Reliability analysis of quay walls using metamodelling

by

J. van der Werf

in fulfilment of the requirements for the degree of
Master of Science
at the Delft University of Technology,
to be defended publicly on September 16, 2021.

Student number:	4520394	
Project duration:	March, 2020 – Month Day, Year	
Committee:	Dr. ir. Mandy Korff (Chair)	*TU Delft*
	Dr. ir. Timo Schweckendiek	*Deltares & TU Delft*
	Ir. Mark Post	*Deltares*
	Dr. ir. Bram van den Eijnden	*TU Delft*
	Dr. ir. Alfred Roubos	*Port of Rotterdam*

Delft University of Technology
Faculty of Civil Engineering and Geo-Scinces
Section Geo-Engineering, Civil Engineering

Preface

This thesis is written to fulfil the graduation requirement of the Master of Science in Geotechnical Engineering at the faculty of Civil Engineering and Geosciences. The research, titled "Reliability analysis of quay walls using metamodelling" was conducted at Deltares, in agreement with TU Delft and the Port of Rotterdam. Metamodelling has the potential to efficiently perform a reliability analysis where conventional methods become infeasible. This thesis focuses on the applicability of the metamodelling method ERRAGA on reliability based assessments of quay walls within two case studies in the port of Rotterdam. The outcome of this research shows how ERRAGA performs on the case studies compared to conventional probabilistic methods and give an insight in the working of ERRAGA and how to use the method for engineering practice for reliability assessment of quay walls.

I would like to thank my committee for their guidance, support and patience. My gratitude goes to Mandy Korff who helped me find this research topic and for all the constructive feedback and motivation during the writing of this thesis. I would like to thank my daily supervisors Mark Post and Timo Schweckendiek for the support and sharing your knowledge during the one on one feedback sessions. Mark, without you I would not have been able to perform reliability analysis on this level, I am grateful for all the ideas and guidance you gave me. Timo, you always provided me with new insights during our meetings, your knowledge and research skills were a great help. I want to thank Bram van den Eijnden for all the feedback and insights you gave me about ERRAGA. Alfred Roubos, thanks you for the valuable feedback you provided me during the progress meetings.

I am grateful to my fellow students and all the people from the university I met during my study period in Delft. A final thank goes out to my family and friends for all the support and motivation they gave me during my studies.

Jilles van der Werf
Amsterdam, September 2021

Summary

Reliability analysis is a rational method for dealing with uncertainties. It is increasingly used for the design and assessment of (civil) structures. Reliability analysis, possibly in combination with performance data, allows for instance to update (extend) the lifetime of existing quay wall structures. This gives economic advantages when assessing quay walls, figuring there are thousands of kilometres of quay walls worldwide. Soil-structure analysis (e.g. analysis of quay walls) is however complex and in general requires finite element (FE) models. Conventional reliability methods are mostly incapable of dealing with FE models because they have long (infeasible) calculation times e.g. Monte Carlo, or they cannot deal with the typical noisy and incomplete FE model output e.g. FORM. One way of dealing with these challenges is by using a metamodelling approach, i.e. build up and make use of a response surface on a limited number of model evaluations. One specific application which uses metamodelling is ERRAGA, which is an abbreviation for Efficient and Robust Reliability Analysis for Geotechnical Applications. This thesis investigates the potential of ERRAGA/metamodelling for reliability analysis of quay walls in engineering practice.

In this thesis ERRAGA is tested on two realistic case studies located in the port of Rotterdam. The first case is called the 'Sleepbootkade', a quay constructed from a combi wall with anchors. The second case study is called the HHTP-quay, a combi wall with relieving platform and MV-piles. In the first case focus is placed on getting to understand the details and workings of the ERRAGA method. In the second case focus has been put on the creation of the reliability model and the practical applicability of ERRAGA using the experiences of the first case. Two critical limit states are tested within the case studies: wall failure in bending and geotechnical failure. Especially the limit state geotechnical failure is challenging as relevant output parameters are known to be noisy and/or incomplete.

The main finding of this thesis is that reliability analysis of quay walls using ERRAGA in combination with a numerical model shows good potential. ERRAGA can generate reliable and accurate results within only hundreds of realizations, hereby overcoming the main drawbacks of existing reliability methods. Main drawback of the ERRAGA method seems to be its inherent complexity and 'black box' nature. For the user to employ the full potential of the method some more in depth knowledge and especially the user-defined settings are required. Based on the gained experiences in this thesis some first recommendations are presented for the use and application of ERRAGA in projects.

List of Figures

Contents

Chapter 1

Introduction

1.1 Problem definition

Like most structures, quay walls are built with the requirement to fulfil their services for a minimum period of time (lifetime). However, in many cases, quay walls are designed stronger than minimally required, which means that the quay wall is able to serve its purpose over a longer timespan or is able to endure higher loads. Optimizing the design of a quay wall, or - in the case of an existing quay wall - to compute the extension of lifetime or the ability to endure higher loads, can be substantiated by using probabilistic design.

The use of probabilistic design for the purpose of performing a reliability analysis of quay walls as described above has several implications. To start, the interaction between soil and quay walls can only be modelled accurately by using a numerical model. The output of numerical models has several complexities. It can be noisy, incompatible and non-linear. In addition, the numerical model requires an extensive computation time. Furthermore, in order to model a realistic case, often a number of relevant random variables often between 10-20 is required. This makes the conventional probabilistic methods, such as Monte Carlo Simulations (MCS) or Importance Sampling (IS) infeasible due to the extensive computation time. Methods such as FORM are neither suitable, because they are unable to deal with the mentioned complexities which consist of data that is noisy, incompatible or non-linear. In short, there is lack of a practical and easily applicable tool.

Metamodelling may fill this gap, as it is expected to reduce the computation time drastically and is able to deal with the complexities connected to the use of a numerical model. ERRAGA is such a metamodelling method. ERRAGA trains and fits a computational less-expensive replacement of the performance function by using Kriging (or Gaussian process regression). It only searches for data points in the parameter space which are expected to add significantly to the overall failure probability. This reduces the amount of the numerical simulations needed.

Although ERRAGA seems to be a promising tool, insight to the use thereof is currently lacking. It is sometimes perceived as a black box, because it is rather new and complex. It also has a large amount of settings which can highly influence the results and computation time. This means that further insight into ERRAGA is required before its use for reliability analyses can be implemented on a larger scale.

1.2 Research questions

This paragraph states the objectives of this thesis and to fulfil these objectives, main- and sub questions are formulated.

The following objectives are formulated:

- 1: To gain insight in the concept of reliability based design and assessment of quay walls.
- 2: To test the applicability of metamodelling for reliability analysis of quay walls and where necessary further develop the method.

The research question and the sub-questions are listed below.

What is the potential of metamodelling for reliability-based assessment of quay walls in engineering practice?

1. What are essential parts of the reliability assessment of quay walls?
2. Which properties of reliability assessments of quay walls make reliability analysis with conventional methods challenging?
3. How does a metamodelling approach perform in realistic cases compared to conventional methods on the aspects of robustness, efficiency and accuracy?
4. What are the recommended 'user defined' settings for metamodelling to obtain optimal performance?

1.3 Scope

In order to answer the research questions, the following approach is handled. To start, this research aims to provide insight in reliability analyses of quay walls in combination with metamodelling, more specifically ERRAGA. It is discussed i) what is needed to perform a reliability analysis of quay walls and ii) how metamodelling works and iii) how are these two elements are combined.

As a part of reliability analysis, the different failure mechanisms must be defined. The definitions of such failure mechanisms is described by a limit state function (LSF). Only the most critical and thus relevant limit states have to be evaluated. Therefore, it is important to gain insight in which LSF's are relevant and how they can be formulated realistically. In addition, the influential parameters must be selected, which are represented by a variable in the reliability analysis.

This results in the assessment of two case studies located in the port of Rotterdam. The first case study is referred to as Sleepbootkade and consists of a combi-wall with anchors. This case study focuses on gaining experience with the metamodel and its user specified settings. This case study has started prior to this thesis by the GRAPA project (Geotechnical Reliability Analysis for Practical Applications).

The second case study, located at the Maasvlakte, is called the HHTP-quay. It consists of a combi-wall with relieving platform and MV-piles with a more complex soil-structure interaction. The goal of case two is to implement the gained knowledge of the first case study in order to perform a reliability analysis on a more complex structure.

Due to the exploratory nature of this research, the numerical models are simplified to reduce the amount of variables and computation time of the numerical model. As a result, both case studies do not fully reflect the factu+al circumstances of the quay wall. However, this research has the intention to approach the actual situation as closely as possible given the scope of the research. Therefore, all components used in these case studies will be present in a fully realistic case study as well.

Lastly, to analyse the convergence behaviour of ERRAGA's reliability analysis, a dump file is implemented that provides detailed information about each model evaluation with a focus on the degree of convergence. In this thesis, this information from the dumpfile is used. In a later stadium there is also a visualisation implemented to show ERRAGA's convergence, which provides information about ERRAGA's reliability analysis. This is elaborated upon further in this report.

1.4 Outline of Report

This thesis is structured as follows. Chapter 2 discusses the theory regarding the reliability of quay walls and elaborates on the LSF's used in the case studies. In chapter 3, the covers the functioning and the attributes of the metamodel and how reliability analyses are performed in combination with the numerical model. Chapter 4 introduces the first case study, whereby first a reduced two-variable model is analysed followed by an analysis of a more extensive model. Then, chapter 5 describes the second case study. Geotechnical decisions were made for the set-up of the reliability analysis. The obtained insights of the first case study are used to perform a reliability analysis with metamodelling on case study 2. In chapter 6 conclusions are drawn from the case study results and these findings are used to formulate answers to the research questions. This is followed by several points of discussion, a list of original contributions and some final recommendations.

1.5 Original contributions

For the execution of this research, the following steps have been taken:

- The output of a dumpfile is implemented in the ERRAGA code which is used to obtain detailed information about the convergence of the runs.
- A detailed visualization of convergence criteria, noise term and hyperparameters (length scale) is developed.
- Case 1 has been reduced to a two-variable problem and it has been investigated what complexities cause non-convergence.
- ERRAGA is tested on two case studies.
- ERRAGA is tested on a artificial model.
- The numerical model of case 2 has been simpliefied and a reliability analysis has been set up (including determination of the (stochastic) variables and adjusting the communicating files).
- The problem for unjustified convergence after the IS update has been solved.

Chapter 2

Reliability of Quay Walls

To determine the reliability of quay walls, different failure modes are assessed. This chapter describes which failure modes are most relevant for the reliability assessment of quay walls and are therefore further elaborated upon in this thesis. Furthermore, the challenges related to probabilistic reliability analysis for limit states are discussed.

2.1 Types of quay walls

Quay walls are soil retaining marine structures for ships to dock and then (un)load their cargo or passengers. There are many different types of quay walls. De Gijt and M.L.Broeken, 2003 divide them into four main types:

1. Gravity walls
2. Sheet pile walls
3. Sheet pile walls with relieving platforms
4. Open berth quays

Ruggeri et al., 2019 illustrates the four types of quay walls as depicted below.

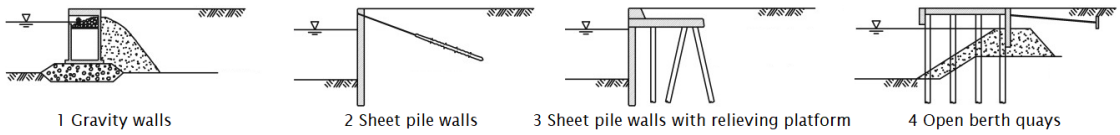


Figure 2.1: Four main quay wall types, (Ruggeri et al., 2019)

The two case studies in this thesis contain respectively a sheet pile wall and a sheet pile wall with relieving platform. Hence, this thesis solely focuses on these two types.

2.2 Failure modes

A quay wall can fail in many different ways, which means that multiple limit states have to be considered. However, not all limit states can be analysed with a numerical model (e.g. internal erosion) and some have little influence and are therefore irrelevant. This means only the possible and most critical limit states are considered in the reliability analysis. This section reviews what failure modes are defined in literature and discusses which are most relevant.

An overview of all possible failure mechanisms is generally displayed in a fault tree. An example of a fault tree is shown in Figure 2.2. Since this is a highly detailed overview, a more general list is shown in Table 2.1. This table considers sheet pile walls with 13 different failure modes and sheet pile walls with relieving platforms with 12 different failure modes.

Failure modes (geotechnical and structural)	**Sheet pile wall**	**Sheet pile wall with relieving platforms**
vertical pile bearing capacity (compression)	x	x
vertical pile bearing capacity (tension)		x
horizontal bearing force subsoil	x	
horizontal soil resistance	x	x
vertical soil fracture (heave)	x	
tension resistance anchorage	x	x
overall stability	x	x
local stability/high sliding plane (e.g. Kranz)	x	
structural strength wall	x	x
structural strength piles		x
structural strength anchorage	x	x
structural strength other elements	x	x
failure trough very large deformations	x	x
internal erosion	x	x
under and back seepage and piping	x	x

Table 2.1: Failure mechanisms for sheet pile walls and sheet pile walls with relieving platforms (De Gijt & M.L.Broeken, 2003), Table 6.2

Previous studies have reduced the list of Table 2.1, because not all the failure modes are relevant for a reliability analysis. Only a few are dominant and determine the reliability of the quay wall. Wolters, 2012 uses four failure modes for reliability analyses of a sheet pile wall and a sheet pile wall with relieving platforms:

1. ULS: Anchor failure
2. ULS: Wall failure in bending
3. SLS: Excessive deformations
4. ULS: Global geotechnical failure

In his research, it appeared that global geotechnical failure is the most critical mode for the reviewed cases. This is partly caused by the use of too short sheet piles. Furthermore there is a strong correlation between global geotechnical failure and wall failure in bending, as both modes are mainly caused by the failure of the passive soil wedge. In case of long sheet piles, global geotechnical failure is becoming less critical, because then wall failure in bending will become the dominant failure mode.

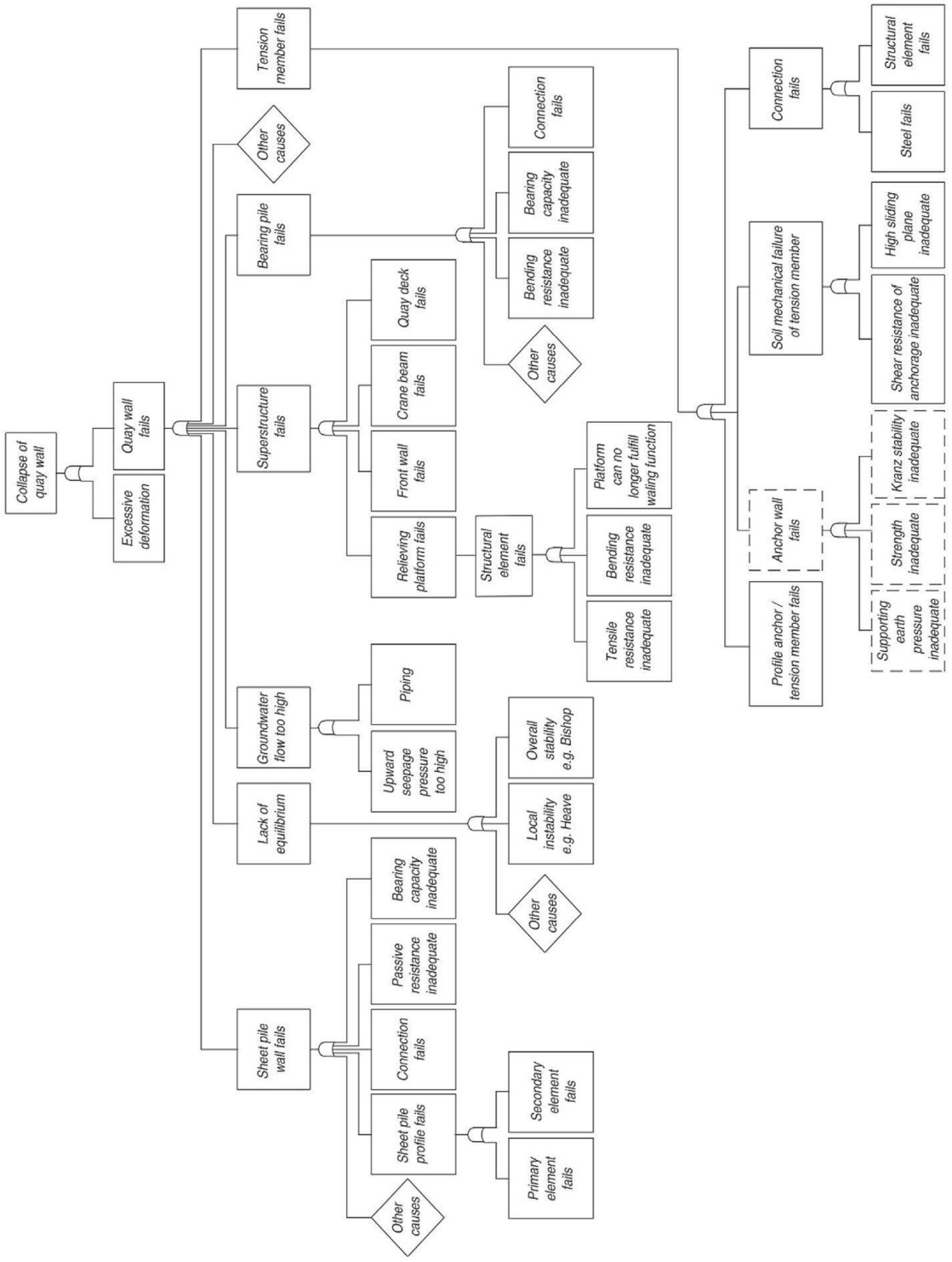


Figure 2.2: An example of a fault tree for a quay wall with a relieving platform De Gijt and M.L.Broeken, 2003, Figure 6.11

Roubos, 2019 evaluates the most relevant failure modes in terms of reliability (Figure 2.3):

1. Wall failure by yielding $Z_{STR;yield}$
2. Wall failure by buckling $Z_{STR;buckling}$
3. Anchor failure $Z_{STR;anchor}$
4. Grout failure $Z_{STR;grout}$
5. Geotechnical failure $Z_{GEO;globalsafety}$

An important finding is that the reliability level of a structural component is generally dominated by one specific failure mode. The structural failure modes of the retaining wall play an important role for combi-walls (Figure 2.3 A) and the geotechnical stability has been found to be an important failure mode for quay walls with a relieving platform (Figure 2.3 D1, D2, D3). The reliability of anchor systems seems to be fairly high, which means it is not considered to be a dominant failure mode (Roubos, 2019). The anchor has a higher reliability because it should not fail first, because the failure of an anchor is very sudden and gives no visible warning. On the contrary, for instance geotechnical failure will give these warnings, as deformations will be visible prior to failure.

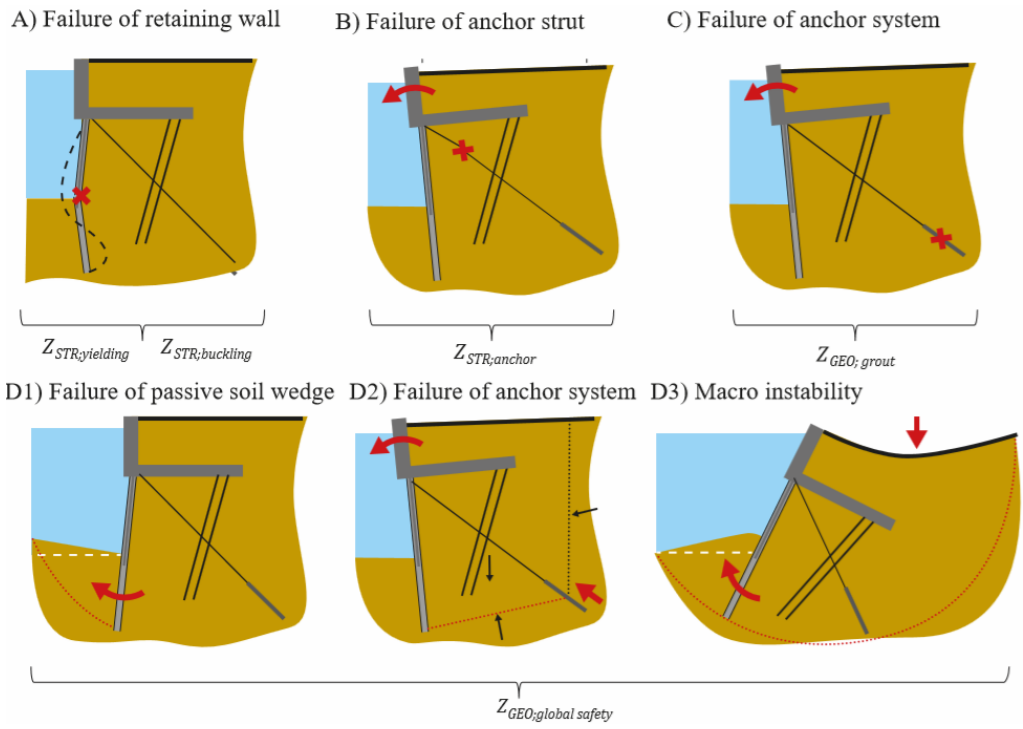


Figure 2.3: Most relevant failure modes of quay walls in terms of reliability (Roubos, 2019)

The relevant failure modes are categorised in three main modes, some divided in submodes:

1. Front wall failure
    - Excessive deformation $Z_{SLS;deformation}$
    - Wall failure by yielding $Z_{STR;yield}$
    - Wall failure by buckling $Z_{STR;buckling}$
2. Anchor failure
    - Anchor strut failure $Z_{STR;anchor}$
    - Grout failure $Z_{GEO;grout}$
3. Geotechnical failure $Z_{GEO;globalsafety}$ (or Z_{GEO})

Geotechnical failure has many different failure modes in practice, however in theory the failure mode accounts for all geotechnical limit states. It is important to note that the limit states are represented by limit state functions (Z-functions).

Both Wolters, 2012 and Roubos, 2019 conclude that global geotechnical failure and failure of the retaining wall by yielding are dominant failure modes. Wall failure by yielding $Z_{STR;yield}$ is a well-known limit state and is widely used in previous research (Roubos, 2019). Post et al., 2021 was able to solve $Z_{STR;yield}$ with FORM. Therefore, this limit state is suitable to use as a starting point for the case studies. Z_{GEO} is known to be more challenging in terms of convergence, thus the metamodelling-based approach can show its value here if it can determine the reliability of this limit state. Therefore, this thesis focusses on the limit states $Z_{STR;yield}$ and Z_{GEO}.

2.3 Limit state functions

This section discusses the limit state functions (LSF) for wall failure by yielding $Z_{STR;yield}$ and global geotechnical failure Z_{GEO}, which have been reasoned in 2.2 to be the most important LSF's to consider. Multiple formulations of the limit state functions are compared, with the goal to select the most appropriate limit state functions in terms of being the most robust in reliability analysis while adequately modelling the limit state. Furthermore, this section looks into the dominant uncertainties, which will help choosing the important parameters to be considered as random variables.

2.3.1 Wall failure by yielding

This limit state is defined as the yield stress of the retaining wall that is exceeded by the maximum stress, which occurs in the wall caused by the bending moment and the axial stress. Wolters, 2012 takes the maximum bending moment and maximum axial stress regardless of the depth to calculate the maximum stress. This is considered a conservative approach. Schweckendiek, 2006 takes a more realistic approach and looks at the maximum stress in the wall, based on depth. In addition Roubos, 2019 added two factors to account for model uncertainties for bending moments and axial forces:

$$Z_{STR;yield} = f_y - max\left(\frac{\theta_M M_{wall}(z)}{W_{wall}} + \frac{\theta_N N_{tube}(z)}{A_{tube}}\right) \tag{2.1}$$

Where

$Z_{STR;yield}$	Limit state function of wall failing by yielding [kN/m2]
f_y	Yield stress steel, combi-wal [kN/m2]
$M_{wall}(z)$	Bending moment, combi-wall [kNm/m]
$N_{tube}(z)$	Axial force, combi-wall [kN/m]
W_{wall}	Section modulus, combi-wall [m3/m]
A_{tube}	Sectional area, tube [m2]
θ_M	Factor to account for model uncertainty for bending moments [-]
θ_N	Factor to account for model uncertainty for axial forces [-]

Post et al., 2021 added a corrosion variable to this equation which simply reduces the thickness of the combi-wall on the waterside of the quay wall. In this research, only Case 1 uses corrosion to compare the results of Post et al., 2021 as a starting point. For the sake of simplification, corrosion is not included in Case 2.

2.3.2 Geotechnical failure

Geotechnical failure is a more challenging failure mode because there are multiple ways in which the soil can fail. Therefore, it is hard to define a criterion. Wolters, 2012 discusses four ways to determine geotechnical failure:

1. **MSF 1.0:** Using ϕ -C reduction Msf: $Z = \Sigma Msf - 1.0$
2. **MSF 2.0:** Multiplying all ϕ -C by 2.0 and using ϕ -C reduction to get result without getting failure in the plastic calculations: $Z = \Sigma Msf - 2.0$. Wolters, 2012 did not test this during his research.
3. **Mobilized shear resistance:** Deriving a LSF based on the mobilized shear resistance $\tau_{mob} = \frac{\tau_{max}}{\tau_{yield}}$
4. **ValidCalc** Defining a LSF using the PLAXIS definition of a soil body collapse during calculations (Numerical software returns Z=1.0 or Z=-1.0)

Both option 1 and 2 are reasonable approaches to use. Option 3 is considered to be too complicated because it is hard to determine in which node or cluster (in the numerical model) the LSF should be evaluated. Geotechnical failure can occur in many different ways, which makes this prediction difficult. Option 4 has a discontinuous z-function which makes it unsuitable for methods such as FORM. It is particularly interesting for line search algorithms like the one used in Directional Sampling, because this method requires too many calculations. Discontinuity also gives difficulties in the determination of influence coefficients (Wolters, 2012). Schweckendiek, 2006 used ϕ-c-reduction on a rather simple example and proved it to be a very convenient method for LSF: $Z = \Sigma Msf - 1.0$. As FORM shows convergence problems due to an apparently unstable limit state function, more advanced methods must be used. This is confirmed by some deterministic studies which conclude that the ΣMsf that is obtained by ϕ-c-reduction leads to slightly unstable results. Roubos, 2019 also uses option 1 from Wolters, 2012 and adds a factor to account for model uncertainty.

When using $Z = \Sigma Msf - 1.0$ and if the structure does not fail, the function gives information about the stability of the structure: $\Sigma Msf > 1.0$, which gives: $Z > 0$ (in case of no failure). A higher Z means a more stable structure. However, the failure domain receives no information about the magnitude of failure because $Z = 0$ when failure occurs. To provide more information in the failure domain, Post et al., 2021 added PLAXIS output parameter $\Sigma Mstage$. This parameter assigns a value to the instability of the soil: $0 < \Sigma Mstage \leq 1.0$, which gives: $-1.0 < Z \leq 0$. This results in the limit state function for geotechnical failure shown in Equation 2.2. The combination of $\Sigma Mstage$ and ΣMsf provides information for the failure- and non-failure domain which has the potential to lead to a faster and more efficient reliability analysis.

$$if\ \Sigma Mstage < 1.0: \qquad Z_{GEO} = \theta_{soil}\Sigma Mstage - 1.0 \tag{2.2a}$$

$$else: \qquad Z_{GEO} = \theta_{soil}\Sigma Msf - 1.0 \tag{2.2b}$$

Where

$\Sigma Mstage$	Global instability ratio related to load increase [-]
ΣMsf	Global stability ratio related to ϕ -c reduction [-]
θ_{soil}	Factor to account for model uncertainty for $\Sigma Mstage$ and ΣMsf [-]

2.4 Summary of starting points for this thesis

The case studies of this thesis asses a sheet pile wall and a sheet pile wall with relieving platforms. The case studies focus on limit state wall failure by yielding (Z_{yield}) and global geotechnical failure (Z_{GEO}). Z_{yield} is a well-known limit state and can usually be assessed by FORM. This makes it a good function to start with as discussed in section 2.2. Z_{GEO} is a challenging limit state where FORM exhibits convergence problems and methods like MCS are too time consuming. Under such circumstances, the metamodelling-approach can show its worth.

Wolters, 2012 assesses two case studies, a anchored sheet-pile and a quay wall with relieving platform. These case studies have a comparable soil profile compared to the case studies of this thesis. The stiffness modulus, the internal friction angle and the soil weight are dominant for both sand layers for wall failure by yielding and only for the lower sand layer for geotechnical failure. Roubos, 2019 found that time-independent variables, such as material properties of soil, steel and grout, as well as model uncertainty, significantly influence the reliability of a quay wall.

Chapter 3

Reliability analysis with metamodelling

This chapter aims to illustrate the methods and software used for this research.

3.1 General Concepts of Reliability Analysis

Reliability Analysis

The goal of reliability analysis of (geotechnical) structures is to determine the probability of failure and which factors influence this.

Failure (undesired event)

Whether failure of a structure occurs is evaluated by a Limit State Function (LSF), which is also called the performance function $g(\mathbf{X})$:

$$Z = g(\mathbf{X}) \tag{3.1}$$

Where negative values of Z represent the failure domain:

$$F = \{Z < 0\} = \{g(\mathbf{X}) < 0\} \tag{3.2}$$

Where $\mathbf{X}$ is the vector of random variables

Probability of Failure

The probability of failure P_f is the chance of failure according to the LSF, and is determined by solving the integral:

$$P_f = P(F) = P(Z < 0) = P(g(\mathbf{X}) < 0) = \int_{g(\mathbf{X})<0} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \tag{3.3}$$

where $f_{\mathbf{X}}(\mathbf{x})$ is the probability density function (PDF) of $\mathbf{X}$. In probabilistic research, the reliability is often expressed as the reliability index also referred to as β:

$$\beta = -\Phi^{-1}(P_f) \tag{3.4}$$

For the design of structures, limit state functions have such a high level of complexity that it is always required to use probabilistic methods to solve Equation 3.3. The conventional probabilistic methods used/named in this research are FORM (First Order Reliability Method), IS (Importance Sampling), DS (Directional Sampling) and MCS (Monte Carlo Simulation). These methods are assumed to be generally known and will not be further explained in this thesis.

3.2 ERRAGA

3.2.1 Principle of ERRAGA

This section is mainly based on Van den Eijnden et al., 2021

ERRAGA (Efficient and Robust Reliability Analysis for Geotechnical Applications) is a metamodelling method which can be used for reliability analysis. The need for more realistic and accurate reliability analysis makes the calculation more complex. Complexities such as non-linearity, incomplete model response or noisy model response may occur, which means that only strong probabilistic methods like Monte Carlo Simulation (MCS) are suitable (the complexities are illustrated in Figure 4.6). However, the disadvantage of a reliability analysis with MCS, in combination with numerical performance functions, is that this process is highly time consuming. Therefore, a metamodel is used. The goal of this metamodel is to make a representation of the performance function and minimise the amount of numerical calculations. In order to generate training data that contributes to providing a solution, an active training scheme is used. Active learning is used based on potential reduction of uncertainty of the metamodel. Numerical calculations of the performance function will be performed where the largest uncertainty in the metamodel is. Kriging (a method also known as Gaussian process, GP) is capable of quantifying these uncertainties of the metamodel, which makes it very suitable for these types of problems. Kriging originates from the mining business, where it was used to estimate the distribution of gold grades in an area based on the data of a limited amount of boreholes (Krige, 1951). This principle is illustrated in Figure 3.1, where an estimation is made of the whole area based on a limited amount of data points. The values of the unknown area (the white area) is fitted by the value of the known data points.

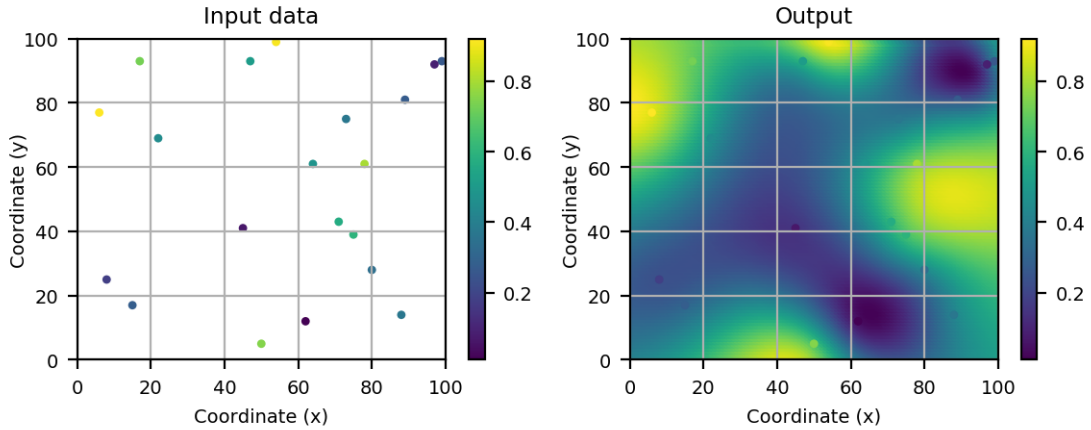


Figure 3.1: Illustration of Kriging, whereby the left side of the illustration shows the input data and the right side shows the estimation of the whole area with the input data slightly visible. The colour bar displays the gold grades when used in the mining industry and would it be used for reliability analysis, it would represent the Z-value.

Echard et al., 2011 linked GP metamodelling to MCS integration for reliability analysis. This is known as the "Active learning reliability method combining Kriging and Monte Carlo Simulation" (AK-MCS). This has formed the foundation for more active learning schemes for reliability analysis. This has resulted in multiple AK-MCS alternatives. ERRAGA is an enhanced implementation of AK-MCS and is aimed at dealing with complexities such as non-linearity, incomplete model responses and noisy response data from the numerical model, combined with very small target probabilities of failure ($P_f < 10^{-6}$), whereas conventional probabilistic methods are not able to deal with these issues. This makes ERRAGA stand out compared to other methods.

3.2.2 Description of methods

As ERRAGA uses multiple existing methods such as the Gaussian process and Importance Sampling, ERRAGA is rather complex. This section discusses the relevant methods and how they are used by ERRAGA.

Gaussian process regression for metamodelling

The response of a computational model $\mathcal{F}(\boldsymbol{x})$ in combination with a LSF like shown in Equation 3.1 can be described by a metamodel $\hat{g}(\boldsymbol{x})$:

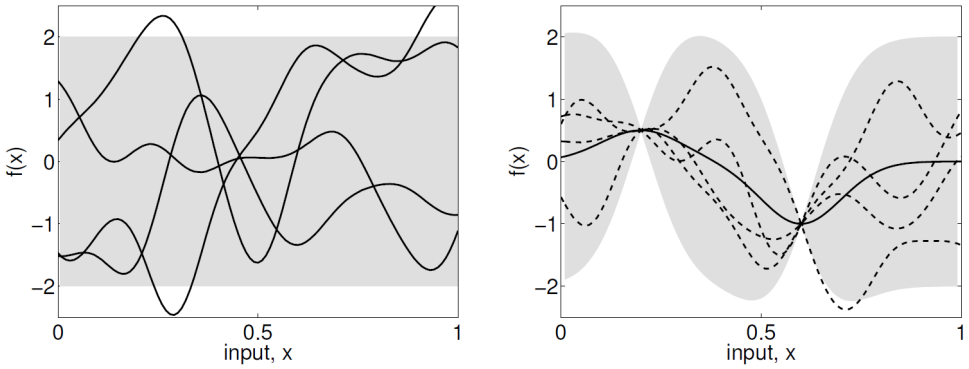
$$\hat{g}(\boldsymbol{x}) \approx g(\boldsymbol{x}) \tag{3.5}$$

Variables $\boldsymbol{x}$ are realizations of stochastic variables $\boldsymbol{X}$ with a joint probability distribution. The metamodel is defined as a GP, which is given by:

$$\hat{g}(\boldsymbol{x}) = m(\mathbf{x}) + \mathbf{K}^{1/2}\boldsymbol{\xi} \tag{3.6}$$

In this equation, the first part $m(\boldsymbol{x})$ is the mean function. In the second part of the equation, $\xi \sim N(\mathbf{0}, \mathbf{I})$ and the covariance function or kernel $\mathbf{K} = k(\mathbf{x}, \mathbf{x} \mid \boldsymbol{\theta})$ describes the covariance of the variables in the field. For convenience, $\boldsymbol{x}$ is transformed into $\boldsymbol{u}$, so the GP is formulated in standard normal space with uncorrelated variables $\boldsymbol{u} \in \boldsymbol{U}$. If there is no information available, the metamodel will tend to the mean. ERRAGA uses simple Kriging, such as $m(\boldsymbol{u}) = 0$, which causes the metamodel to regress to $g = 0$ in regions with no training data. There are many different types of kernels, which can deal with different forms of data. Kernels contain internal parameters ($\boldsymbol{\theta}$), known as hyperparameters, which define the shape of the kernel (e.g.: the correlation length l). As part of the training process, the hyperparameters are optimised to maximise the likelihood of the training data.

Figure 3.2 from Rasmussen and Williams, 2006 shows a simple 1-D regression problem. Four random samples from the prior distribution are shown in Figure 3.2 (a). With the addition of two data points the posterior is made, shown in Figure 3.2 (b). Only functions that pass through these two points are considered. The confidence bounds (shaded region) are zero at these points and grow larger as the distance to the data points increases. This means there is a greater uncertainty on locations further from the known data points. Note that the mean (solid line) tends to go to 0 if no training data is available (left and right part of the graph).



(a), prior (b), posterior

Figure 3.2: Panel (a) shows four samples drawn from the prior distribution. Panel (b) shows the situation after two data points have been observed. The mean prediction is shown as the solid line and four samples from the posterior are shown as dashed lines. In both plots the shaded region denotes twice the standard deviation at each input value x. (Rasmussen & Williams, 2006), Figure 1.1

Performance function $g(\boldsymbol{u})$ consists of known (training) data $\boldsymbol{g}_{\mathrm{t}}$ and unknown (predicted) data $\boldsymbol{g}_{\mathrm{p}}$. The training data $\boldsymbol{g}_{\mathrm{t}}$ is also referred to as Design of Experiment (DoE). The previous section has shown that the performance function $g(\boldsymbol{u})$ is formulated as a GP, such as Equation 3.6. The GP can be rewritten in a way which the unknown predicted data $\boldsymbol{g}_{\mathrm{p}}$ is solved. $\boldsymbol{g}_{\mathrm{p}}$ has two components, the best estimate $\hat{\boldsymbol{g}}$ and its variance $\boldsymbol{\sigma}^2_{\hat{g}}$.

