HAW
HAMBURG

Masterthesis

Max Sahlke

Development of an optimised design of experiments methodology for the creation of surrogate models

Faculty of Engineering and Computer Science
Department Information and Electrical Engineering

Max Sahlke

Development of an optimised design of experiments methodology for the creation of surrogate models

in the study course *Master of Science Automatisierung*
at the Department Information and Electrical Engineering
at the Faculty of Engineering and Computer Science
at University of Applied Science Hamburg

Supervisor: Prof. Dr.-Ing Dipl.-Kfm. Jörg Dahlkemper
Supervisor: Dr. Jasone Garay García

Submitted on: 30. September 2022

Max Sahlke

Thema der Arbeit

Entwicklung einer optimierten statistischen Versuchsplanung für die Erstellung von Surrogatmodellen

Stichworte

Design of experiments, adaptive sampling, Luftfahrt, maschinelles Lernen

Kurzzusammenfassung

Diese Arbeit behandelt die Entwicklung einer optimierten statistischen Versuchsplanungs Methodik zur Erstellung von surrogate Modellen. Ziel dieser Methodik ist es die durchschnittliche Wertigkeit der mittels Simulationen erstellter Datenpunkte zu erhöhen. Dies führt dazu, dass Modelle der gleichen Güte mit weniger Datenpunkten erstellt werden können oder bessere Modelle mit der gleichen Datenmenge.

Max Sahlke

Title of Thesis

Development of an optimised design of experiments methodology for the creation of surrogate models

Keywords

Design of experiments, adaptive sampling, aviation, machine learning, surrogate model

Abstract

This report describes the development of an optimised design of experiment methodology for the creation of surrogate models. The goal of this method is to increase the average value of the simulated data samples. This yields to the possibility to generate models with the same quality with less data, or models with better quality with the same amount of data. Hence, there is the potential to save cost and time in many development processes while increasing the overall quality.

Contents

List of Figures vii

List of Tables ix

1 Introduction 1

1.1 Motivation 1

1.2 Aim of this thesis 2

1.3 Structure of this thesis 2

2 Theoretical principles 3

2.1 Design of Experiments 3

2.1.1 Full Fractorial Design (FFD) 4

2.1.2 Pseudo-Monte Carlo Sampling 4

2.1.3 Halton sequence 5

2.1.4 Latin Hypercube Sampling (LHS) 8

2.2 Surrogate Models 10

2.2.1 Radial Basis Function(RBF) 10

2.2.2 Kriging 12

2.2.3 Machine Learning Surrogate Models 13

2.3 Adaptive Design of Experiments 14

2.3.1 Bayesian optimization 14

2.3.2 Power Function and Curvature Sampling Criteria 17

2.3.3 Maximum Mean Squared Error and Expected Improvement Function Sampling Criteria 19

3 Analysis of the requirements 24

3.1 Stability 24

3.1.1 Stability of the DoE method 24

3.1.2 Stability of the improvement for the ML-model 25

3.2 General applicability 25
3.3 Calculate more than one sample position at once 26
3.4 Short execution time 26
3.5 Advantage over LHS sampling 26
3.6 Applicability to high-dimensional problems 27
3.7 Cost reduction 27
3.8 Balance of the sample distribution 27

4 Conception 28
4.1 Stability 28
4.2 General applicability 30
4.3 Calculate more than one sample position at once 31
4.4 Short execution time 31
4.5 Advantage over LHS sampling 32
4.6 Applicability to high-dimensional problems 33
4.7 Cost reduction 34
4.8 Balance of the sample distribution 34
4.9 Summary 35

5 Development of an adaptive DoE method 37
5.1 First implementation to test feasibility 43
5.2 Analysis and improvement 46
5.2.1 Normalisation of the design space 46
5.2.2 Offset 48
5.2.3 Variable LHS reference points 49
5.2.4 Reference points outside the design space 50
5.3 First test 53
5.3.1 Setup 53
5.3.2 Results 54
5.4 Further improvements 63
5.4.1 Determine an optimized shaping factor 63
5.4.2 Identifying a stack of samples 65
5.5 Second Test 65
5.5.1 Results 67
5.6 Test on 4 dimensional BF 69
5.6.1 Setup 70

5.6.2 Evaluation . . . 70

6 Test on real application 75

6.1 FAME-W . . . 75

6.2 Setup . . . 77

6.3 FAME-W asynchron test . . . 77

6.3.1 Reduction of the MSE . . . 77

6.3.2 Reduction of the amount of samples . . . 79

6.4 FAME-W synchron test . . . 80

6.4.1 Reduction of the MSE . . . 81

6.4.2 Reduction of the amount of samples . . . 82

6.5 Assessment of the results . . . 83

7 Summary 85

7.1 Conclusion . . . 85

7.2 Outlook . . . 85

Bibliography 87

A Appendix 91

A.1 Code . . . 91

A.2 Data and images investigations and benchmarks . . . 91

A.3 Data and images FAME-W . . . 91

Declaration of Autorship . . . 92

List of Figures

2.1 Example FFD [3] 5
2.2 Example Pseudo-Monte Carlo Sampling [3] 6
2.3 Example Halton [3] 7
2.4 Example LHS [3] 9
2.5 UCB acquisition function [19] [5] 16
2.6 EI Acquisition function [19] [5] 21
2.7 MaxMSE Acquisition function [19] [5] 22

4.1 Sampling behavior[14] 29
4.2 Function 4.1 [14] 30
4.3 Function 4.2 32
4.4 Sampling positions 33

5.1 Used benchmark functions 39
5.2 Comparison of ML topologies for the BFs 43
5.3 Combined criterion 2D surface 45
5.4 Flowchart first implementation 47
5.5 First results 48
5.6 Effect of the offset parameter 49
5.7 Function 1 with LHS reference points 51
5.8 RBF approximation benchmark function 1 52
5.9 Effect of reference points out of design space 53
5.10 Effect of the shaping factor on benchmark function 2 55
5.11 Effect of the shaping factor on benchmark function 1 55
5.12 Data set comparison BF3 Test 1 57
5.13 Effect of the offset on BF2 58
5.14 Effect of the offset on BF1 59
5.15 Error LHS vs adaptive BF1 60
5.16 Error LHS vs adaptive BF2 61

5.17 Comparison RBF with fixed and optimized shaping factor 64
5.18 Comparison RBF Laplace with fixed and optimized shaping factor 65
5.19 Flowchart updated implementation 66
5.20 BF3 in reduced design space 70
5.21 Results BF1_3 adaptively sampled and LHS 71
5.22 BF1_3 sample distribution for 954 samples, 30 reference points 72
5.23 Results BF1_3 adaptively sampled and LHS, 100 reference points 72
5.24 BF1_3 sample distribution for 954 samples, 100 reference points 73
5.25 Results BF1_3 adaptively sampled and LHS NN 74

6.1 Adaptive samples FAME-W MTOW over Wing span 78
6.2 FAME-W comparison of 1000+500 adaptive samples and 1500 LHS samples 79
6.3 FAME-W comparison of 500+400 adaptive samples and 1500 LHS samples 80
6.4 Comparison of 1000+500 adaptive samples asynchronous and synchronous 81
6.5 Comparison of 500+400 adaptive samples asynchronous and synchronous . 82

List of Tables

2.1 Example Halton . . . 7
2.2 Most commonly used RBFs [16] [15] . . . 11

4.1 Comparison of the MSE . . . 33
4.2 Comparison of the methods . . . 36

5.1 Used benchmark functions . . . 37
5.2 NN hyperparameter . . . 40
5.3 PSO hyperparameter settings . . . 44
5.4 Test 1 parameters . . . 53
5.5 Test 1 ML results . . . 56
5.6 Test 1 ML shaping factor . . . 57
5.7 Test 1 ML offset . . . 58
5.8 Test 1 effect of the offset compared to LHS . . . 62
5.9 Test 1 effect of number of reference points . . . 62
5.10 Test 2 parameters . . . 67
5.11 Test 2 ML results . . . 67
5.12 Test 2 effect of number of reference points without BF3 . . . 68
5.13 Test 2 effect of number of reference points with BF3 . . . 68
5.14 Test 2 effect of the offset compared to LHS . . . 68
5.15 Test 2 effect of the stack size compared to LHS . . . 68

6.1 FAME-W used inputs and output . . . 76
6.2 FAME-W improvement achieved compared to LHS benchmark . . . 83

List of Acronyms

Acronym	Meaning
BF	Benchmark Funktion
CFD-CSM	Computional Fluid Dynamics- Computional Structure Mechanics
DACE	Design and Analyses of Computer Experiments
DoE	Design of Experiments
DT	Desicion Tree
EI	Expected Improvement Function
FAME-W	Fast and Advanced Mass Estimation - Wing
FEM	Finite Element Method
FFD	Full Fractorial Design
KPI	Key Performance Indicators
LHS	Latin Hypercube Sampeling
SLE	System of Linear Equations
MAE	Mean Average Error
MaxMSE	Maximum Mean Squared Error
MSE	Mean Square Error
MTOW	Maximum Take Off Weight
MZOW	Maximus Zero Fuel Weight
NaN	Not a Number
NN	Neural Network
PSD	Partical Swarm Optimisation
RF	Random Forest
RREF	Reduced Row - Echelon Form algorithm
CPU	Central Processing Unit
TC ratio	Thicness-to-Chord ratio
UCB	Upper Confidance Bound
USD	United States Dollar

1 Introduction

At Airbus, surrogate models are used in the area of multidisciplinary analysis and optimization. These surrogate models have the advantage over simulations (e.g. CFD-CSM or FEM) that they can approximate the effects of parameter variations in real time. This enables savings in terms of time and costs when developing a new aircraft or aircraft variants. In addition, more variations of the parameters can be performed. The optimal point of design can be searched for and the necessary knowledge about the interrelationships of the parameters at the point of design is provided.

