Computational Geotechnics, edited by R. B. J Brinkgreve, 281–296. London: Routledge.

Schöbi, R., B. Sudret, and S. Marelli. 2016. "Rare Event Estimation Using Polynomial-chaos Kriging." *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering* 3: D4016002.

Schweckendiek, T., A. C. W. M. Vrouwenvelder, and E. O. F. Calle. 2014. "Updating Piping Reliability with Field Performance Observations." *Structural Safety* 47: 13–23.

Schweckendiek, T., A. C. W. M. Vrouwenvelder, E. O. F. Calle, W. Kanning, and R. Jongejan. 2013. "Target Reliabilities and Partial Factors for Flood Defenses in the Netherlands." In *Modern Geotechnical Design Codes of Practice*, edited by P. Arnold, G. A. Fenton, M. A. Hicks, T. Schweckendiek, B. Simpson, 311–328. Lansdale: IOS Press.

Sivakumar Babu, G. L., and A. Srivastava. 2007. "Reliability Analysis of Allowable Pressure on Shallow Foundation Using Response Surface Method." *Computers and Geotechnics* 34: 187–194.

Soubra, A. H., T. Al-Bittar, J. Thajeel, and A. Ahmed. 2019. "Probabilistic Analysis of Strip Footings Resting on Spatially Varying Soils Using Kriging Metamodeling and Importance Sampling." *Computers and Geotechnics* 114: 103107.

Straub, D. 2011. "Reliability Updating with Equality Information." *Probabilistic Engineering Mechanics* 26: 254–258.

Teixeira, R., M. Nogal, and A. O'Connor. 2021. "Adaptive Approaches in Metamodel-based Reliability Analysis: A Review." *Structural Safety* 89: 102019.

van den Eijnden, A. P., and M. A. Hicks. 2017. "Efficient Subset Simulation for Evaluating the Modes of Improbable Slope Failure." *Computers and Geotechnics* 88: 267–280.

Vermeer, P. A., and H. P. Neher. 1999. A Soft Soil Model That Accounts for Creep." In *Beyond 2000 in Computational Geotechnics*, edited by R. B. J. Brinkgreve, 249–261. London: Routledge.

Zhu, C., R. H. Byrd, P. Lu, and J. Nocedal. 1997. "Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-scale Bound-constrained Optimization." *ACM Transactions on Mathematical Software (TOMS)*23: 550–560.

Appendices

Appendix 1. Multiple design point Gaussian mixture importance sampling

Starting from a metamodel and a sample pool $\mathbf{u}_{\mathrm{MC}}$ with corresponding metamodel predictions $\hat{I}_c$ and $\hat{g}$, the importance resampling distribution is generated as follows:

(1) select $\mathbf{u}_S$ as the subset of all points $\mathbf{u}_{\mathrm{MC}}$ for which $(\hat{I}_c = 1 \vee \hat{g} < 0)$
(2) select design point $\vec{u}_1^{\mathrm{des}}$ from $\mathbf{u}_S$ as the point with lowest norm $|\vec{u}|$
(3) define angle ω_i as the radial angle between a sample $\vec{u}$ and quasi-design point $u^{\vec{\mathrm{des}}}{}_i$ as

$$\omega_i = \cos^{-1}\left(\frac{\vec{u}\cdot u^{\vec{\mathrm{des}}}{}_i}{|\vec{u}||u^{\vec{\mathrm{des}}}{}_i|}\right) \tag{A1}$$

(4) iteratively select new design points $u^{\vec{\mathrm{des}}}{}_{k+1}$ from $\mathbf{u}_S$ for which $\omega_{i=1..k} > \omega_T$, until no more design points can be found
(5) repeat step 4 for different threshold angles ω_T to find the smallest ω_T such that exactly k quasi-design points are found (k=20 is used here)
(6) compute normalised weights w_i as

$$\lambda_i = \frac{\Phi(-|u^{\vec{\mathrm{des}}}{}_i|)}{\sum_{j=1}^{k}\Phi(-|u^{\vec{\mathrm{des}}}{}_j|)} \tag{A2}$$

(7) define a multiple importance distribution model with PDF

$$q_{\mathrm{mult}}(\vec{u}) = \sum_{i=1}^{k}\lambda_i q_i(\vec{u}) = \sum_{i=1}^{k}\lambda_i p(\vec{u}|\vec{\mu} = u^{\vec{\mathrm{des}}}{}_i, \boldsymbol{\sigma}^2 = I) \tag{A3}$$

where $p(\vec{u}|\vec{\mu}, \sigma^2)$ is the multivariate normal PDF.