Main algorithm

This section gives a general explanation of the algorithm of ERRAGA. The later sections give a more detailed description of some components which are interesting for this research. Figure 3.3 shows a flowchart of the algorithm, the following steps are taken in the process:

1. Initial (random) samples are taken, which form the initial DoE $\sqcap_{DoE}$.
2. Taking a training sample requires an evaluation of the computational model $\mathcal{F}(\sqcap_{DoE})$.
3. The sampling pool for MCS/IS is defined $\mathbf{u}_{\mathrm{MC}}$.
4. The learning decision is made which model is going to be trained.
5. The learning function of prediction- or classification model is used based on the learning decision, $\mathcal{L}_{\mathrm{p}}$ or $\mathcal{L}_{\mathrm{c}}$. The chosen training function will return a training point, $\sqcap_{learn}$.
6. A new training sample is taken $\mathcal{F}(\sqcap_{learn})$ and added to the set of training data.
7. The classification model $\mathcal{M}_{\mathrm{c}}$ is trained with the updated set of training data. The MCS/IS integration of the updated classification model gives the classification index $\hat{\boldsymbol{I}}_{\mathrm{c}}$.
8. The prediction model $\mathcal{M}_{\mathrm{p}}$ is trained with the updated set of training data. The MCS/IS integration of the updated prediction model gives the performance function prediction $\boldsymbol{g}_{\mathrm{p}}$.
9. The probability of failure $P_{f|0}$ and the probability of incompatibility P_1 are determined.
10. Both models are assessed on convergence based on the previous results. If both or one of the models is not converged, return to 4.
11. If both models are converged the MCS/IS integration is assessed on convergence. If the MCS/IS integration is not converged, a new learning cycle is started, return to 3.
12. If the MCS/IS integration is also converged it means there is overall convergence and the final results are returned.

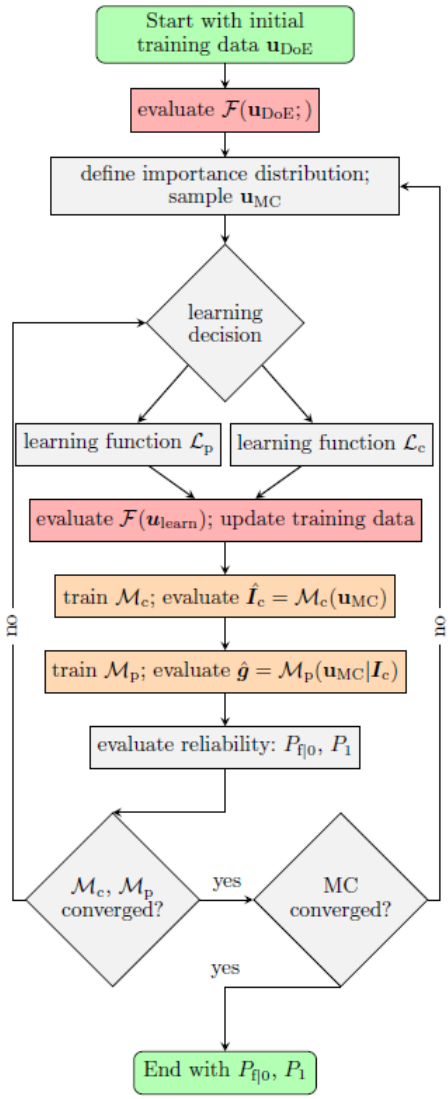


Figure 3.3: Flowchart of the Learning algorithm of ERRAGA. The red parts indicate the steps with high computational costs, the orange components have moderate computational costs. (Van den Eijnden et al., 2021)

GP metamodelling for reliability analysis

The Prediction metamodel ($\mathcal{M}_{\text{p}}$) returns a prediction $\hat{g}(\boldsymbol{u})$ which is used to determine the probability of failure P_{f} by MCS integration over a sample set $\mathbf{u}_{\text{MC}}$. Because $\hat{g}(\boldsymbol{u})$ is a prediction, there is an uncertainty in the determination of P_{f}. The metamodel is checked by three convergence criteria and one of these convergence criteria judges if the predicted metamodel is good enough. There are multiple ways to define a convergence criterion for the prediction metamodel. As default, the uncertainty of P_{f} is used as convergence criterion. An upper- and lower 95% confidence bound of P_{f} is determined using the variance (Schöbi et al., 2016):

$$\hat{P}_{\text{f}}^{\pm} = \mathbb{P}\left[\hat{g}(\boldsymbol{u}) \mp 1.96\sigma_{\hat{g}}(\boldsymbol{u}) \leq 0\right] \tag{3.7}$$

The relative uncertainty is determined using Equation 3.8. As default, the criterion converges if it is under a value of 0.05, because this is the default value for the Convergence Requirement

(ConvReq). This value is considered to be averagely strict according to Van den Eijnden et al., 2021. The convergence criterion is elaborated upon further in this chapter.

$$\epsilon_{P_f,M} = \frac{\hat{P}_f^+ - \hat{P}_f^-}{\hat{P}_f} < \text{ConvReq} = 0.05 \text{ (default value)} \tag{3.8}$$

The second convergence criterion checks if the MCS integration is converged, therefore it uses:

$$\text{CoV}_{\text{P}_\text{f}\text{MC}} = \frac{\sigma_{\hat{P}_f}}{\hat{P}_\text{f}} = \sqrt{\frac{1-\hat{P}_\text{f}}{N_\text{MC}\hat{P}_\text{f}}} < 5\% \tag{3.9}$$

There must be a sufficient amount of samples in the MCS pool, in order to reach this criterion. Reaching the threshold of 5% is therefore mainly a matter of using many samples (by default 100,000). If this criterion is not met, an update of the sample pool is needed,**'Model for Multiple Importance Sampling'** explains this in further detail.

The third convergence criterion checks if the metamodel dealing with incomplete data is converged. This so-called Classification Metamodel ($\mathcal{M}_\text{c}$) will only be used if incomplete training data is encountered. Section **'Incomplete (incompatible) training data'** explains more about this.

If the Prediction- and/or Classification metamodel are not converged, a new sample is taken and added to the DoE based on a learning function which aims to select the most informative next realization. This is repeated until both metamodels are converged. After this, the convergence of the MCS integration is checked. If the MCS integration is converged, all three criteria are met and the calculation is finished. If this is not the case, an update of the sample pool is needed. The convergence- and learning decisions are shown in Figure 3.3. This figure visualizes the process of ERRAGA in a flowchart.

Model for Multiple Importance Sampling

In case of very small probabilities (e.g. $P_f < 10^{-6}$), brute-force MCS may be unable to evaluate the probabilistic calculation because there are not enough or no samples at all taken in the failure domain. Importance sampling solves this problem by sampling from a distribution that overweights the important region (i.e. the failure domain) (Melchers, 1989). Figure 3.4 shows the different sample area of both methods in a typical two parameter domain. The combination of importance sampling (IS) and AK-MCS is known as AK-IS. The convergence criterion of is similar to Equation 3.9. If increased variance sampling over the MCS-sample-size (default: max 100,000 samples) is insufficient to obtain convergence of the MCS integration, the sampling pool is updated. This means a new sampling pool, located around the design points of the current predicted limit state and incompatible domain boundaries is picked based on the current failure- and incompatible domain. As a result, both metamodels have to converge again, after which the IS integration convergence is evaluated again (see Figure 3.3).

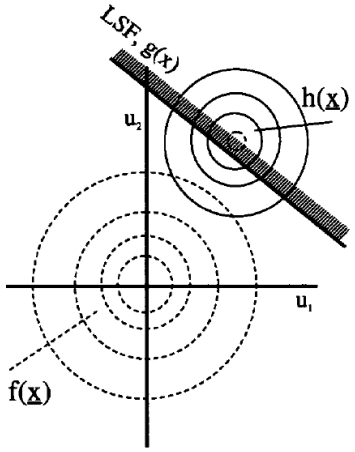


Figure 3.4: Two parameter sample area of Monte Carlo Simulation f(x) and Importance Sampling h(x) (Waarts, 2000)

Incomplete (incompatible) training data

Computational models do not always provide a response, which can lead to incomplete training data. Van den Eijnden et al., 2021 found in a case study of a sheetpile wall in a dyke, that incompatible realizations occur if the combination of input parameters leads to slope failure in the initialisation phase, prior to the installation or the evaluation of the sheetpile wall loading. The incompatible domains form a serious pitfall for the learning algorithm, as it is likely to get stuck if incompatible results occur. In order to deal with this problem, the GP predictive metamodel ($\mathcal{M}_\mathrm{p}$) is combined with a GP classification metamodel ($\mathcal{M}_\mathrm{c}$), which classifies the incompatible domain. A binary method is used for the classification, with ($I_c = 0$) for feasible and ($I_c = 1$) for incompatible realizations.

An important parameter to check the convergence of the classification model is the estimate of the probability of incompatibility $\hat{P}_1$:

$$\hat{P}_1 = \frac{1}{N_\mathrm{MC}} \sum_{i=1}^{N_\mathrm{MC}} \hat{I}_{\mathrm{c},i} w_{\mathrm{imp},i} \tag{3.10}$$

Where N_MC is the number of MCS or IS samples, $\hat{I}_{\mathrm{c},i}$ indicates the number of incompatible data and $w_{\mathrm{imp},i}$ is the weight factor for IS.

Two criteria are used for the convergence of the classification model. The first criteria is met if the relevant convergence of $\hat{P}_1$ is converged for four subsequent iterations. The second is based on the relevance of incompatible data with respect to the probability of failure $\hat{P}_f$:

$$\max_{i=1..4} \left(\frac{|\hat{P}_1^{(k_c=N_c-i} - \hat{P}_1|}{\hat{P}_1} \right) < \epsilon_{P_1} = \mathrm{ConvReq} = 0.05 \text{ (default value)} \quad \text{or} \quad \frac{\hat{P}_1}{\hat{P}_f} < 0.01 \tag{3.11}$$

Learning of the combined metamodel

The classification metamodel M_c and the prediction metamodel M_p are combined into a two-staged metamodel M. Both the metamodels have to be trained so active learning includes a learning function for the classification metamodel L_c and for the prediction metamodel L_p. This also means that a learning decision has to be made (see Figure 3.3). If one of the metamodels

is converged, the learning function of the not-converged metamodel is applied. If none of the metamodels are converged, the learning decision is made based on priority. More elaboration on the learning functions is provided in subsection 3.2.3.

With the inclusion of the incompatible domain, the entire domain can be divided into four sections (see Figure 3.5. Both the failure- and the compatibility indicator are binary which are defined as:

I_f failure indicator

- $I_f = 0$: No failure $g > 0$
- $I_f = 1$: Failure $g < 0$

I_c compatibility indicator

- $I_c = 0$: Feasible value
- $I_c = 1$: Incompatible value

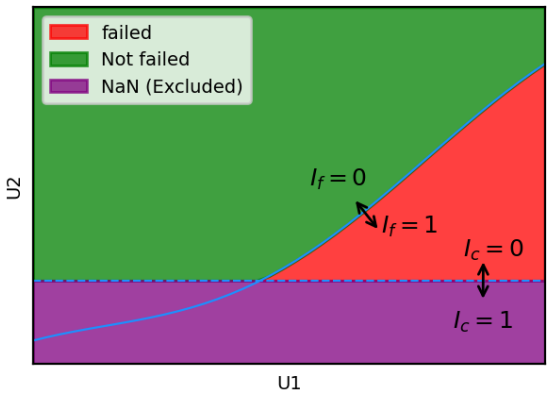


Figure 3.5: Indication of failure, non-failure and incompatible domain.

In accounting for the combination of incompatible- and failure domain, three estimates of probabilities are defined. These are used to asses convergence of the prediction- and classification model and when converged it will be the final result, whereby N_{MC} is the number of MCS/IS samples and w_{imp} is the weight applied when using IS:

- **Probability of incompatibility** $P_1 := P(I_c = 1)$: The probability of incompatible data (Note that this is the same as Equation 3.10)

$$P_1 = \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} (\hat{I}_{\text{c},i} = 1) \cdot w_{\text{imp},i} \tag{3.12}$$

- **Imputed probability of failure** $P_f^* := P(I_f = 1)$: Incompatible MCS/IS samples are predicted/determined by the classification model (all MCS/IS data points are used):

$$\hat{P}_f^* = \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} (\hat{I}_{\text{f},i} = 1) \cdot w_{\text{imp},i} \tag{3.13}$$

- **Updated probability of failure** $P_{f|0} := P(I_f = 1 | I_c = 0)$: Only the samples in the feasible domain are used:

$$\hat{P}_{f|0} = \frac{1}{1 - \hat{P}_1} \frac{1}{N_{\text{MC}}} \sum_{i=1}^{N_{\text{MC}}} (\hat{I}_{\text{f},i} = 1) \cdot (\hat{I}_{\text{c},i} = 0) \cdot w_{\text{imp},i} \tag{3.14}$$

The standard output of ERRAGA is the imputed probability of failure P_f^*. However, when encountering incompatible data the updated probability of failure $P_{f|0}$ should be used and is the default output in the final version of ERRAGA. Note that P_f^* and $P_{f|0}$ are equal if no incompatible data is encountered.

Noisy data metamodel

Computational models can show a certain degree of noise. This makes it extra complicated to fit a metamodel on the training data and causes overfitting when using a noise-free kernel, as shown by Van den Eijnden et al., 2021 in Figure 3.6a. A white noise component is added in Figure 3.6b (Rasmussen & Williams, 2006), the prediction of the metamodel is a better representation of the training data.

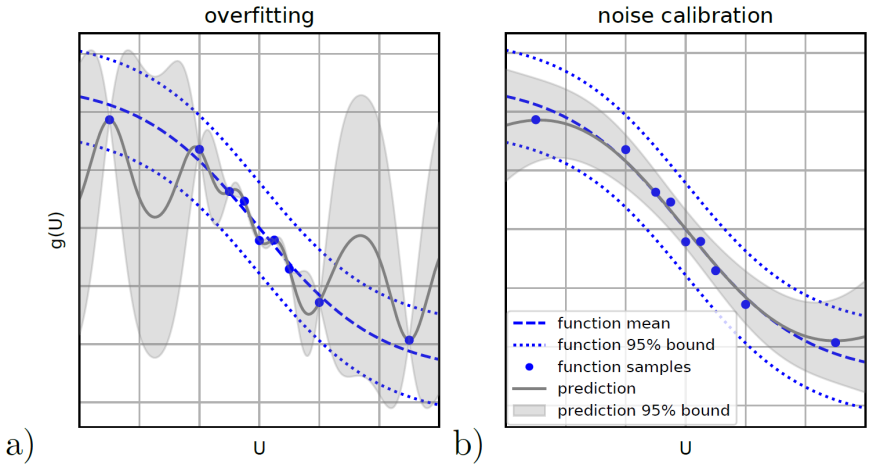


Figure 3.6: Van den Eijnden et al., 2021: Samples are taken from a smooth function after which noise is added. No noise term is added to the kernel in (a) and (b) has a white noise term which results in a smoother and more representative prediction with less uncertainty.

3.2.3 ERRAGA attributes

ERRAGA has multiple attributes and settings which can influence the results, convergence behaviour or the amount of training samples needed. The correct use of the attributes can be essential in the success of reliability assessments. Therefore, this section will give an insight in how different attributes work and how they influence the calculation. Note that not all attributes are mentioned in this section, but only the ones relevant for this thesis.

Initial realizations

Attribute: NinitED

The user can set the number of realizations (training samples) which are initially taken. These samples are randomly picked and form the initial DoE. The learning algorithm is not used during this process. The amount of training samples taken initially can influence the convergence, which will be further discussed in the section 'Learning Function'.

Minimum realizations

Attribute: NminED

A minimum can be set to the number of training samples which have to be taken before the calculation can stop. This minimum number of realizations include the initial realizations. If a problem is too 'simple' it is possible that it is converged based on the convergence criteria, but since there are not enough samples to produce a final answer, the calculation will fail.

Maximum realizations

Attribute: NmaxED

The maximum number of realizations determines how many training samples are taken during each learning cycle before the MCS/IS sampling set is updated. The user has to take into consideration that a lower amount of maximum realizations will force MCS/IS updating to occur earlier, which can lead to faster convergence. However, if the amount of maximum realizations is too low, this can prevent convergence because the model simply needs more training data.

Convergence Criterion Prediction metamodel

Attribute: ConvCrit

The convergence of the prediction metamodel M_p is determined by a stopping criterion. This criterion judges whether the predicted metamodel is accurate and reliable enough. This section discusses two stopping criteria which both base their criteria on the uncertainty of the prediction. Only MCS/IS samples in the feasible domain are used because the convergence of the prediction model is assessed.

PfStop
As default PfStop is used, this is a relative stopping criterion based on the probability of failure P_f, note that this criterion is already shown in the previous section:

$$\epsilon_{P_f,M} = \frac{\hat{P}_f^+ - \hat{P}_f^-}{\hat{P}_f} < \text{ConvReq} = 0.05 \text{ (default value)} \qquad \text{(3.8 revisited)}$$

The relative criterion is also available for the reliability index β. This criterion is similar to PfStop and therefore not further discussed.

BetaAbsStop
An option for an absolute criterion is added to ERRAGA:

$$\epsilon_{\beta,M} = \beta^0 - \beta^- < \text{ConvReq} = 0.05 \text{ (default value)} \qquad (3.15)$$

This criterion is based on the difference between the lower uncertainty bound of the reliability index value (β^-) and the best estimate of the reliability index value (β^0). Only the lowest possible reliability index is interesting when assessing the reliability of a structure. That is why the difference with the lower bound is sufficient enough to focus on.

For problems with a very low probability of failure - and thus a high reliability index - the absolute criterion will always converge first. Convergence of the relative criterion can be difficult for these kind of problems, as is shown in the following example:

For low possibilities of failure the convergence value of 0.05 can be too narrow. The upper- and lower bound and best estimate of the probability of failure are:

$P_f^+ = 8 \cdot 10^{-6}, \qquad P_f^0 = 7 \cdot 10^{-6}, \qquad P_f^- = 5 \cdot 10^{-6}$

Using equation 3.8: $\epsilon_{P_f,M} = 0.50$, which means there is no convergence, because the value is 10 times larger than the default limit value of 0.05. To give an idea of the difference between the upper and lower bound, the probability of failure over 50 years is calculated for both values. Hereby it is assumed that the given P_f-values are the annual probability of failure. To calculate the total probability of failure, a standard equation is used:

$$P_{f;total} = 1 - (1 - P_f)^n \qquad (3.16)$$

Where n is the lifetime of a structure
Upper bound: $P_{f;total}^+ = 4.0 \cdot 10^{-4}$
Lower bound: $P_{f;total}^- = 2.5 \cdot 10^{-4}$

When using the absolute convergence criterion (equation 3.15), the model is converged:
$P_f^0 = 7 \cdot 10^{-6} \quad \rightarrow \quad \beta^0 = 4.34$
$P_f^+ = 8 \cdot 10^{-6} \quad \rightarrow \quad \beta^- = 4.31$

$\beta^0 - \beta^- = 0.03 < 0.05$

This example shows that while the relative criterion PfStop is still far from convergence, the absolute criterion BetaAbsStop is already converged. The question is which criteria should be used to get the most efficient results. The relative criterion can be considered to be too narrow and therefore needs many more training samples compared to the absolute criterion. The precision of the absolute criterion is considered to be sufficient when using a Convergence Requirement of 0.05, figuring that the numerical models and limit state functions are also not perfectly accurate.

Convergence Requirement

Attribute: ConvReq

The Convergence Requirement (ConvReq) is a convergence limit for the prediction model and for one convergence criterion of the classification model:

- Prediction model, see Equation 3.8 and Equation 3.15
- Classification model, see Equation 3.11

As default, a value of 0.05 is used for ConvReq.

Learning Function

Attribute: LearnFnc

A learning function $L(\mathbf{u}_{\mathrm{MC}})$ is an important feature of the metamodel. If merely training data is produced which contributes to generate a solution, it reduces the amount of time consuming learning samples needed for convergence. There are multiple learning functions available for the prediction metamodel. U-learn is the standard learning function for the classification metamodel L_c. The aim of all learning functions is to select the most informative next realization ($\boldsymbol{u}_{\mathrm{learn}}$). It depends on the problem what the most informative realization is and therefore, ERRAGA contains eight different learning functions. Two of these functions are discussed in this section.

U-learn
The default learning function for the prediction metamodel L_p is U-learn. This function selects the next realization at the location with the highest probability of misclassification (i.o.w.: find the largest uncertainty with respect to the absolute predicted value (i.e. $|\hat{g}(\boldsymbol{u})|$)). U-learn function according to Echard et al., 2011:

$$\boldsymbol{u}_{\mathrm{learn}} = L_{\mathrm{p}}(\mathbf{u}_{\mathrm{MC}}) = \underset{\boldsymbol{u} \in \mathbf{u}_{\mathrm{MO}}}{\arg\min}(U(\boldsymbol{u})), \qquad U(\boldsymbol{u}) = \frac{|\hat{g}(\boldsymbol{u})|}{\sigma_{\hat{g}}(\boldsymbol{u})} \tag{3.17}$$

The probability of misclassification is reduced to zero at the location of the training sample (see Figure 3.6.a). However, this is only the case when the training data is noise-free and the training data has equal sample weights. When encountering noisy data or when using IS, it is advised to use UNIS-learn, which is discussed below.

UNIS-learn
When dealing with noise, there is a noise term added to the kernel and therefore the probability of misclassification of the training sample will not reduce to zero (see Figure 3.6.b). If IS is used, Equation 3.17 does no longer hold for selecting the sample with the highest probability of misclassification. The unequal sample weights should be included in the learning function, otherwise samples can be taken in regions far from the design point and thus do not contribute

much to the improvement of the prediction model. To deal with these effects, a learning function is developed which uses the concept of maximising utility. The utility function looks at the difference in probability of misclassification before and after taking a learning sample $\boldsymbol{u}_i$:

$$U_{\mathrm{NIS}}\left(\boldsymbol{u}_i\right) = \left(\Phi\left(-\frac{|\hat{g}\left(\boldsymbol{u}_i\right)|}{\sigma_{\hat{g}}}\right) - \Phi\left(-\frac{|\hat{g}\left(\boldsymbol{u}_i\right)|}{\sigma_{\hat{g}+1}}\right)\right) w_{\mathrm{imp},i} \tag{3.18}$$

The left hand probability is the current probability of misclassification based on the current prediction variance. The right hand probability is the probability of misclassification based on the prediction variance after taking a learning sample $\boldsymbol{u}_i$.

$$\boldsymbol{u}_{\mathrm{learn},P} = L_{\mathrm{P}}\left(\mathbf{u}_{\mathrm{MC}}\right) = \underset{\boldsymbol{u}\in\mathbf{u}_{\mathrm{MC}}}{\arg\max}\left(U_{\mathrm{NIS}}(\boldsymbol{u})\right) \tag{3.19}$$

To sum up, this function accounts for noise and unequal sample weight. The noise is accounted for by looking at the difference in misclassification, if noise is negligible, the UNIS-learn function would be equal to the U-learn function with an importance weight reduction. UNIS-learn is suited when encountering noisy data or when using IS. However, Kentrop, 2021 found that a metamodel with a limited DoE size can affect the convergence behaviour but a DoE of 50 samples always resulted in convergence for the case studies he considered.

Classification model

Attribute: Classifier

A classification model is used to classify the incompatible domain as is explained in subsection 3.2.2. In the incompatible domain, there is no data response from the computational model and no unnecessary samples are taken from this domain when it is classified. ERRAGA has access to two types of classification models:

1. Gaussian process classification (GPC), which uses the logistic regression function
2. Support vector machine (SVM), which is used for classification

For the calculations performed for this thesis, there is no preference between both classification models.

Noise Term

Attribute: Noise

Noisy data metamodelling is explained in subsection 3.2.2. Adding a white noise term to the kernel prevents overfitting when noisy training data is received. This is implemented in ERRAGA by using a Gaussian process which is called 'WhiteKernel'. Therefore, it is advised to always turn the noise term on when performing reliabiliy analysis with computational models like FEM.

Noise Bounds

Attribute: NoiseBounds

The noise term represents the variance of white noise σ_{wn}^2 which is determined by the WhiteKernel. ERRAGA's WhiteKernel component tries to fit the optimal level of noise which is added to the kernel (Rasmussen & Williams, 2006):

$$k_{\mathrm{wn}}\left(\boldsymbol{u}, \boldsymbol{u}'\right) = \sigma_{\mathrm{wn}}^2 \delta_{\boldsymbol{u}\boldsymbol{u}'} + k\left(\boldsymbol{u}, \boldsymbol{u}' \mid \sigma^2, \boldsymbol{\theta}\right) \tag{3.20}$$

With noise added to the kernel, the prediction of the metamodel does not have to fit precisely through the obtained data points (see Figure 3.6).

The user can influence the amount of noise by adjusting the upper- and lower value of the noise term, the so-called noise bounds. The noise term will lie between these bounds and this feature comes into play when the user wants ERRAGA to add more or less data to the kernel. It is possible that ERRAGA finds an optimal noise term which is too low, which can cause overfitting and will slow down the convergence drastically, resulting in an increase in the amount of required training samples. Kentrop, 2021 investigated this behaviour on a case study with 14 variables and found that a noise term of $\sigma_{wn} = 0$ causes overfitting and that this can be prevented by adding a small amount of noise. On most of the 14-variable case studies, it was not necessary to set a lower bound, although Kentrop, 2021 recommends to always check the limit state function for logical behaviour and overfitting. If overfitting is suspected it is advised to increase the lower bound of white noise which may give better and faster results. However, it is important to keep in mind that the case study used by Kentrop, 2021 is a specific case study and the required amount of white noise may differ for other cases figuring the varying magnitude of noise produced by different computational methods. Van den Eijnden et al., 2021 found that the computation model showed a noise response up to approximately 1% of total variation in the system response. It is advised to keep this as a rule of thumb, because if this percentage gets higher, miscalibration (everything is seen as noise) can occur.

Noise Variance Reduction Ratio (NVRR)

Attribute: NoiseVarianceReductionRatio

The observed white noise can also be used to artificially decrease the influence of the white noise on the convergence criteria, by (partly) subtracting the noise from the uncertainty of the prediction metamodel. This means that a part of the uncertainty from the prediction model is ignored and this uncertainty is used to determine the upper and lower bounds of the stopping criterion (see Equation 3.8 and Equation 3.15). This will result in faster convergence, especially if NVRR=1.0 and a high degree of noise occurs with respect to the uncertainty. If there is no noise in the model response the uncertainty of the prediction model will stay unchanged. The 'new' uncertainty $\sigma_{optimist}$, which is used for the stopping criterion for the prediction model, is given by:

$$\sigma_{optimist} = \sqrt{\sigma^2 - NVRR \cdot \sigma_{wn}^2} \tag{3.21}$$

The NVRR can range from 0 to 1, the reduction of uncertainty is shown by:

$NVRR = 0.0$	No noise accepted/ignored
$NVRR = 0.75$	Noise standard deviation reduced by 50%
$NVRR = 1.00$	All noise accepted/ignored

It is doubtful if this approach is acceptable because the upper- and lower bounds of the convergence criterion of the prediction metamodel are not completely representative any more. Furthermore, the use of NVRR is not implemented in the learning functions which are available.
Under-fitted models, which are smooth models, with a high white-noise component will benefit from the use of NVRR. Where the over-fitted model will benefit from the noise term which is added to the kernel.

Size of MCS/IS pool

Attribute: Nlearn

The size of MCS/IS pool for sampling domain integration can be adjusted by the user, which is set to 100000 samples by default. The MCS/IS pool size has to be large enough to determine an accurate probability of failure Pf. A smaller Pf will result in less MCS/IS samples in the predicted failure domain, which can give a bad estimate of Pf if this amount is too low. However, this

is solved by ERRAGA by performing an IS refinement. Another option is to use an increased variance of the initial sampling pool (see 'Beta Prior'), which will also increase the amount of samples in the failure domain and correct it by a weight factor. If the classification metamodel has difficulties with convergence an increase in the MCS/IS pool might solve this problem (see Equation 3.11).

Internal Hyperparameter optimisations

Attribute: OptIter

The kernel is the covariance function used by ERRAGA, by default this is Matérn 5/2. As explained in the previous section, the hyperparameters are the parameters of the covariance function and define the shape of the kernel. Each time when new training data is added to the DoE, the hyperparameters are updated. The optimal hyperparameters are determined based on a likelihood function. However, the likelihood function can have multiple local optima which are used to determine the hyperparameters instead of a global optimum. To increase the chance of using the global optimum, the determination of the optimal likelihood is repeated multiple times with random starting points. As default this amount is set to 10, but it can be changed in the attribute OptIter. A too small OptIter can result in convergence criterion which show an erratic behaviour. This means that ERRAGA is probably switching between different prediction models. An increase in OptIter may result in smoother and more stable behaviour.

Beta Prior

Attribute: beta

If Beta Prior is not zero, ERRAGA starts directly with IS with increased variance instead of standard-normal MCS. By giving an estimate of the final beta, a factor U_{amp}, is determined which increases the standard deviation of the training samples:

$$U_{amp} = \sqrt{1 + \frac{BetaPrior^2}{n_{dimensions}}} \tag{3.22}$$

All the standard normal variables then become: $N(\mu,\, \sigma \cdot U_{amp})$. In this way there is a more optimal initial variance for the IS integration. This means that also the initial DoE has a higher variance (the random samples are taken over a wider area). This increases the chance that ERRAGA finds failure samples during the first learning cycle, something which can occur when dealing with a low probability of failure. On the other hand BetaPrior should not be set too high because this could lead to extreme importance sampling (IS is taken too far into the failure domain). This includes high importance factors, which will lead to inaccurate estimations, even though many failing realisations are predicted.

3.3 Software

3.3.1 Probabilistic Toolkit (PTK)

The Probabilistic Toolkit is a software program developed by Deltares. It is able to analyse the effects of uncertainty to any model. For this thesis, it is used to perform reliability analysis on the case studies. The PTK is the central component of the reliability analysis where the variables and settings are inserted and where the results are displayed (see Figure 3.7). The PTK has multiple reliability methods at its disposal (e.g. FORM, MCS) and the metamodelling method ERRAGA is added to its inventory. In the PTK, the user defines the variables, the limit state function (LSF) and settings for the reliability method. The finite element method PLAXIS (PLX) performs the geotechnical calculations. The communication between PTK and PLX takes place through the use of Python files.

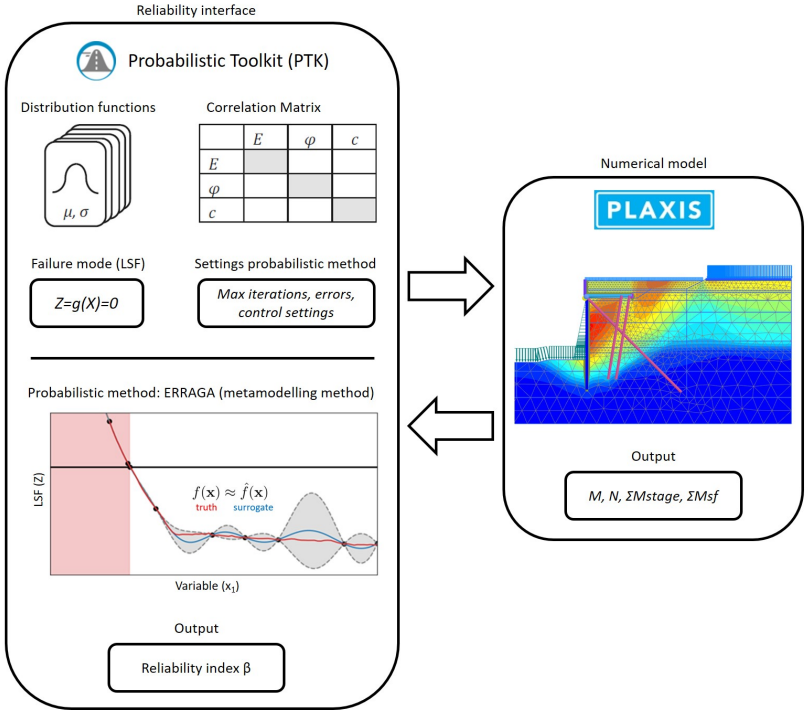


Figure 3.7: Connection between the probabilistic toolkit, metamodelling method and finite element model, inspired by Roubos, 2019 Figure 3.2

3.3.2 PLAXIS (PLX)

PLAXIS is software for geotechnical analysis and is used to model the quay wall designs of the case studies. PLAXIS uses a Finite Element Method (FEM), which is necessary to model the complex soil-structure interaction in the case studies.

The output parameters of the model are used to calculate the LSF's, for this research this includes i) max deformation (quay wall), ii) max bending moment (quay wall), iii) max normal force (quay wall), and iv) stability factors of soil. It is known that the output of PLAXIS can be noisy, especially stability factors of soil ($\Sigma Mstage$ and ΣMsf) seem to entail noise. There are two different sources of noise in the numerical model:

- Identical input gives slightly different output. This seems to occur when running on different computers or using a different amount of cores. There also seems to be some randomness in the Hardening Soil model.

- A very small difference in input gives unexpected and relatively large differences in output. This occurs at the threshold of failure where $\Sigma Mstage$ and ΣMsf show undulating behaviour. Figure 3.8 shows an example of ΣMsf close to failure. The final value is at the orange circle. Looking at the course of the graph, it seems rather arbitrary at what point the calculation stops. In combination with a maximum amount of iterations, this may cause noise.

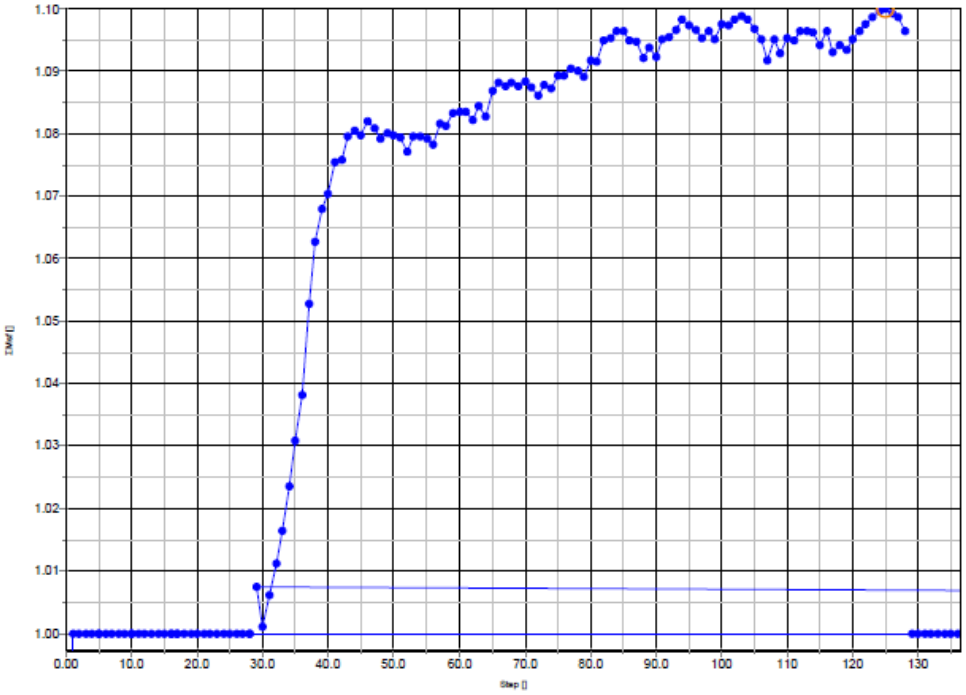


Figure 3.8: Example of ΣMsf (almost at failure). Where ΣMsf on the vertical axis and the number of step on the horizontal axis

There are ways to make the results of PLAXIS more stable, however this increases the computation time and the calculation can thus become infeasible and it is also not sure if it reduces the noise enough.

3.4 Conclusions

This chapter shows how the reliability assessment with a metamodelling approach is set up for this research. The metamodelling method ERRAGA and its attributes are explained in order to get an insight in the method and how do use it in order to perform a successful reliability assessment. Because of its complexity, understanding this method is crucial, the same goes for the attributes. Therefore, the different attributes are discussed separately and it is shown that the attributes can be used for different purposes, mainly in order to deal with noisy- and incompatible data.

Chapter 4

Case 1: Sleepbootkade

In this chapter, the first case study is discussed. It concerns the Sleepbootkade, which is located in the port of Rotterdam. This case study builds upon the work of the GRAPA (Geotechnical Reliability Analysis for Practical Applications) (Post et al., 2021). Up to subsection 4.4.1 shows the work which has been performed by Post et al., 2021. The aim of the GRAPA project is to perform full-probabilistic reliability analyses for Geotechnical structures, and in this project more particularly quay walls. Furthermore, the GRAPA project aims to make the metamodelling methods more accessible to engineers willing to practice this. The goal of this case study is to gain insight in the details on how ERRAGA performs a reliability analysis on a realistic case. In addition, the complexities encountered during the reliability analysis are analysed in this chapter.

4.1 Case description

The Sleepbootkade (in English: Thugboat-quay) is part of a small harbour (Sleepboothaven) located in the port of Rotterdam. Currently, the quay wall is used for the transport of cargo. Figure 4.1 shows a top view of the Sleepbootkade. The Port of Rotterdam concluded that the old quay wall reached the end of its serviceability life span. Therefore, engineering firm Witteveen+Bos was instructed to design a new quay wall structure. Figure 4.2 displays the new design, in which the old front wall in this cross section is visible. The front wall consists of a combiwall which is anchored with grout anchors. The existing front wall will be removed to a level in order to enable the anticipated anchorage to pass above.

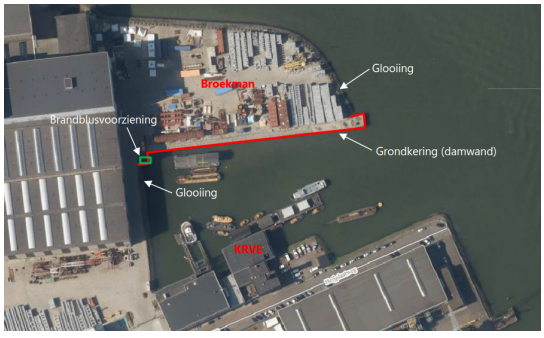


Figure 4.1: Location of the Sleepbootkade (red line) (Post et al., 2021)

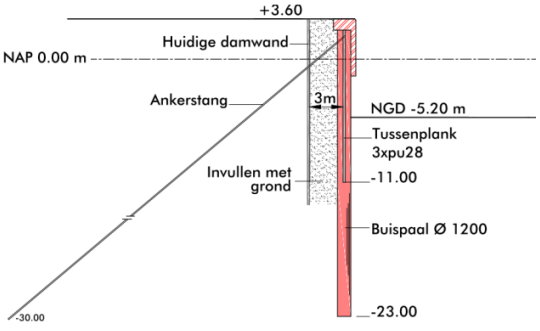


Figure 4.2: Cross section of the new Sleepbootkade (Post et al., 2021)

4.2 PLAXIS model

Witteveen+Bos provided a PLAXIS model used for the design of the quay wall. This model is simplified because it was considered to be too complex for the reliability assessment (Figure 4.3). The adjustments and simplifications were made on:

- Phasing
- Soil layering
- Geometry of the structure
- Numerical settings per phase

As it is very unlikely that global failure will occur, there are some adjustments made for LSF Geo because in the original situation, the quay wall has a relative deep sheet pile wall. To counter this, the front wall has been shortened from a pile tip at NAP -23 m to NAP -16.5 m. The anchor rod has been set from elastic to elastic-plastic properties so it may yield.