Surrogate models are mathematical models that approximate the relationships using existing data. Accordingly, data must be generated for the creation of these models. In engineering use cases, it is important that these data points are distributed in the parameter space in such a way that the model can learn all the required physical relationships. Another requirement is that as few data points as possible are generated for this purpose, as their generation consumes significant time and financial resources. This subject area is called Design of Experiments (DoE). The DoE should cover the complete information of the parameter space without actually having recorded every data point. The aim of this work is to develop an optimised DoE method.

1.1 Motivation

Generating the necessary data for surrogate models requires considerable financial and time resources. Internal estimates show that the cost of a CFD-CSM data point is in the four-digit USD range. Accordingly, there is considerable potential for savings by reducing the number of simulations required for obtaining still optimal results.

The reduction of the number of simulations has different advantages, such as a reduced computing power requrement, not only for the data production, but also for the model training. Therefore, this reduction makes it possible to generate the necessary data in a sufficiently short time.

Furthermore, there are cases where companies do not want data to be in or created in the cloud for security reasons. In these cases, less computing power is available. Accordingly, a reduction in the amount of data is also advantageous here.

1.2 Aim of this thesis

The objective of this thesis is to provide an overview of some of the most relevant existing methods in the literature, and a further research to determine which DoE methods have an advantage over the currently used Latin Hypercube Sampling (LHS) for the described application. The Key Performance Indicators (KPIs) used in this thesis to compare and describe the performance of a DoE are two, namely the number of data points required to obtain the desired performance, and the magnitude of the error on a test data set.
Subsequently, a benchmark is defined in which the various methods are applied to one or more test functions. The results are then compared to each other as well as to the original DoE.
Finally, the most suitable approach will be determined to create a DoE for a given real aviation-specific problem. The surrogate model based on this will be compared to the currently available one.

1.3 Structure of this thesis

Chapter 2 of this thesis discusses the state of the art and the current implementation. Chapter 3 analyses the requirements for the method to be implemented. Subsequently, the methods in question are evaluated with regard to these requirements in chapter 4. Chapter 5 describes the implementation and optimisation of the method. In Chapter 6, the results are presented in relation to a real application and compared with the current method. Finally, a summary and conclusion of this thesis is presented in Chapter 7.

2 Theoretical principles

In this section, the basic methods and properties of DoE and surrogate models will be discussed. This serves as a basis for the following scientific sections.

2.1 Design of Experiments

As explained in Section 1, the aim of creating a DoE is the optimal data-point generation in order to populate a design space out of which different properties need to be studied and analyzed. There are two main areas in which DoEs are applied. One of them is the testing of interrelationships in a physical test environment. The other one is the research of interrelationships via simulations. In both cases, this methodology can be applied to physical, chemical, or other technical fields. The aim is to gain as much information as possible from the available number of experiments.

For this thesis, only the simulation based type of implementation is relevant. In some sources the phrase “design and analysis of computer experiments” (DACE) is used as a synonym. The main difference to DoE in physical experiments is that simulations are not distorted by random errors due to environmental influences.

In the development phase of products or processes, often only the extreme points are considered with regard to their design. For example, the parameter configuration that leads to the maximum load of a component. These are relevant, but alone cannot provide any information about the system behaviour in the considered design space. If the importance of the analysis does not only rely on the design based solely on the extreme points, but also on the system behaviour, the DoE methodology offers significant added value. The exclusive use of certain input data constellations, where just a small part of the design space is sampled, should be avoided in order to be able to represent relationships as generally as possible. For this purpose, the samples should be distributed relatively evenly, as well as in slightly different distances and constellations to each other.

Another important aspect in planning a DoE is the choice of inputs. It is only possible to determine relationships between inputs for which a set of different values are known. For example, friction, drag and power are relevant values for the prediction of acceleration. However, it is not possible to make predictions about accelerations with variable mass based on a DoE that only contains the aforementioned three values, as the information about the mass is missing and the complete physical correlation cannot be built. In practice, the choice of suitable inputs is often more difficult than for basic physical problems. Furthermore, it should be noted, that the complexity of the model increases with the number of selected inputs.

Finally, before creating the DoE, it should be ensured, that the entire area of interest matches the design space. Outside of this design space, it will not be possible to make valid predictions about the behavior of the system.

Since various methods have been developed over time, only the most important and relevant ones for this thesis will be discussed in the following sections. [2]

2.1.1 Full Fractorial Design (FFD)

The most straightforward way to create a DoE is the FFD methodology. Here, all samples are distributed in a grid (sometimes called grid design) across the design space. This ensures an even distribution of the samples, but sampling with a fixed step size leads to the fact that correlations in non-linear areas are either not at all or only insufficiently captured. [2] [24] [3]

Another disadvantage of this method is the fact, that the possible number of samples depends on the number of dimensions of the design space, i.e. only multiples of $2n$, were n is the number of dimensions, can be used as the number of samples. For high-dimensional problems, this can lead to the fact that even the smallest possible number of possible samples already exceeds the number of affordable samples, although only the vertices of the hyperspace are examined.

A basic example of the FFD design with two input variables is shown in Figure 2.1.

2.1.2 Pseudo-Monte Carlo Sampling

Pseudo-Monte Carlo Sampling (also called pseudo-random sampling or just Monte Carlo sampling) is a DoE method based on a pseudo-random number generation algorithm. The samples are distributed in the design space based on these generated pseudo random numbers. [2]

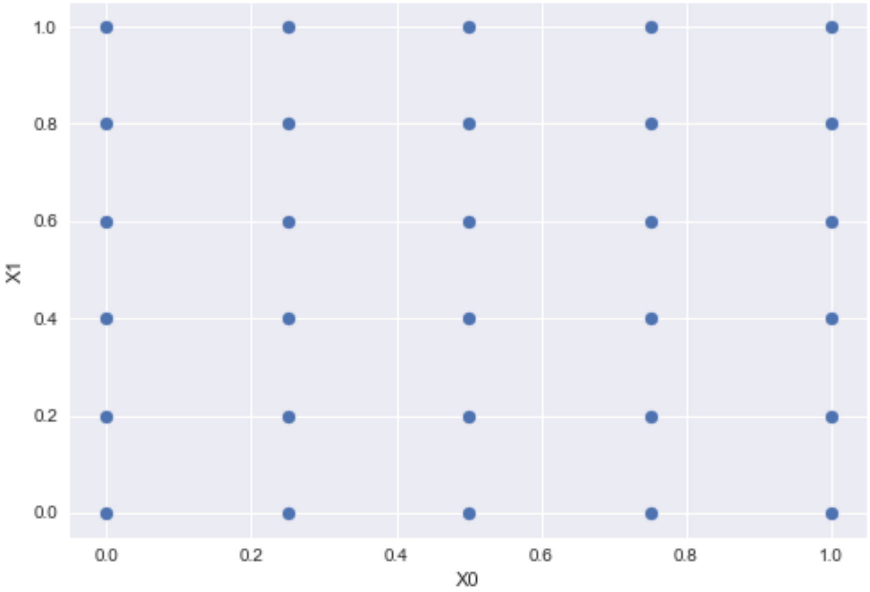


Figure 2.1: Example FFD [3]

This leads to the fact that more different input values and combinations of input values are processed compared to the FFD method. This is because the input values are not fixed to values on a grid. Since there are pseudo randomly chosen, the chance that a specific input value is used more than once is low.
Hence, the generation of close to real random numbers in a deterministic system is a known technical problem. In cases where the data is correlated either strongly or in other forms, the quality of the distribution of the samples is determined by the quality of the pseudo-random number generation algorithm.
This leads to the fact that pseudo-Monte Carlo sampling will often leave large regions of the design space unexplored. [2].
An example for pseudo-Monte Carlo sampling with two input variables is shown in Figure 2.2

2.1.3 Halton sequence

The Halton sequence is a deterministic method to distribute samples in an interval from 0 to 1. It is part of the low discrepancy sequences (also called quasi-Monte Carlo sampel-

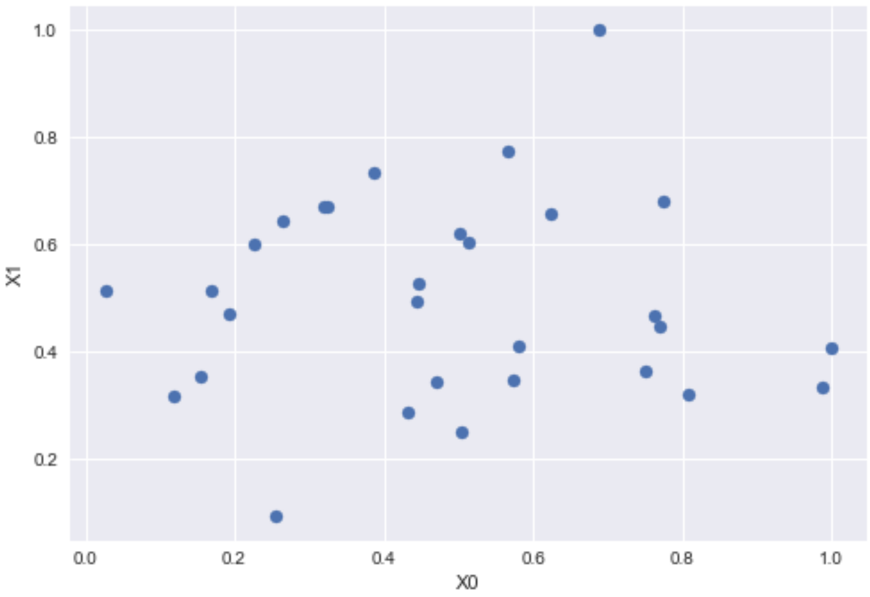


Figure 2.2: Example Pseudo-Monte Carlo Sampling [3]

ing). Based on a prime number per dimension of the design space, each sample is placed at a sub-interval of the previous sample, as shown in Figure 2.3 and Table 2.1.