The multiple importance distribution model $q_{\text{mult}}(\vec{u})$ is then used for the importance sampling of a new MC integration sampling pool $\mathbf{u}_{\text{MC}}$, with corresponding importance weights defined as:

$$w_{\text{imp}} = \frac{q_{\text{mult}}(\vec{u})}{p(\vec{u} \,|\, \vec{\mu} = \vec{0},\, \boldsymbol{\sigma}^2 = \mathbf{I})} \tag{A4}$$

Appendix 2. Variance after updating

Starting from an experimental design $\{\vec{g}_{\text{t}}, \mathbf{u}_{\text{t}}\}$ and a prediction $g_{\text{p}}(\vec{u}_{\text{p}})$ at sample location $\vec{u}_{\text{p}}$, the GP metamodel is formulated as a realisation of a multivariate normal distribution, as formulated in Equations (4) and (5):

$$\hat{g} = \mathbf{K}_{\text{tp}}^{\top}\mathbf{K}_{\text{tt}}^{-1}\vec{g}_{\text{t}} \tag{A5}$$

$$\sigma_{\hat{g}}^2 = \sigma^2 - \mathbf{K}_{\text{tp}}^{\top}\mathbf{K}_{\text{tt}}^{-1}\mathbf{K}_{\text{tp}} \tag{A6}$$

When g_{p} is evaluated and added to the training data, the variance at $\vec{u}_{\text{p}}$ reduces, giving a new non-zero prediction variance. For evaluating the new prediction variance at this point, the updated situation is written in terms of its multivariate normal distribution with a subsequent prediction at the same point $\vec{u}_{\text{p}}$:

$$\begin{pmatrix} \vec{g}_{\text{t}} \\ \hat{g} \\ g_{\text{p}} \end{pmatrix} \sim N\left(\vec{0},\, \begin{bmatrix} \mathbf{K}_{\text{tt}} & \mathbf{K}_{\text{tp}} & \mathbf{K}_{\text{tp}} \\ \mathbf{K}_{\text{tp}}^{\top} & \sigma^2 + \sigma_{\text{wn}}^2 & \sigma^2 \\ \mathbf{K}_{\text{tp}}^{\top} & \sigma^2 & \sigma^2 \end{bmatrix}\right) \tag{A7}$$

In the same way as above, the variance is written as

$$\sigma_{\hat{g}+1}^2 = \sigma^2 - \begin{bmatrix} \mathbf{K}_{\text{tp}}^{\top} & \sigma^2 \end{bmatrix} \begin{bmatrix} \mathbf{K}_{\text{tt}} & \mathbf{K}_{\text{tp}} \\ \mathbf{K}_{\text{tp}}^{\top} & \sigma^2 + \sigma_{\text{wn}}^2 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{K}_{\text{tp}} \\ \sigma^2 \end{bmatrix} \tag{A8}$$

Some algebraic manipulations and using Equation (A6) then leads to

$$\begin{aligned} \sigma_{\hat{g}+1}^2 &= \sigma^2 - \begin{bmatrix} \mathbf{K}_{\text{tp}}^{\top} & \sigma^2 \end{bmatrix} \begin{bmatrix} \mathbf{K}_{\text{tt}}^{-1}\mathbf{K}_{\text{tp}}\left(1 - \frac{\sigma_{\hat{g}}^2}{\sigma_{\hat{g}}^2 + \sigma_{\text{wn}}^2}\right) \\ \frac{\sigma_{\hat{g}}^2}{\sigma_{\hat{g}}^2 + \sigma_{\text{wn}}^2} \end{bmatrix} \\ &= \sigma_{\hat{g}}^2 \frac{\sigma_{\text{wn}}^2}{\sigma_{\hat{g}}^2 + \sigma_{\text{wn}}^2} \end{aligned} \tag{A9}$$

This means that the relative variance reduction at any added point is given by $\frac{\sigma_{\text{wn}}^2}{\sigma_{\hat{g}}^2 + \sigma_{\text{wn}}^2}$.