As this is the first test case with ERRAGA, it is convenient to keep the calculation time low, because a lot of test runs have to be performed. After simplifications of the Geometry and phasing, there are some relatively large deviations observed between the original and the simplified model. These differences are accepted for this case because the goal is to test reliability analysis with metamodelling, and therefore it is not important to obtain the exact correct answer.

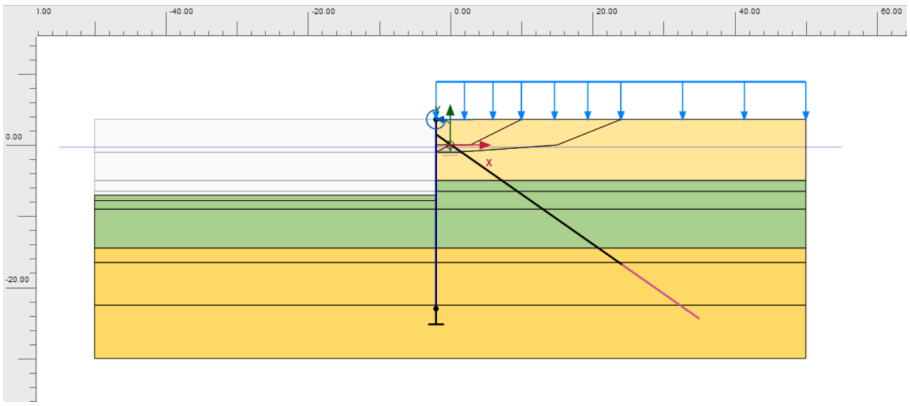


Figure 4.3: Simplified PLAXIS model case 1 (Post et al., 2021)

4.3 Statistical distributions

4.3.1 Stochastic variables

For this case study, 10 to 15 stochastic variables are selected. A real case, may have a similar amount of variables, but may also have up to 30-40 variables. This depends on how many variables the user considers to be influential and need to be taken into account. In this case study, 10 to 15 variables are taken to limit the computation time. Therefore, only the variables that are expected to be most dominant are selected. The values of the variables are taken from Post et al., 2021, who based them on experience. There are some known influential variables like the water- and bottom level which are not selected for simplification reasons. According to the findings of chapter 2, the stiffness modulus and the soil weight of the lower sand layer are also dominant variables for limit state Wall failure by bending and for Geotechnical failure. However, to limit the amount of soil variables, they are not selected for this case study. To resemble the case study of Post et al., 2021, the same variables with their distribution, mean and standard deviation are selected, as can be seen in Table 4.1.

Variable	**Unit**	**Distribution**	μ	σ	V
Unsaturated volumetric weight: Sand, loose $\gamma_{unsat;sandloose}$	[kN/m3]	Normal	17	0.85	0.05
Saturated volumetric weight: Sand, loose $\gamma_{sat;dandloose}$	[kN/m3]	Normal	19	0.95	0.05
Internal friction angle: Sand, loose $\phi_{sandloose}$'	[degrees]	Normal	35	3.5	0.1
Secant stiffness for: Sand, loose $E^{ref}_{50;sandloose}$	[kN/m2]	Lognormal	15000	3000	0.2
Internal friction angle: Clay ϕ_{clay}'	[degrees]	Normal	29	2.9	0.1
Secant stiffness: Clay $E^{ref}_{50;clay}$	[kN/m2]	Lognormal	2000	400	0.2
Cohesion: Clay c^{ref}_{clay}	[kN/m2]	Lognormal	5	1	0.2
Internal friction angle: Sand, medium $\phi_{sandmedium}$'	[degrees]	Normal	38	3.8	0.1
Annual surcharge $Q_{surface}$	[kN/m2]	Gumbel	26	5.2	0.2
Yield stress tube $f_{y;tube}$	[kN/m2]	Lognormal	337000	13500	0.04
Tube diameter D_{tube}	[m]	Normal	1.219	0.06095	0.05
Tube thickness t_{tube}	[m]	Uniform	0.016	0.0008	0.05
Corrosion combiwall $\Delta t_{water;tube}$	[m]	Normal	0.002	0.0005	0.25
Model uncertainty θ_M θ_N θ_{soil}	[-]	Lognormal	1	0.1	0.1

Table 4.1: Stochastic variables used for case 1

4.3.2 Correlations

The correlations between the variables are taken from Post et al., 2021 and shown in Table 4.2. There is only a correlation between variables of the same soil layers, the other stochastic variables are assumed uncorrelated, for simplification reasons these variables are not displayed. Note that the saturated and unsaturated volumetric weight of the loose sand layer are fully correlated, which means that these two variables function as one variable.

#	Variable	1	2	3	4	5	6	7
1	$\gamma_{unsat;sandloose}$	1	1	0.5	0.5	-	-	-
2	$\gamma_{sat;sandloose}$	1	1	0.5	0.5	-	-	-
3	$\phi_{sandloose}$'	0.5	0.5	1	0.25	-	-	-
4	$E^{ref}_{50;sandloose}$	0.5	0.5	0.25	1	-	-	-
5	ϕ_{clay}'	-	-	-	-	1	0.25	-0.65
6	$E^{ref}_{50;clay}$	-	-	-	-	0.25	1	0.12
7	c^{ref}_{clay}	-	-	-	-	-0.65	0.12	1

Table 4.2: Correlations case 1

4.4 Limit state wall failure by yielding

4.4.1 Results from previous analyses (GRAPA)

This section builds upon the work of (Post et al., 2021). The main results are shown below. Several reliability calculation are performed before and during this research. Those 'runs' are named after the project name 'GRAPA', followed by a version indicator, such as 'v2t' or 'v3t'. The number refers to either limit state wall failure by yielding (2) or Geotechnical failure (3).

- ERRAGA shows similar results compared to conventional methods (FORM and DS).
- ERRAGA uses the same order of calculations as FORM and is a few orders of magnitude faster than MC-like approaches such as DS.
- Activating the 'noise' option seems to give an improved performance and calculation time drops from 4 hours to 1.5 hours (with a slightly higher β-values).l

4.4.2 Results benchmark case

Verification of the results

This run is adopted from Post et al., 2021 and the goal is to obtain (practically) the same results as a benchmark case for this research. This verifies if the finite element model and software couplings are correct. Three runs are performed to check if there is any randomness in the results, as shown in section B.1. Note that the convergence criterion (prediction model) is not displayed correctly, this software bug that was fixed afterwards. A FORM run is performed, which gives an indication of the beta-value and the influence factors. Lastly, a final ERRAGA run (GRAPA v2y) is performed to obtain the correct ERRAGA prediction of the influence factor. On a side note, LSF failure by buckling is still included in this calculation. This is acceptable because the goal of this run is to verify settings.

ERRAGA settings:		
Initial realisations	10	default
Min realisations (N Min)	50	
Max realisations (N Max)	5000	
Convergence Criterion	PfStop	default
Convergence Requirement	0.05	default
Learning Function	Ulearn	default
Classification Model	None	default
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0.0	default
Size MCS pool	100000	default
Internal hyperparameters	10	default
Beta Prior	1.0	

Table 4.3: ERRAGA settings for GRAPA v2t

The verification runs are considered to be well within range of the FORM and Post et al., 2021. As such the results can be considered accurate. The maximum deviation is 0.04, which proves that the results are reliable. The observed scatter between the three verification runs is less then the order of 0.1 beta (max 0.07). For practical application of the method, these are considered to be small deviations. This proves that the results are reliable. This outcome gives enough confidence to work further with the software set-up and ERRAGA settings. As a side note, the convergence criterion of the FORM run is more precise than ERRAGA's, which explains the difference in

required realisations. As Post et al., 2021 states, FORM and ERRAGA are expected to need the same order of realisations for having the same reliability.

	Results			
Method	**Converged**	**N**	β	**Run id**
ERRAGA	YES	50	2.94	GRAPA v2t (Post et al., 2021)
ERRAGA	YES	50	2.98	GRAPA v2t run2
ERRAGA	YES	50	2.91	GRAPA v2t run3
ERRAGA	YES	50	2.92	GRAPA v2t run4
FORM	YES	272	2.94	GRAPA v2y FORM
ERRAGA	YES	51	2.91	GRAPA v2y

Table 4.4: The results of the reliability analysis for LSF Frontwall Case 2, no model uncertainty

Influence factors

Table 4.5 compares the influence factors between FORM and ERRAGA. There are some large differences in the prediction of the influence factors, for the higher influential parameters there is a difference up to 12%. However, there are four parameters which are clearly the most influential, which is confirmed by both methods. As a conclusion, the results of both FORM and ERRAGA are not identical, but the values can be considered close enough to get an idea of the major influential parameters. According to the comparable case of Wolters, 2012 in chapter 2, the stiffness modulus, the internal friction angle and the soil weight of the upper- and lower sand layer should be the dominant parameters. However, these parameters are part of the least dominant parameters for this case. The Geology is mainly the same, small Geological differences will not have such an impact on the influence of the parameters. It is hard to say what causes the unexpected influence factors, a reason could be the simplification of the PLAXIS model. Another remarkable fact is the large influence of model uncertainty factor of the bending moment, this could indicate that there is a very small normal force in the wall and failure is dominated by the bending moment.

Influence Factors:	**FORM**	**ERRAGA**
$\gamma_{unsat;sandloose}$	1	3
$\phi_{sandloose}$	1	1
$E^{ref}_{50;sandloose}$	0	3
ϕ_{clay}	13	15
$E^{ref}_{50;clay}$	3	0
c^{ref}_{clay}	0	0
$\phi_{sandmedium}$	0	4
$Q_{surface}$	27	18
$f_{y;tube}$	5	4
D_{tube}	11	23
t_{tube}	3	0
$\Delta t_{water;tube}$	0	6
θ_M	36	22
θ_N	0	0

Table 4.5: Influence factors (in %), comparison between FORM and ERRAGA

4.5 Limit state Geotechnical

4.5.1 Results from previous analyses (GRAPA)

The LSF Geo runs of Post et al., 2021 were not able to converge, even though the same ERRAGA settings of LSF Front wall were used. Multiple reliability assessments have been performed on the LSF Geo using ERRAGA. The final result is shown in Figure 4.4, which is the starting point of this section.

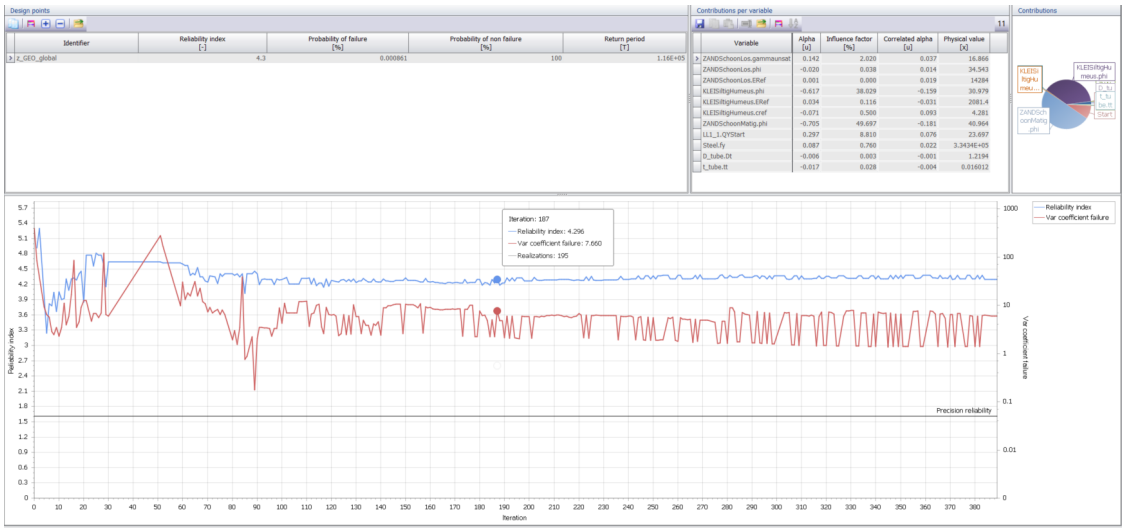


Figure 4.4: Run GRAPA v3n, reliability assessment of LSF Geo including a numerical model with 11 stochastic variables. Visualizing the development of beta (blue line) and the convergence criterion of the predicion model (red line) (Post et al., 2021)

The main results found from the simulations performed by Post et al., 2021 are listed below:

- It seems as if a representative beta was found, but the convergence seems questionable in Figure 4.4. The convergence value is rather far from the convergence criteria. The erratic is shown because ERRAGA is switching between different prediction models (i.e. switching between a different set of hyperparameters) because each realisation different optima are found, as can be seen under Internal Hyperparameter optimisations in section 3.2. The β is checked by an alternative approach and multiple beta-values are determined by using LSF $Z = dx_{max} - dx_{occured}$ and increase the allowable deformation (dx_{max}) for each run (see Figure 4.5). The result of Figure 4.4 and the horizontal asymptote of Figure 4.5 are in the same order of magnitude.

- LSF Geo uses soil (in)stability factors $\Sigma Mstage$ and ΣMsf from PLAXIS where LSF Front wall uses the bending moment and normal force from PLAXIS. It is doubtful whether combining both $\Sigma Mstage$ and ΣMsf in one LSF is able to create a smooth LSF. Limit state Geo shows a way more difficult convergence behaviour compared to limit state front wall (Section: 4.4), it is possible that this is caused by the noise in $\Sigma Mstage$ and ΣMsf.

- The usage of the 'noise' term (see, section 3.2) reduces the required model iterations (in one case by a half).

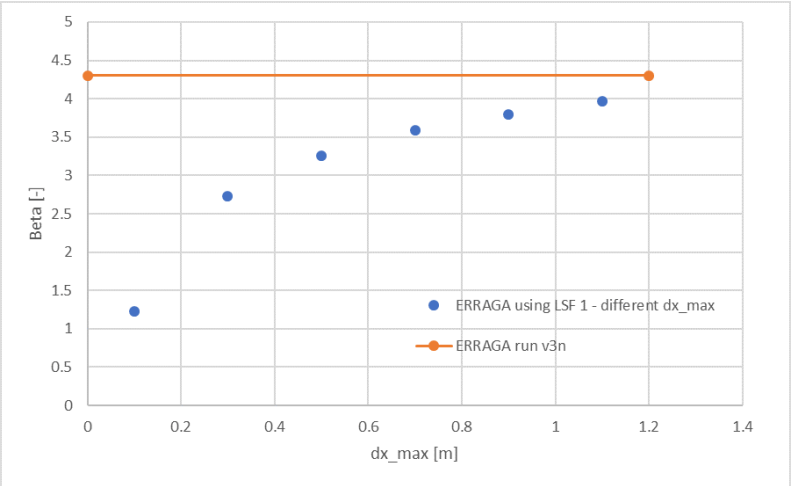


Figure 4.5: dx_{max} vs. β to verify the β found for LSF Geo (Post et al., 2021)

4.5.2 Two-variables model

The main issue of the previous section (subsection 4.5.1) is the fact that the reliability analysis is not converging, at least not within a reasonable amount of iterations. This is most likely caused by complexities such as incompatibility, non-linearity, noise and multiple variables. ERRAGA offers multiple settings which can be adjusted to deal with this complexities, as is visible in section 3.2. To determine what complexities are playing a role in the model, the amount of variables is decreased from 11 to 2. If the two-variables model also encounters convergence problems, it is analysed to see what is causing these issues and solutions can be tested.

This section analyses the reduced two-variable version of the LSF Geotechnical model. Having a model with two-variables gives the opportunity to plot the samples and metamodel (2D). In this way, the behaviour of the LSF can be visually analysed. Appendix C contains detailed information about all the runs performed with this model. In this section, the process and results are summarised.

Numerical two-variables model

The two stochastic variables are the internal friction angle of one of the sand layers and the annual expected surcharge load on surface level. The results of the run are shown in section C.1. It is very clear that the limit state is non-linear and there is a lot of incompatible data. There is no visual proof of any noise, however the data is not strong enough to exclude the presence of noise. It is reasonable to assume that noise is not visible with the available information.

Metamodel

Visualizations of the metamodel in section C.2 show that ERRAGA makes a failure- non-failure prediction of the incompatible domain. When plotting the data samples over the metamodel, it shows an incorrect prediction near the incompatible data. Looking at the data samples in the incompatible domain, it seems as if ERRAGA is searching for compatible data near the predicted limit state. The uncertainty is not enough reduced in the incompatible area, which withholds convergence. There is no clear sign of any noise, but this can be so small it might not be visible on this scale.

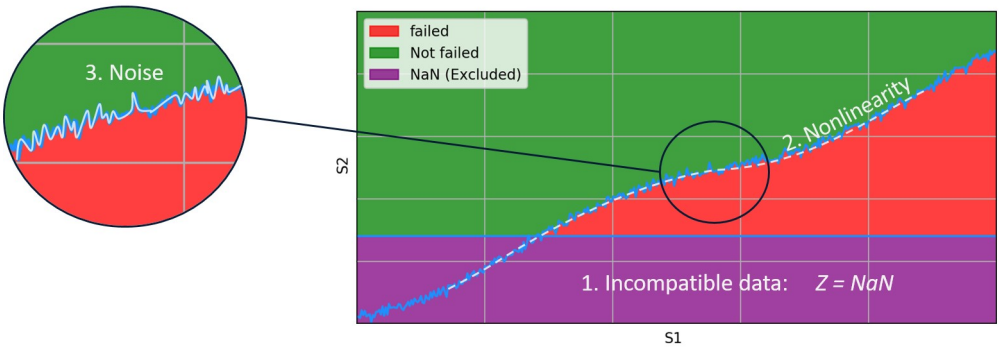


Figure 4.6: Complexities in the limit state of the two-variable model, also the visualisation of the artificial model (but with increased noise)

Artificial two-variables model

Figure 4.6 illustrates the complexities which are present in the two-variable model. More tests have to be performed to find out which complexities cause the most issues in terms of convergence. However, a 400 iterations run with PLAXIS takes about two days of computation time. Therefore, an artificial model without a numerical model is made. This artificial model is based on the results of the two-variables model. The artificial limit state function is a rather simple equation with the three complexities embedded in it. Section C.3 shows the exact details on how the model is made. Figure 4.6 gives an idea what the model looks like, the only difference is that this figure has an increased noise for visual purposes.

Note that it is not feasible to make an exact copy of the two-variable model limit state, the aim is solely to make a rough imitation with the artificial model. Most important for this research is that the artificial model shows the same convergence behaviour as the two-variables model, this is verified in section C.4. The artificial model is not converging, just like the two-variables model and the run is interrupted after 3282 iterations (top row of Table 4.6). Two other similarities are the steady convergence line and the numerous amount of incompatible samples.

In order to see what complexity gives the most issues for convergence, three runs are performed whereby the complexities are left out one by one. Section C.5 to section C.7 show the runs in detail, whereas the main results are visible in Table 4.6. Most interesting is the fact that all the three runs converge within a reasonable amount of realisations. This means that a combination of the three complexities causes no convergence. However, if one complexity is left out, the problem converges. Besides this main result, it appears that FORM can deal with all three cases if there is no incompatible data found, and if so, FORM stops and does not return an answer. The third finding is the erratic convergence behaviour, which appears in two of the three runs and is filtered out by PTK in one case.

Complexities	β **ERRAGA**	β **exact (IS)**	**Realisations**
NaN + nonlinear + noise	3.32 (no convergence)	-	3282
nonlinear + noise	3.31	3.31	158
NaN + noise	3.36	3.35	223
NaN + nonlinear	3.31	3.31	90

Table 4.6: Results of the artificial model for testing the influence of complexities

Discussion on complexities

To make the artificial model with all three complexities able to converge, it is necessary to eliminate or deal with at least one of the complexities. All three complexities are discussed on magnitude of complexity and multiple solutions are addressed on how to deal with them.

1. **Incompatible data (NaN):** has a high degree of complexity as comes forth from the first run with the artificial model (Appendix C.3). The majority of the samples are taken in the incompatible area. To solve this, two solutions are suggested:
   - To use a classification model from ERRAGA (see section 3.2).
   - To define the variable distributions in a way so no incompatible data can be produced, for example by using truncated distributions.
2. **Nonlinear limit state:** the nonlinearity has a smooth shape and is not angular which means ERRAGA should not have any problems with it. Also FORM can deal with this degree of nonlinearity as is shown in the previous tests.
3. **Noise:** this is the most complex based on Table 4.6. If noise is excluded in the limit state, ERRAGA only needs 90 realisations to converge. The noise is caused by PLAXIS-output: ΣMsf and $\Sigma Mstage$. During the tests, the 'noise term' of ERRAGA was already in use. Two other solution are suggested to deal with noise:
   - To use the 'Noise Variance Reduction Ratio' (NVRR) of ERRAGA
   - To make ΣMsf and $\Sigma Mstage$ more reliable. This requires a longer computation time (subsection 3.3.2)

From the solutions suggested above, it makes most sense to first implement the use of a classification model. As describe above, it is important to deal with incompatible data because otherwise, a lot of samples are fruitless. Besides that, the classification model is already implemented in ERRAGA and ready to use. If the classification model is not sufficient enough, noise may be an interesting aspect to look at.

Artificial two-variables model with SVM classification model

Appendix C.8 shows the detailed results of two ERRAGA runs with the SVM classification model. Note that for this run all three complexities are involved. The runs converge within 369 and 267 realisations (see Table 4.7). The results are very precise when comparing both beta's with the exact value. The use of the SVM model has reduced the amount of realisations massively. However, it might be necessary to reduce the realisations even more when scaling up from two to eleven variables. Logically, the number of required realisations will increase as well (Waarts, 2000). It is experienced during the use of the SVM model that it only starts working when incompatible data is found. Therefore, it is suggested to always use one of the classification models figuring it does not have a negative effect on the calculation when no incompatible data is found

Method	β	**Ucrit/CoVar**	**Realisations**
ERRAGA run 1	3.31	0.05	369
ERRAGA run 2	3.31	0.05	267
Exact (IS)	3.31	0.016	1000000

Table 4.7: Results of the artificial model using the SVM classification model

Artificial two-variables model with SVM and NVRR

What stands out in the previous runs is the fact that the beta value is steady for a long time before convergence occurs. This suggests that the uncertainty in the metamodel is too large for convergence of the prediction model and it is reduced very slowly. One way to reduce the uncertainty caused by (white) noise is the use of the Noise Variance Reduction Ratio (see section 3.2). It should be noted that there are some doubts about using this attribute as it can influence the final result. Three runs with the use of NVRR of 0.10, 0.75 and 0.99 are performed (see Table 4.8). The results are shown in detail in section C.9 - section C.11. Table C.9 shows that the use of NVRR reduces the amount of realisations drastically and the beta stays very close to the exact value. Section C.11 also zooms in on the visualisations of the PTK. The current display of the PTK does not show all the data that is necessary to fully understand the convergence behaviour of ERRAGA. The application of an external and more elaborate graph is therefore started. However, for this graph it is necessary to export a datafile. The code to export this datafile is implemented in the ERRAGA code in a later stage, which means that it is not possible to plot the external graph for the runs made before.

Method	β	**Ucrit/CoVar**	**Realisations**
ERRAGA (NVRR=0.10)	3.31	0.05	208
ERRAGA (NVRR=0.75)	3.32	0.05	85
ERRAGA (NVRR=0.99)	3.32	0.05	69
Exact (IS)	3.31	0.016	1000000

Table 4.8: Results of the artificial model using the SVM classification model

Main results

- A combination of the three complexities causes issues for convergence.
- The standard use of a classification model is recommended, as it deals very effectively with incompatible data.
- The use of the Noise Variance Reduction Ratio (NVRR) reduces the amount of required realisations, however, there are some doubts about using this attribute as it may influence the final result.

4.5.3 Numerical model (GRAPA)

This section discusses several runs of the numerical model for LSF Geotechnical which continue on the run named GRAPA v3n, shown in Figure 4.4. This model is more complex compared to the artificial model used in the previous section because it uses a numerical model and it has eleven stochastic variables instead of two. The settings which are tested in the previous section are used in this section, whereby the goal is to obtain convergence and to get a correct result. Table 4.9 gives an overview of the ERRAGA settings used for the runs. Note that there are three settings which are varied (Convergence Criterion, Noise Variance Reduction Ratio (NVRR) and Beta Prior). A run is considered converged if all three convergence criteria are converged. Table 4.10 to Table 4.12 shows the results where each table focusses on a different attribute. Table 4.13 compares the result of ERRAGA with conventional methods FORM and IS. Appendix D displays a selection of the runs in detail.

ERRAGA settings:		
Initial realisations	10	default
Min realisations (N Min)	50	
Max realisations (N Max)	100	
Convergence Criterion	varied	varied
Convergence Requirement	0.05	default
Learning Function	Ulearn	default
Classification Model	SVM	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	varied	varied
Size MCS pool	100000	default
Internal hyperparameters	10	default
Beta Prior	varied	varied

Table 4.9: ERRAGA settings for the reliability analysis for LSF Geo Case 1

Convergence criterion (prediction model)

The convergence criterion of the prediction model is one of the three convergence criteria, as described in subsection 3.2.2. This convergence criterion is not able to converge in many cases even if the the value of β remains steady for a large number of iterations. This means that ERRAGA takes training samples to reduce the uncertainty in the metamodel, but the reduction of this uncertainty is very small and barely influences the prediction of β. The convergence criterion of the prediction model is based on β and its uncertainty, as has been explained in subsection 3.2.3. That section also describes that the convergence criterion PfStop is too precise and the precision of BetaAbsStop is sufficient for the reliability analyses considered in this research. Table 4.10 shows that PfStop has trouble with convergence within a reasonable amount of realisations, whereas BetaAbsStop is able to achieve this. The degree of NVRR is also displayed in the table because this has a influence on the possibility of convergence as well. Remarkable is that there is only one run converged when using PfStop within only 62 realisations. However, this run encountered no incompatible data which makes the metamodel less complex and in theory, easier to converge. The degree of arbitrariness in the performance of reliability analysis with ERRAGA appears when a run with identical settings is unable to achieve convergence. Overall is the use of BetaAbsStop preferred for LSF Geo. The results of β seem to be influenced by NVRR, Table 4.11 further elaborates thereon.

	Attributes		**Results**			
Method	**ConvCrit**	**NVRR**	**Converged**	**N**	β	**Run id**
ERRAGA	PfStop	0.0	No	400 (Max)	4.43	GRAPA v3o
ERRAGA	PfStop	0.25	No	300 (Max)	4.35	GRAPA v3r
ERRAGA	PfStop	0.99	Yes no NaN	62	4.42	GRAPA v3p
ERRAGA	PfStop	0.99	No	400 (Max)	4.41	GRAPA v3s
ERRAGA	BetaAbsStop	0.0	No	1000 (Max)	4.41	GRAPA v3w
ERRAGA	BetaAbsStop	0.25	Yes	213	4.29	GRAPA v3u
ERRAGA	BetaAbsStop	0.99	Yes	160	4.23	GRAPA v3t

Table 4.10: Results of the reliability analysis for LSF Geo Case 1, thereby focussing on the influence of the convergence criterion (ConvCrit)

NVRR

With the use of BetaAbsStop there is a higher chance of convergence. Nevertheless, to achieve convergence it is also necessary to use NVRR. The effects of NVRR are clearly visible, whereby full usage of NVRR seriously reduces the amount of realisations. Full usage of NVRR should affect the outcome negatively, however, this is only slightly visible when comparing the beta values. There is a maximum difference between the converged β's of 0.15. BetaAbsStop is 0.05, this means that the difference between the prediction of β and its lower uncertainty bound is 0.05. One could expect this value as the maximum difference between the β's. This is not the case here, because there is a chance that the runs converge on a different design point or the metamodel is just slightly different but satisfies the convergence criterion. This results in a slightly different answer.

	Attributes		Results			
Method	**ConvCrit**	**NVRR**	**Converged**	**N**	β	**Run id**
ERRAGA	BetaAbsStop	0.0	No	1000 (Max)	4.41	GRAPA v3w
ERRAGA	BetaAbsStop	0.10	Yes	507	4.38	GRAPA v3y
ERRAGA	BetaAbsStop	0.25	Yes	213	4.29	GRAPA v3u
ERRAGA	BetaAbsStop	0.25	Yes	267	4.30	GRAPA v3u2
ERRAGA	BetaAbsStop	0.75	Yes	224	4.33	GRAPA v3x
ERRAGA	BetaAbsStop	0.99	Yes	160	4.23	GRAPA v3t

Table 4.11: Results of the reliability analysis for LSF Geo Case 1, thereby focussing on the influence of the noise variance reduction ratio (NVRR)

Beta Prior

Beta Prior is used to increase the chance to encounter failed and/or incompatible data. It is still possible to encounter failed and/or incompatible data when not using Beta Prior (=0.0), however, it looks like this is partly coincidental. Table 4.12 shows that only no failure samples were encountered when using a Beta Prior of 0.0. When using a Beta Prior of 2.0 or 5.0, all the runs found failed samples.

	Attributes	Results			
Method	**BetaPrior**	**Converged**	**N**	**Failure samples**	**Run id**
ERRAGA	0.0	No	100	No	GRAPA v3q
ERRAGA	0.0	Yes no NaN	62	Yes	GRAPA v3p
ERRAGA	0.0	No	100	No	GRAPA v3v
ERRAGA	2.0	No	300 (Max)	Yes	GRAPA v3r
ERRAGA	2.0	No	400 (Max)	Yes	GRAPA v3s
ERRAGA	5.0	Yes	160	Yes	GRAPA v3t
ERRAGA	5.0	No	1000 (Max)	Yes	GRAPA v3w
ERRAGA	5.0	Yes	224	Yes	GRAPA v3x
ERRAGA	5.0	Yes	507	Yes	GRAPA v3y

Table 4.12: Results of the reliability analysis for LSF Geo Case 1, thereby focussing on the influence of BetaPrior

Comparison with conventional methods

To verify the results of ERRAGA, a FORM and a IS reliability analysis is performed. A starting design point (DP) for the IS run is obtained by using the metamodel of the ERRAGA displayed in Table 4.13. The beta-value of the IS run is slightly higher compared to the results of ERRAGA. This is remarkable, but it is not a significant difference. Comparison of the influence factors in Table 4.14 show a large deviation at the parameters $Q_{surface}$ or $\phi_{sandmedium}$. Possibly there is

no clear DP for this model resulting in multiple possible locations. This might require a much higher accuracy of the calculations, because the IS run has a coefficient of variation of 0.1 and it converged within 2389 realisations which is fast for an IS run. The value of IS is higher than the result of ERRAGA, which might suggest that the IS run has its DP at a local minimum where ERRAGA is at the global minimum. This is also visible looking at the IF pie charts in Appendix D, where - despite of a software bug - the influence factors are not completely correct. However, the most influential parameter is very clear, either $Q_{surface}$ or $\phi_{sandmedium}$ is the most dominant parameter. This indicates that there are (at least) two DP's which have almost the same β-value.

	Results			
Method	**Converged**	**N**	β	**Run id**
ERRAGA	Yes	507	4.38	GRAPA v3y
FORM	No	600 (Max)	4.56	FORM v3z
IS	Yes	2389	4.59	IS v3u

Table 4.13: Results of the reliability analysis for LSF Geo Case 1, thereby focussing on the results with conventional methods.

Influence Factors:	**IS**	**ERRAGA**
$\gamma_{unsat;sandloose}$	0	1
$\phi_{sandloose}$	17	9
$E^{ref}_{50;sandloose}$	0	0
ϕ_{clay}	12	19
$E^{ref}_{50;clay}$	1	2
c^{ref}_{clay}	1	3
$\phi_{sandmedium}$	4	40
$Q_{surface}$	64	25
$f_{y;tube}$	0	0
D_{tube}	1	0
t_{tube}	0	0

Table 4.14: Influence factors (in %) of IS and ERRAGA (GRAPA v3u2)

Main results

- The use of BetaAbsStop is advised, for it results in faster convergence with accurate and reliable results.
- The use of Beta Prior is advised (if the user is able to make an estimate of beta), for it reduces the chance of not finding failed samples.
- The use of NVRR reduces the computation time and might be necessary when encountering noisy models.
- Getting a converged result seems to be partly arbitrary, which might be caused by finding the correct DP immediately instead of first finding a local minimum and then the global minimum. The encounter of noise is arbitrary and with less realisations there is a smaller chance of noise.
- There is a blank spot in the PLAXIS output of the IS-run (for Z_{Geo}), this is not the case for the ERRAGA runs.

4.6 Conclusions

The following conclusions can be drawn from this case study. In this section, a distinction made between concluding remarks about the PLX model and the effectiveness of the method.

PLX model

- The PLX model is reduced in such a way that it is not representative for the original model. Therefore, the results are too uncertain to draw any conclusions about the constructed structure. However, it is apparent which are in general the most influential factors per LSF.
- The LSF Frontwall shows four main influence factors; friction angle clay (middle layer), surface load, diameter of the combiwall tube and the model uncertainty of the bending moment. The influence of the friction angle of clay depends on the Geological layering and is thus project-specific.
- The LSF Geo shows four main influence factors; friction angle loose sand (upper layer), friction angle clay (middle layer), friction angle medium sand (bottom layer) and surface load.
- For Case 2 it is advised to add the variables stiffness modulus and volumetric weight of the lower soil layers (at the tip of the quay wall) according to chapter 2.

Effectiveness of the method

- It is advised to turn a Classification model on as default to deal with incompatible data. The classification model does not have any negative effect on the calculation when turned on and there is no incompatible data found.
- By default, the convergence criterion for the prediction model is PfStop. This is a relative criterion which makes it very precise when dealing with a high reliability index. BetaAbsStop is used as an alternative, where the boundary between the best estimate and the lower bound is 0.05 at the most.
- It is recommended to use Beta Prior to increase the chance of finding failed samples (especially when dealing with a high reliability index).
- The use of the Noise Variance Reduction Ratio (NVRR) reduces the required amount of realisations, however, it might reduce the reliability of the final answer.
- LSF Geo converges slower compared to LSF frontwall, this seems to be caused by white noise. The second graph of Figure B.5 shows a very small amount of noise (i.e. Sigma noise) compared to the average uncertainty (i.e. Sigma mean) for LSF Frontwall. On the other hand, LSF Geo shows a much higher degree of noise compared to the average uncertainty. The difference in average uncertainty between both LSF's is caused by the definition of the LSF. The Z-values of LSF Frontwall ranges in the order of 100,000 and LSF Geo ranges in the order of 1.
- Extending the Noise Bounds will not help the convergence of LSF Geo. The second graph of e.g. Figure D.9 shows that the white noise is already within the bounds of the default settings.
- Of all the runs of LSF Geo where no convergence occurred, the beta-value has been stable during the last 75% of the realisations. In all runs, this is caused by the convergence criterion of the prediction model which shows very slow improvement along the runs.

As a final conclusion, with these ERRAGA settings the reliability analysis achieves a credible answer with a short runtime (matter of hours for LSF wall failure by yielding and matter of days for LSF Geotechnical failure).

Chapter 5

Case 2: HHTT-quay

In this chapter, the second case study is discussed. It concerns the HHTT-quay, which is located in the port of Rotterdam at the Maasvlakte. The goal of this case study is to perform a probabilistic analysis on two limit states with the use of ERRAGA. In contrast to case study 1, this case needs some work in the set-up for reliability analysis. This means that the PLAXIS model has to be simplified, adjusted and verified and constant- and variable parameters and correlations have to be derived. To perform reliability analyses, the experience gained with ERRAGA of Case 1 will be taken into account.

5.1 Case description

The HES Hartel Tank Terminal (HHTT) quay wall is located at the first Maasvlakte and shown in Figure 5.1. The terminal has different sections with different types of quay walls, but only the part which contains a relieving platform is considered in this case. This part will be referred to as the HHTT-quay. It is a deep-sea quay for large sea-going vessels. The HHTT-quay is a combined wall with a relieving platform supported by MV-piles (Müller Verpress piles) and SI-piles (screwed displacement piles). The quay wall is equipped with sensors to monitor the behaviour of the structure (Schouten, 2020).

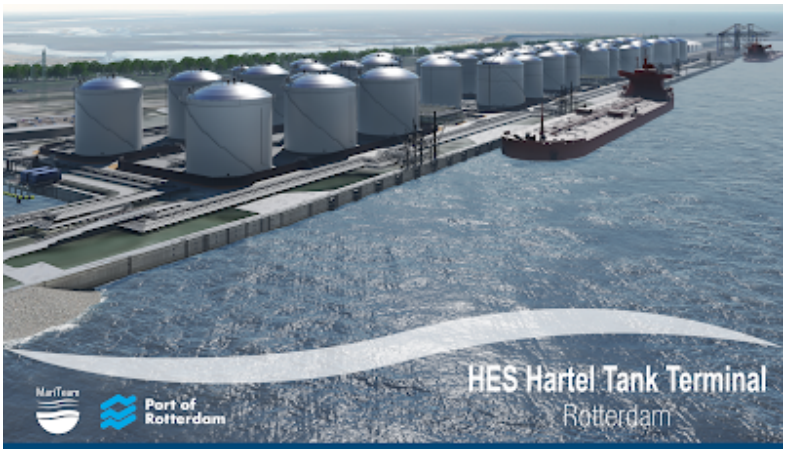


Figure 5.1: An illustration of the HHTT-quay, www.maps123.net

The quay wall is designed with Consequence Class 2 (CC2) which means the β should be around and above 3.8. Additionally, the quay wall is designed to have a lifespan of 100 year which will increase the reliability index slightly up to 4.0. This reliability has to be achieved for every individual limit state.

5.2 PLAXIS model

The Plaxis model of Schouten, 2020 is used for this case study, however, the model is simplified and necessary adjustments are made. The chosen relative settings and simplifications are discussed below.

Constitutive soil model

Similar to Case 1 (see chapter 4), case 2 also used the Hardening Soil (HS) model. This model accounts (in contrast to e.g. the Mohr-Coulomb model) for stress-dependency of stiffness moduli, which means that all stiffnesses increase with pressure. Schouten, 2020 also used the Hardening Soil model with small-strain stiffness (HSsmall). This model additionally accounts for the increased stiffness of soils at small strains. Compared to the HS model, the HSsmall model needs two extra parameters and it requires more calculation time. However, the differences in results between the HS and the HSsmall are small, and it is preferred to use the simpler HS model.

End-bearing

In the Plaxis model, the combiwall is modelled as a plate element which has no vertical bearing capacity. However, in reality the combiwall has a bearing capacity which plays a role of interest because of the (plugged) pipes. This means there is an upward force from the soil. In the Plaxis model, this is represented with a vertical spring (or end-bearing) to prevent very large vertical deformations of the plate element. This vertical spring has (almost) no unrealistic influence on the LSF Geo. However, it can play an unrealistic role in the simulation of LSF Front Wall. If the wall starts bending, the spring brings an unjustified contribution to the bending moment. In order restrict excessive deformations of the plate element, there are two options. It is possible to exclude the end-bearing element for both limit states or to use the prevent punching option. The prevent punching option restricts the soil volume surrounding the tip of the plate element from undergoing plasticity. This means that there will be no vertical displacement, which is not realistic.