The Halton Sequence is created for each dimension based on a different prime number (here 2 and 3) using Equation 2.1.
This makes it possible to map more different values of the inputs with the same number of samples than with FFD. Accordingly, interrelationships can be better identified. The number of samples can be freely chosen.
The disadvantage of this method is, that only a limited number of different Halton sequences are available for a specific design space due to the limitation on the underlying prime numbers. [25] [11]
Furthermore, the distribution of samples using the Halton sequence results in a distribution across the diagonals of the design space, as shown in Figure 2.3. This distribution is not as strict as with FFD.
One advantage that the Halton sequence has over many other DoEs is, that the sample number can be increased even after it has been determined or after the experiments have

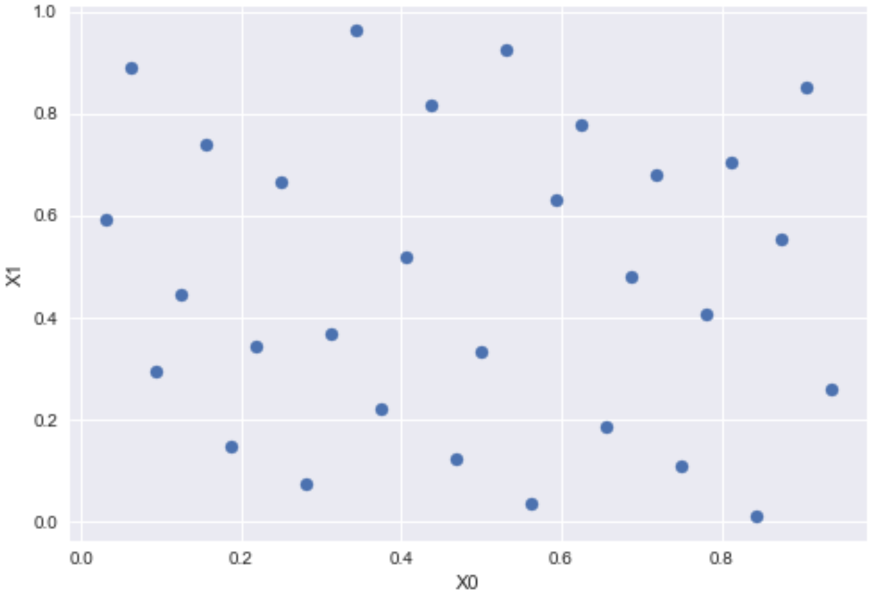


Figure 2.3: Example Halton [3]

performed, without decreasing the quality of the distribution of the points.

$$Halton(p)_k = \sum_{i=0}^{log_p k} \frac{1}{p^{i+1}}([\frac{1}{p^i}]mod(p)) \tag{2.1}$$

Sample number	$Halton(2)$	$Halton(3)$
1	0.5000	0.3333
2	0.2500	0.6667
3	0.7500	0.1111
4	0.1250	0.4444
5	0.6250	0.7777
6	0.3750	0.2222
7	0.8750	0.5556
8	0.0625	0.8889
9	0.5626	0.0370
$\vdots$	$\vdots$	$\vdots$

Table 2.1: Example Halton

2.1.4 Latin Hypercube Sampling (LHS)

LHS is also a method of distributing the samples pseudo-randomly in the design space. For a given set of p samples, the range of each parameter is divided into p bins of equal probability. For n dimensions, this partitioning results in a total of p^n bins in the design space. Then, the pseudo-randomly selected p samples from the parameter space, subject to the following two constraints. Firstly each of the one-dimensional projections of the p samples and bins, there is only single sample in each bin. Secondly each sample is pseudo-randomly placed in a bin. [14]

In a two-dimensional example like the one shown in Figure 2.4, the sample placement rules for LHS guarantee, that only one bin can be selected in each row and column. [2]

As with Halton sequence sampling, the number of samples can be arbitrarily chosen. Due to the distribution of the samples, LHS is more precise and effective than FFD, especially in capturing non-linear correlations. Since the samples are subject to a pseudo-random distribution, the advantage over sampling using Halton sequence is, that more different distributions are possible for a given design space and sample size. [24] [3]

LHS is the state of the art methodology for DoE design. It is also the actual used DoE, within the industry area this thesis is based on.

LHS can be supplemented with samples at the corner points of the design space in order to increase the information density at the boundaries of the design space.

It is not possible to increase the number of samples without worsening the distribution after completion of the experiments, as is the case with the Halton sequence.

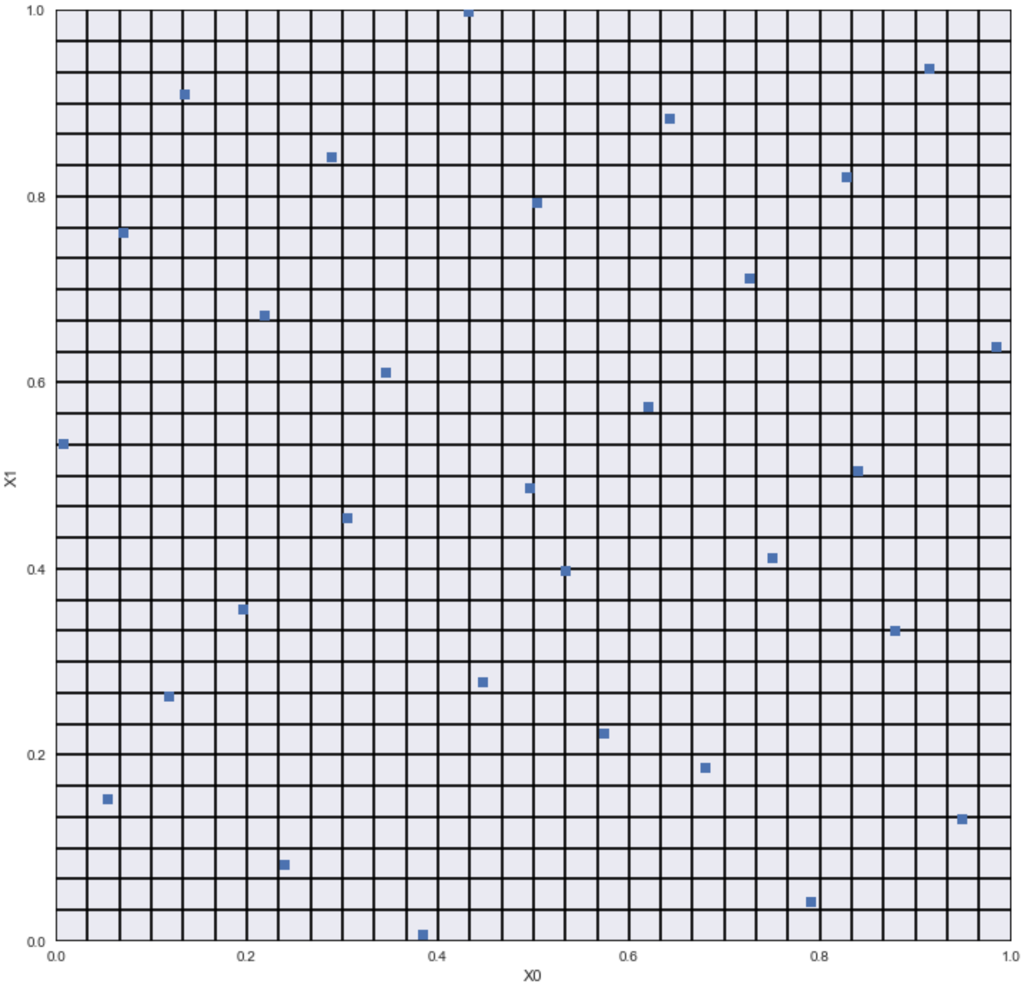


Figure 2.4: Example LHS [3]

2.2 Surrogate Models

Surrogate models are analytical models that predict the input and output behavior of complex, higher-order systems. Surrogate models provide a continuous function, which approximates the response of the underlying complex system. The primary benefit from using Surrogates is, when the underlying function is expensive to evaluate, either computational cost or physical testing. The surrogate models can be created from a limited number of discrete samples of the expensive function and be used to rapidly approximate the response. [23]
There are many possible uses for these surrogate models. Engineers can understand and analyse the relationships of the model despite the small amount of data. Since the relationship between inputs and outputs can be determined with much less computer power, many more parameter variations can be tested. This leads to desighn points being reached that better fit the requirements. [1]
In the following, some of the most relevant methods for creating surrogate models are explained.