Soil layers

To reduce the number of variables the number of soil layers is reduced from twelve to four layers. The layers are merged based on geotechnical parameters and geological background (Table 5.1).

Layer	Description
Sand fill	Mainly consists of fine sands
Sand 1	Consists of multiple fine and silty sand layers with slightly different characteristics. For simplification of the model, these layers are merged into one sand layer.
Clay	This layer is part of the clay layer of Wijchen. The upper part is a sandy clay layer and the lower part is an organic clay layer. These layers are merged as one clay layer.
Sand 2	This layer is part of the Kreftenheye formation and consists out of moderate to coarse sand.

Table 5.1: Description of soil layers in the simplified model

Figure 5.2 shows the cross-section of the original model used by Schouten, 2020 and the simplified model used in this research.

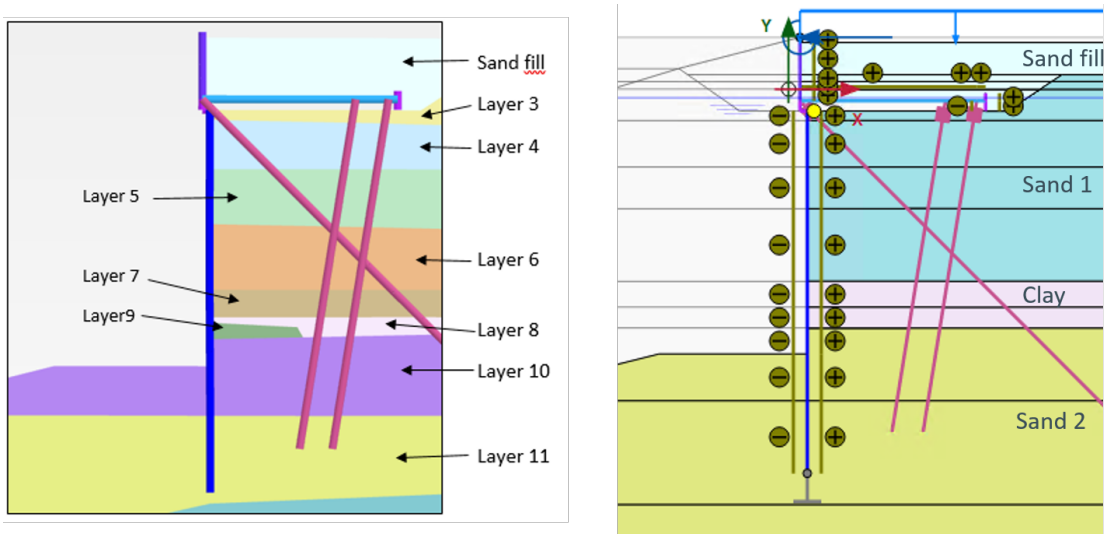


Figure 5.2: Left, the original soil profile by (Schouten, 2020). Right, the simplified model

Table 5.2 shows the characteristic soil parameters per soil layer, wherby the values are based on the design reports of Mariteam. The parameters of layer Sand 1 and Clay are determined by getting a weighted average of the involved layers. A few of these parameters are used as deterministic values in the PLAXIS model, but the majority will be represented by stochastic distributions.

		Sand fill	**Sand 1**	**Clay**	**Sand 2**
γ_{unsat}	[kN/m3]	18	18.5	18	18
γ_{sat}	[kN/m3]	20	20	18	19
ϕ' SLS	[degrees]	40	33	26	36
ϕ' ULS	[degrees]	-	27	21	30
ψ	[degrees]	5	3	0	6
c'	[kN/m2]	1	1	7.5	1
E_{50}^{ref}	[kN/m2]	40000	28000	6500	40000
E_{oed}^{ref}	[kN/m2]	40000	28000	3250	40000
E_{ur}^{ref}	[kN/m2]	120000	106000	26000	160000
m	[-]	0.500	0.573	0.700	0.510

Table 5.2: The soil parameters which are based on the design reports of Mariteam.

External loads

There are three external loads, as can be seen in Figure 5.3:

- **Tank load:** A permanent load caused by storage tanks
- **Annual surcharge:** The variable annual expected surcharge load
- **Bollard load:** Load on bollards caused by moored ships, the load is modelled as a point load and a bending moment

For LSF Frontwall and LSF Geo the annual and permanent surcharge is active and the bollard force and momentum is inactive, which reduces the momentum in the combi wall.

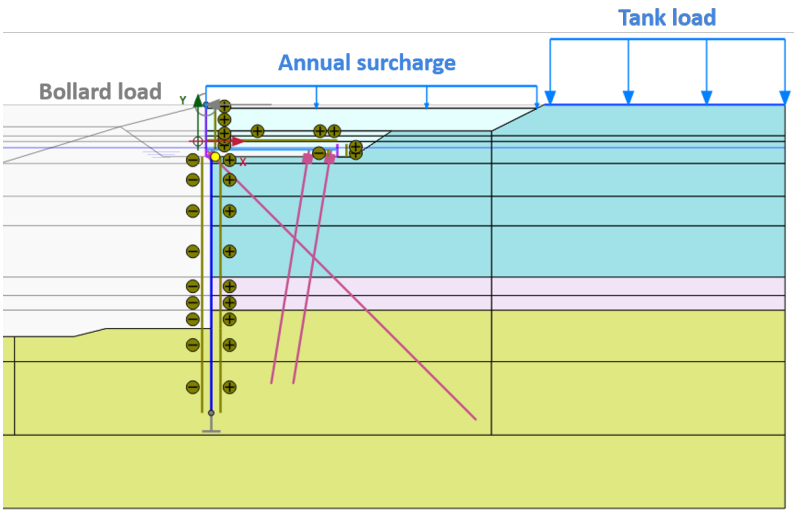


Figure 5.3: The simplified PLAXIS model

5.2.1 Failure of the PLAXIS model

The failure of the quay wall in the PLX model is interesting to analyse, for it shows if the quay fails in a way we would expect, if not, this is a point of interest. A reliability analysis contains multiple failed model evaluations and therefore cannot all failed model evaluations be analysed in detail. Two failed evaluation of the ERRAGA runs used later in this chapter are carefully chosen. Important is that the first failure mode only fails by LSF Front wall and not by LSF Geo, the second failure mode vice versa. These failure modes are expected to be interesting because of their differences. The failure modes are shown in Appendix G (beware of the distance- and colour scale differences between the two situations).

The failure mode of LSF Front wall shows a great displacement behind the middle of the wall. This indicates that the wall is heavily deforming and bending at this point. There is also little displacement at the tip of the combi wall, this indicates that there is some geotechnical instability.

The failure mode of LSF Geo is comparable to the failure mode of LSF Front wall, however there is relatively more deformation around the tip of the pile. This is expected because of the shallow depth of the pile tip compared to the harbour floor. The stability of the quay wall is therefore mainly dependent on the strength of the lower layer, that is why the parameters of the lower layer have such high influence factors

5.3 Statistical distributions

5.3.1 Stochastic variables

The sand fill is the top layer and it is not interacting with the combi wall (only with the relieving platform). All the parameters of the sand fill are deterministic. The sand fill is considered to make the volumetric weight variable, however, because of the relative small thickness of the layer and to limit the amount of variables, the choice is made to keep this parameter deterministic.

Variable	**Unit**	**Distribution**	μ	σ	V
Unsaturated volumetric weight: Sand 1 $\gamma_{unsat;sand_1}$	[kN/m3]	normal	18.5	0.925	0.05
Saturated volumetric weight: Sand 1 $\gamma_{sat;sand_1}$	[kN/m3]	normal	20	1.0	0.05
Internal friction angle: Sand 1 ϕ_{sand_1},	[degrees]	normal	39	3.9	0.1
Secant stiffness for: Sand 1 $E^{ref}_{50;sand_1}$	[kN/m2]	LogNormal	40000	8000	0.2
Internal friction angle: Clay ϕ_{clay},	[degrees]	normal	31	3.1	0.1
Secant stiffness: Clay $E^{ref}_{50;clay}$	[kN/m2]	LogNormal	10000	2000	0.2
Cohesion: Clay c^{ref}_{clay}	[kN/m2]	LogNormal	7	1.4	0.2
Unsaturated volumetric weight: Sand 2 $\gamma_{unsat;sand_2}$	[kN/m3]	normal	18	0.9	0.05
Saturated volumetric weight: Sand 2 $\gamma_{sat;sand_2}$	[kN/m3]	normal	19	0.95	0.05
Internal friction angle: Sand 2 ϕ_{sand_2},	[degrees]	normal	43	4.3	0.1
Secant stiffness for: Sand 2 $E^{ref}_{50;sand_2}$,	[degrees]	normal	60000	12000	0.2
Annual surcharge $Q_{surface}$	[kN/m2]	Gumbel	26	5.2	0.2
Yield stress tube $f_{y;tube}$	[kN/m2]	LogNormal	475000	19000	0.04
Tube diameter D_{tube}	[m]	normal	1.42	0.071	0.05
Tube thickness t_{tube}	[m]	Uniform	0.024	0.0012	0.05
Model uncertainty θ_M θ_N θ_{soil}	[-]	LogNormal	1	0.1	0.1

Table 5.3: The stochastic variables used for case 2

Volumetric weights

Sand 1 and Sand 2 are the soil layers for which the saturated- and unsaturated volumetric weight γ_{unsat} and γ_{sat} is represented by a stochastic variable. This forms a contrast with case 1, where only the upper sand layer has variable weights, but the lower sand layer is added according to the findings of chapter 2. It is important to note that the saturated and unsaturated volumetric weight of the loose sand layer are fully correlated, which means that these two variables function as one variable in the probabilistic analysis. The coefficient of variation is taken from Roubos, 2019.

Friction angle

The friction angle ϕ' is determined by assuming:

- Coefficient of variation: V = 0.1 (Roubos, 2019)
- The SLS value (Table 5.2) is the value at 5% of the distribution (CDF = 0.05), which means that the SLS value is at -1.64 standard deviation from the mean value.

By using these assumptions, the mean value of the distribution is calculated by:

$$\mu = \frac{SLS}{1 - 1.64 \cdot V} \tag{5.1}$$

The values are rounded to half number. With this method, the ULS values in Table 5.2 have probability of occurrence of 0.1% which is assumed to be reasonable.

Stiffness parameters

The values of E_{50}^{ref} shown in Table 5.2 are the characteristic values. The mean is determined in the same way as the Friction angle. The Coefficient of variation V = 0.2 (Roubos, 2019) and the mean value is determined with Equation 5.1.

E_{oed}^{ref} and E_{ur}^{ref} are determined by using a ratio with respect to E_{50}^{ref}. In this case, the design document of Mariteam is used:

- For sand: $E_{oed}^{ref} \approx E_{50}^{ref}$
- For clay: $E_{oed}^{ref} \approx \frac{1}{2} E_{50}^{ref}$
- For sand and clay: $E_{ur}^{ref} \approx 4E_{50}^{ref}$

These ratios are not always possible because only ratios within certain ranges are valid depending on the mutual values and by parameters such as ϕ. By using the SoilTest function in PLX, the minimum and maximum ratios between $E_{50}^{ref} : E_{oed}^{ref} : E_{ur}^{ref}$ are determined:

Sand	1	:	1	:	4
Clay	1	:	0.54	:	4

Cohesion

Only the cohesion c' of the clay layer is variable. The mean is determined with the use of Equation 5.1 and the Coefficient of variation V = 0.2 (Roubos, 2019). The sand layers have a cohesion of 1 kPa, otherwise the sand layers are completely cohesionless which can cause unrealistic behaviour.

Annual surcharge

Mariteam uses an annual surcharge $Q_{surcharge}$ of 40 kN/m2 for SLS and 44 kN/m2 for ULS. On the gumbel distribution used in Case 1 the value of 44 kN/m2 is on 99% of the distribution which suits SLS and ULS.

Yield stress steel tube

The yield stress of the steel tube f_y is 483 MPa, this value is divided by 1.016 to account for the design life of 100 years (design reports of Mariteam) The reduced value is taken as the mean value of the LogNormal distribution. The standard deviation is based on the same variance used in case 1 and originates from Roubos, 2019.

Dilation Angle (constant)

The Dilation Angle ψ is used to determine the plastic strain rate, for sands it is defined by using the following rule of thumb: $\psi = \phi - 30$, where ψ canNot be smaller than 0 for this case. The dilation angle of clay is set to 0.
The determination of ψ is in practice more complicated than the rule of thumb used here. In the probabilistic analysis it is possible that an internal friction angle above 50°is used in the numerical model, which results in an ψ of 20°. In the lower sand layer, this will not cause any problems because the ϕ is such a dominant parameter that the structure will not fail for a ψ of 20°. However, if the upper sand layer can have a negative influence on stability of the quay wall when having a high ϕ and thus a high ψ:

- Sand 1: Phi=39.5, Psi=9.5, u max = 0.2134m, dx max = 0.2028m, M max = -2447 kNm/m
- Sand 1: Phi=39.5, Psi=19.5, u max = 0.3527m, dx max = 0.3422m, M max = -3669 kNm/m
- Sand 1: Phi=49.5, Psi=19.5, u max = 0.3498m, dx max = 0.3380m, M max = -3651 kNm/m

There is an increase in the maximum deformation and in the bending moment because high ψ means a high volumetric strain, the 'extra' volumes of soil deforms the wall. However, these values are still too low to cause failure and therefore there is not a maximum value set for ψ.

Tank load (constant)

There are large storage tanks for fuel located 47 meters from the edge of the quay wall. These tanks have a characteristic load of 168 kPa (SLS) (design reports of Mariteam). This is the maximum load in normal use, as the load is based on tanks completely filled with water. This characteristic value is used as the deterministic value in the PLX model, because the use of an ULS would be too unfavourable.

5.3.2 Correlations

There are only correlations between variables of the same soil layers, the other stochastic variables are assumed uncorrelated and are for reasons of simplicity not displayed in Table 5.4. note that the saturated and unsaturated volumetric weight of the sand layers are fully correlated which means that these two variables function as one variable.

#	Variable	1	2	3	4	5	6	7	8	9	10	11
1	$\gamma_{unsat;sand_1}$	1	1	0.5	0.5	-	-	-	-	-	-	-
2	$\gamma_{sat;sand_1}$	1	1	0.5	0.5	-	-	-	-	-	-	-
3	ϕ_{sand_1}'	0.5	0.5	1	0.25	-	-	-	-	-	-	-
4	$E^{ref}_{50;sand_1}$	0.5	0.5	0.25	1	-	-	-	-	-	-	-
5	ϕ_{clay}'	-	-	-	-	1	0.25	-0.65	-	-	-	-
6	$E^{ref}_{50;clay}$	-	-	-	-	0.25	1	0.12	-	-	-	-
7	c^{ref}_{clay}	-	-	-	-	-0.65	0.12	1	-	-	-	-
8	$\gamma_{unsat;sand_2}$	-	-	-	-	-	-	-	1	1	0.5	0.5
9	$\gamma_{sat;sand_2}$	-	-	-	-	-	-	-	1	1	0.5	0.5
10	ϕ_{sand_2}'	-	-	-	-	-	-	-	0.5	0.5	1	0.25
11	$E^{ref}_{50;sand_2}$	-	-	-	-	-	-	-	0.5	0.5	0.25	1

Table 5.4: The correlations of case 1

5.4 Limit state Front wall

5.4.1 Settings and results

The chosen settings in ERRAGA are shown in Table 5.5. The main difference in settings compared to the LSF Front wall of case 1 is the use of the UNIS learning function instead of Ulearn. UNIS learn is designed to work with IS (see subsection 3.2.3), which is used from the start of the run because Beta Prior is activated. The results of the reliability assessments of LSF Front wall without model uncertainty are shown in Table 5.6 and results with model uncertainty are shown in Table 5.7. Further details of the runs are shown in Appendix E.

ERRAGA settings:		
Initial realisations	10	default
Min realisations (N Min)	30	
Max realisations (N Max)	100	
Convergence Criterion	varied	varied
Convergence Requirement	0.05	default
Learning Function	UNIS	
Classification Model	GPC	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0	default
Size MCS pool	varied	varied
Hyperparameter Optimisations (Opts)	10	default
Beta Prior	varied	varied

Table 5.5: The selected ERRAGA settings for the reliability analysis for LSF Front wall Case 2

	Attributes		**Results**			
Method	**ConvCrit**	**BetaPrior**	**Converged**	**N**	β	**Run id**
ERRAGA	BetaAbsStop	5.0	Yes	93	4.17	GRAPA2 v2i
FORM	-	-	CRASH	-	-	FORM v2i

Table 5.6: The results of the reliability analysis for LSF Front wall Case 2, no model uncertainty

	Attributes		**Results**			
Method	**ConvCrit**	**BetaPrior**	**Converged**	**N**	β	**Run id**
ERRAGA	PfStop	10.0	Yes	199	4.04	GRAPA2 v2f2
ERRAGA	BetaAbsStop	5.0	Yes	101	4.18	GRAPA2 v2j
ERRAGA	PfStop	8.0	Yes	96	4.20	GRAPA2 v2j2

Table 5.7: The results of the reliability analysis for LSF Frontwall Case 2, with model uncertainty

5.4.2 Discussion of the results

Based on the performed case study, the following remarks can be made.

The runs have resulted in a β around 4 which is a reasonable answer and expected according to the design requirements.

Model uncertainty seem to have a very small or no effect on the results. The LSF includes two model uncertainty factors, one for the normal force and one for the bending moment. These might cancel each other out for a reliability analysis with a low amount of realisations (up to 200). Therefore, it might be better to put a combined uncertainty factor in front of the equation. Another possibility is that the model uncertainty has almost no influence and the deviation in the β-values is larger than the deviation caused by the influence factors.

An increase in Beta Prior was necessary from 5 to 8 and 10 because no failure samples were found. Experience has learned that encountering failed- or incompatible data during the first 100 realisations is partly arbitrary. Therefore, it is assumed that failed- or incompatible samples would be encountered if performing sufficient attempts. It is not advised to use a Beta Prior much higher than the resulting β, therefore a good alternative would be the use of a manual user input of a priori variable set to force the encounter of failure samples.

5.4.3 Main results

The performed analysis in this section has led to the following main results.

- When using PfStop there are four times as many realisations needed for convergence of LSF Frontwall when comparing case 2 to Case 1 (200 vs 50). This is probably caused by the encounter of incompatible data.
- A β-value around 4 is an expected value and a reasonable answer.
- LSF Front wall encounters very little noise which makes PfStop a suitable convergence criterion.
- Even though PfStop is suitable, BetaAbsStop is considered to be reliable enough as well.
- FORM encounters too much incompatible data and therefore crashes.
- Model uncertainty has little to no influence on the LSF.
- The number of stochastic variables increase from 13 to 15 when including model uncertainties. This increase in variables does not show a clear increase in the number of required realisations. There might be a link to the very low influence of the model uncertainty factors.

5.5 Limit state Geotechnical

5.5.1 Settings and results

The selected settings of ERRAGA for LSF Geo are shown in Table 5.8. The main difference in settings compared to LSF Geo of case 1 is the use of the UNIS learning function instead of U learn. Also the number of internal hyperparameter optimisations are increased, which may decrease the erratic behaviour of the convergence. In contrast to the quay wall of case 1, the quay wall with a relief platform is expected to fail as a passive wedge when looking at LSF Geo. The results of the reliability assessments of LSF Geo without model uncertainty are shown in Table 5.9 and results with model uncertainty are shown in Table 5.10. Details of the runs are shown in Appendix F.

ERRAGA settings:		
Initial realisations	10	default
Min realisations (N Min)	50	
Max realisations (N Max)	varied	varied
Convergence Criterion	BetaAbsStop	
Convergence Requirement	0.05	default
Learning Function	UNIS	
Classification Model	GPC	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	varied	varied
Size MCS pool	100000	default
Hyperparameter Optimisations (Opts)	varied	varied
Beta Prior	5.0	

Table 5.8: The selected ERRAGA settings for the reliability analysis for LSF Geo Case 2

	Attributes		**Results**			
Method	**NVRR**	**Opts**	**Converged**	**N**	β	**Run id**
ERRAGA	0.10	10	Yes	624	4.18	GRAPA2 v3g2
ERRAGA	0.10	20	Yes no naN	157	4.13	GRAPA2 v3h2
ERRAGA	0.10	20	Yes no naN	156	4.06	GRAPA2 v3h3
ERRAGA	0.10	20	Yes no naN	393	4.16	GRAPA2 v3h4

Table 5.9: The results of the reliability analysis for LSF Geo Case 2, no model uncertainty

	Attributes		**Results**			
Method	**NVRR**	**Opts**	**Converged**	**N**	β	**Run id**
ERRAGA	0.25	20	no	400	3.12	GRAPA2 v3k
ERRAGA	0.75	20	Yes no naN	70	3.13	GRAPA2 v3j

Table 5.10: The results of the reliability analysis for LSF Geo Case 2, with model uncertainty

Influence Factors	Model uncertainty					
	Excluded				Included	
$\gamma_{(un)sat;sand_1}$	1	0	2	3	4	0
ϕ_{sand_1}	0	0	1	0	2	0
$E^{ref}_{50;sand_1}$	8	0	0	0	0	3
ϕ_{clay}	2	0	1	0	0	0
$E^{ref}_{50;clay}$	6	0	1	0	0	2
c^{ref}_{clay}	0	3	0	1	1	4
$\gamma_{(un)sat;sand_2}$	28	48	27	41	15	23
ϕ_{sand_2}	31	44	63	50	6	13
$E^{ref}_{50;sand_2}$	1	1	0	2	0	1
$Q_{surface}$	4	4	0	1	1	6
$f_{y;tube}$	7	0	1	1	4	0
D_{tube}	4	0	0	2	2	1
t_{tube}	9	1	4	0	0	2
θ_{soil}	-	-	-	-	63	45
Run id:	v3g2	v3h2	v3h3	v3h4	v3k	v3j

Table 5.11: The influence factors (in %) of LSF Geo Case 2. The four runs left have no model uncertainty included, the two runs on the right include model uncertainty.

5.5.2 Discussion of the results

The increase of the number of Internal Hyperparameter Optimisations from 10 to 20 is done to reduce the erratic behaviour of the metamodel. Looking at the graphs shown in Appendix F, the erratic behaviour is not reduced. However, most remarkable is the encounter of incompatible data when using 10 Hyperparameter Optimisations and no encounter of incompatible data when using 20 Hyperparameter Optimisations. Of course there is an arbitrary factor for the encounter of incompatible data, but on the other hand an increase in Hyperparameter Optimisations will give a better estimate of the DP. If the DP is far from the incompatible domain, the change to encounter incompatible data decreases.

Comparing the β-values of Table 5.9, the results are considered to be reliable. The maximum deviation is 0.12 which is considered to be reasonable.

The encounter of incompatible data shows that the complexity of the model increases which therefore requires more training samples for convergence. The runs which encountered incompatible data (convergence or no convergence) had a stable β for the last 75% of the run, and the final value is very close to the result of the other runs. This stable *beta* means that ERRAGA takes training samples but they do not contribute to the convergence of one or more convergence criteria. For all LSF Geo runs it is the convergence criterion of the prediction model which has trouble with convergence. This is also observed in Case 1 and because the convergence criterion PfStop was found to be too precise, the less precise BetaAbsStop was introduced. However, based on the results of Case 2, it is possible that BetaAbsStop can be still too precise.

The influence factors are included in Table 5.11 and show that there is a clear similarity between the runs. In addition, it is clear which variables are most influential. However, the deviations between the runs are large (up to 32%), the most logical explanation is that the runs have a different DP. A remarkable outcome is the large influence of the model uncertainty in contrast to the model uncertainty of LSF Front wall. chapter 2 describes that in the partly comparable case of Wolters, 2012 the most influential parameters for LSF Geo are the stiffness modulus, the internal friction angle and the soil weight of the lower sand layer. That corresponds with the results of this case, except that the stiffness modulus is not dominant for this case. However, the stiffness modulus has not had a dominant role in the results of both, Case 1 and Case 2.

For both LSF, the β represents a reasonable value. However, not all influential parameters are made stochastic, for instance the parameters for the water and soil levels. When adding these parameters, uncertainties are added too, which would make one expect that the β becomes smaller. However, the current water and soil level parameters (constants) used in the calculations are already on the lower boundary of their uncertainty spectrum. Consequentially, it is expected that making these parameters stochastic will not have a substantial impact on the β-value.

5.5.3 Main results

The performed analysis in this section has led to the following main results.

- The β-value is around 4, which is the expected value.
- Similar to case 1, LSF Geo encounters a high degree of noise (in comparison to LSF Front wall).
- The use of nVRR is desirable because the convergence behaviour of the prediction model drops exponentially. The use of nVRR is debatable, however, from the gained result it does not influence the results drastically.
- When incompatible data is encountered, an increased amount of training samples is required.
- In many runs there is no improvement in the results (*beta*-value) for the last 50%-75% of the run, this is clearly visible in section F.3. For all LSF Geo runs it is the prediction model convergence criterion which has trouble with convergence.
- ERRAGA is able to perform a reliability analysis of LSF Geo with reasonable results and within a short amount of time.
- Model uncertainty θ_{soil} has a very high influence on the obtained results.
- The number of stochastic variables increase from 13 to 14 when including model uncertainties. Based on these results it is difficult to draw any firm conclusions on how the number of stochastic variables will influence the required amount of realisations.

5.6 Lowered harbour floor

The previous section found that the simplified model of the HHTT-quay has a reliability index (β) of 4 which suits a CC2 design with a required lifetime of 100 years. In practice it is possible to allow a lower β for quay walls, which makes it possible to e.g. lower/dredge the harbour floor or increase the surcharge on top of the quay. To see what potential metamodelling has in practice, a reliability analysis is performed on a situation where the harbour floor is lowered by 2.60 meters (from 25.40m-NAP to 28.00m-NAP) as illustrated in Figure 5.4 to see what effect it has on the reliability index (β) of LSF Geo.

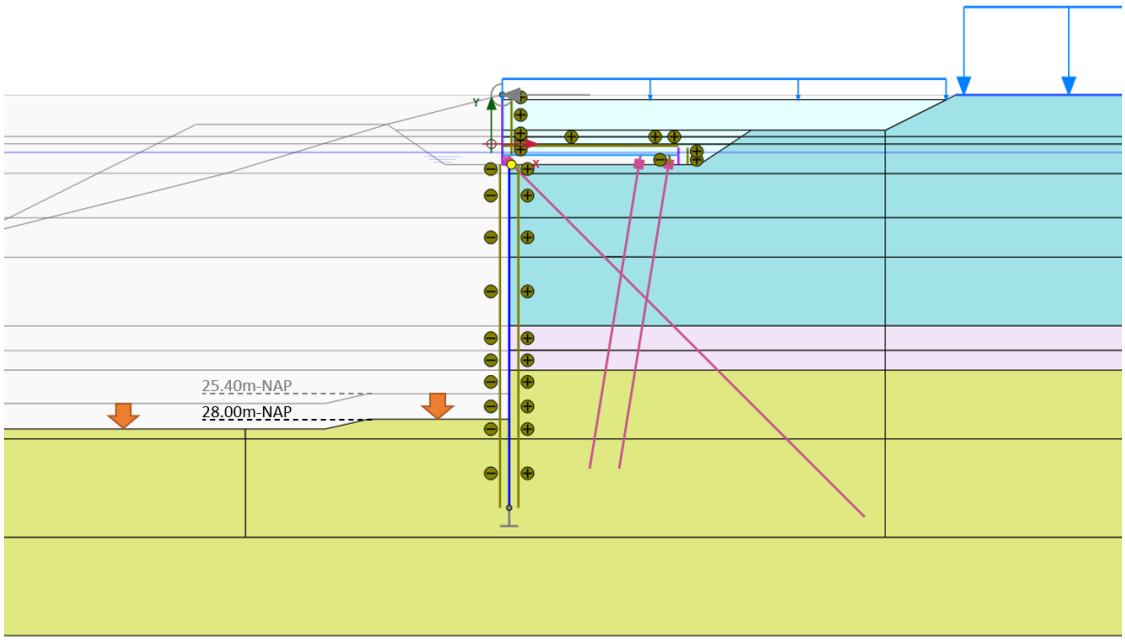


Figure 5.4: The harbour floor is lowered by 2.60 meters, from 25.40m-NAP to 28.00m-NAP

Results

The β of the original level harbour floor is based on the results of Table 5.9. Table 5.12 shows that the lowering of the harbour floor reduces the β to a value of 3.66, intuitively this is a reasonable reduction. Details of the reliability analysis are shown in section F.5. The reliability analysis has all the aspects of a real case, with a calculation time of a few days this has the potential to be a useful tool in practice.

Level harbour floor	β
25.40m-NAP	4.15
28.00m-NAP	3.66

Table 5.12: The effect of lowering the harbour floor on the reliability index β

5.7 Conclusions

The following conclusions can be drawn from this case study. In this section, a distinction made between concluding remarks about the PLX model and the effectiveness of the method.

PLAXIS model

- LSF Geo shows two main influence factors: friction angle sand 2 (lower layer) and volumetric weight sand 2 (lower layer) as concluded in chapter 2. When model uncertainty is included, this becomes the most influential variable.
- For both LSF's a β-value around 4 is an expected value and a reasonable answer.
- Looking at the values of β and the influence factors, the distributions of the variables are considered to be realistic enough.
- Despite the simplifications of the numerical model and the fact that not all influential parameters are variable, the results of the reliability assessments look reasonable. This gives confidence in the performance of ERRAGA on this case study.

Application in engineering practice, lowering the harbour floor, LSF Geo

- With the use of ERRAGA it is possible to see the effects of the lowering of the harbour floor for LSF Geo. This takes relatively low effort for the user and the calculation is completed within a reasonable amount of time (200 model evaluations in 60 hours)
- The harbour floor is lowered 2.60m which is a sizeable amount, the β drops from 4.15 to 3.66 which is logic consequence but acceptable figuring a β of 3.66 is still above CC1 and slightly below CC2.

Effectiveness of the method

- With experiences of Case 1 it was possible to analyse Case 2 in an efficient manner.
- For the convergence of LSF Geo it is necessary to use BetaAbsStop. Still this convergence criterion seems to be too precise in some cases. An increase of the convergence value can of course solve this.
- When comparing the results of LSF Front wall and LSF Geo there is a clear correlation between the degree of noise and the amount of required realisations.
- The encounter of incompatible data doubles the amount of required realisations.
- There is no clear difference observed when increasing the number of stochastic variables. The increase is relatively small, LSF Front wall was increase from 13 to 15 and LSF Geo from 13 to 14. However, an increase would be expected, the influence of the added stochastic variable might play an important role in the effects. More research needs to be performed to judge the impact of an increase in stochastic variables.

The recommended ERRAGA attribute settings are shown in Table 5.13, which are based on the experience from both case studies.

ERRAGA settings:	LSF Front wall	LSF Geo
Initial realisations	10	10
Min realisations (N Min)	50	50
Max realisations (N Max)	100	100
Convergence Criterion	PfStop	BetaAbsStop
Convergence Requirement	0.05	0.05
Learning Function	UNIS	UNIS
Classification Model	GPC	GPC
Noise Term	True	True
Noise Bounds	1E-10, 0.01	1E-10, 0.01
Noise Variance Reduction Ratio (NVRR)	0	0.75
Size MCS pool	100000	100000
Hyperparameter Optimisations (Opts)	10	20
Beta Prior	4.0	4.0

Table 5.13: The recommended default settings for ERRAGA

Chapter 6

Conclusion and recommendations

6.1 Assessment of the research questions

This research has two objectives. To start, it aims to gain insight in the concept of reliability based design and assessment of quay walls. The second objective is to test the applicability of metamodelling for reliability analysis of quay walls and where necessary to further develop the method. These objectives were achieved by performing and investigating reliability analysis on two case studies situated in the port of Rotterdam. With this objective, the main research question was formulated in chapter 4, along with four sub-questions. This section answers the sub-questions, and is followed by an answer to the main question.

What are essential parts of the reliability assessment of quay walls?

With respect to quay walls, several essential parts of the reliability assessments have been identified in the course of this research. To start, the Limit state functions (LSF) and Failure mechanisms play an important role, because they define failure. However, it is key to select the relevant LSF's and to formulate them as realistically as possible. Secondly, the probabilistic method plays an essential role, because a complex LSF can only be solved with the use of probabilistic methods. In combination with a time-consuming numerical model, this even requires a metamodelling approach. Thirdly, the stochastic variables constitute the uncertainty in the reliability analysis, which makes them an important element. Therefore, the stochastic variables must be formulated with care. The correlation variables in their turn, correlate the stochastic variables and as such form the fourth essential part in performing reliability assessment. The fifth and final key element to model a realistic response of the quay wall is the numerical model.

Which properties of reliability assessments of quay walls make reliability analysis with conventional methods challenging?

The following properties may pose an obstacle to make a reliability analysis. The first challenging property is the computation time, which is the pitfall for strong probabilistic methods such as MCS. These methods are able to solve every reliability assessment if computation time did not play a role. Of course will the amount of variables influence the required computation time. Based on the results of the cases, it is hard to make an accurate statement about the ratio between the number of variables and the computation time because there is little variation in the amount of variables used in the cases.

Secondly, the LSF of the two-variable model of case 1 showed three types of complexities, namely Incompatible data, Non-linearity and (white) noise. These three subjects are now discussed separately.

To start, the non-linearity can provide a challenge. The degree of non-linearity shown in the two-variable model is clearly visible. This means that the multi-variable models also have a degree of non-linearity, but this is hard to visualise. Therefore, it is uncertain what the degree of non-linearity is and how it contributes to the complexity of the reliability analysis. The two-variable model shows a smooth line without sharp angles, so this should pose no issues for ERRAGA.

Furthermore, incompatible data, which is encountered when the numerical model does not return an answer, may hinder a successful reliability assessment. The encounter of incompatible data mainly happens when failure occurs before the last construction phase. Methods like FORM will crash when encountering too many incompatible samples. The two-variable model of case 1 clearly shows that ERRAGA takes many fruitless samples from the incompatible domain when no classification model is used.

Lastly, a noisy model response complicates convergence. The presence of noise is not clearly visible on the two-variable model of case 1. However, it is observed by ERRAGA in both cases. LSF Front wall shows a negligible amount of noise, where LSF Geo shows a high degree of noise.

How does a metamodelling approach perform in realistic cases compared to conventional methods on the aspects of robustness, efficiency and accuracy?

In terms of robustness there is one major point of attention for ERRAGA, which is the finding of the failure domain. When no failed (or incompatible) data is found during the beginning of the run (10 to 15 realisations) ERRAGA starts to reuse one of the first samples until it reaches the maximum amount of realisations and returns no results. This can be solved by increasing the value of Beta Prior or by implementing an option for the user to manually insert a set of variables. A major advantage of ERRAGA compared to many conventional methods is that it can deal with incompatible data by using a classification metamodel.

The efficiency of ERRAGA is very high compared to strong probabilistic methods such as MCS or IS. ERRAGA needs about 100-1000 realisations (depending on the accuracy) for LSF Geo of case 2, which takes one to twelve days. A similar simulation with IS or MCS will take months or even years. Compared to FORM it shows a similar or even better efficiency. Results show that ERRAGA needs less realisations compared to FORM. However, FORM has a higher accuracy set and therefore it is not completely possible to compare the results in a detailed way.

The accuracy of ERRAGA is hard to prove as it is almost impossible to get a 100% certain exact answer with conventional methods. From the one comparison with FORM, ERRAGA shows a good accuracy in terms of β, with a maximum deviation of 0.04. The influence factors have a lower accuracy, however, the same order of magnitude is shown. The reliability of β falls within the reasonable bounds and can be more narrow if lower convergence criteria are applied. It is uncertain how this will affect the computation time. Influence factors show a lot more deviations between the runs, but it is possible to derive the most influential variables.

What are the recommended 'user defined' settings for metamodelling to obtain optimal performance?

The recommended ERRAGA settings are based on experience of both cases. The most relevant attributes are explained below:

Convergence Criterion (prediction model)
The prediction model of LSF Front wall converges fast and usually has very low uncertainty. PfStop is preferred because the criterion is easily met and it is more accurate compared to BetaAbsStop. For LSF Geo, it is necessary to use the less precise convergence criterion BetaAbsStop to achieve convergence. Even though the use of BetaAbsStop, there are many runs which had difficulties to converge. Many runs show a stable β for the last 50%-75% of the run, which indicates that ERRAGA takes training samples but they barely contribute to the convergence of the model.

Therefore it is possible that BetaAbsStop is still too precise or the convergence value is too small. BetaAbsStop has a default convergence value of 0.05 but can easily be increased to e.g. 0.1 which is still an acceptable precision (depending on the requirements of the user).

Learning Function
When encountering noisy data and when using beta prior (and thus the use of IS is activated) it is advised to use the UNIS learn function.

Classification model
With regard to the settings for the classification model, it is advised to always switch on one of the two classification models in order deal with incompatible data. Convergence is very likely to be prevented if the incompatible domain is not classified. The classification model does not have any negative effect on the calculation when switched on and there is no incompatible data encountered.

Noise Bounds
The upper bound of ERRAGA's noise term has a default value of 0.1. LSF Geo has its range of Z-value in the magnitude where it fits this upper bound. LSF Front wall has Z-values in a much higher order of magnitude. If a higher degree of noise would have been encountered it is advised to raise this upper bound to a Sigma which suits the magnitude of the Z-values.

NVRR
It is advised to set the NVRR of LSF Geo to 0.75, which means that half the amount of observed noise is ignored. If convergence is reached easily, the user is advised to rerun the reliability analysis with a lower NVRR of e.g 0.5, 0.1 or even 0.

Beta Prior
Beta Prior should be an estimate of the expected β, but this value is project dependent. During this research a higher value is often used to make sure failed samples were found in the initial phase of the run. This problem can also be avoided if it is possible for the user to make an a-priori estimate of initial realisations which are in the failure domain.

What is the potential of metamodelling for reliability-based assessment of quay walls in engineering practice?

This research has investigated the potential of reliability-based assessment of quay walls. Although this research question does not allow for long analyses, it must be concluded that ERRAGA can provide a value for the assessment of quay walls. It is possible to determine the effects on the β of LSF Geo when lowering of the harbour floor and the calculation is performed within a reasonable amount of time. This is exceptional because the reliability analysis has 13 stochastic variables and is performed on a complex quay wall with relieving platform, modelled with a numerical model. This can possibly be applied on other aspects as well, e.g. increasing the surcharge on the quay or looking at the effects of extreme water levels. Therefore, ERRAGA has the potential to become a very useful tool for engineering in practice.

Using ERRAGA does require some specific knowledge, but this thesis tries to contribute thereto by explaining the method and more specifically by providing the recommended default settings. Assuming that the user possesses the required knowledge and knows how to deal with the discussed complexities, metamodelling for reliability-based assessment of quay walls has the potential to add value in engineering practice.

6.2 Recommendations

In this section, several recommendations are made regarding the improvement of the methodology and further research:

A priori input of variable sets: An a-priori estimate of the initial realizations could be inserted manually, while currently the initial realizations are picked randomly. If there is a low probability of failure, there is a very small chance that ERRAGA will find failed samples, also during the first learning cycle. That will stop/fail the calculation. If the user can insert a set of parameters that will cause failure, this will be prevented. Even though this can also be avoided by using other attributes like beta prior, this is not the main purpose of beta prior and manually inserting data points might also assist in this issue.

Improved visualisations: The PTK shows only one convergence criterion (prediction model) of ERRAGA. However, ERRAGA contains three convergence criteria and to make a good analysis of the convergence behaviour or to judge why convergence is prevented, all three convergence criteria should be visualized. Another recommendation is to visualize the white-noise observed by ERRAGA together with the average uncertainty of the metamodel. Additionally, a visualisation of the hyperparameters can be valuable, because this indicates how the metamodel changes and if it maybe switches between completely different metamodels during the calculation. This visualisation is developed and used during this research, however, it could be convenient to implement this in the PTK.

Pause the PTK and adjust the ERRAGA settings: The possibility to pause the PTK and to adjust settings during a run would be a valuable improvement. This is possible when using ERRAGA without the PTK, however, during the calculations performed for this research, ERRAGA is controlled by the PTK and thus an intermediate stop is not possible or not user friendly. The PTK calculation can be paused, however, it is currently not possible to continue the calculation afterwards.

ERRAGA in system reliability analysis: This research aimed to gain insight on the use of ERRAGA, but has not included system reliability analysis in its scope. However, it would be beneficial if this would be possible for the assessment of reliability of quay walls. In order to do so, the question must be answered whether ERRAGA is able to deal with multiple LSF's in one reliability assessment or if it is better to make a separate metamodel for each LSF and combine these metamodels prior to performing a reliability assessment on the combined metamodel. This is something to assess in future research.

Investigating the noise of LSF Geo: In both case studies, there is a considerable amount of noise encountered. This research has been able to deal with this noise, by using ERRAGA's attributes. However, it is not completely certain what causes the noise, but it is clear that this noise originates from the numerical model. Therefore, it is suggested to test the combination between $\Sigma Mstage$ and ΣMsf in future research to verify if this does give a smooth LSF.

Number of stochastic variable versus the computation time: The effect of the number of stochastic variables on the required amount of realisations and thus the computation time are not clearly visible in this research. Based on the results of this research it seems that the influence factor of the added stochastic variable plays an important role in the effects, but more research should be performed to get an insight in these effects.

Investigation on the accuracy and reliability of the influence factors: In this research, it became apparent that different runs on the influence factors may lead to different values per run. Although the found values are broadly similar, they are sometimes far from identical. It is unclear what causes these differences, but the logical explanation seems to be that the convergence criteria are not sufficiently accurate. It would therefore be useful to determine what is necessary to improve the estimate of the influence factors.

Bibliography

De Gijt, J., & M.L.Broeken. (2003). *Handboek kademuren*. Gouda : CUR.

Echard, B., Gayton, N., & Lemaire, M. (2011). Ak-mcs: An active learning reliability method combining kriging and monte carlo simulation. *Structural Safety*, *33*, 145–154. https://doi.org/10.1016/j.strusafe.2011.01.002

Kentrop, D. (2021). *A metamodelling approach to reliability updating with dike construction survival* (MSc Thesis). Technical University Delft.

Krige, D. (1951). A statistical approach to some basic mine valuation problems on the witwatersrand. *J Chem Metall Min Soc S Afr*, *94*, 95–111. https://doi.org/10.2307/3006914

Melchers, R. (1989). Importance sampling in structural systems. *Structural Safety*, *6*, 3–10.

Post, M., Schweckendiek, T., Roubos, A., & van de Greef, J. (2021). *Reliability analysis of quay walls*. Deltares.

Rasmussen, C., & Williams, C. (2006). *Gaussian processes for machine learning*. MIT Press.

Roubos, A. (2019). *Enhancing reliability-based assessments of quay walls* (Doctoral dissertation). Technical University Delft.

Ruggeri, P., Fruzzetti, V. M. E., & Scarpelli, G. (2019). Renovation of quay walls to meet more demanding requirements: Italian experiences. *Coastal Engineering*, *147*, 25–33. https://doi.org/https://doi.org/10.1016/j.coastaleng.2019.01.003

Schöbi, R., Sudret, B., & Marelli, S. (2016). Rare event estimation using polynomial-chaos kriging. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, *500*, D4016002. https://doi.org/10.1061/AJRUA6.0000870

Schouten, O. (2020). *Optimising the functionality of smart quay walls using measurement data obtained during the construction process* (MSc Thesis). Technical University Delft.

Schweckendiek, T. (2006). *Structural reliability applied to deep excavations. coupling reliability methods with finite elements.* (MSc Thesis). Technical University Delft.

Van den Eijnden, A., Schweckendiek, T., & Hicks, M. (2021). *Metamodelling for geotechnical reliability analysis with noisy and incomplete models.*

Waarts, P. H. (2000). Structural reliability using finite element analysis - an appraisel of dars: Directional adaptive response surface sampling.

Wolters, H. (2012). *Reliability of quay walls* (MSc Thesis). Technical University Delft.

Appendices

Appendix A

Limit State Functions

This appendix describes the complete Limit State Functions for both LSF Wall and LSF Geo.

LSF FrontWall

Post et al., 2021 split the limit state function in two formulas for efficiency reasons, one for the water side and one for the land side. This formula is capable to include the effects of corrosion, this is only considered for the benchmark case (LSF Front wall, case 1).

deterministic values
3 * PU28 tussenplanken from W+B report
$L_{sheet} = 1.8$ m
$I_{sheet} = 106490.0E-8$ m4
$E_{steel} = 2.1E8$ kN/m2
$dx_{max} = 1.10$ m
$A_{anchor} = 6442E-6$ m2 anchor anchor 101,6x28mm

Formulas to assess a tubular pipe in steel class 3

determine the corroded cross sectional area of the tube

$$A_{shell;water;corr} = \frac{1}{8}\ \pi\ ((D_{tube} - 2dt_{water;tube})^2 - (D_{tube} - 2t_{tube})^2) \tag{A.1}$$

$$A_{shell;land} = \frac{1}{8}\ \pi\ ((D_{tube})^2 - (D_{tube} - 2t_{tube})^2) \tag{A.2}$$

$$A_{tube;corr} = A_{shell;water;corr} + A_{shell;land} \tag{A.3}$$

determine the corroded moment of inertia of the combi wall

$$I_{shell;tube;water;corr} = \frac{1}{128}\ \pi\ ((D_{tube} - 2\ dt_{water;tube})^4 - (D_{tube} - 2\ t_{tube})^4) \tag{A.4}$$

$$I_{shell;tube;land} = \frac{1}{128}\ \pi\ (D_{tube}^4 - (D_{tube} - 2\ t_{tube})^4) \tag{A.5}$$

$$I_{y;corr} = I_{shell;tube;water;corr} + I_{shell;tube;land} \tag{A.6}$$

$$Z_{shell;tube;water;corr} = \frac{2}{3\ \pi}\ \frac{(D_{tube} - 2\ dt_{water;tube})^3 - (D_{tube} - 2\ t_{tube})^3}{(D_{tube} - 2\ dt_{water;tube})^2 - (D_{tube} - 2\ t_{tube})^2} \tag{A.7}$$

$$Z_{shell;tube;land} = \frac{2}{3\ \pi}\ \frac{D_{tube}^3 - (D_{tube} - 2\ t_{tube})^3}{D_{tube}^2 - (D_{tube} - 2\ t_{tube})^2} \tag{A.8}$$

$$Z_{y;tube} = \frac{Z_{shell;tube;land}\ A_{shell;land} - Z_{shell;tube;water;corr}\ A_{shell;water;corr}}{A_{tube;corr}} \tag{A.9}$$

$$I_{tube;corr} = I_{y;corr} - Z_{y;tube}^2\ A_{tube;corr} \tag{A.10}$$

determine the values per m1 quay wall (indicated by means of double capitals)

$$AA_{tube;corr} = \frac{A_{tube;corr}}{L_{system}} \tag{A.11}$$

$$II_{combiwall;corr} = \frac{I_{tube;corr} + I_{sheet}}{L_{system}} \tag{A.12}$$

$$WW_{combiwall;water;corr} = \frac{II_{combiwall;corr}}{0.5\ D_{tube} - dt_{water;tube} + Z_{y;tube}} \tag{A.13}$$

$$WW_{combiwall;land;corr} = \frac{II_{combiwall;corr}}{0.5\ D_{tube} - Z_{y;tube}} \tag{A.14}$$

NO corrosion included for now, cross section per anchor NOT per m1 quay

LSF dx: deformations of front wall

$$Z_{SLS} = dx_{max} - \theta_U\ dx_{actual} \tag{A.15}$$

LSF Wall: strength of front wall, using combination M and N per node for land or water side. from PLX model, negative bending moment = compression land side, negative normal force is compression

$$Z_{STR;yield;land} = fy - abs\left(\frac{\theta_M\ MM_{plaxis;land}}{WW_{combiwall;land;corr}} + \frac{\theta_N\ NN_{plaxis;land}}{AA_{tube;corr}}\right) \tag{A.16}$$

$$Z_{STR;yield;water} = fy - abs\left(-\frac{\theta_M\ MM_{plaxis;water}}{WW_{combiwall;water;corr}} + \frac{\theta_N\ NN_{plaxis;water}}{AA_{tube;corr}}\right) \tag{A.17}$$

LSF Geo

$\Sigma MstageResult$ is a stability indicator where the soil body is stable if $\Sigma MstageResult = 1$. Another criterion for a stable soil body is that $\Sigma Mstage$ has to be larger than 0.995.

$$if\ \Sigma Mstage > 0.995\ and\ \Sigma MstageResult = 1: \qquad Z_{GEO} = \theta_{Soil}\ \Sigma Msf - 1.0 \tag{A.18a}$$

$$else: \qquad Z_{GEO} = \theta_{Soil}\ \Sigma Mstage - 1.0 \tag{A.18b}$$

Appendix B

Case 1: LSF Front wall, Numerical model

B.1 GRAPA v2t

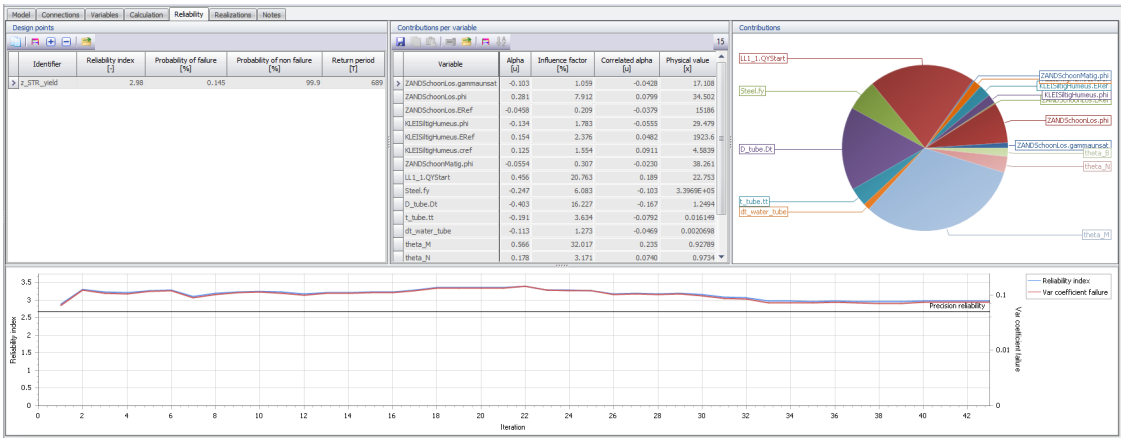


Figure B.1: Convergence graph of ERRAGA run: GRAPA v2t run 2

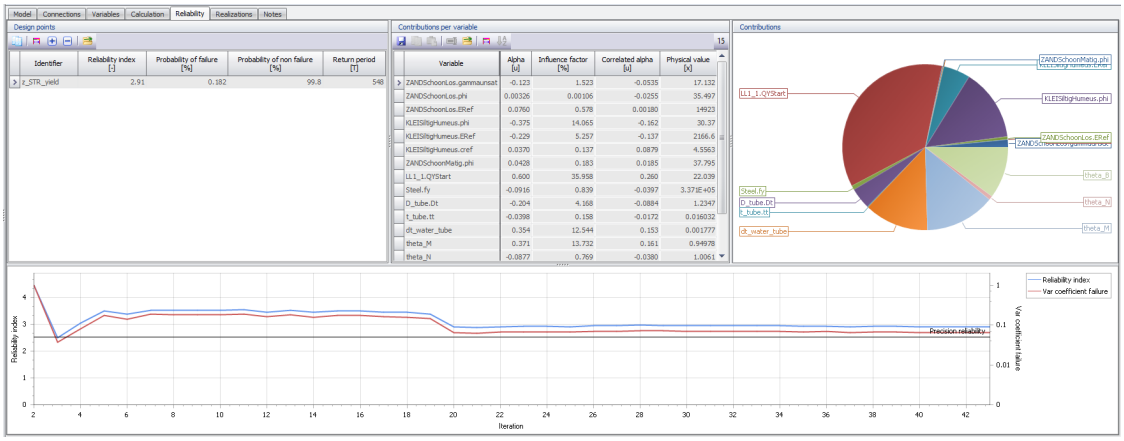


Figure B.2: Convergence graph of ERRAGA run: GRAPA v2t run 3

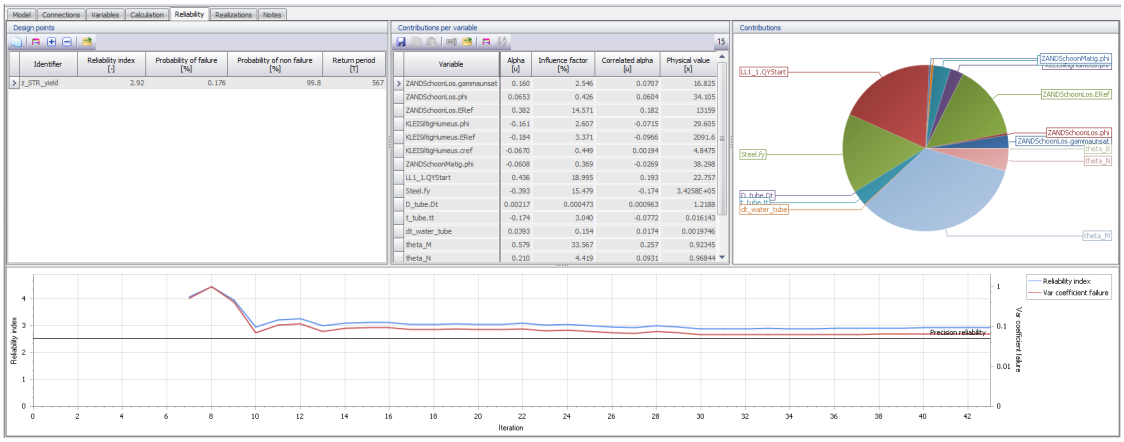


Figure B.3: Convergence graph of ERRAGA run: GRAPA v2t run 4

B.2 GRAPA v2y

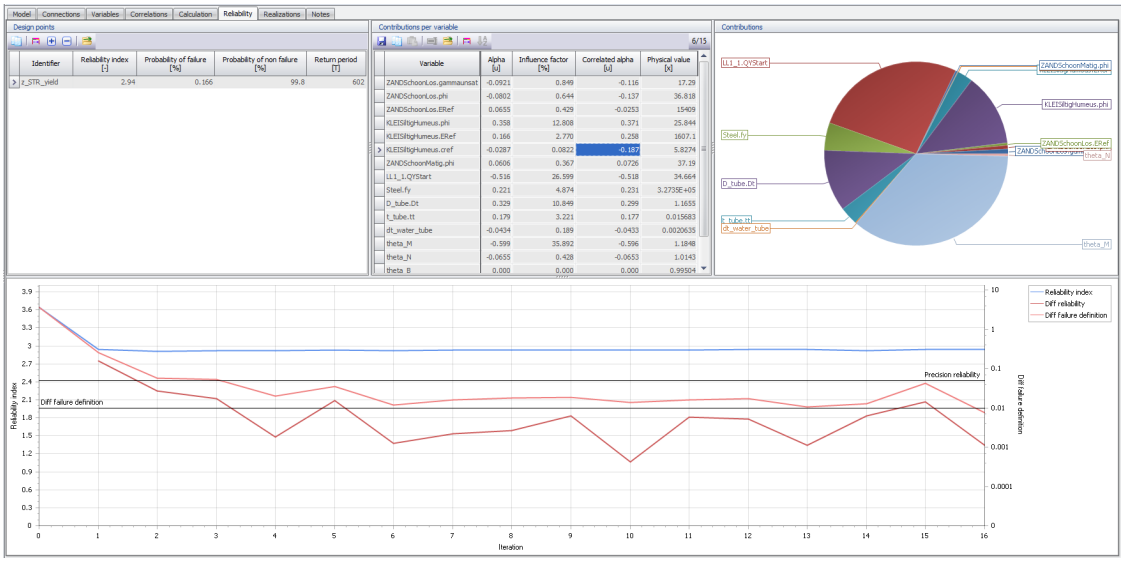


Figure B.4: Convergence graph of FORM run: GRAPA v2y FORM

ERRAGA settings:		
Initial realisations	10	default
Min realisations (N Min)	50	
Max realisations (N Max)	200	
Convergence Criterion	PfStop	default
Convergence Requirement	0.05	default
Learning Function	Ulearn	default
Classification Model	GPC	default
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0.0	default
Size MCS pool	100000	default
Hyperparameter Optimisations (Opts)	10	default
Beta Prior	0.0	default

Table B.1: ERRAGA settings for GRAPA v2y

Method	**Beta**	**Realisations**	**Convergence**
FORM	2.94	272	YES
ERRAGA	2.91	53	YES

Table B.2: Results of GRAPA v2y

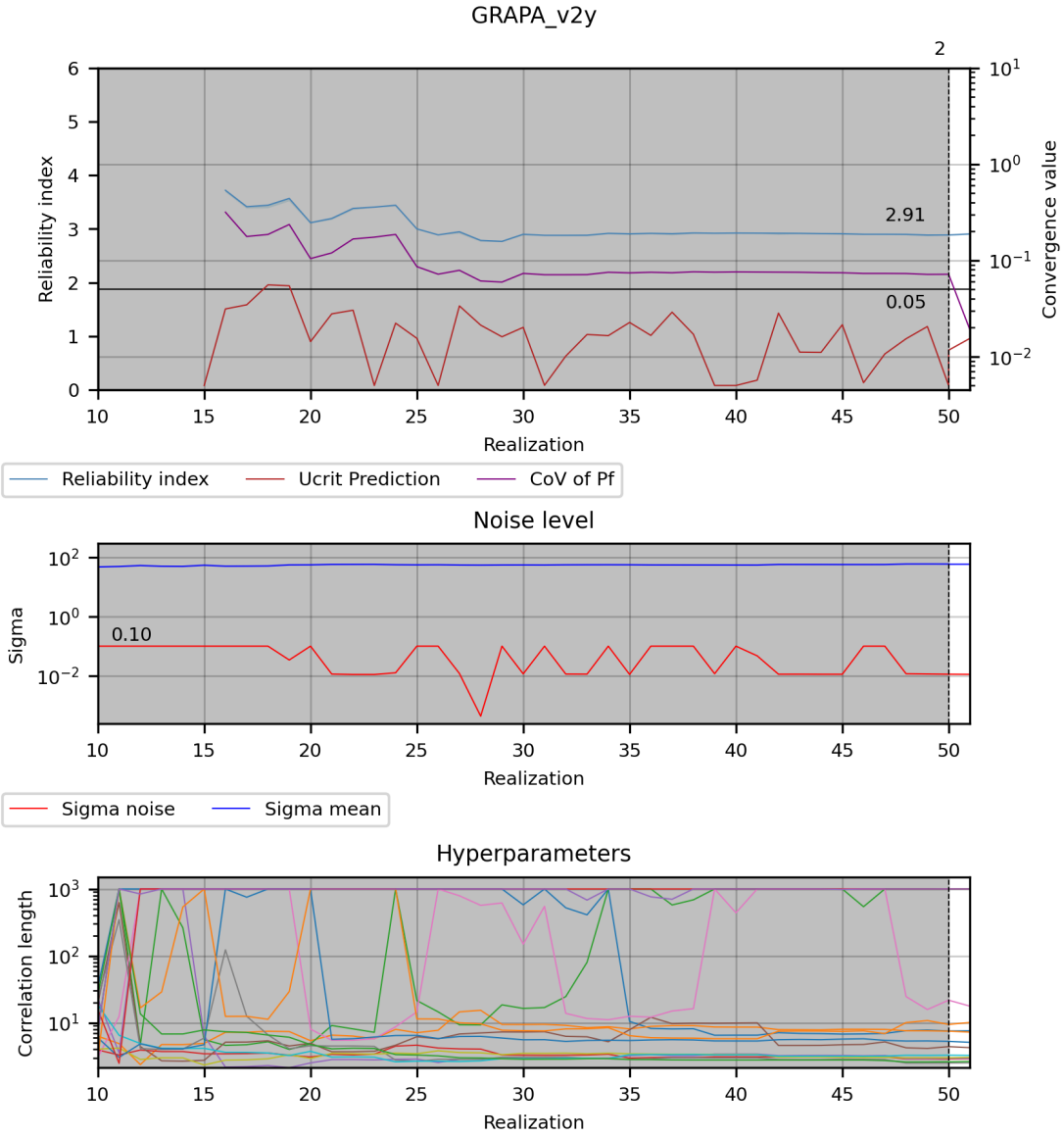


Figure B.5: Convergence graph of ERRAGA run: GRAPA v2y

Findings

- Convergence of the prediction model converged after 21 realisations.
- Convergence of the MCS/IS run converges after one IS-update
- The average uncertainty of the metamodel (Sigma mean) is high (approximately 60) compared to the reliability index
- The observed white noise fluctuates between 0.10 and 0.01 (0.10 is the default upper bound)

Appendix C

Case 1: LSF Geotechnical, two-variables Model

For this analysis, the GRAPA v3n model is reduced to a two-stochastic variables model. Figure C.1 shows the list of defined variables. Only the ZANDSchoonLos.phi and LL.1_1.QYStart are stochastic variables, the rest is set on deterministic. ZANDSchoonLos.phi represents the internal friction angle of one the sand layers, for this analysis the mean value is reduced from 35° to 30° because in the first test run no failure points were found. LL.1_1.QYStart is the annual expected surcharge load on surfacelevel. Both parameter were picked because of their high influence factors in the original model.

Variable	Distribution	Inverted	Truncated	Mean [x]	Deviation	Unit deviation	Shift [x]	Scale [-]	Quantile [x]	From measurements
Model: plaxisController										
ZANDSchoonLos.gammaunsat	Deterministic			17					17	
ZANDSchoonLos.gammasat	Deterministic			19					19	
ZANDSchoonLos.phi	Normal			30	3,5	[x]			30	
ZANDSchoonLos.ERef	Deterministic			15000					15000	
KLEISiltigHumeus.phi	Deterministic			29					29	
KLEISiltigHumeus.ERef	Deterministic			2000					2000	
KLEISiltigHumeus.cref	Deterministic			5					5	
ZANDSchoonMatig.phi	Deterministic			38					38	
LL1_1.QYStart	Gumbel			26	5,2	[x]	23,66	4,0544	25,146	
Steel.fy	Deterministic			3,37E+05					3,37E+05	
D_tube.Dt	Deterministic			1,219					1,219	
t_tube.tt	Deterministic			0,016					0,016	
Model: Internal										
dt_water_tube	Deterministic			0					0	
theta_U	Deterministic			1					1	
theta_M	Deterministic			1					1	
theta_N	Deterministic			1					1	
theta_F	Deterministic			1					1	
theta_Soil	Deterministic			1					1	
fy_a	Deterministic			5,15E+05					5,15E+05	

Figure C.1: List of variables, two-variable model LSF geotechnical failure

C.1 Primary results

A first run is performed with the same ERRAGA settings (Table C.1) as used for the original GRAPA v3n model in Figure 4.4. This gives the possibility to compare the results of both runs with each other.

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.0	default
Classification Model	None	default
Beta Prior	0.0	default
Min Realizations	50	
Max Realizations	100	

Table C.1: ERRAGA settings for two-variable model of GRAPA v3n

Figure C.2 shows the final visualisations from the PTK. The run is not converted as can be seen in the upper plot where the red line is not below the black line. The lower scatter plot shows all the samples. Striking is the amount of samples taken in the incompatible area (purple dots) which provide no information about failure/non-failure. This incompatible (or excluded) area is below a ZANDSchoonLos.phi value of ±18°. A low internal friction angle causes failure in one of the construction phases of the Plaxis model, which returns NaN (not a number) to the PTK a result. Noteworthy is the fact that ERRAGA tries to search for failure/non-failure points in the incompatible area where the method thinks the limit state is, as the purple dots form lines, which could be an extension of the known limit state. As noted above, these samples are fruitless, nevertheless there are ways to deal with this problem which will be discussed later on.

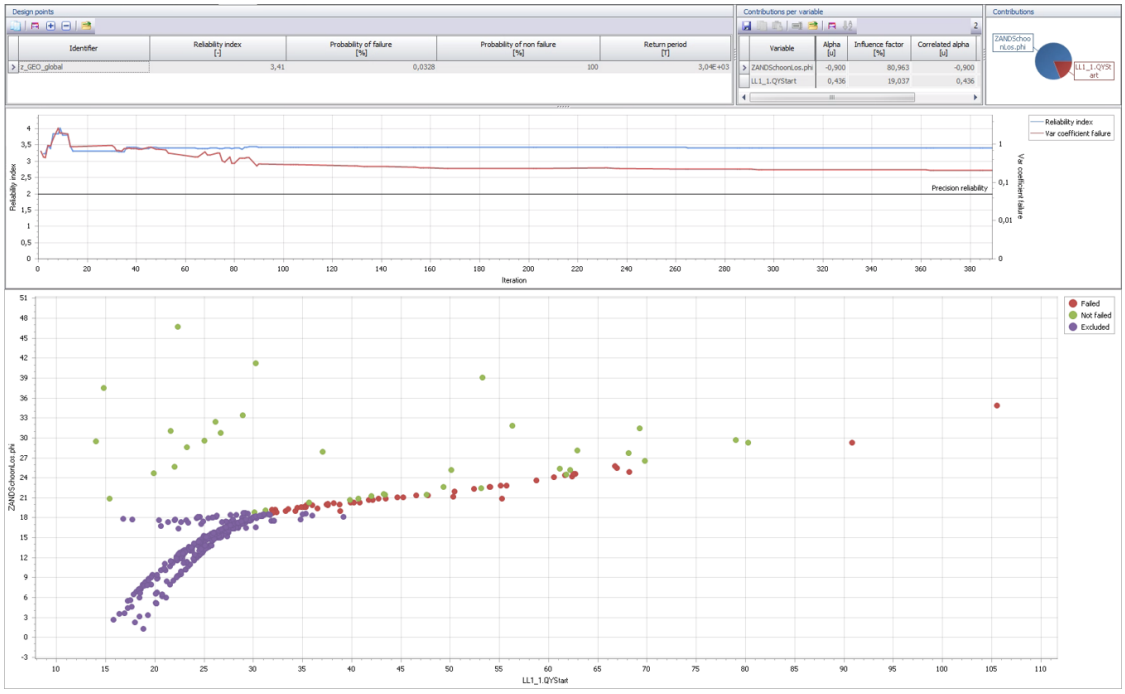


Figure C.2: The PTK visualisations of two-variables model of GRAPA v3n

The numerical samples and the resulting metamodel of this run are analysed since there might

be other complexities besides the incompatible area. Figure C.3 zooms in on the lower graph of Figure C.2. It is hard to tell if there is any noise, because the green and red dots are on the same line and not mixed like one would see if there is a high amount of noise. On the other hand, there might be a small noise which cannot be determined visually. Another remark is the limited amount of samples which do not give enough information to rule out any chances over-fitting by ERRAGA. The next section analyses the metamodel to see if the limit state determined by ERRAGA agrees with the obtained samples.

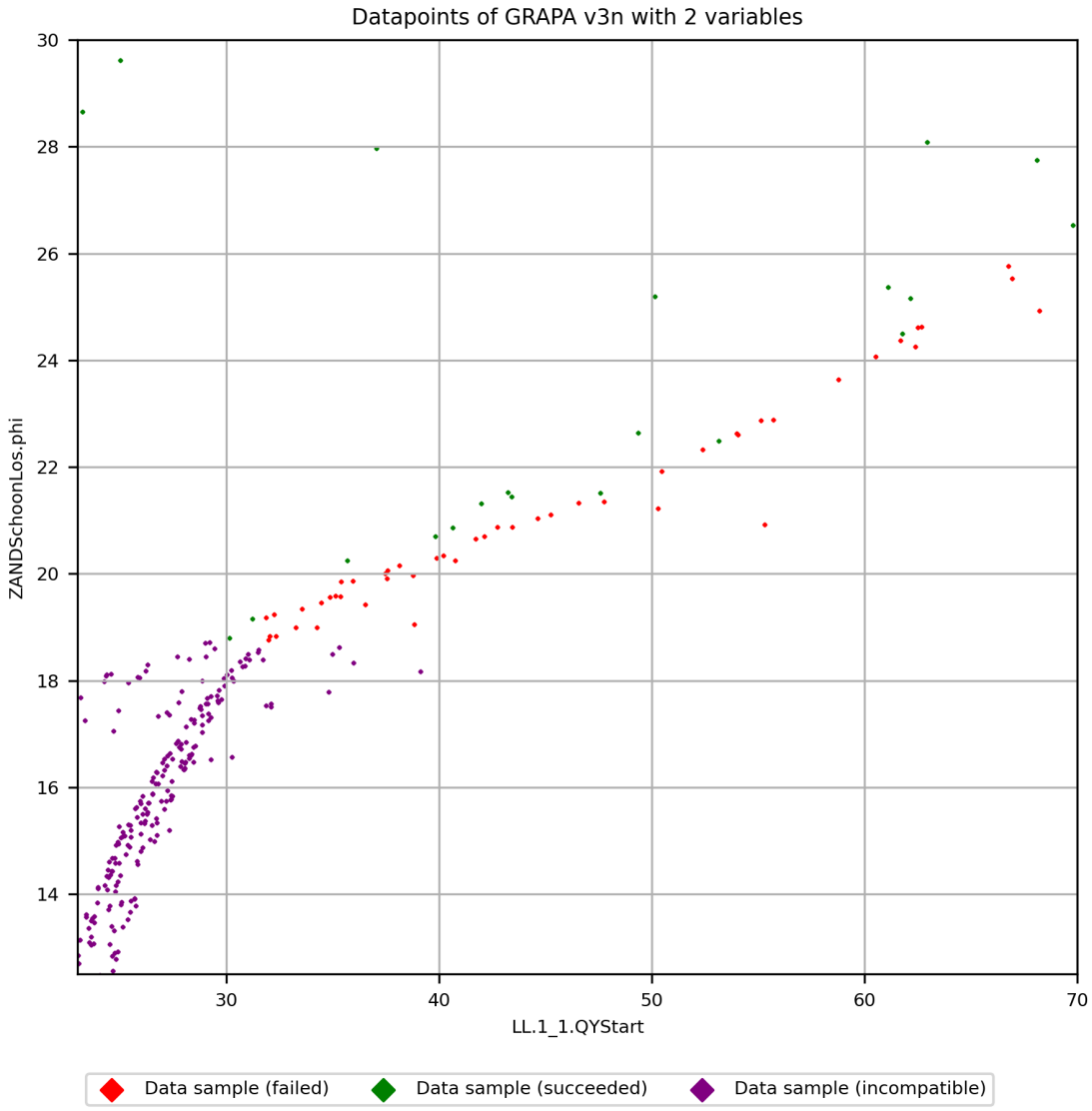


Figure C.3: Datapoints of the two-variables model (GRAPA v3n), zoomed in on Figure C.2

C.2 Metamodel analysis

A MCS is performed on the metamodel to make visual analysis possible (Figure C.4). The datapoints are displayed in a U-space where S1 and S2 represent LL.1_1.QYStart and ZANDSchoonLos.phi, respectively. To make the limit state better visible, S1 and S2 have a normal distribution N(0,3) instead of the default N(0,1). Note that the incompatible area is divided into a failure and non-failure area by ERRAGA, the purple line indicates the original NaN area. This means that the metamodel has no incompatible domain and all MCS/IS samples are used to determine the (Imputed) probability of failure, see section 3.2.

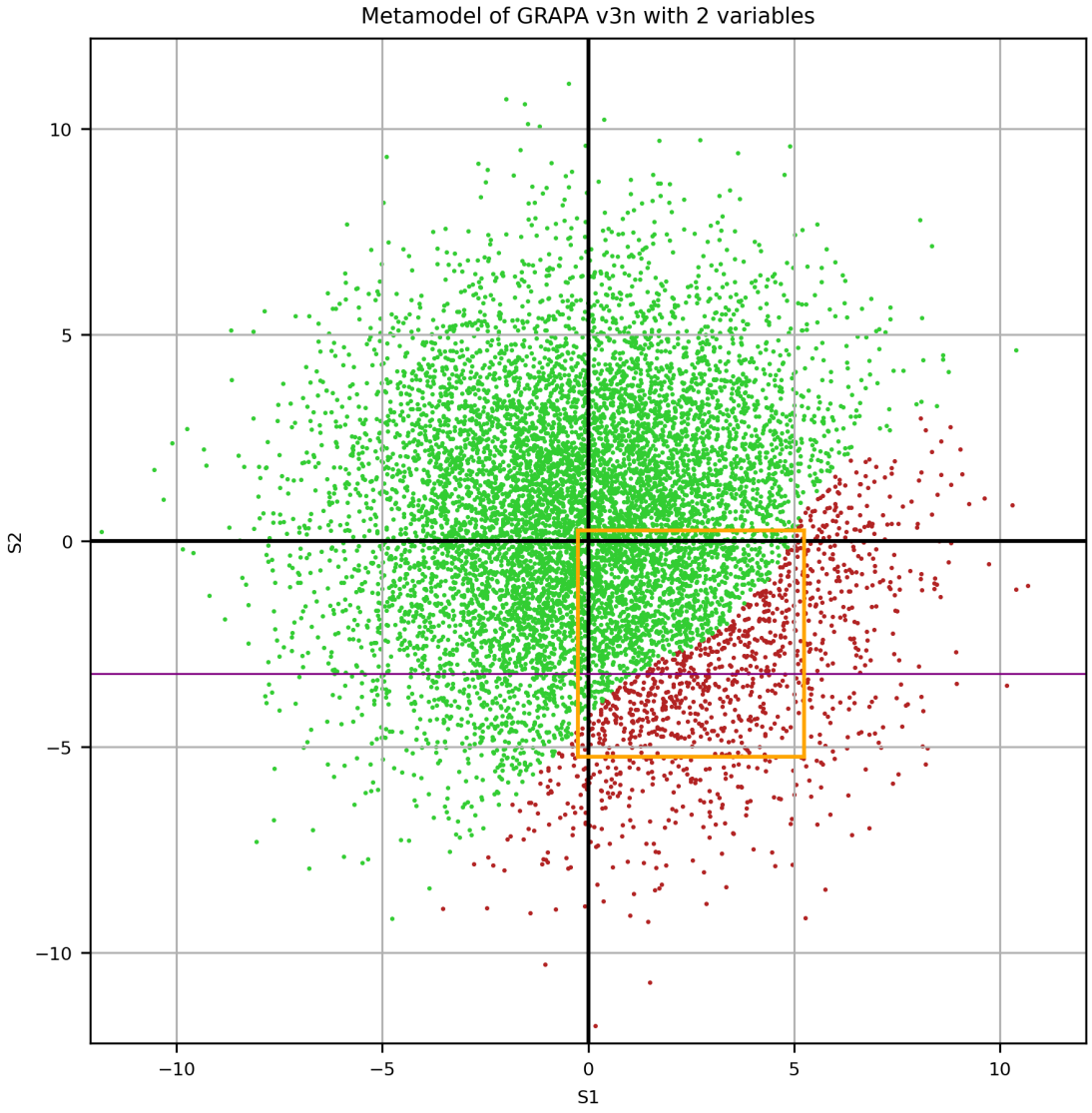


Figure C.4: Metamodel of the two-variables model (GRAPA v3n)

Figure C.4 shows a very clear non-linear limit state as expected based on the previous part. Figure C.5 zooms in on the area indicated by the orange square in Figure C.4. The LSF clearly shows some non-linearity. The limit state of the metamodel does not completely corresponds with the obtained samples in Figure C.3 at the location of the orange circle. This indicates that ERRAGA has trouble to fit a good prediction, but figuring this is near the incompatible

domain, this might be the reason. It is very likely that the model is not converged because of the incompatible data as discussed in the beginning of this chapter.