2.2.1 Radial Basis Function(RBF)

In RBF approximations, several basis functions are placed in superposition to each other. In order to optimise the basis functions such that the model approximates reality as closely as possible, the Euclidean distance between the known samples and the reference points is used. The mathematical output of this model is described by Equation 2.2. [14] [11]

$$f(x) = \sum_{i=1}^{M} c_i \phi(r_i) = \sum_{i=1}^{M} c_i \phi(\|x - \xi_i\|) \tag{2.2}$$

where M is the amount of basis functions (number of reference points), c_i is the weight of the actual basis function, ϕ is the basis function, x is the position of the input and ξ_i is the position of the actual reference point.[15]
Any function fulfilling the condition that its output is solely dependent on a single input variable is a mathematically valid RBF function. In this case the input variable is the Euclidean distance between the point to be predicted and the reference point. The most

common functions used as RBF in practice are shown in the Table 2.2. [16] [15]

RBF	$\phi(r)$
Gauss function	$e^{-(\alpha r)^2}$
Inverse Quadric (IQ)	$\frac{1}{1+(\alpha r)^2}$
Thin-Plate Spline (TSP)	$(\alpha r)^2 log(\alpha r)$
Wendland's $\phi_{3,2}$	$(1-\alpha r)^6 + (35(\alpha r)^2 + 18\alpha r + 3)$

Table 2.2: Most commonly used RBFs [16] [15]

In the equations of Table 2.2 $\alpha > 0$ is a shaping parameter, and r is the input value of the function. The vector of weights $c = (c_1, ..., c_M)^T$ needs to be determined by making the approximation as accurate as possible. Either Lagrange multipliers or solving a system of linear equations (LSE) can be used for this. [15]

In the following, the methodology using an LSE to perform the approximation of c is explained, as it is the methodology used during the thesis when RBFs are performed. The LSE equation is shown in Equation 2.3.

$$Ac = h, \tag{2.3}$$

where A is the matrix of this system, i.e. it contains all outputs of the RBF from all reference points to all known points, c is the vector of weights required, and h are the values of the known points. This leads to Equation 2.4.

$$\begin{pmatrix} \phi_{1,1} & \dots & \phi_{1,M} \\ \vdots & \ddots & \vdots \\ \phi_{N,1} & \dots & \phi_{N,M} \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_M \end{pmatrix} = \begin{pmatrix} h_1 \\ \vdots \\ h_N \end{pmatrix} \tag{2.4}$$

where M is the number of RBFs and N is the number of known points. For a good approximation, $N \gg M$ is required.

This leads to an overdetermined system of equations. To solve it, it can be extended as shown in Equation 2.5.

$$A^T A c = A^T h \tag{2.5}$$

To determine c, the Reduced Row-Echelon Form algorithm (RREF) is used. [15] Alternatively, other algorithms such as singular value decomposition can be used as well, but this is out of the scope of this thesis, since the RREF method is also the preferred method in the literature. [14]
For a good approximation, it is important to choose a suitable shaping factor. If this is not the case, the approximation might become unusable.
Furthermore, there are methods that enable not only the determination of a globally shaping factor, but also a specific shaping factor for each RBF. This translates into improvement of the approximation, but is highly computationally expensive and usually only applicable to low-dimensional problems. [16] [20]

2.2.2 Kriging

The Kriging algorithm describes the modelling of interrelationships through the correlation of stationary Gaussian fields, based on Gaussian process modeling. In contrast to the RBF methodology, Kriging is a stochastic process. [7]
The Kriging model is described by Equation 2.6.

$$\hat{y}(x) = \mu(x) + \epsilon(x), \tag{2.6}$$

where $\hat{y}$ is the estimated value at position x, μ is the mean value, and ϵ is a normally distributed error term with expected value 0 and variance r^2. The mean value μ is the mean of previously defined functions by the user. The functions are often linear functions, but higher order terms can be used as well. [14] [7]
The universal Kriging model is written as:

$$\hat{y}(x) = \sum_{k=0}^{n} \gamma_k f_k(x) + \epsilon \tag{2.7}$$

where γ is an n-dimensional vector of regression coefficients, and n is the number of used known functions. The error term ϵ represents a random process correlated to the distance between the reference point of the function and actual point.[14]

To use the model, regression coefficients must be calculated accordingly, and Equation 2.8 can be used for this purpose. [14]

$$\gamma = (\mathbf{F}^T\mathbf{R}^{-1}\mathbf{F})^{-1}\mathbf{F}^T\mathbf{R}^{-1}\mathbf{y}, \tag{2.8}$$

with

$$\mathbf{F} = \begin{pmatrix} f_0(x_0) & \dots & f_n(x_0) \\ \vdots & \ddots & \vdots \\ f_n(x_0) & \dots & f_n(x_n) \end{pmatrix} \tag{2.9}$$

$$\mathbf{R} = \begin{pmatrix} \phi_{0,0} & \dots & \phi_{0,n} \\ \vdots & \ddots & \vdots \\ \phi_{n,0} & \dots & \phi_{n,n} \end{pmatrix}, \tag{2.10}$$

where y is the known data, and ϕ is the correlation function. The form of ϕ can be chosen. A possible way to ensure that ϵ is a Gaussian process and the correlation function ϕ has a radial basis is described in equation 2.11:

$$\phi_{i,j} = e^{d(x_i,x_j)}, \tag{2.11}$$

with

$$d(x_i, x_j) = \sum_{k=1}^{n} \theta_k |x_{i,k} - x_{j,k}|^2, \tag{2.12}$$

where θ_k describes the importance of the kth coordinate. [14]

2.2.3 Machine Learning Surrogate Models

Machine Learning (ML) models have the ability to learn the relationships between inputs and outputs. ML models are built based on sample data, known as train data. This train data contains a vector of inputs and outputs for each sample. In the case of surrogate models, these vectors contain numerical data, on the basis of simulation values of the system to be investigated. An algorithm is used to train the model so that it can approximate the input output behavior as well as possible. For example, the gradient descent method is used for this.

Different model topologies can be used for this purpose, e.g. Random Forests (RF) or Neural Networks (NN). However, NNs must be tuned to each data set by adapting the so called hyperparameters. This leads to the fact that the evaluation of the quality of a data set using NNs requires a lot of time. Since the aim of this work not is to find the best model, but the best data set, RF are predominantly used in the following. This is considered in more detail in Chapter 5.

2.3 Adaptive Design of Experiments

The basic methods used to produce DoEs explained in Section 2.1 always distribute the samples without using any knowledge of the physics involved. Furthermore, they can only be partially extended. For systems that exhibit a continuous behavior, sufficiently accurate surrogate models can be created with a predefined number of samples.

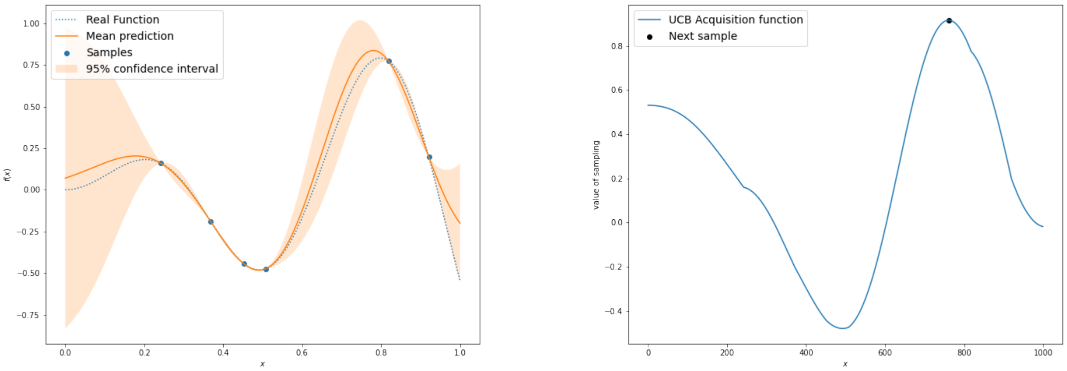
However, for systems that do not have a continuous behavior or for models that are not sufficiently accurate, it is advantageous if additional samples can be added. In the best case scenario, these samples are distributed at positions that lead to the extrema mathematical outputs of a function or positions where small deviations lead to large changes in the output.

There are various mathematical methods for creating function approximations such as those represented by surrogate models, and to further optimize these approximations using additional data. This enables enriching an existing data set with adaptively added data contain as much domain information as possible. In the following, some of these methods are presented.

2.3.1 Bayesian optimization

Bayesian optimization is an approach to optimize objective functions that take a long time to be evaluated. It is based on a set of functions with different shapes.

All these functions are parameterised such that they have no deviations with respect to the existing samples. This implies that all the functions have the same value at the position of these known samples. Between those samples, the functional values differ. Based on this difference, this standard deviation is calculated. The mean prediction is also



(a) First approximation

(b) First determined adaptive sample

calculated. Making use of that information, the optimal position for the next sample can be determined based on a so called acquisition function. There are different acquisition functions available, such as probability of improvement, Expected Improvement (EI), Upper Confidence Bound (UCB) and entropy search. In the following example UCB is used. [8]

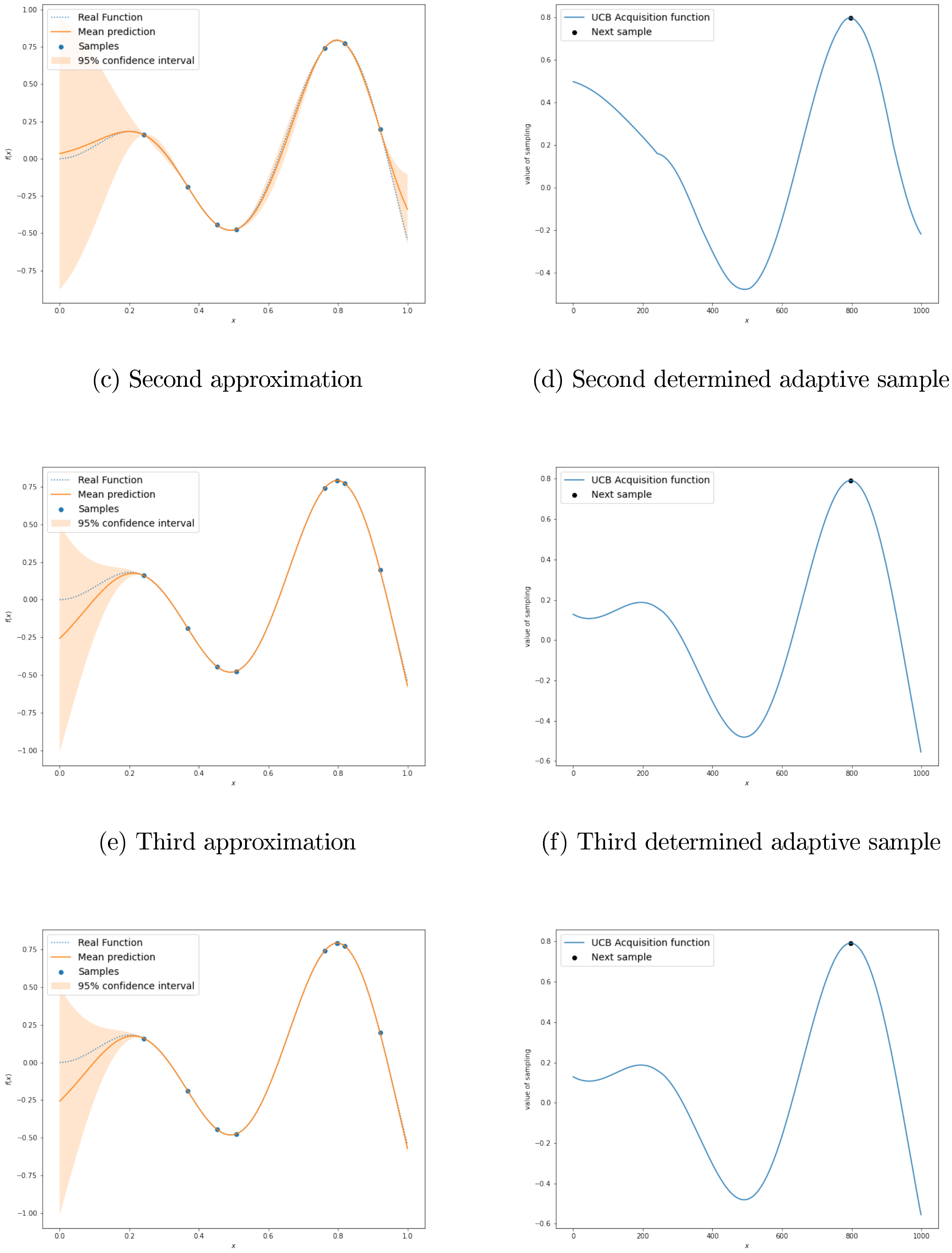
Using UCB, the value of the acquisition function is calculated using Equation 2.13.

$$a_{UCB} = \mu(x) \pm \beta\sigma(x) \tag{2.13}$$

$$\sigma(x) = \sqrt{K(x,x)}, \tag{2.14}$$

Where K is the number of used functions, $\beta > 0$ is a hyperparameter, $\mu(x)$ is the mean value, and $\sigma(x)$ is the standard deviation. In this example $\beta = 1$ is chosen. It is possible to use UCB to search for maxima or minima of functions. The unknown function to optimize in this example is function 2.15. At the beginning of this example 5 samples are randomly chosen. Then 4 more samples are selected based on the Bayesian optimization process. The results of this example are shown in Figure 2.5.

$$f(x) = x\sin(10x) \tag{2.15}$$



(c) Second approximation

(d) Second determined adaptive sample

(e) Third approximation

(f) Third determined adaptive sample

(g) Fourth approximation

(h) Fourth determined adaptive sample

Figure 2.5: UCB acquisition function [19] [5]

As it can be seen, the goal of the Bayesian optimization is to fit the mean prediction with just a few adaptive samples as close as possible to the real function.

It should be noted that there are also variants of the Bayesian optimisation that take into account noise on the data. However, as the subject of this thesis is the handling of simulation data that is assumed to be noise-free, the effect of noise is out of the scope of this thesis.

2.3.2 Power Function and Curvature Sampling Criteria

This sampling criterion is based on the RBF approximation surrogate model. As with the previous methods, an initial data set is needed for this method. The goal of this method is on the one hand to place samples in the locations, where all the local non-linearities are, and on the other hand to explore the design space. [14] [10]

To achieve this, the criteria applied to sample the method is based on two different sub aspects. The first aspect is the description of the value of curvature at any point of the design space. For this purpose, the Laplace operator (Equation 2.16) is used. The Laplace operator describes how much the value of the curvature of the point currently observed differs from the average value of the curvature of the surrounding points. [14]

$$\nabla^2 \hat{y} = \sum_{i=1}^{n} \frac{\partial^2 \hat{y}}{\partial x_i^2} \tag{2.16}$$

The second aspect is how close the currently observed point is to the next known sample. For this purpose, the square of the power function (equation 2.17) is used. The input here is the Euclid distance between the currently observed point and the closest known sample. Function 2.17 results in very low values for low inputs. This way the adaptive samples get separated from both known samples, as well as from known clusters of samples. [14]

$$P = \sqrt{||(\hat{y}, y)||} \tag{2.17}$$

The combined criterion results in Equation 2.18

$$C = (|\nabla^2 \hat{y}| + \epsilon)P, \tag{2.18}$$

where $\epsilon > 0$ is an offset parameter that ensures a nonzero value when $|\nabla^2 \hat{y}| = 0$. The absolute value of the Laplace operator is used, as it is not decisive whether the non-linearity is a right or left curvature of the function.[14]
Since the design space is in most cases defined by real numbers, there are infinitely many different possible positions for the next adaptive sample at each time. For this reason Particle Swarm Optimization (PSO) is used to find a sample position with a high value in a sufficiently short time on the combined criterion. [14]

Particle Swarm Optimization

PSO is an evolutionary algorithm that improves a candidate sample based on a givan measure of quality. A swarm of particles is moving around in the given design space. Each particle represents a potential solution of the search for the best solution. Every particle is affected by its own best solution and the global known best solution for the problem. Both of these factors effect the acceleration of the particle. This way all particles converge in the direction of the global known best point while also each single particle is moving to its own known best solution. This improves both the global known optimum and minimises the likelihood of a better optimum being missed. [6]
The effect of the position of a particle being charged after each generation is described in Equation 2.19.

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1), \tag{2.19}$$

where $x_{ij}(t)$ is the actual position of the actual particle i in the dimension j, and $v_{ij}(t+1)$ is the new velocity of the actual particle in the dimension j. In the following, the global best PSO algorithm is explained, as it is used in this thesis. There are also different algorithms in the literature, like local best PSO. [6]
The new velocity is calculated by equation 2.20.

$$v_{ij}(t+1) = w \cdot v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}[\hat{y}_{ij}(t) - x_{ij}(t)] \tag{2.20}$$

where $v_{ij}(t)$ is the velocity of the particle i in the dimension j at time step t, $0 < w < 1$ is a prefactor to prevent the velocity of the particles from increasing to infinity, c_1 is the

cognitive component, which is a hyperparameter describing how relevant the effect of the best own known solution is, c_2 is the social component, which is a hyperparameter describing how relevant the influence of the best global known solution is, $\hat{y}_j$ is the global known best solution, $r_{1j}(t)$ and $r_{2j}(t)$ are random values in the range $[0, 1]$.[6]
The optimal choice of these factors ensures the efficiency of this method. The positions of the particles are initialized to uniformly cover the design space. This ensures that the swarm optimisation method covers all the areas of the design space and find the global optimum. The initial velocities are initalised to zero. The uniform initialisation of the position already ensures random directions of movement for the particles. A random initialisation of the velocities leads to the risk that particles might leave the design space.[6]