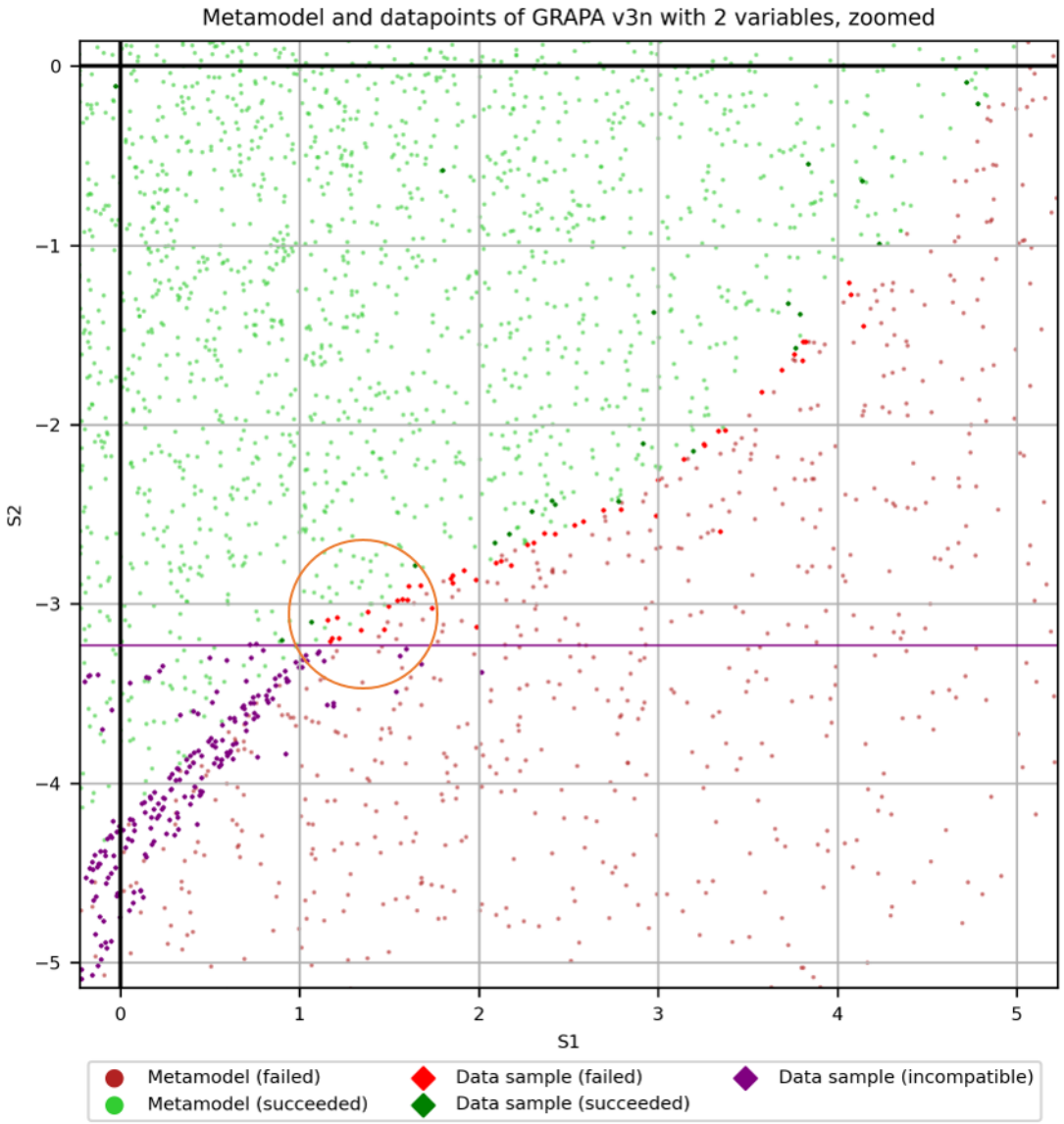


Figure C.5: Metamodel and datapoints, zoomed

C.3 Artificial model

For further analysis on the complexities of the Limit state functions, an artificial model is used as shown in Figure C.6. The big advantage thereof is the largely reduced computation time. The artificial model is based on the model discussed in the previous parts, however the values on the axis and the limit state do not correspond. The variables are:

- S1: gumbel distribution (15, 5)
- S2: normal distribution (20, 4)

The limit state function for the artificial model contains 3 complexities. Incompatibel data and non-linearity are proven in the previous part, noise is also added based on subsection 3.3.2:

1. Incompatible data (NaN area): $S2 < 7 : \quad Z = NaN$
2. Non-linearity: $sin\left(10 + \frac{S1}{5}\right)$
3. Noise (random): $random.normal(0.0, 0.1)$

The limit state function is defined as:

$$Z = 2\ S2 - 0.4\ S1 - 6 - sin\left(10 + \frac{S1}{5}\right) + random.normal(0.0, 0.1) \tag{C.1}$$

$$S2 < 7 : \quad Z = NaN \tag{C.2}$$

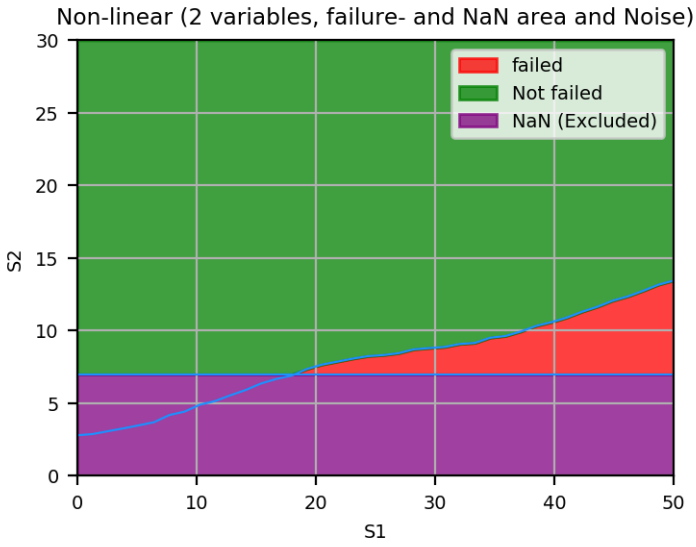


Figure C.6: Visualisation of the artificial model

C.4 Artificial model verification (015-1)

The artificial model should show the same behaviour as the original model in order to perform useful tests. Even though it is impossible to perfectly copy the original model, this artificial gives better insight on the influence of different complexities in limit state functions. The same ERRAGA settings are used as for the original model (Table C.1 and Table C.2).

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.0	default
Classification Model	None	default
Beta Prior	0.0	default
Min Realizations	20	
Max Realizations	10000	

Table C.2: ERRAGA settings for artificial model

The results of the run are shown in Figure C.7. The run is interrupted after 3282 realisations (after 3.5 hrs) as no convergence has occurred or will occur looking at the gradient of the convergence criterion. The majority of the samples are taken in the incompatible area. Both these characteristics correspond with the original model, this gives enough trust to perform further tests with this artificial model.

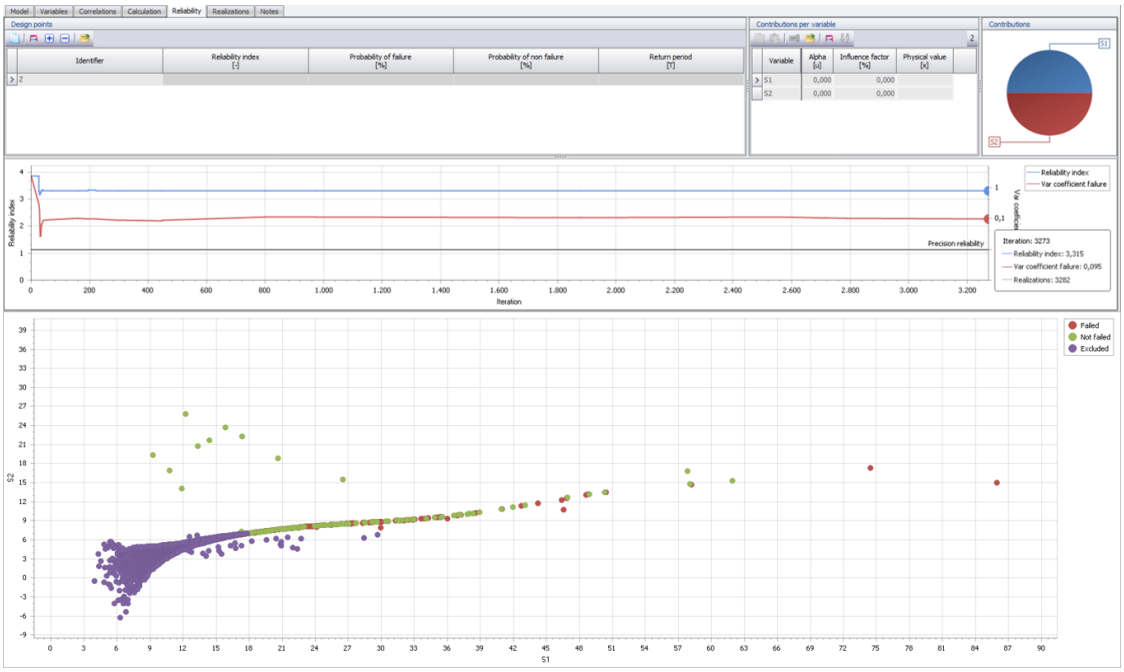


Figure C.7: The first run of the artificial model

Using this artificial model, further research will be performed in the following sections. Section C.5 to section C.7 analyses the influence of each complexity by turning them off one by one. Further, some ERRAGA settings, e.g. Classification model and Noise Variance Reduction Ration (NVRR) will be tested.

C.5 Artificial model without incompatible data (015-2)

ERRAGA settings are the same as Table C.2

The limit state function is defined as:

$$Z = 2\ S2 - 0.4\ S1 - 6 - sin\left(10 + \frac{S1}{5}\right) + random.normal(0.0, 0.1) \tag{C.3}$$

$$\cancel{S2 < 7 : Z = NaN} \tag{C.4}$$

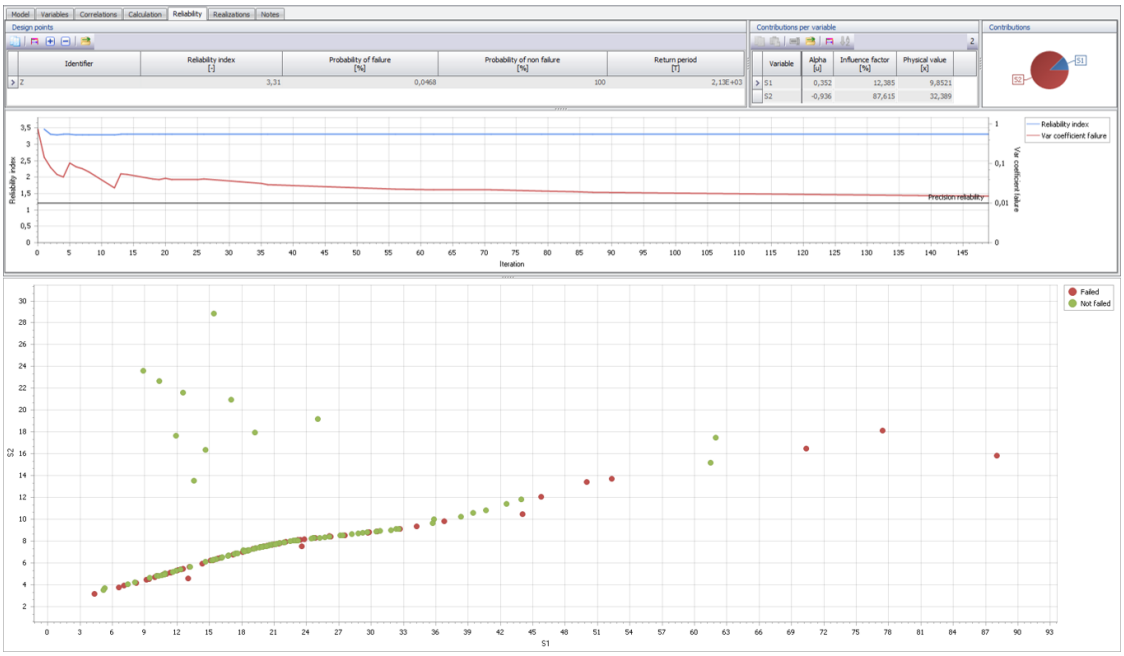


Figure C.8: Artificial model without incompatible data

Method	Beta	Ucrit/CoVar	Realisations
ERRAGA	3.31	0.05	158
IS (Exact)	3.31	0.016	1000000
FORM	3.30	0.01	42

Table C.3: Results of the artificial model without NaN

Findings

- Fast and constant convergence of ERRAGA and FORM
- FORM is able to handle the problem

C.6 Artificial model without non-linearity (015-3)

ERRAGA settings are the same as Table C.2

The limit state function is defined as:

$$Z = 2\ S2 - 0.4\ S1 - 6 - \cancel{sin\left(10 * \frac{S1}{5}\right)} + random.normal(0.0, 0.1) \tag{C.5}$$

$$S2 < 7: \qquad Z = NaN \tag{C.6}$$

Figure C.9: Artificial model without nonlinearity

Method	Beta	Ucrit/CoVar	Realisations
ERRAGA	3.36	0.05	223
IS (Exact)	3.35	0.015	1000000
FORM	CRASH or 3.41	0.01	CRASH or 21

Table C.4: Results of the artificial model without nonlinearity

Findings

- Fast but erratic convergence
- FORM is able to handle the problem unless incompatible data is encountered

C.7 Artificial model without noise (015-4)

ERRAGA settings are the same as Table C.2

The limit state function is defined as:

$$Z = 2\ S2 - 0.4\ S1 - 6 - sin\left(10 + \frac{S1}{5}\right) + \cancel{random.normal(0.0, 0.1)} \tag{C.7}$$

$$S2 < 7: \qquad Z = NaN \tag{C.8}$$

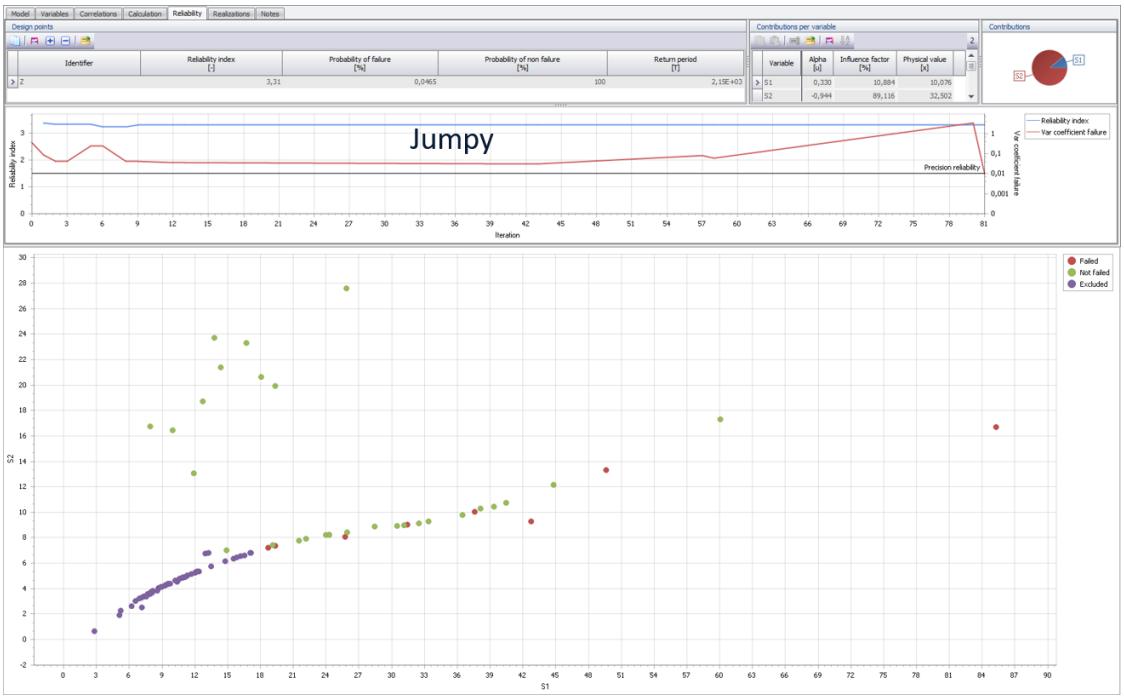


Figure C.10: Artificial model without noise

Method	Beta	Ucrit/CoVar	Realisations
ERRAGA	3.31	0.05	90
IS (Exact)	3.31	0.016	1000000
FORM	3.30 or CRASH	0.01	18

Table C.5: Results of the artificial model without noise

Findings

- Fast but erratic and sudden convergence
- PTK has filter out the erratic convergence
- FORM is able to handle the problem unless incompatible data is encountered

C.8 Artificial model using SVM (015-5 and 015-6)

This section shows the results of two runs of the artificial model using the classification model SVM with the goal to see the effects of SVM and to investigate the (erratic) convergence behaviour.

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.0	default
Classification Model	SVM	
Beta Prior	0.0	default
Min Realizations	20	
Max Realizations	10000	

Table C.6: ERRAGA settings for artificial model using the SVM classification model

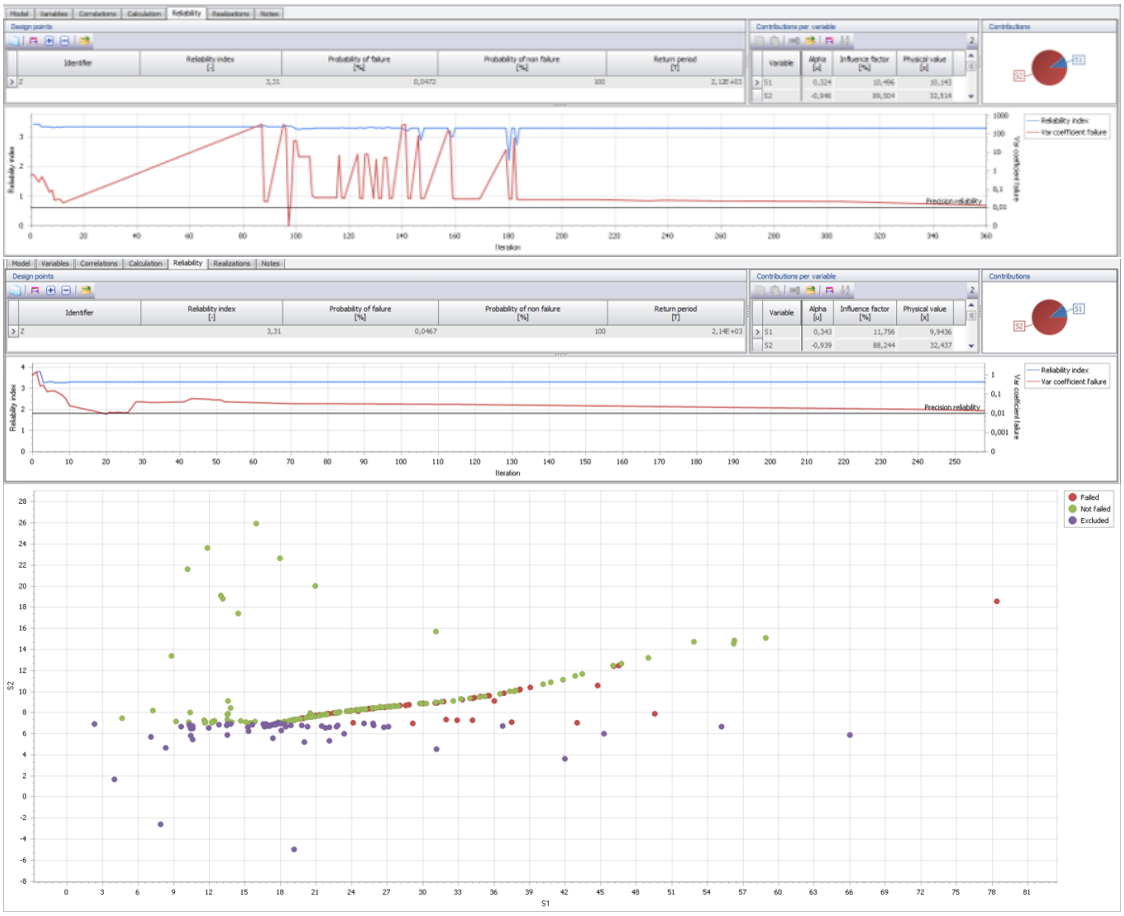


Figure C.11: Artificial model with SVM classification model, whereby both runs have the exact same settings and input values. Only one graph with samples is shown, in essence this graph was similar for both runs.

Method	Beta	Ucrit/CoVar	Realisations
ERRAGA run 1	3.31	0.05	369
ERRAGA run 2	3.31	0.05	267
IS (Exact)	3.31	0.016	1000000

Table C.7: Results of the artificial model using the SVM classification model

Findings

- With the use of the SVM classification model the problem converges
- Both runs have a very accurate answer and a reasonable amount of realisations
- The erratic behaviour of the convergence line seems to be arbitrary

C.9 Artificial model using SVM and NVRR=0.10 (015-7)

Section C.9, C.10 and C.11 test the ERRAGA setting Noise Variance Reduction Ratio (NVRR) (see section 3.2), respectively a NVRR of 0.10, 0.75 and 0.99 is used.

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.10	
Classification Model	SVM	
Beta Prior	0.0	default
Min Realizations	20	
Max Realizations	10000	

Table C.8: ERRAGA settings for artificial model using SVM and NVRR=0.10

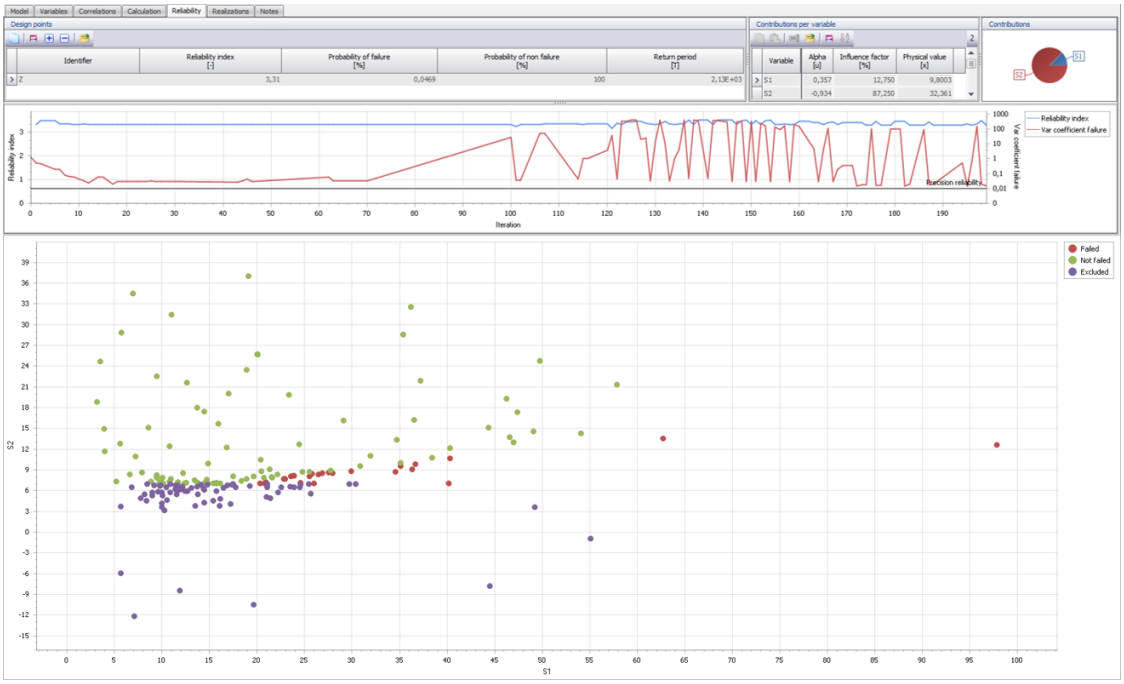


Figure C.12: Artificial model with SVM and NVRR=0.10

Method	**Beta**	**Ucrit/CoVar**	**Realisations**
ERRAGA	3.31	0.05	208
IS (Exact)	3.31	0.016	1000000

Table C.9: Results of the artificial model using the SVM classification model

Findings

- The number of realisations has decreased when using NVRR=0.10
- Very accurate result looking at beta

C.10 Artificial model using SVM and NVRR=0.75 (015-8)

Section C.9, C.10 and C.11 test the ERRAGA setting Noise Variance Reduction Ratio (NVRR) (see section 3.2), respectively a NVRR of 0.10, 0.75 and 0.99 is used.

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.75	
Classification Model	SVM	
Beta Prior	0.0	default
Min Realizations	20	
Max Realizations	10000	

Table C.10: ERRAGA settings for artificial model using SVM and NVRR=0.75

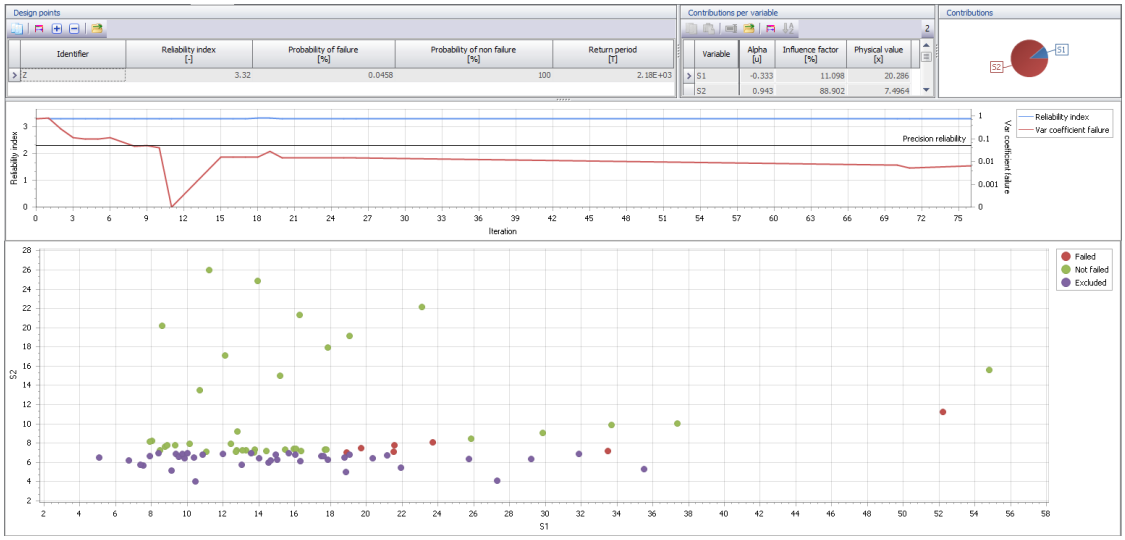


Figure C.13: Artificial model with SVM and NVRR=0.75

Method	**Beta**	**Ucrit/CoVar**	**Realisations**
ERRAGA	3.32	0.05	85
IS (Exact)	3.31	0.016	1000000

Table C.11: Results of the artificial model using SVM and NVRR=0.75

Findings

- The number of realisations has decreased again when using NVRR=0.75
- Very accurate result looking at beta
- Remarkable convergence behaviour, the run seems to be converged after 10 realisations

C.11 Artificial model using SVM and NVRR=0.99 (015-9)

Section C.9, C.10 and C.11 test the ERRAGA setting Noise Variance Reduction Ratio (NVRR) (see section 3.2), respectively a NVRR of 0.10, 0.75 and 0.99 is used.

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.99	
Classification Model	SVM	
Beta Prior	0.0	default
Min Realizations	20	
Max Realizations	10000	

Table C.12: ERRAGA settings for artificial model using SVM and NVRR=0.99

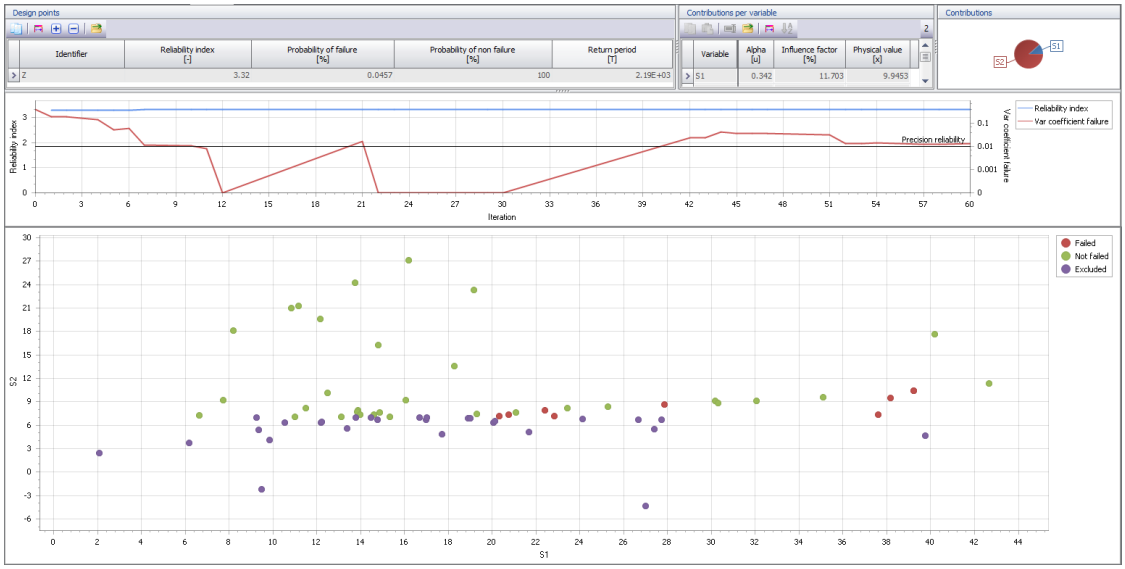


Figure C.14: Artificial model with SVM and NVRR=0.99

Method	**Beta**	**Ucrit/CoVar**	**Realisations**
ERRAGA	3.32	0.05	69
IS (Exact)	3.31	0.016	1000000

Table C.13: Results of the artificial model using SVM and NVRR=0.99

Findings

- The number of realisations has decreased drastically when using NVRR=0.99
- The accuracy of beta is still very accurate
- Remarkable convergence behaviour, the run seems to be converged after 22 realisations

The remarkable convergence behaviour is investigated with the dumpfile, Figure C.15 shows the actual course of the convergence lines and the seperation between the cycles

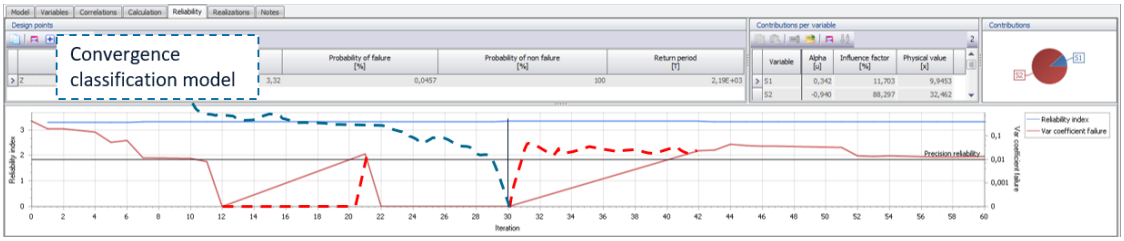


Figure C.15: Artificial model with SVM and NVRR=0.99, showing the actual course of the convergence line of the prediction model (red line) and the convergence line of the classification model (blue line), also the separation between the two cycles is indicated

The following remarks are made, looking at the visualisation of the PTK:

- The convergence line of the prediction model (red line) ignores a certain amount of points in some sections, recognizable by a straight (diagonal) line. These 'blank spots' can be confusing when analysing the run.
- Besides the convergence criterion of the prediction model (red line), the classification model has two convergence criteria, one is (partly) shown in Figure C.15 (blue line). Only one of the two classification criteria has to converge (see section 3.2).
- The fourth criterion is the convergence of the MCS/IS run on the metamodel, this criterion is met if the Coefficient of Variation of the Probability of Failure (PfCoV) is below 0.05 (see section 3.2).
- If the maximum amount of iterations for one cycle is reached, or the prediction- and classification model are converged but the PfCoV is not, a new cycle starts. The start of a new cycle is indicated by the black vertical line in Figure C.15.

The visualization of the PTK does not give enough information to fully understand the convergence behaviour of the ERRAGA run. A more elaborate graph is made for this thesis which will be used further on in this research.

Appendix D

Case 1: LSF Geotechnical, Numerical model

D.1 GRAPA v3o

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.0	default
Classification Model	SVM	
Beta Prior	0.0	default
Min Realizations	50	
Max Realizations	100	

Table D.1: ERRAGA settings for GRAPA v3o

Figure D.1: Convergence graph and sample chart of GRAPA v3o

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.42	0.05	400 (max)	NO

Table D.2: Results of GRAPA v3o

Findings

- The SVM classification model is activated.
- The erratic convergence behaviour is reduced.
- Looking at the convergence criterion, the prediction model is closer to convergence compared to GRAPA v3n.
- It looks like the convergence line is not descending towards the convergence value.

D.2 GRAPA v3p

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.99	
Classification Model	SVM	
Beta Prior	0.0	default
Min Realizations	50	
Max Realizations	100	

Table D.3: ERRAGA settings for GRAPA v3p

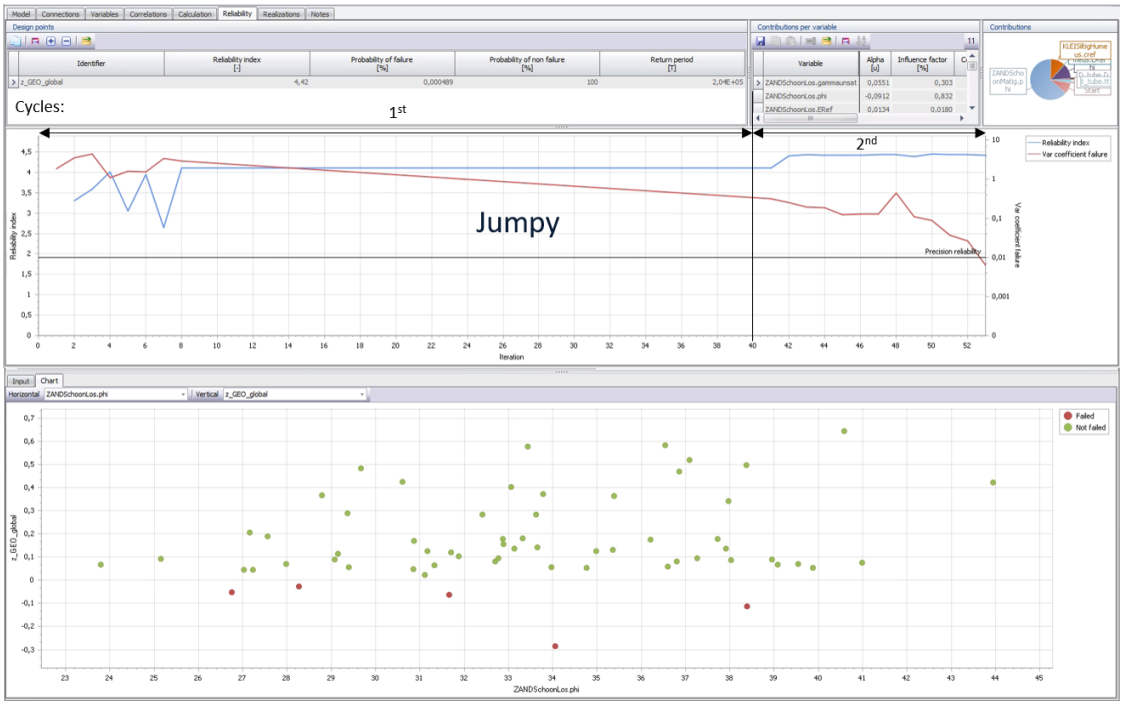


Figure D.2: Convergence graph and sample chart of GRAPA v3p (Indication of the cycles is added afterwards)

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.42	0.05	62	YES

Table D.4: Results of GRAPA v3p

Findings

- Successful convergence within small amount of realisations.
- No incompatible data encountered, this makes the problem less complex.