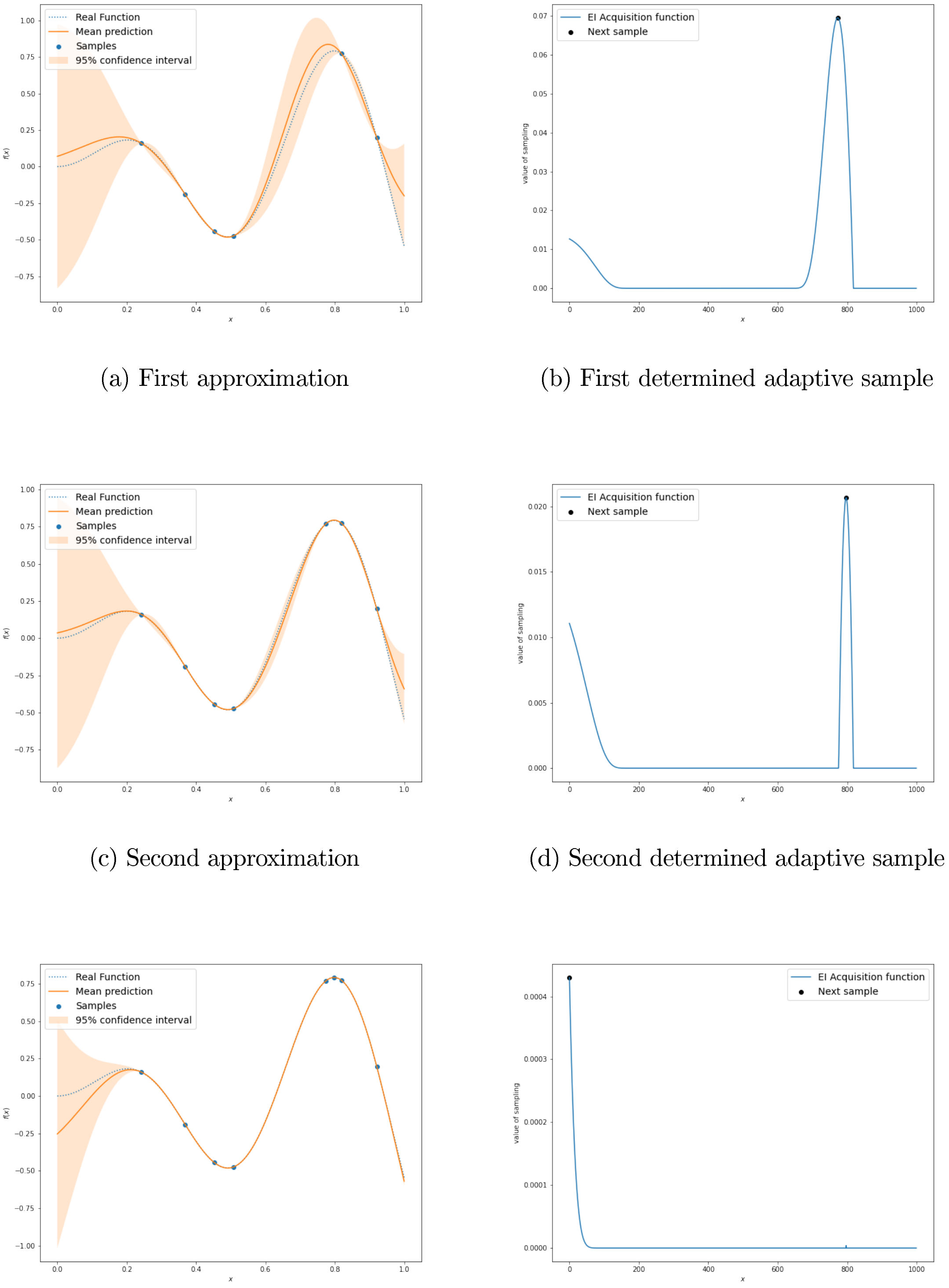
The position of the combined criterium determined by PSO is used as input to generate the next sample. This sample is then used to create a new, more accurate RBF approximation. The process is iterative until a termination criterion is reached.

2.3.3 Maximum Mean Squared Error and Expected Improvement Function Sampling Criteria

Similar to the the method explained in Section 2.3.2, the goal of this method is to find an optimal solution that places the samples where the non-liniaritys are found and enables the exploration of the design space. For this purpose, a kriging model is created. Since a kriging model always consists of several functions, this dose not only return the mean value of a point in the design space, but also the uncertainty of the model at this position. The search for the maximum uncertainty ensures that positions in the design space are found that are far away from the known samples. This sample criterion is called Maximum Mean Squared Error (MaxMSE) [14]
The EI function is used to search for positions at which a maximum or minimum is likely to be found. The EI function can be written as shown in Equation 2.21.

$$E[I(x)] = \begin{cases} (y_{min|max} - \hat{y})\Phi(\frac{y_{min|max}-\hat{y}}{s}) + s\phi(\frac{y_{min|max}-\hat{y}}{s}) & s > 0 \\ 0 & s = 0 \end{cases} \tag{2.21}$$

where Φ and ϕ are cumulative distribution functions, $\hat{y}$ is the mean value, and s is the variance. An example for the EI can be seen in Figure 2.6, and for MaxMSE in Figure 2.7.



(a) First approximation

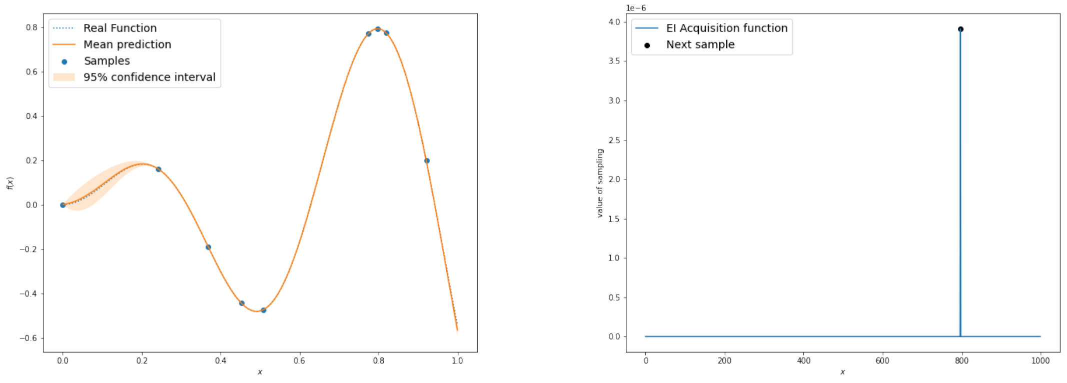
(b) First determined adaptive sample

(c) Second approximation

(d) Second determined adaptive sample

(e) Third approximation

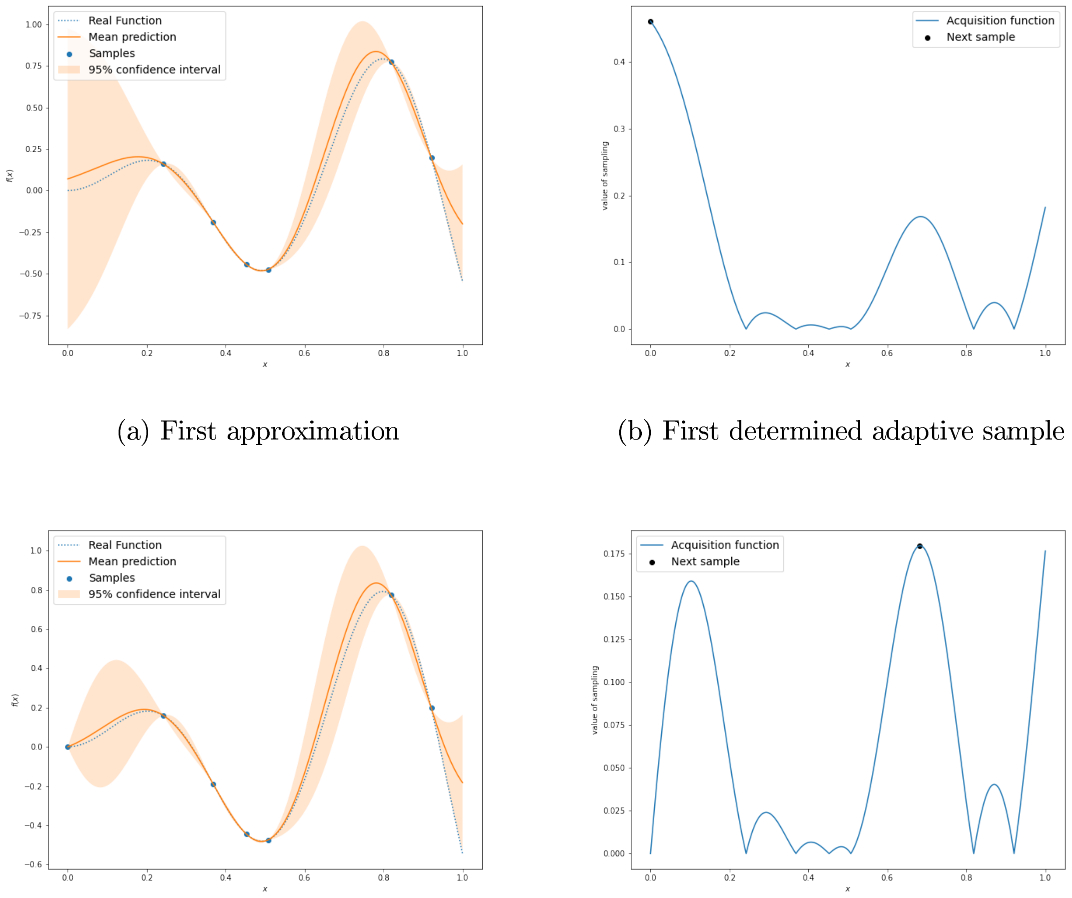
(f) Third determined adaptive sample



(g) Fourth approximation

(h) Fourth determined adaptive sample

Figure 2.6: EI Acquisition function [19] [5]

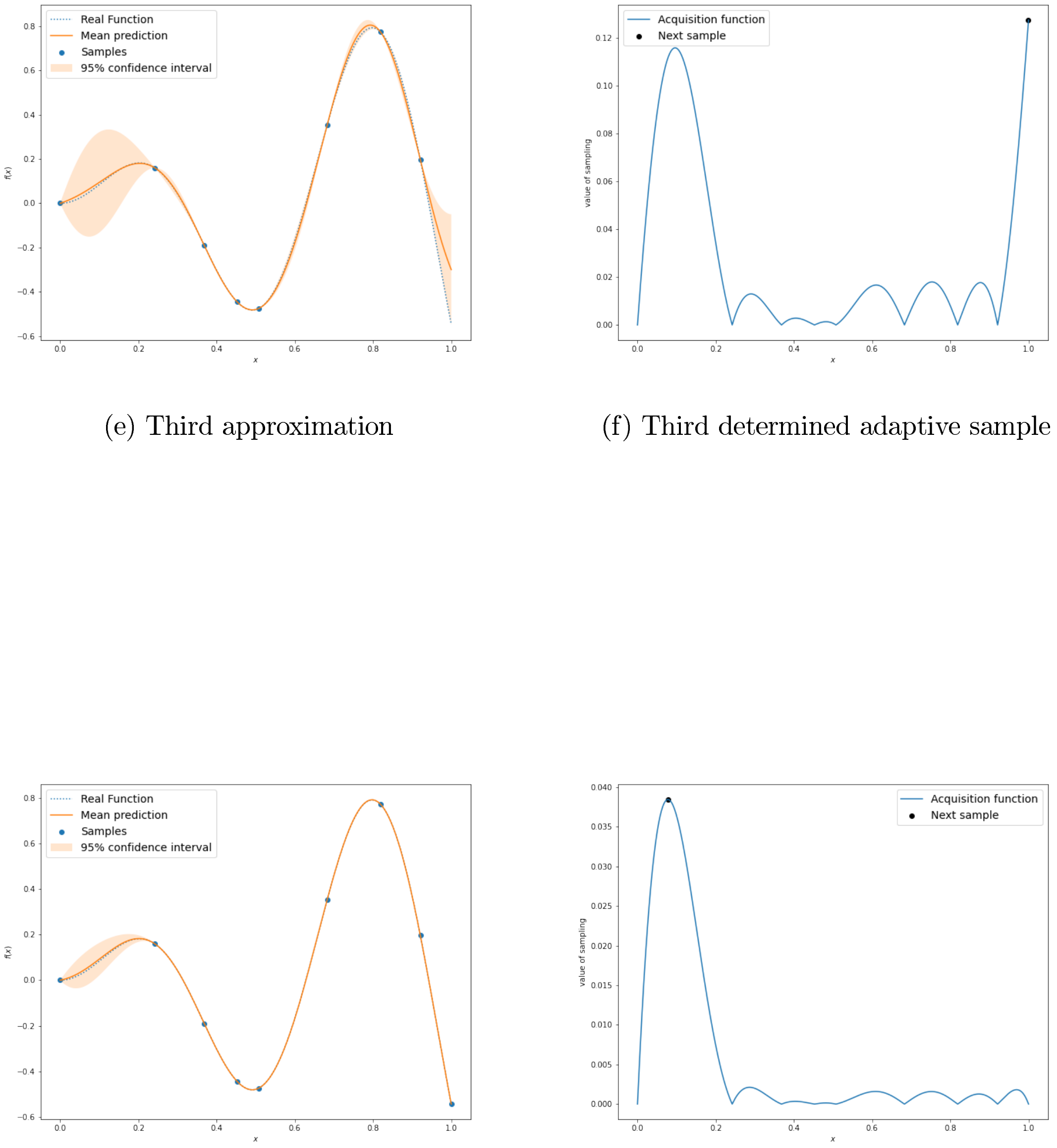


(a) First approximation

(b) First determined adaptive sample

(c) Second approximation

(d) Second determined adaptive sample



(e) Third approximation

(f) Third determined adaptive sample

(g) Fourth approximation

(h) Fourth determined adaptive sample

Figure 2.7: MaxMSE Acquisition function [19] [5]

As can be seen, the two acquisition functions differ in terms of their specific positions for new samples. For the EI acquisitions function, the position where the upper limit of the confidence interval is the largest is always chosen. This leads accordingly to the formation of clusters of samples at these extreme points.

With the MaxMSE acquisitions function, on the other hand, the position where the distance between the upper limit of the confidence interval and the lower limit of the confidence interval is greatest is chosen as the next sample position. This leads to the fact that the new samples do not form clusters but are widely distributed.

For the application of this method, a ratio between MaxMSE and EI samples has to be defined as these are two separate criteria.

3 Analysis of the requirements

The requirements for the method to be developed are described in the following chapter. The list of requirements is based on the requirements set by the company. In addition, there are requirements that can be formulated on the basis of the available information about the simulation tools, without being limited to specific tools.

3.1 Stability

Stability is the most important requirement for any method deployed in the context described in this thesis. Without stability there is a high risk of wasting a significant amount of time, money and computing resources.
In the worst case scenario there are no more resources to redo an DoE. In this case the badly designed DoE leads to a surrogate model with poor or unusable assessments of system properties.
There are two basic mechanisms of effect that need to be considered in terms of stability.

3.1.1 Stability of the DoE method

The first mechanism of effect is the execution of the adaptive DoE method itself. In this case, the problem is that the method is not able to distribute the adaptive samples to the desired areas of the design space based on the defined conditions. This can have various causes. For example, the initial information given is not sufficient or the system behavior is too complex.
In the best case secenario, the adaptive samples are distributed close to randomly within the design space. This implies that the advantages of adaptive sampling are no longer exploited. In such a case, however, it can be assumed that the information content of

the DoE does not deteriorate significantly compared to an LHS sampling.
In the worst case secenario, one or more clusters of samples are formed in areas of the design space that are not of appropriate importance. Since this leads to a decrease in the amount of samples in important areas of the design space compared to an LHS sampling, it can be assumed that the information content of the DoE deteriorates significantly.
This stability requirement is verified by performing tests in which the corresponding areas are known. More detailed information on this can be found in the Section 3.5.

3.1.2 Stability of the improvement for the ML-model

The second mechanism of effect is the ability to place the adaptive samples in the wanted position based on the defined conditions. The problem here is that the conditions do not match the paths of the design space where an ML based surrogate model needs adaptive samples.
It is also possible that the conditions match the paths of the design space where an ML based surrogate model needs adaptive samples for some system behaviors but not for all.
This stability requirement is verified by running tests and then applying the resulting data sets for training ML models. These data sets are compared to an LHS benchmark and the resulting models are compared with respect to their MSE. More information on this can be found in Section 5.3 and Section 5.5.

3.2 General applicability

The method developed for data generation should be applicable to various physical problems. It is explicitly not about optimizing a method for a specific application. The aim is to develop a method that produces a data set that is superior to the current standard on an unknown system.
To ensure this, the method is first tested and optimized on suitable test functions [14] [10] [12]. Testing on a real application is only performed when results for these general test cases are available.

3.3 Calculate more than one sample position at once

A lot of simulations with high computing cost are able to run in parallel, i.e. it is for instance possible to simulate one sample position per core of a CPU. For this reason, it is more efficient to not run a single sample position each time, but a stack of them. This is why it is important that the method is able to determine more than one sample position at once. It is also an explicit requirement of the company.

3.4 Short execution time

A short execution time for the selection of the next samples is also important. A lot of simulations are running on highly expensive computers. It can be assumed that the calculation of the positions of the next samples is much less computing expensive than the simulations themself. But a lot of the simulations run parallel on multiple cores. This implies that while the position of the next samples is calculated a highly expensive computer is not fully utilized. To reduce this time the Method has to be fast.

In this context, fast means that the amount of information gained per time must be higher than in the benchmark, despite the additional computing effort required to determine the position. The goal is to generate a data set in the same time that leads to a lower MSE than the benchmark when training an ML model. The time needed to determine the sample positions must therefore be overcompensated by the information content of the samples.

3.5 Advantage over LHS sampling

As described in Section 2.1.4 the LHS sampling is the currently used method for DoE generation. It is already good in terms of design space exploration and detection of non-linearities. For the extra effort of adaptive sampling to be worth, there must be a measurable advantage. There are two areas in which this advantage can arise.

Reduction of the amount of required data

If it is possible to reduce the amount of data without a change in the behavior of the surrogate model, a reduction of the used resources.

Reduction of the error

The second parameter by which the improvement of a DoE can be evaluated is the reduction of the error. In terms of Surrogate models in addition tho the MSE the Mean Absolute Error (MAE) and mean percentage error are used as well. A reduction of these errors without increasing the amount of used data is equivalent to increased model quality for the same coast.