D.3 GRAPA v3s

ERRAGA settings:		
Convergence Criteria	PfStop	default
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.99	
Classification Model	SVM	
Beta Prior	2.0	
Min Realizations	50	
Max Realizations	100	

Table D.5: ERRAGA settings for GRAPA v3s

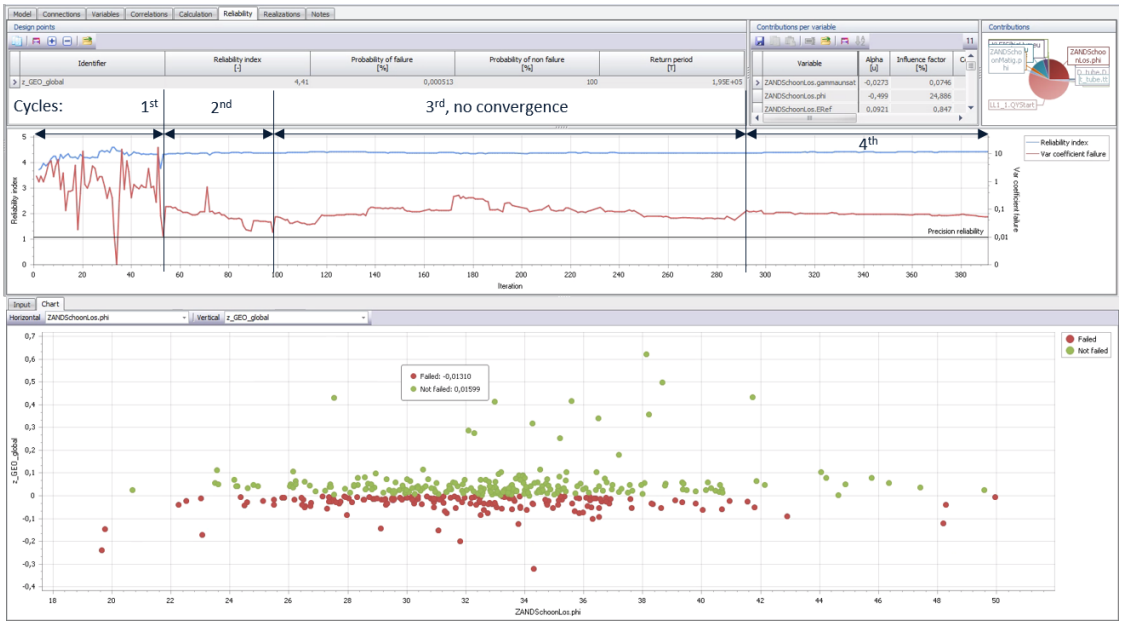


Figure D.3: Convergence graph and sample chart of GRAPA v3s (Indication of the cycles is added afterwards)

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.41	0.05	400 (max)	NO

Table D.6: Results of GRAPA v3s

Findings

- Cycle 1 and 2 converged, cycle 3 and 4 did not converge.
- Compared to the previous run only Beta Prior is increased, this means IS is used instead of MCS during the first cycle.
- Incompatible data is first encountered at realisation 228
- Beta is very steady during the last 75% of the run

D.4 GRAPA v3t

ERRAGA settings:	
Convergence Criteria	BetaAbsStop
Noise Term	True
Noise Variance Reduction Ratio (NVRR)	0.99
Classification Model	SVM
Beta Prior	5.0
Min Realizations	30
Max Realizations	100

Table D.7: ERRAGA settings for GRAPA v3t

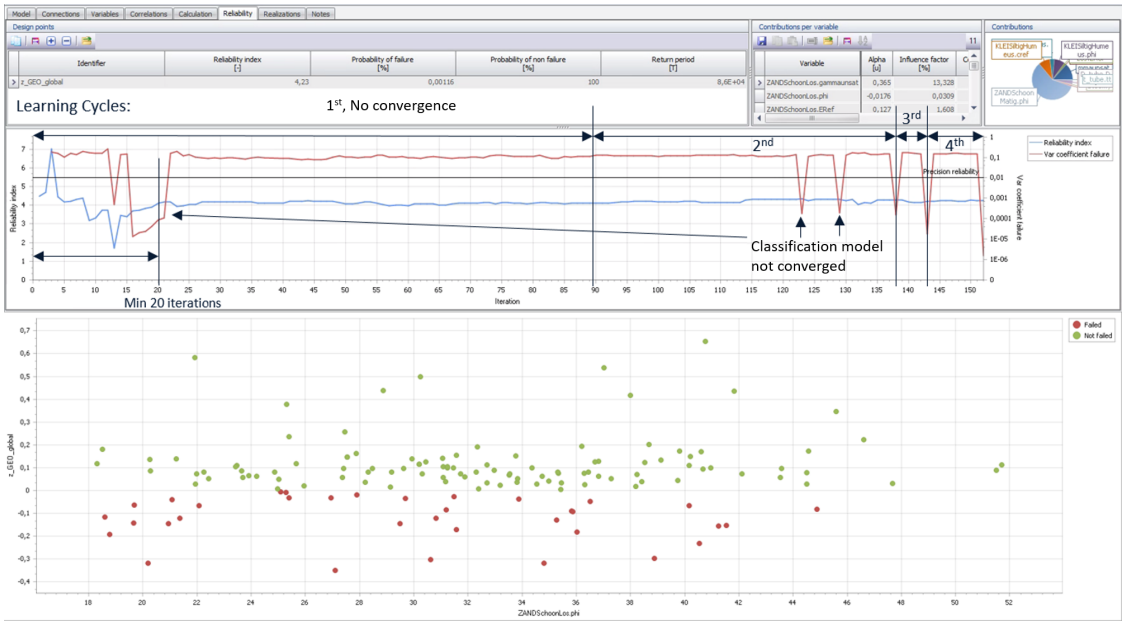


Figure D.4: Convergence graph and sample chart of GRAPA v3t (Indication of the cycles is added afterwards)

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.23	0.05	160	YES

Table D.8: Results of GRAPA v3t

Findings

- The prediction model shows a very sudden convergence.
- Beta is steady during the last 75% of the run

D.5 GRAPA v3u

ERRAGA settings:	
Convergence Criteria	BetaAbsStop
Noise Term	True
Noise Variance Reduction Ratio (NVRR)	0.25
Classification Model	SVM
Beta Prior	5.0
Min Realizations	30
Max Realizations	1000/220

Table D.9: ERRAGA settings for GRAPA v3u

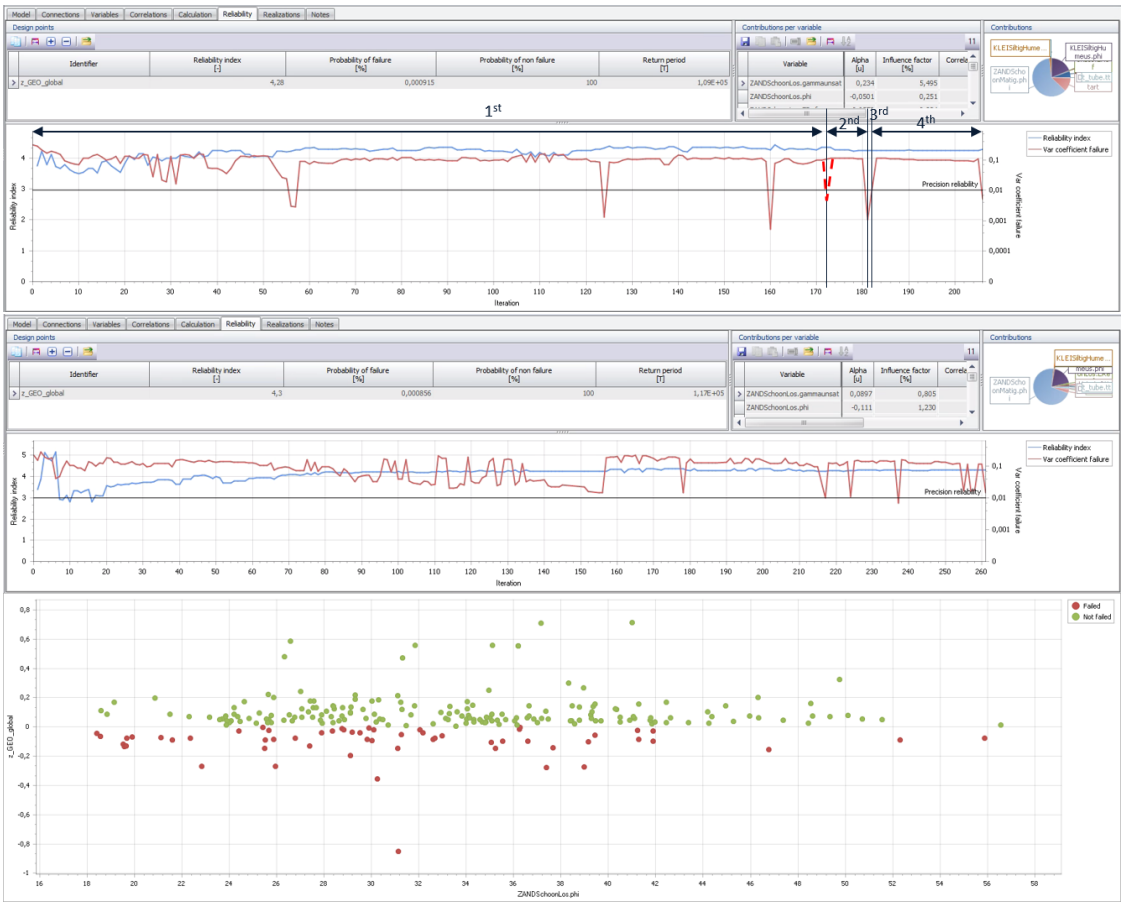


Figure D.5: Convergence graph and sample chart of GRAPA v3u

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.29	0.05	213	YES
ERRAGA	4.30	0.05	267	YES

Table D.10: Results of GRAPA v3u

D.6 GRAPA v3w

This section shows extra, more elaborated visualisations of the run.

ERRAGA settings:		
Convergence Criteria	BetaAbsStop	
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.0	default
Classification Model	SVM	
Beta Prior	5.0	
Min Realizations	50	
Max Realizations	250	

Table D.11: ERRAGA settings for GRAPA v3w

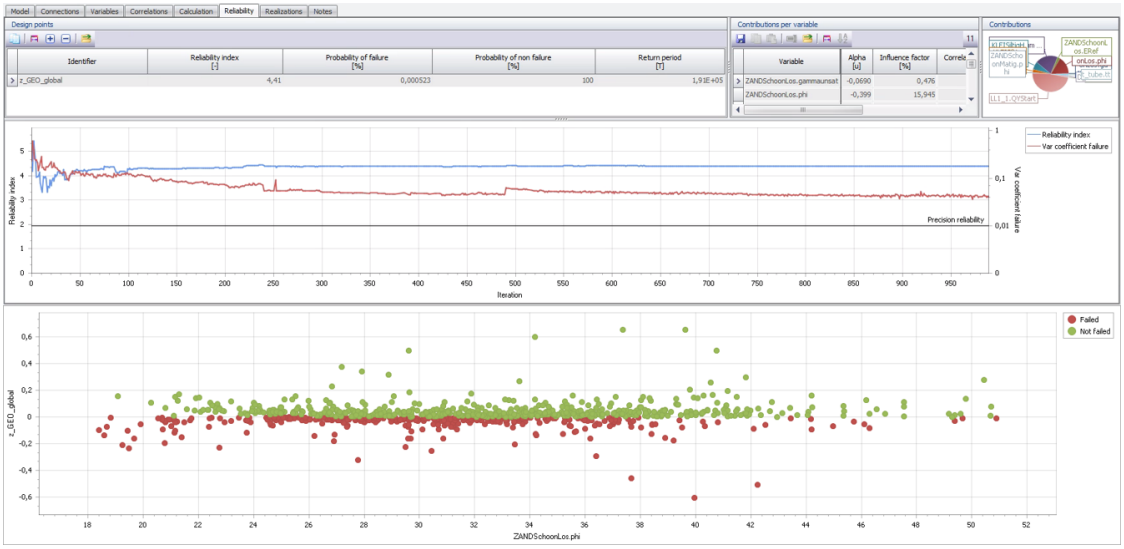


Figure D.6: Convergence graph and sample chart of GRAPA v3w

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.41	0.05	1000 (max)	NO

Table D.12: Results of GRAPA v3w

Findings

- No convergence, prediction model convergence line shows a very small decline.
- Beta is very steady during the last 75% of the run.
- The uncertainty of beta is barely reduced during the last 75% of the realisations (Figure D.7).
- The erratic behaviour during the last 50% of the realisations means that ERRAGA is switching between design point and thus metamodels.
- The degree of uncertainty and white noise are in the same order of magnitude.

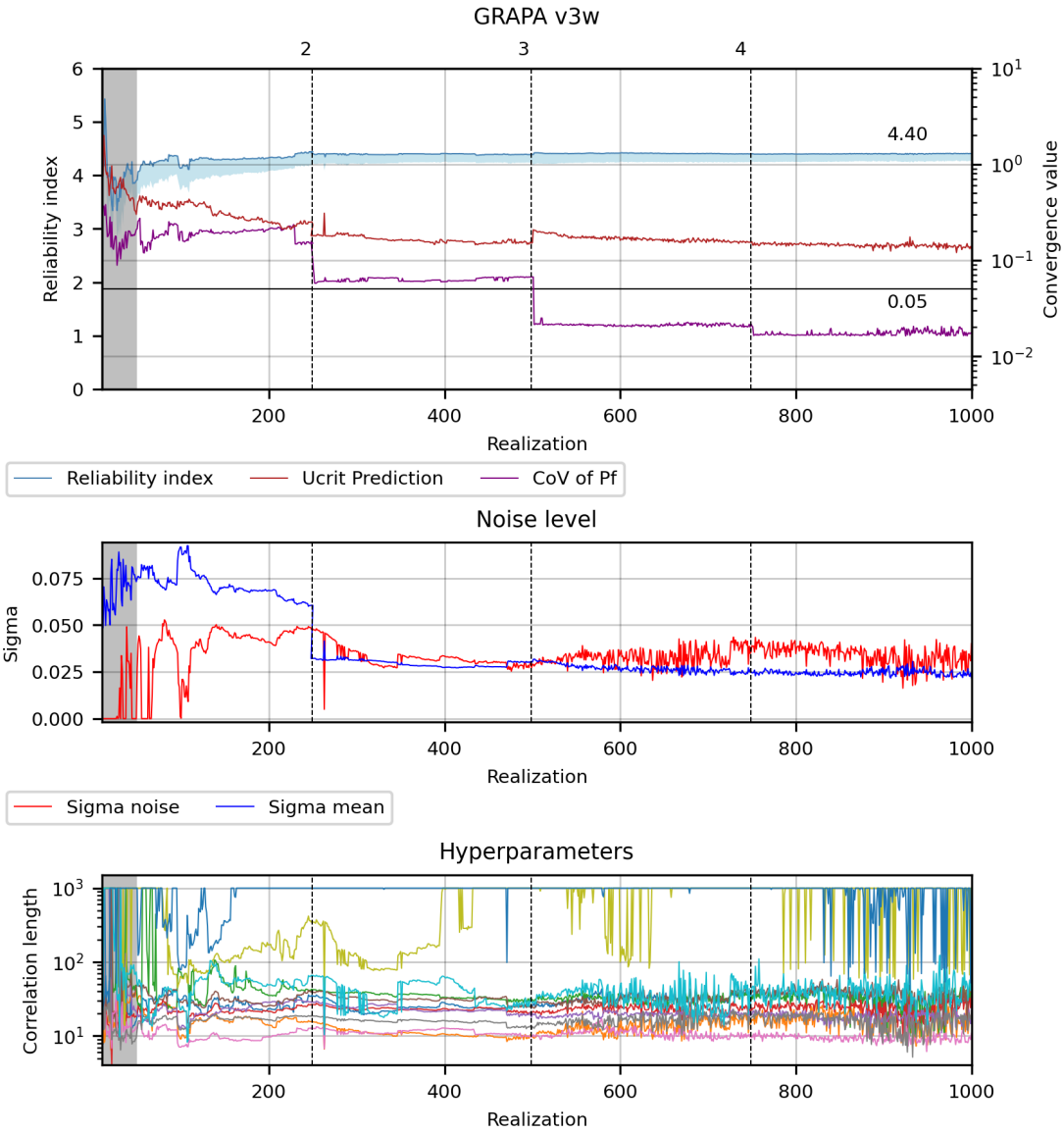


Figure D.7: Convergence-, Noise- and Hyperparameter graphs of GRAPA v3w

D.7 GRAPA v3y

This section shows extra, more elaborated visualisations of the run.

ERRAGA settings:	
Convergence Criteria	BetaAbsStop
Noise Term	True
Noise Variance Reduction Ratio (NVRR)	0.10
Classification Model	SVM
Beta Prior	5.0
Min Realizations	50
Max Realizations	250

Table D.13: ERRAGA settings for GRAPA v3y

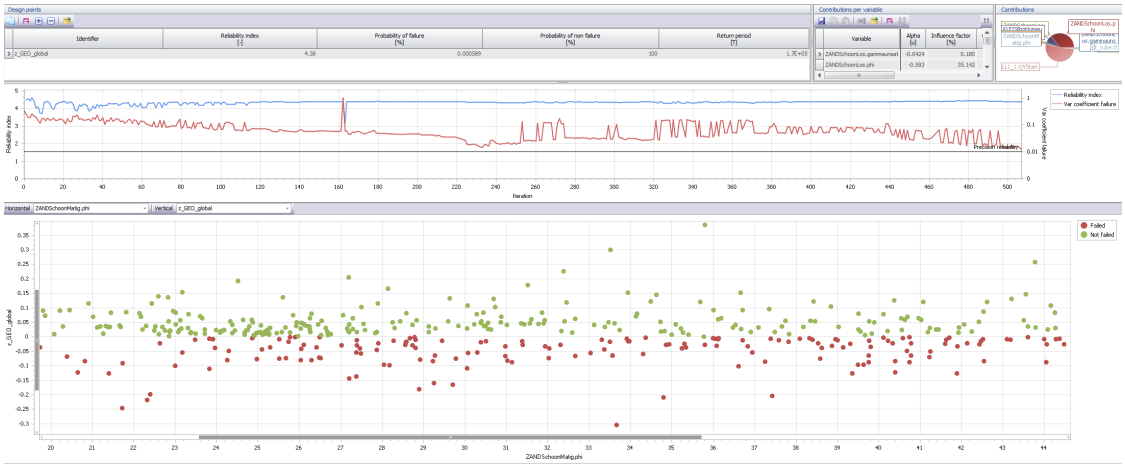


Figure D.8: Convergence graph and sample chart of GRAPA v3y

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.38	0.05	514	NO

Table D.14: Results of GRAPA v3y

Findings

- Erratic convergence behaviour.
- Cycle 3 and 4 converged very fast compared to cycle 1 and 2.
- The degree of uncertainty and white noise are in the same order of magnitude.

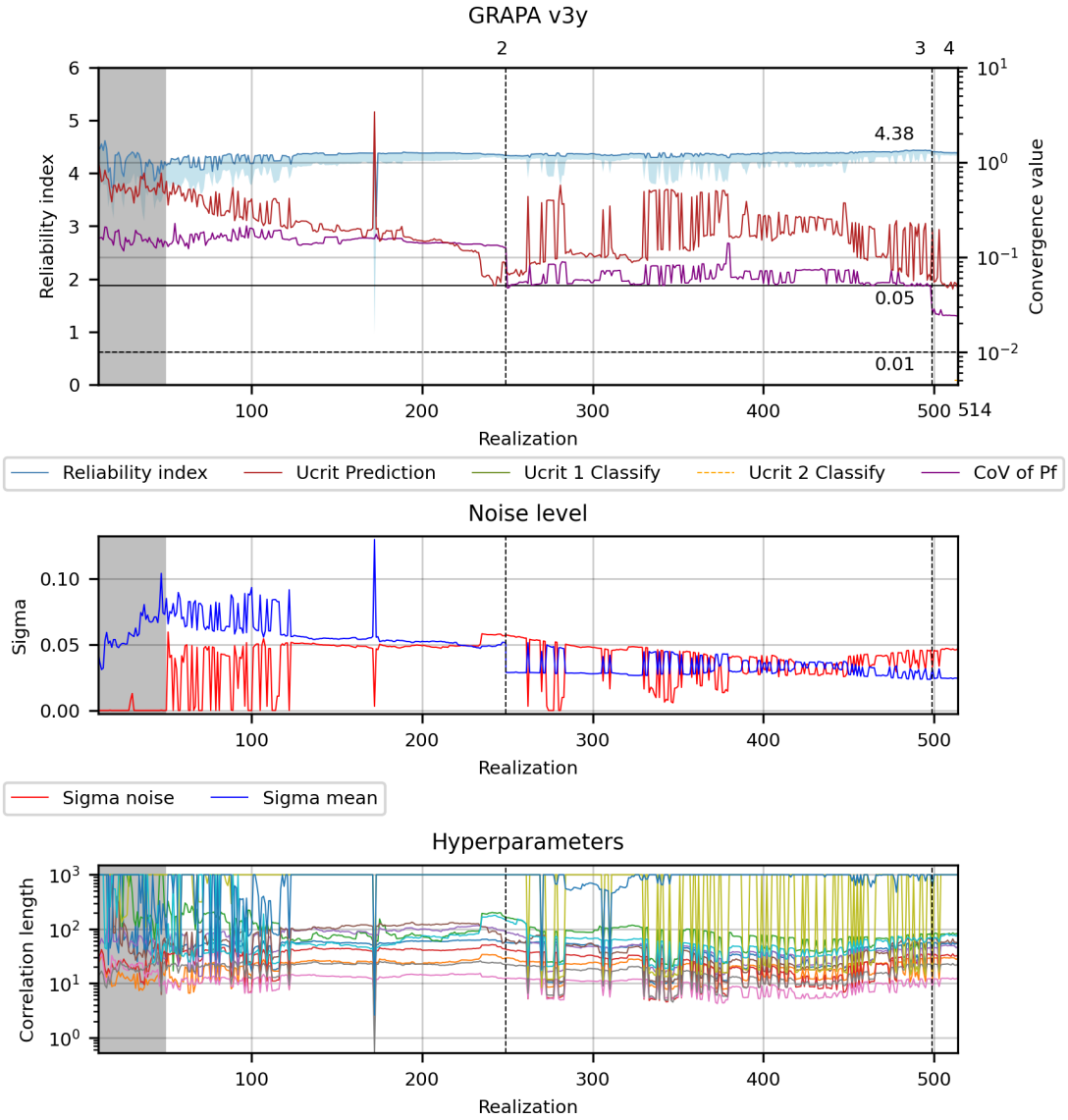


Figure D.9: Convergence-, Noise- and Hyperparameter graphs of GRAPA v3y

The convergence of the classification model is barely visible (lower right) because most of the convergence values are NaN, it converges immediately after convergence of the prediction model.

D.8 GRAPA v3 IS

Figure D.10: Convergence graph and sample chart of GRAPA v3, Importance Sampling

Method	Beta	Ucrit/CoVar	Realisations	Convergence
IS	4.59	0.1	2389	YES

Table D.15: Results of GRAPA v3 IS

Findings

- Gap between just above Z_{Geo}, seems to be a 'blank spot' in the results

Appendix E

Case 2: LSF Front wall, numerical model

E.1 GRAPA2 v2i (No model uncertainty factors)

ERRAGA settings:		
Initial realisations Realisations	10	default
Min realisations	30	
Max realisations	100	
Convergence Criterion	BetaAbsStop	
Convergence Requirement	0.05	default
Learning Function	UNIS	
Classification Model	GPC	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0.0	default
Size MCS pool	100000	default
Hyperparameter Optimisations (Opts)	10	default
Beta Prior	5.0	

Table E.1: ERRAGA settings for GRAPA2 v2i

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.17	0.05	93	YES

Table E.2: Results of GRAPA2 v2i

Findings

- Realistic beta, should be around and above 4 according to the design standards.
- The prediction model converges immediately.
- Two IS updates are needed for convergence.

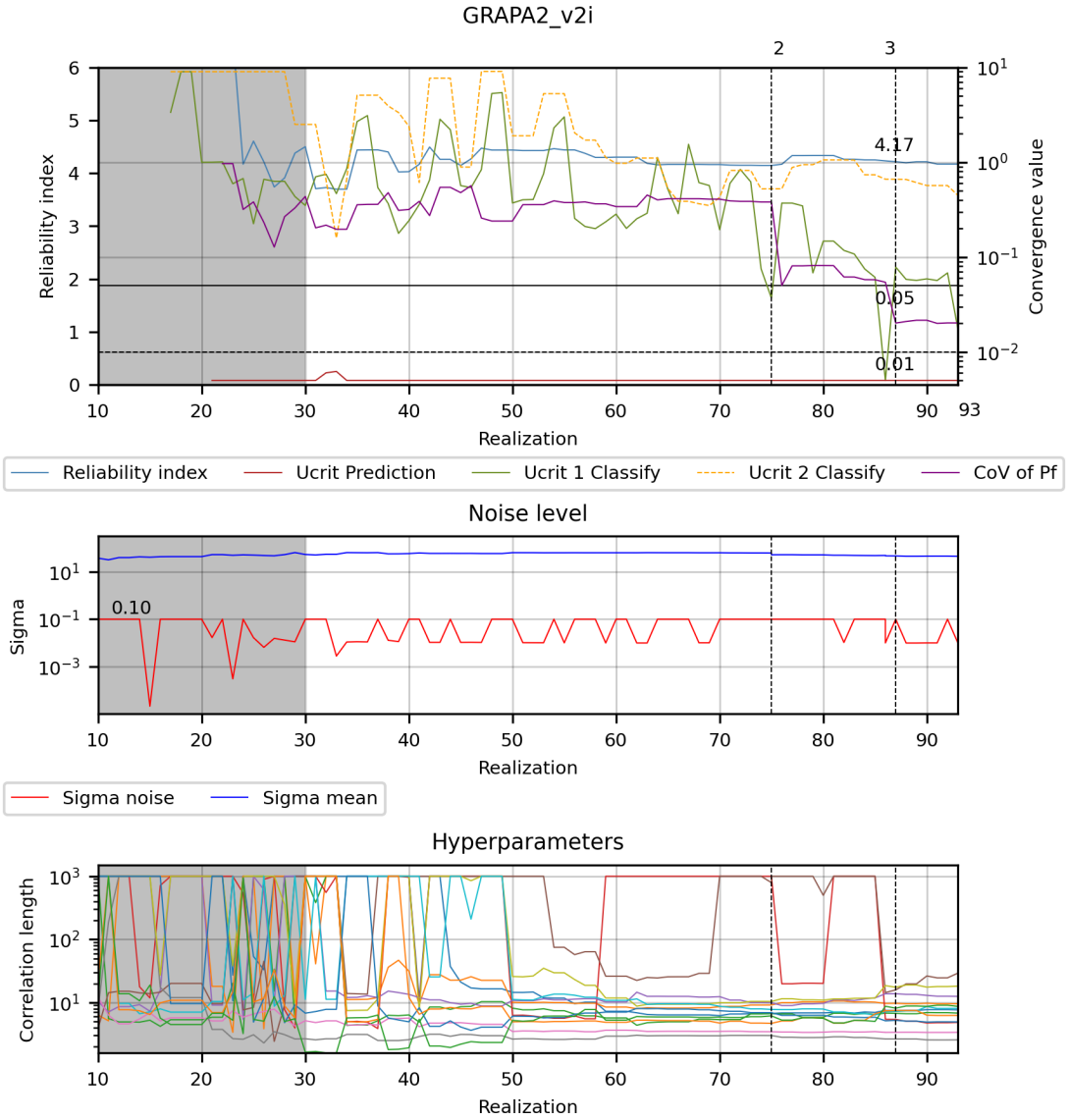


Figure E.1: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v2i

E.2 GRAPA2 v2f2 (Including model uncertainty factors)

ERRAGA settings:		
Initial realisations Realisations	10	default
Min realisations	30	
Max realisations	100	
Convergence Criterion	PfStop	default
Convergence Requirement	0.05	default
Learning Function	UNIS	
Classification Model	GPC	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0.0	default
Size MCS pool	1200000	
Hyperparameter Optimisations (Opts)	10	default
Beta Prior	10.0	

Table E.3: ERRAGA settings for GRAPA2 v2f2

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.05	0.05	199	YES

Table E.4: Results of GRAPA2 v2f2

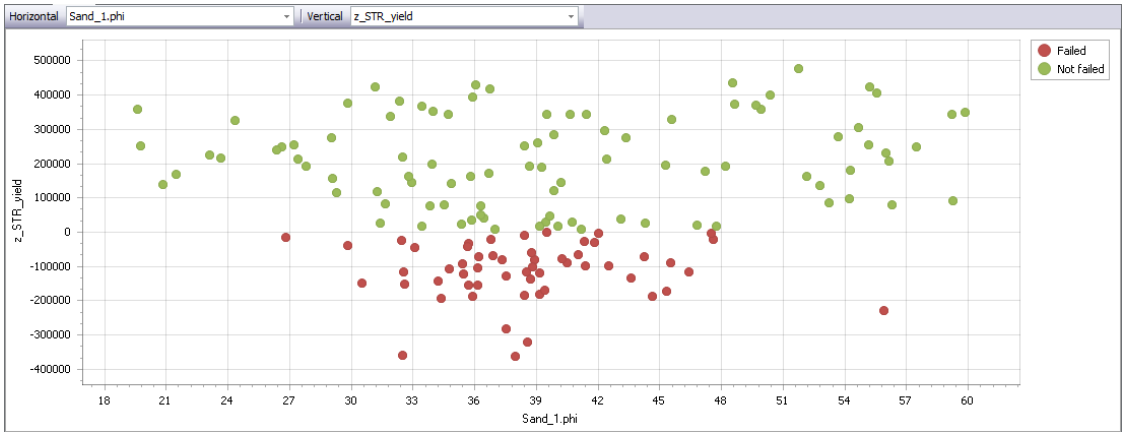


Figure E.2: Sample chart of GRAPA v2f2

Findings

- Beta slightly lower compared to previous run GRAPA v2i which might be the effect of the model uncertainty factors.
- No 'blank spot' in sample domain shown in Figure E.2 (which does occur in IS run of LSF Frontwall Case 1 section D.8)

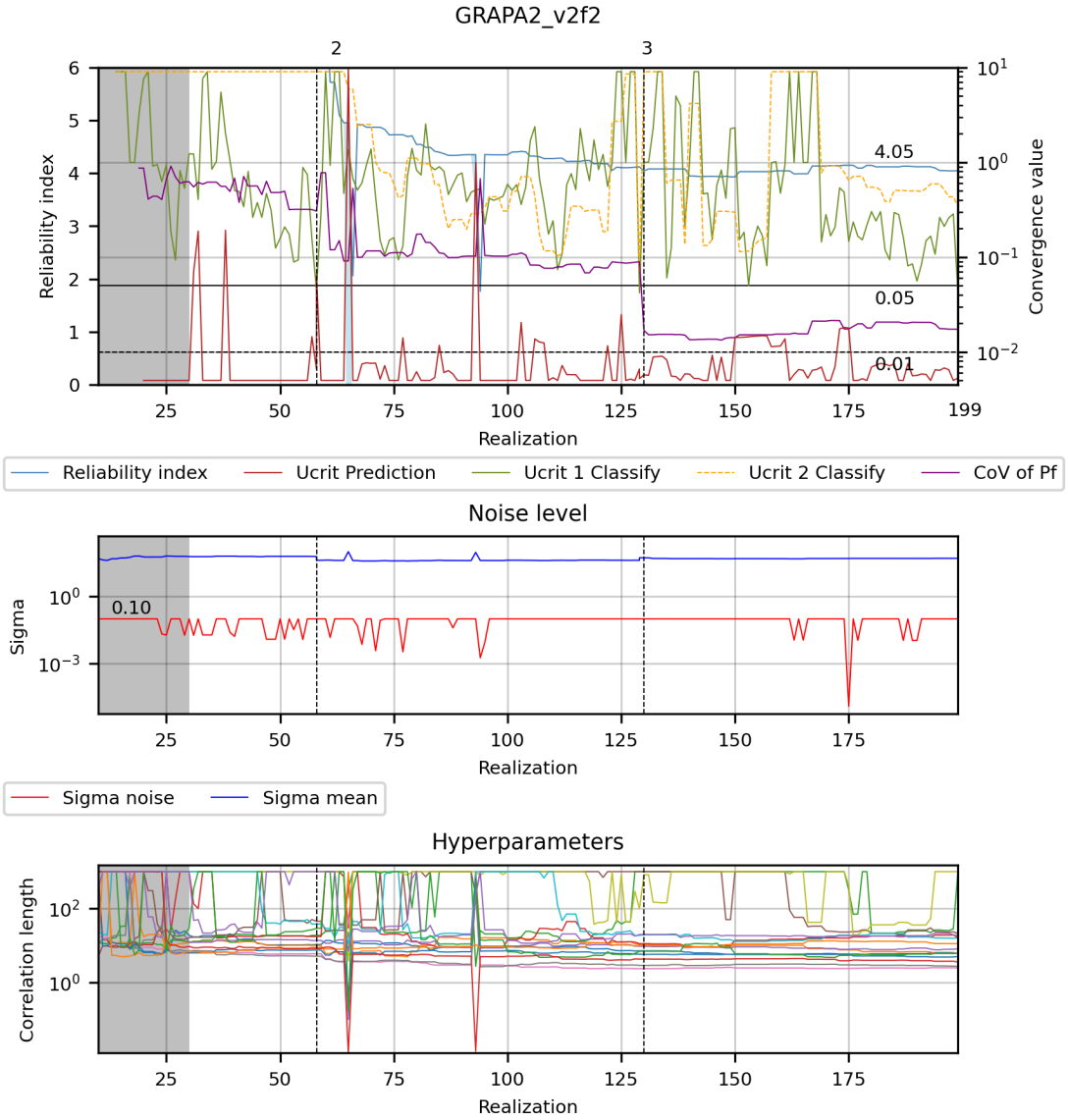


Figure E.3: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v2f2

E.3 GRAPA2 v2j (Including model uncertainty factors)

ERRAGA settings:		
Initial realisations Realisations	10	default
Min realisations	30	
Max realisations	100	
Convergence Criterion	BetaAbsStop	
Convergence Requirement	0.05	default
Learning Function	UNIS	
Classification Model	GPC	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0.0	default
Size MCS pool	100000	default
Hyperparameter Optimisations (Opts)	10	default
Beta Prior	5.0	

Table E.5: ERRAGA settings for GRAPA2 v2j

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.18	0.05	101	YES

Table E.6: Results of GRAPA2 v2j

Findings

- Beta is higher compared to the run with no model uncertainty (GRAPA v2i) which is remarkable because a lower beta is expected when adding uncertainty.

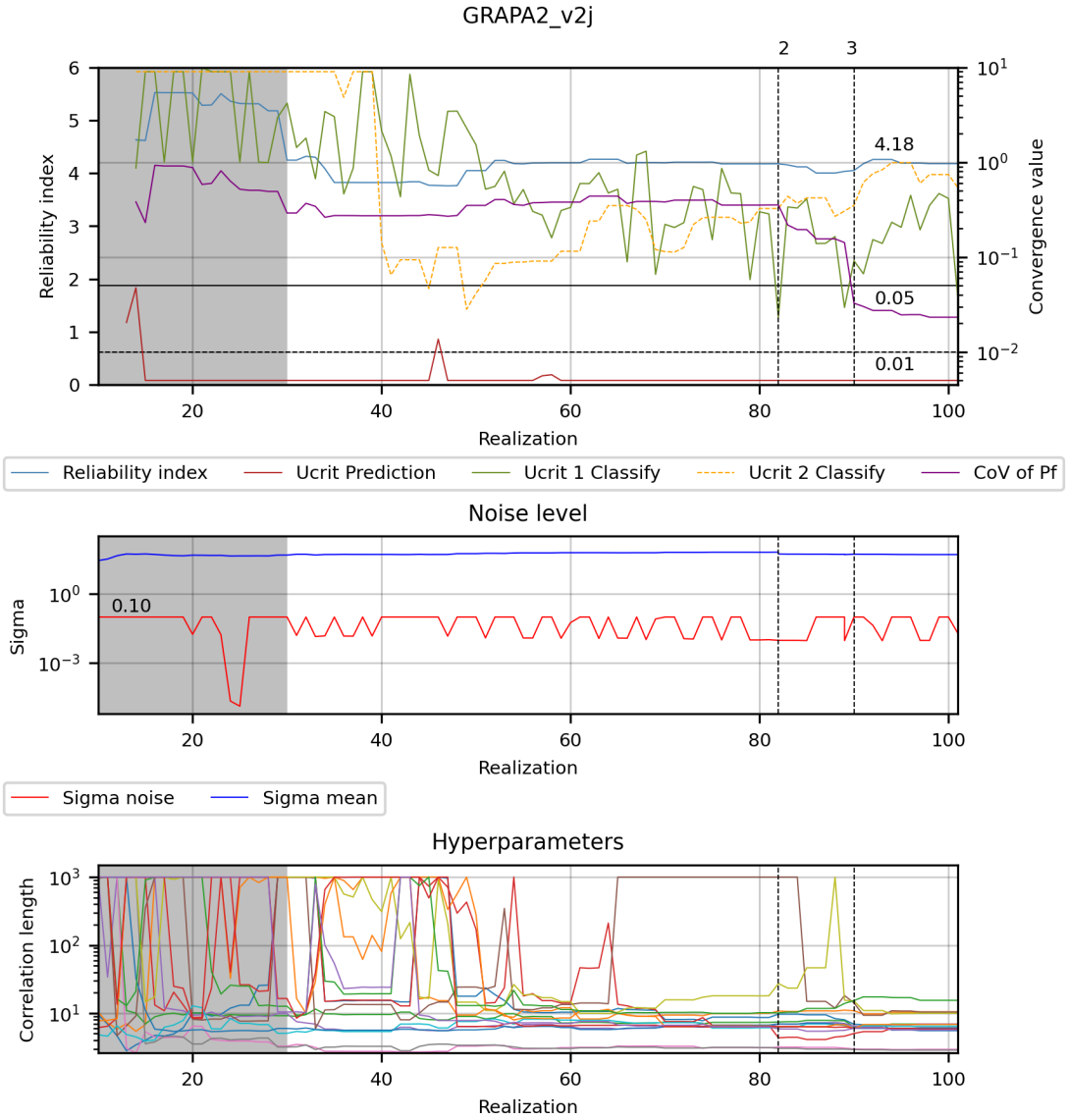


Figure E.4: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v2j

E.4 GRAPA2 v2j2 (Including model uncertainty factors)

ERRAGA settings:		
Initial realisations Realisations	10	default
Min realisations	30	
Max realisations	100	
Convergence Criterion	PfStop	default
Convergence Requirement	0.05	default
Learning Function	UNIS	
Classification Model	GPC	
Noise Term	True	
Noise Bounds	1E-10, 0.01	default
Noise Variance Reduction Ratio (NVRR)	0.0	default
Size MCS pool	100000	default
Hyperparameter Optimisations (Opts)	10	default
Beta Prior	8.0	

Table E.7: ERRAGA settings for GRAPA2 v2j2

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.20	0.05	96	YES

Table E.8: Results of GRAPA2 v2j2

Findings

- Remarkable are the two peaks shown in the upper and lower graph. ERRAGA fitted a completely different metamodel during these realisations which have a large uncertainty.

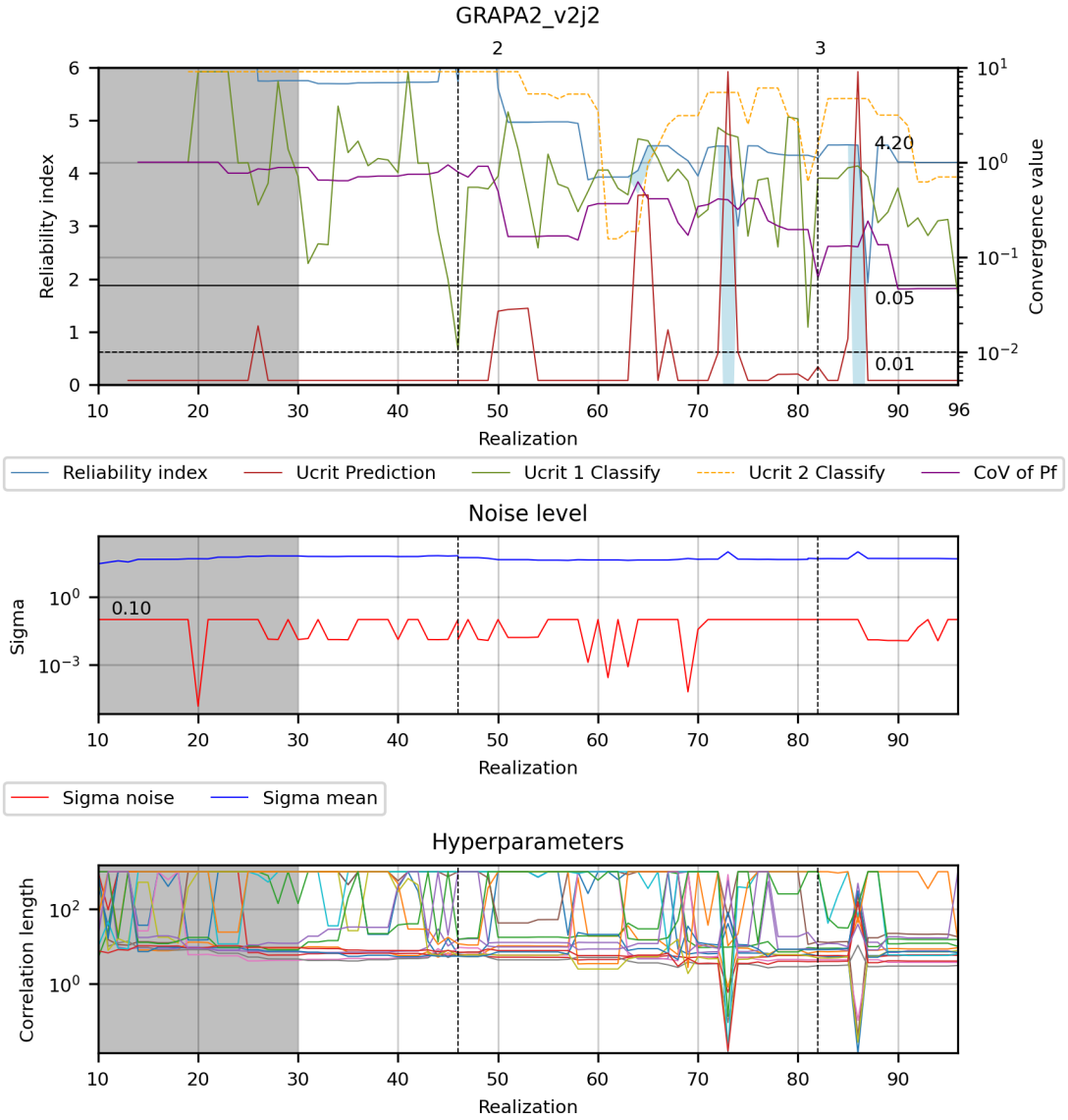


Figure E.5: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v2j2

Appendix F

Case 2: LSF Geotechnical, numerical model

F.1 GRAPA2 v3g2 (No model uncertainty factors)

ERRAGA settings:		
Convergence Criteria	BetaAbsStop	
Learn Function	UNISlearn	
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.1	
Classification Model	GCP	
Beta Prior	5.0	
Size MCS pool	100000	default
Min Realizations	30	
Max Realizations	250	
Hyperparameter Optimisations (Opts)	10	default

Table F.1: ERRAGA settings for GRAPA2 v3g2

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	4.18	0.05	624	YES

Table F.2: Results of GRAPA2 v3g2

Findings

- Erratic convergence behaviour.
- Incompatible data encountered at realisation 130.

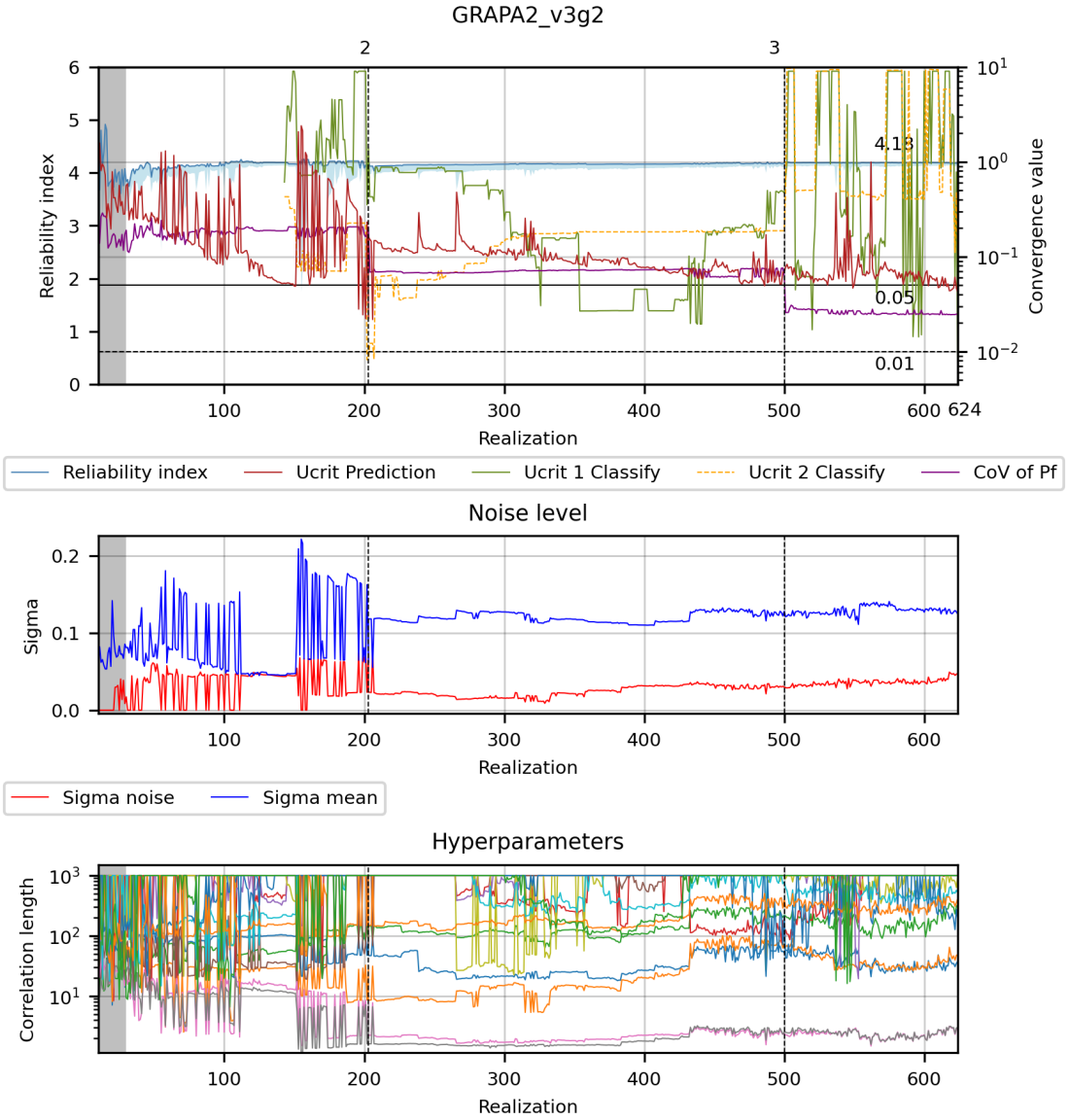


Figure F.1: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v3g2

F.2 GRAPA2 v3h (No model uncertainty factors)

ERRAGA settings:		
Convergence Criteria	BetaAbsStop	
Learn Function	UNISlearn	
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.1	
Classification Model	GCP	
Beta Prior	5.0	
Size MCS pool	100000	default
Min Realizations	50	
Max Realizations	250	
Hyperparameter Optimisations (Opts)	20	

Table F.3: ERRAGA settings for GRAPA2 v3h

Run	**Method**	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
GRAPA v3h2	ERRAGA	4.13	0.05	157	YES
GRAPA v3h3	ERRAGA	4.06	0.05	156	YES
GRAPA v3h4	ERRAGA	4.16	0.05	393	YES

Table F.4: Results of GRAPA2 v3h

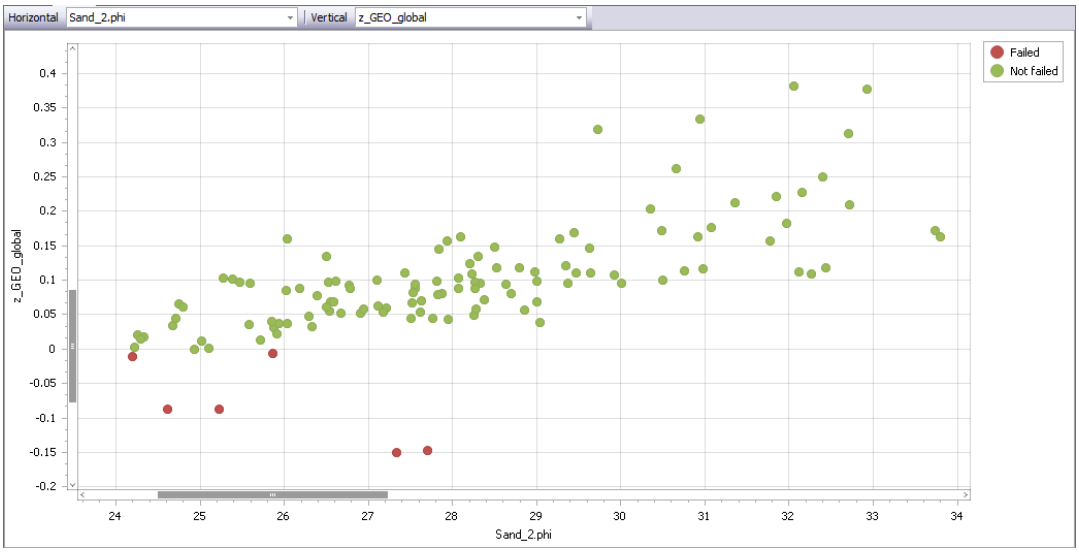


Figure F.2: Sample chart of GRAPA2 v3h3

Findings

- The erratic behaviour is not improved unless the increase of hyperparameter optimisations.
- In all 3 runs is no incompatible data found, this seems to reduce the amount of realisations.
- Looking at the beta value, the results are considered to be reliable.

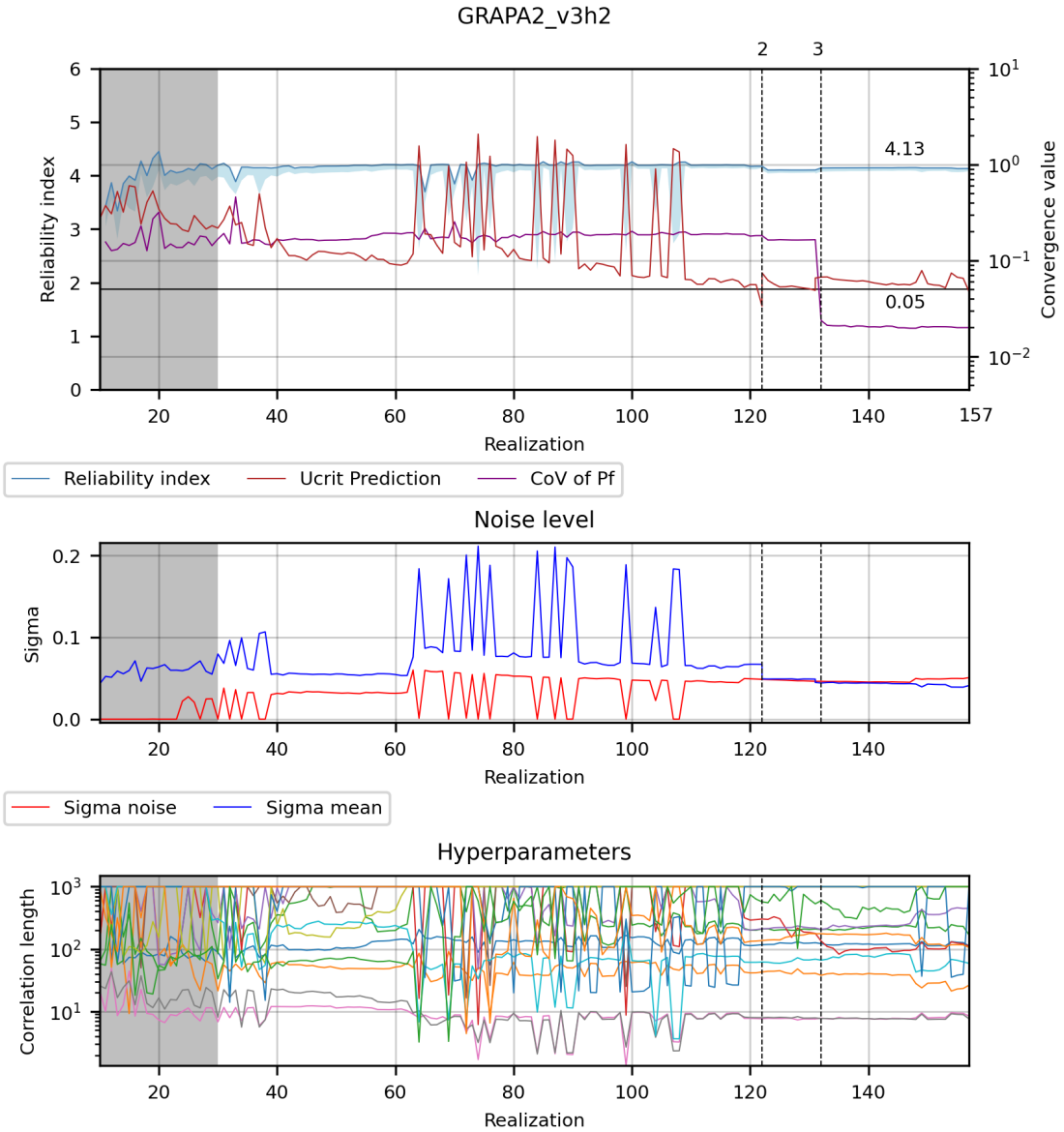


Figure F.3: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v3h2

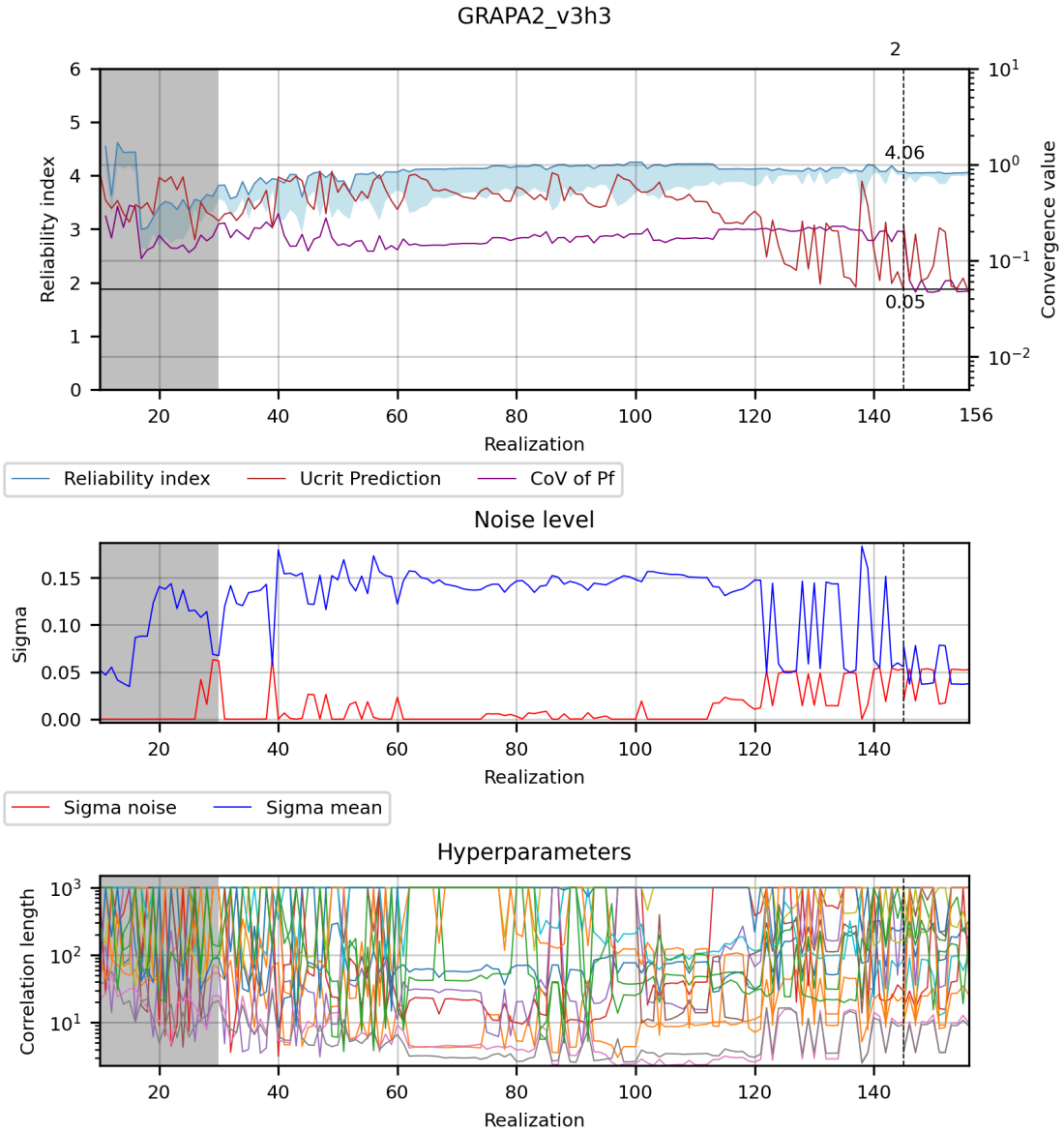


Figure F.4: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v3h3

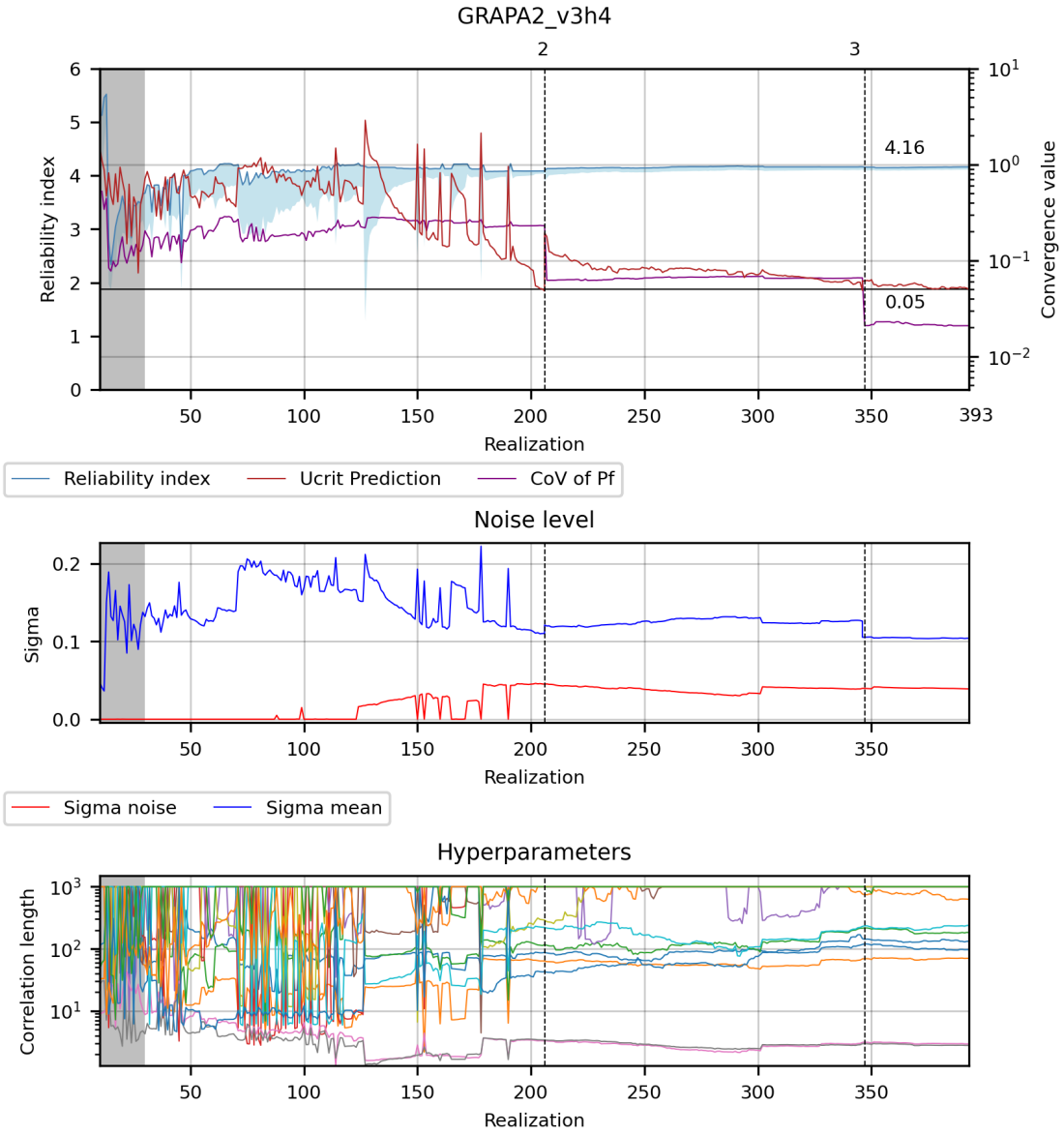


Figure F.5: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v3h4

F.3 GRAPA2 v3k (Including model uncertainty factors)

ERRAGA settings:		
Convergence Criteria	BetaAbsStop	
Learn Function	UNISlearn	
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.25	
Classification Model	GCP	
Beta Prior	5.0	
Size MCS pool	100000	default
Min Realizations	50	
Max Realizations	100	
Hyperparameter Optimisations (Opts)	20	

Table F.5: ERRAGA settings for GRAPA2 v3k

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	3.12	0.05	400	NO

Table F.6: Results of GRAPA2 v3k

Findings

- No convergence, incompatible data is encountered at realisation 155 which increases the number of required realisations.
- The prediction model is almost converged, looking at the graph it seems a few more realisations were needed.
- The degree of noise increases as the run progresses.

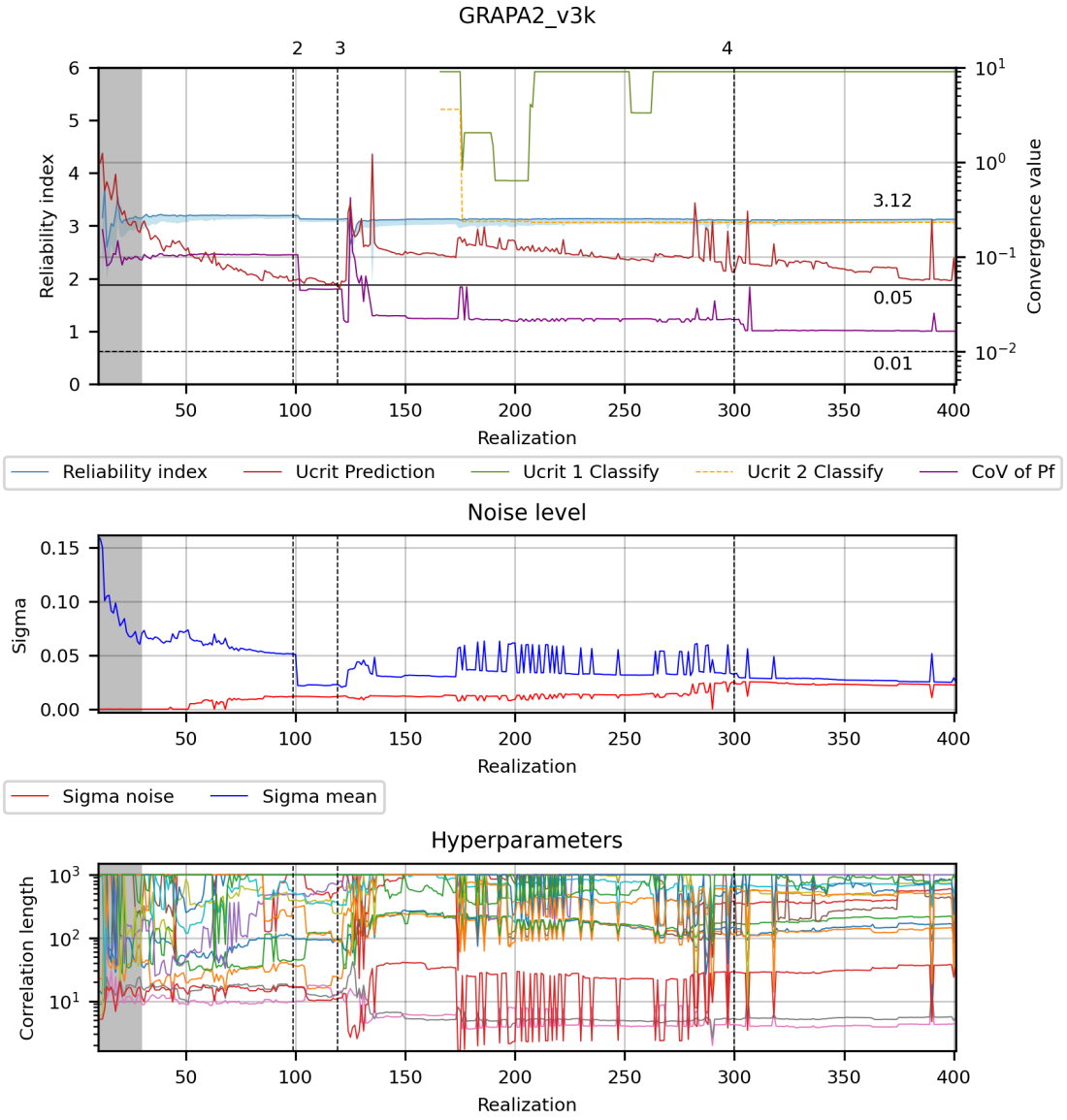


Figure F.6: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v3k

F.4 GRAPA2 v3j (Including model uncertainty factors)

ERRAGA settings:		
Convergence Criteria	BetaAbsStop	
Learn Function	UNISlearn	
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.75	
Classification Model	GCP	
Beta Prior	5.0	
Size MCS pool	100000	default
Min Realizations	50	
Max Realizations	100	
Hyperparameter Optimisations (Opts)	20	

Table F.7: ERRAGA settings for GRAPA2 v3j

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	3.13	0.05	70	YES

Table F.8: Results of GRAPA2 v3j

Findings

- No incompatible data is encountered.
- The run converges before a high degree of noise develops.
- The small degree of noise reduces the effect of NVRR in theory.
- Fast convergence because no incompatible data is encountered and no high degree of noise has developed.

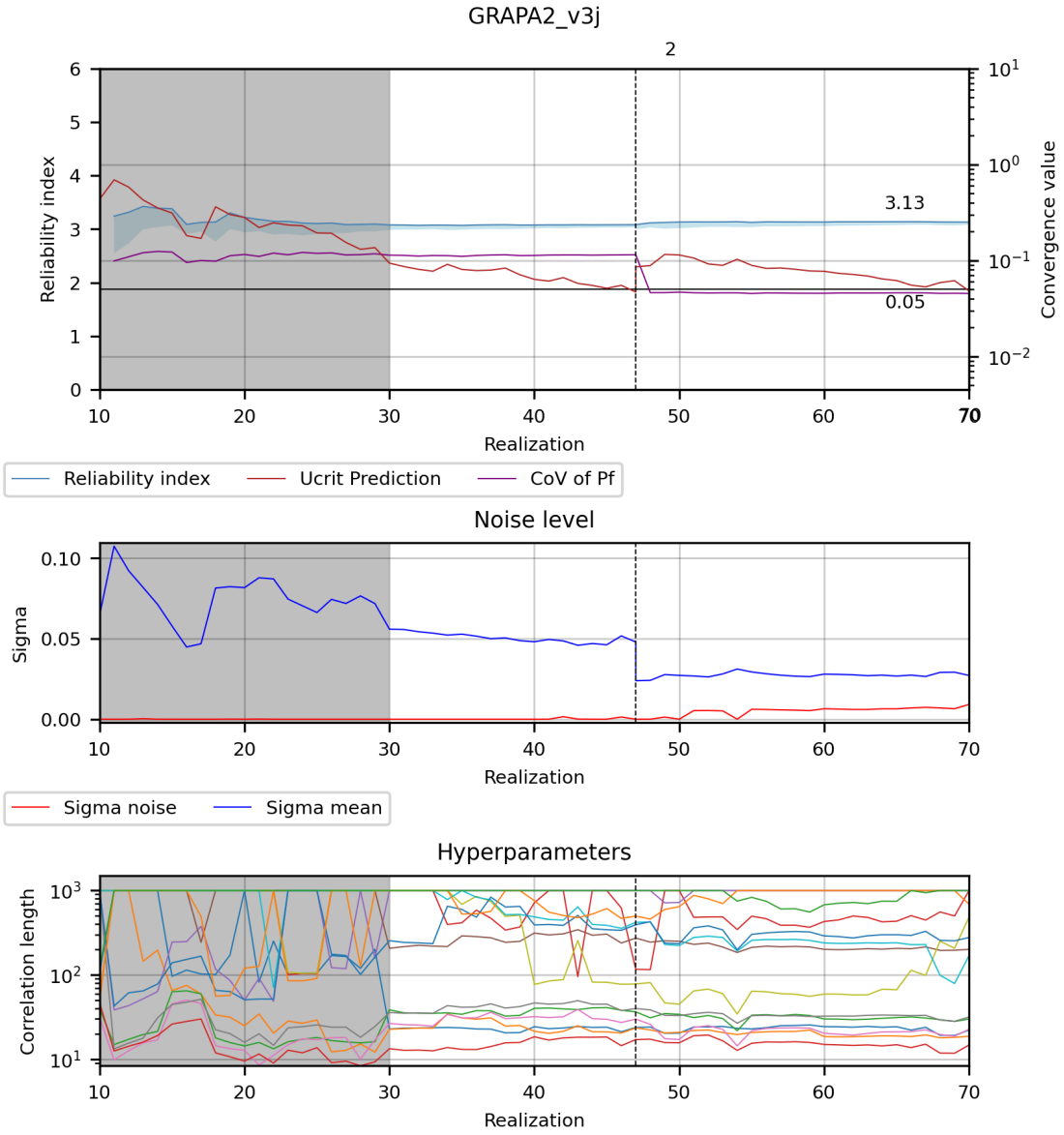


Figure F.7: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v3j

F.5 GRAPA2 v4a (Lowered harbour floor)

ERRAGA settings:		
Convergence Criteria	BetaAbsStop	
Learn Function	UNISlearn	
Noise Term	True	
Noise Variance Reduction Ratio (NVRR)	0.75	
Classification Model	GCP	
Beta Prior	4.0	
Size MCS pool	400000	default
Min Realizations	30	
Max Realizations	100	
Hyperparameter Optimisations (Opts)	20	

Table F.9: ERRAGA settings for GRAPA2 v4a

Method	**Beta**	**Ucrit/CoVar**	**Realisations**	**Convergence**
ERRAGA	3.66	0.05	209	YES

Table F.10: Results of GRAPA2 v4a

Findings

- The harbour floor is lowered by 2.60m which results in a lower β

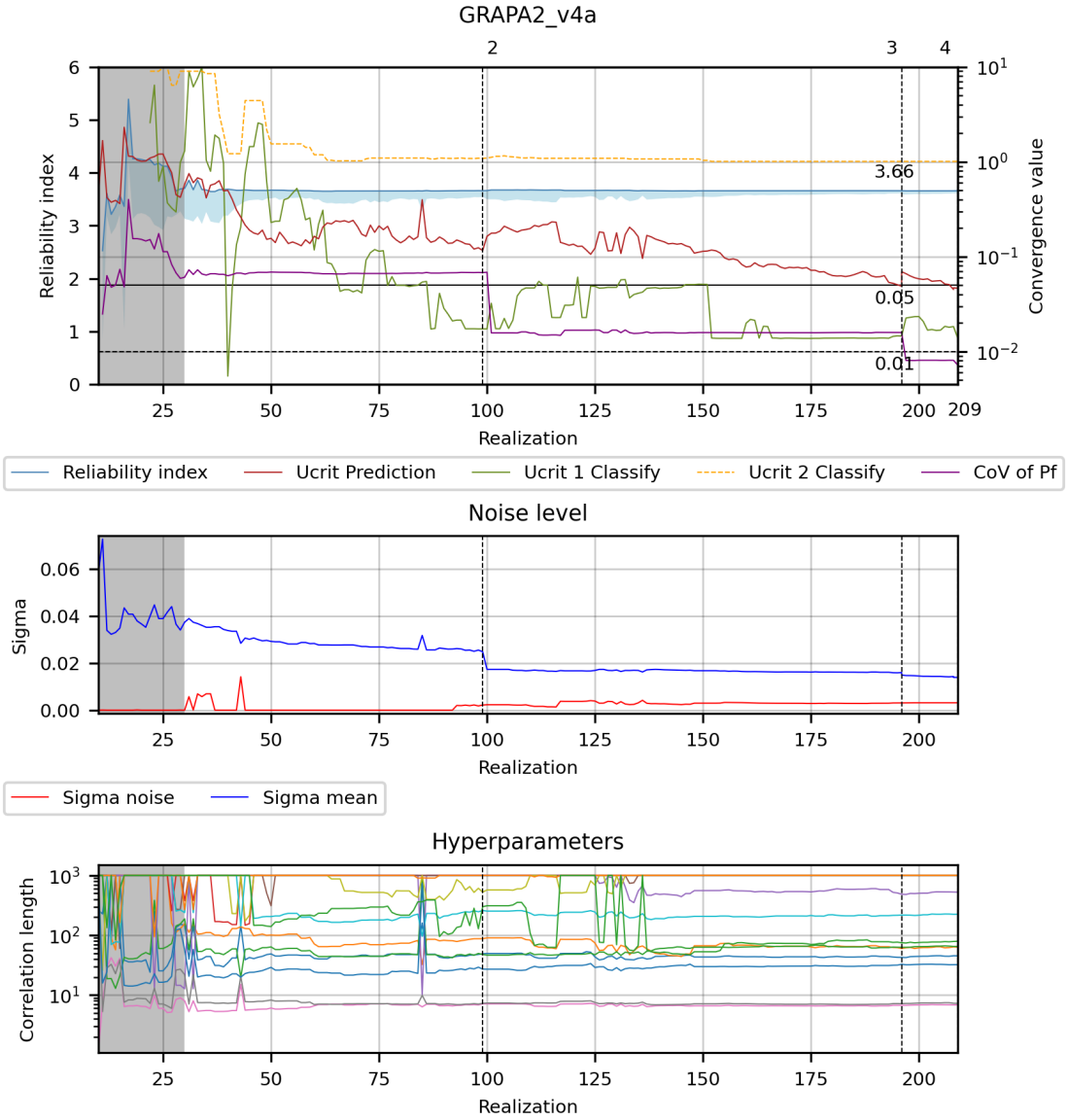


Figure F.8: Convergence-, Noise- and Hyperparameter graphs of GRAPA2 v4a

Appendix G

Failure modes of the PLAXIS model

G.1 Failure mode LSF Front wall

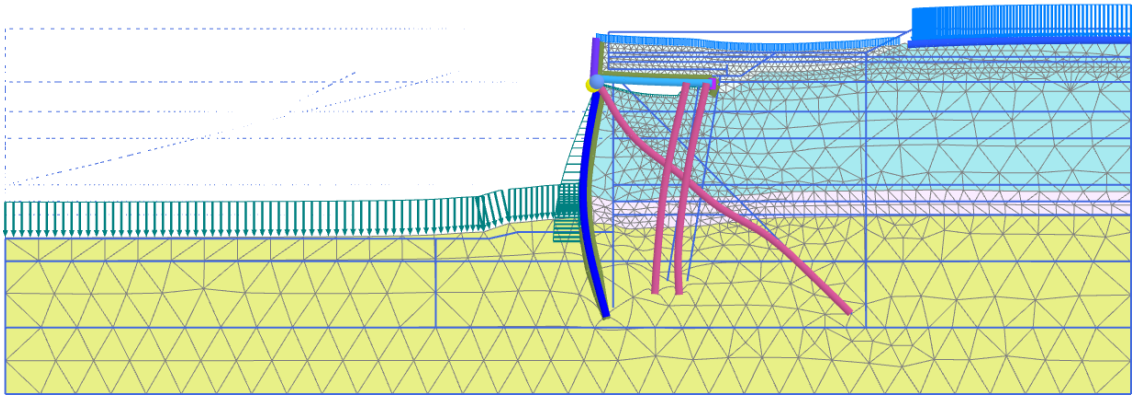


Figure G.1: Deformation mesh (scaled up 10 times) of a failure mode of LSF Front wall. Maximum displacement = 0.4874m

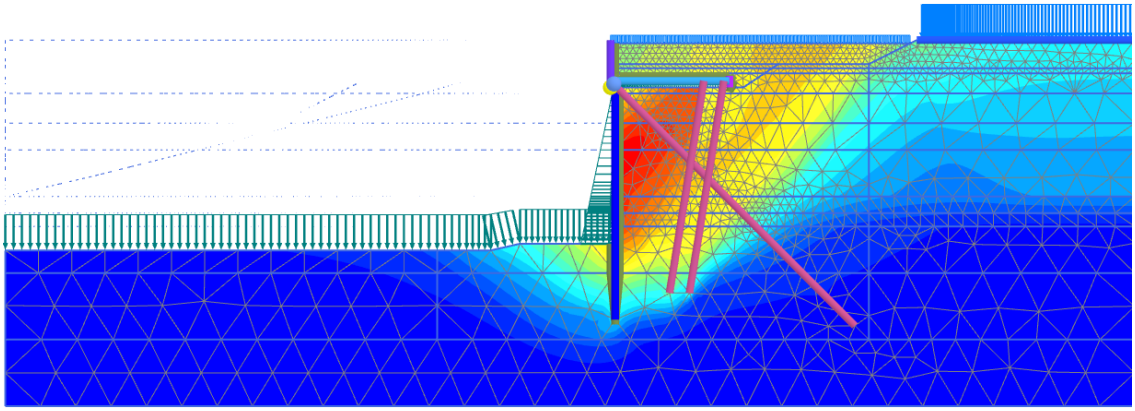


Figure G.2: Total displacements (scaled up 10 times) of a failure mode of LSF Front wall. Maximum displacement = 0.4874m

G.2 Failure mode LSF Geo

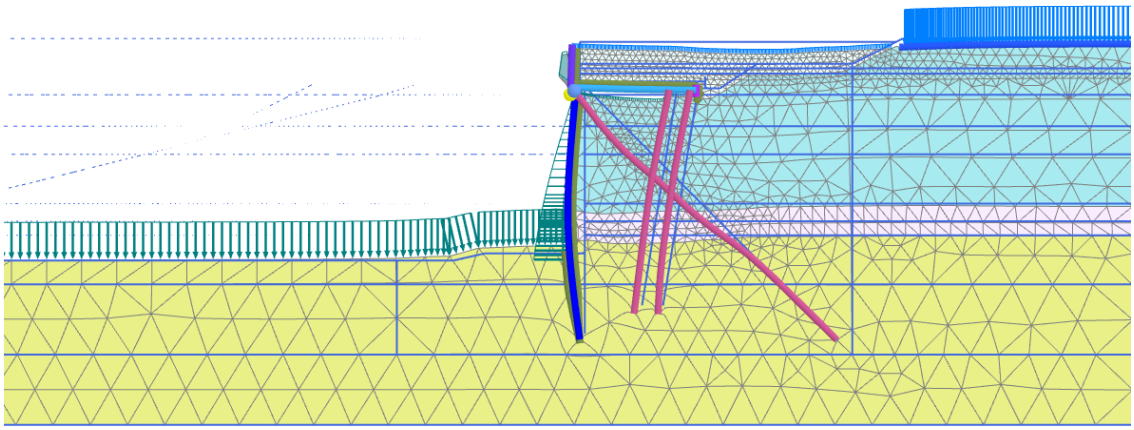


Figure G.3: Deformation mesh (scaled up 5 times) of a failure mode of LSF Geo. Maximum displacement = 1.055m

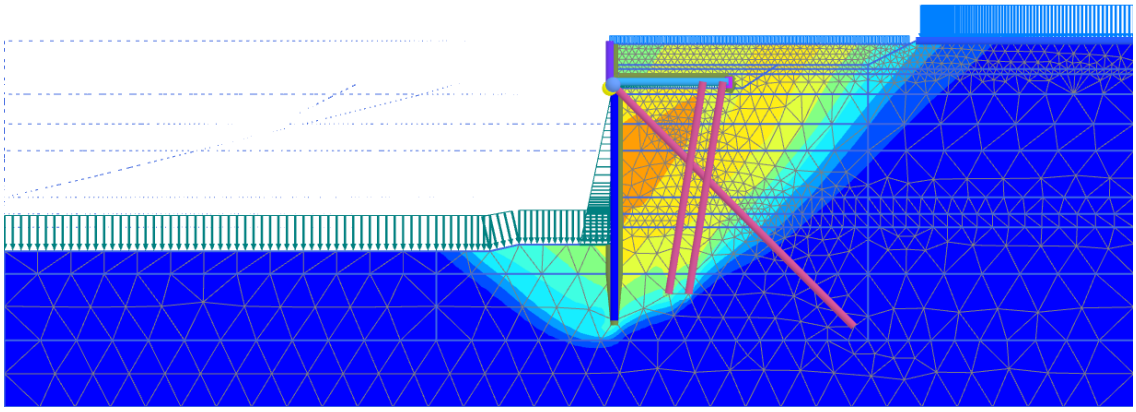


Figure G.4: Total displacements (scaled up 5 times) of a failure mode of LSF Geo. Maximum displacement = 1.055m