3.6 Applicability to high-dimensional problems

Since the method is not designed for a specific application, neither the number of input or output dimensions is known. However, in most cases it can be assumed that both the input vector and the output vector are high-dimensional. Accordingly, it is important that the method created is robust with respect to high dimensional space. The highest known use case has 26 inputs and 14 outputs.

3.7 Cost reduction

A lot of the requirements in the previous sections are leading to an improvement on the resource usage. However, this does not necessarily lead to a reduction in costs. In order to reduce costs, the savings achieved must at least outweigh the costs incurred through development, staff costs and, if applicable, costs for licenses. As it can be induced, the higher the margin between the savings and the development cost, the more business attractive the method will be.

3.8 Balance of the sample distribution

As explained in Section 3.5 the aim of the method is to generate a significant advantage over the LHS sampling. To achieve this the distribution of the samples must be performed in a way that it is optimal for the given ML model. Since the physical behavior of the system is assumed to be unknown, the method must be able to find the right balance between clustering in important areas, and even by distributing the samples in the design space automatically and without the requirement of manualy setting the hyperparameters.

4 Conception

In this chapter, the methods discussed in Chapter 2 will be compared. They will be evaluated on the basis of the criteria presented in Chapter 3. Finally, the method to be implemented will be determined on this basis.

Since the Bayesian optimization method and the Maximum Mean Squared Error and Expected Improvement function Sampling Criteria differ only slightly in terms of model topology, these methods will also differ only slightly in terms of their applicability. The acquisition functions used in each source are different, but there is no reason why they could not be applied in the respective other model. The use of several acquisition functions in alternation is also possible in both model topologies.

4.1 Stability

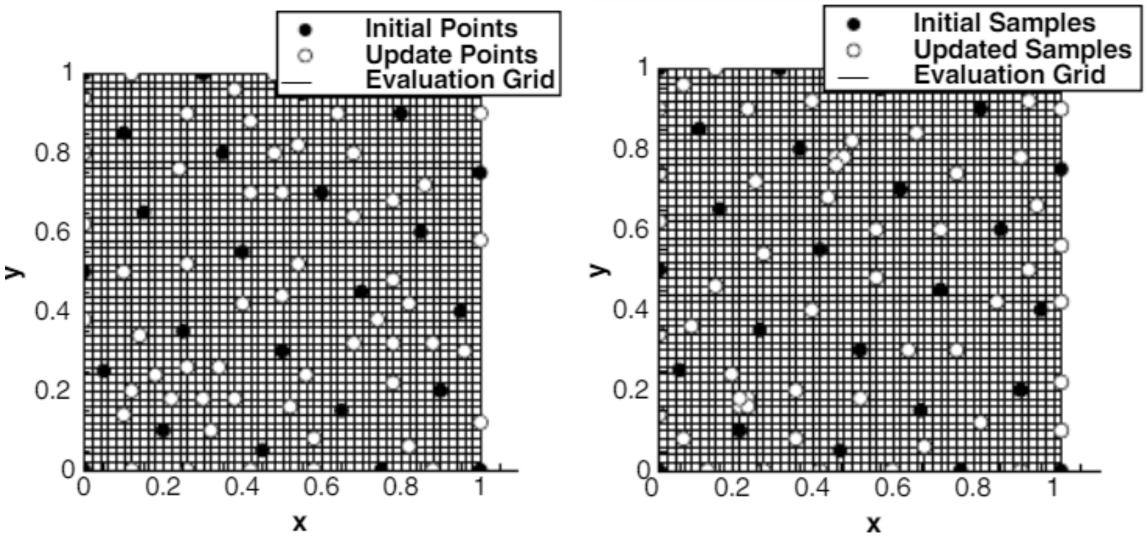
Stability of the DoE method

Regarding the stability of all models, it must be taken into account that there are only a few samples at the beginning of the adaptive process. Regardless of the model topology, this leads to the risk of not being able to represent the behavior of the system sufficiently accurately. This aspect has the same effect on all the methods under consideration in this thesis.

Another important point to consider regarding model stability is the course of dimensionality. This will be discussed in more detail in the Section 4.6.

Stability of the improvement for the ML-model

The sources that discuss adaptive sampling aim to improve the existing model (usually RBF or Kriging) with the new samples. The methods also find points of improvement for their respective model topology. Depending on the underlying behavior of the system,



(a) RBF sampling

(b) Kriging sampling

Figure 4.1: Sampling behavior[14]

sometimes one method or the other is better suited to reduce the error. [14]
Therefore, it is not possible to deduce which method is better suited to reduce the error of an ML model. However, there are clear differences in the way the adaptive samples are distributed. Figure 4.1 shows how each method distributes its adaptive samples in the design space. The corresponding sampled system behavior is the one represented in 4.1, which is shown in Figure 4.2 for better understanding. [14]

$$\begin{aligned} f(x,y) = \frac{3}{4}e^{-\frac{1}{4}((9x-2)^2+(9y-2)^2)} + \frac{3}{4}e^{-\frac{1}{49}(9x+1)^2-\frac{1}{10}(9y+1)^2} \\ +\frac{1}{2}e^{-\frac{1}{4}((9x-7)^2+(9y-3)^2)} - \frac{1}{5}e^{-(9x-4)^2-(9y-7)^2} \end{aligned} \tag{4.1}$$

Although, it can be seen that both methods found the extreme points of the function and generated samples in these area, the way in which these samples are distributed in the area of the respective extreme point is different.
In the method based on the Kriging model, very compact clusters are formed. This is due to the fact that the EI acquisition function used for this purpose uses the information of the newly generated samples to place the following sample even closer to the extreme point. With the method based on the RBF model, however clusters are located in the area of the extreme points, but these clusters are much less compact. This is due to the

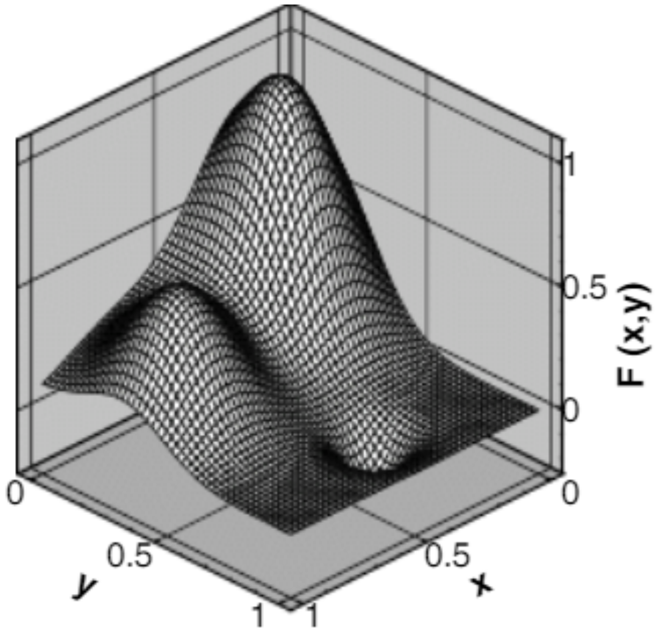


Figure 4.2: Function 4.1 [14]

fact that the probability of a new sample being placed at a position is decreasing with a low Euclidean distance to the next known sample.
Based on the DoE principle that samples must be significantly different from each other in order for them to contain new information, it can be assumed that as in the RBF-based model the samples are less compactly clustered. This leads to the fact that this method contains more information than the Kriggin based one. Furthermore, in-house research has shown that ML models built only on samples located at extreme points have very limited predictive quality with respect to the overall design space. Evaluating the performance of the ML model based only on the extreme point is however just a test performed in order to get an indication on the effect of these sampling methods, but it does not provide a realistic case that can serve as a solid benchmark for a final decision making on the methodology. A more detailed discussion of this topic can be found in Section 4.5

4.2 General applicability

In terms of general applicability, none of the methods shows a significant advantage over the others. All of the methods are widely used in science for various different problems and all of them can be applied without more initial information than a predefined number of initial samples.

4.3 Calculate more than one sample position at once

The methods based on the evaluation of an acquisition function depend on the information of the new sample. Therefore, using several different acquisition functions leading to different sample positions could be considered as a possible solution. This method, however, does not solve the problem to the required degree. Since the acquisition functions always generate samples in the region that serve a specific purpose, exploration of the non-linear regions, or exploration of the most unknown regions, they are not equally suitable for reducing the MSE of ML models. Furthermore, for non ML models the best results were obtained when the acquisition functions were used in specific ratios [14]. Arbitrary use to vary the stack size should therefore be avoided.

In the method based on the combined Lapace and square to the power function, it is more feasible to vary the stack size. Firstly, it can be assumed that the POS algorithm will find different local minima in different runs. Additionaly, even if the output of a sample is not yet known, it is possible to take the position of the sample into account with the square to the power function. This ensures that all positions determined at once are different and the size of the stack can be chosen freely. However, this leads to the fact that with a large stack size many positions with the same amount of information have to be determined. It can therefore be assumed that the quality of these positions is reduced with increasing stack size. Nevertheless, it is possible to vary the stack size. How this affects the information content of the samples must be investigated.

4.4 Short execution time

In all methods, the most computationally intensive step is the process of building the model which is used to determine the position of the next sample. In this respect, the methods are to be considered equivalent.

However, the possibility of determining more than one sample per model created, as described in the Section 4.3, has the advantage that in the method based on the combined Lapace and square to the power function does not require a new model for each sample.

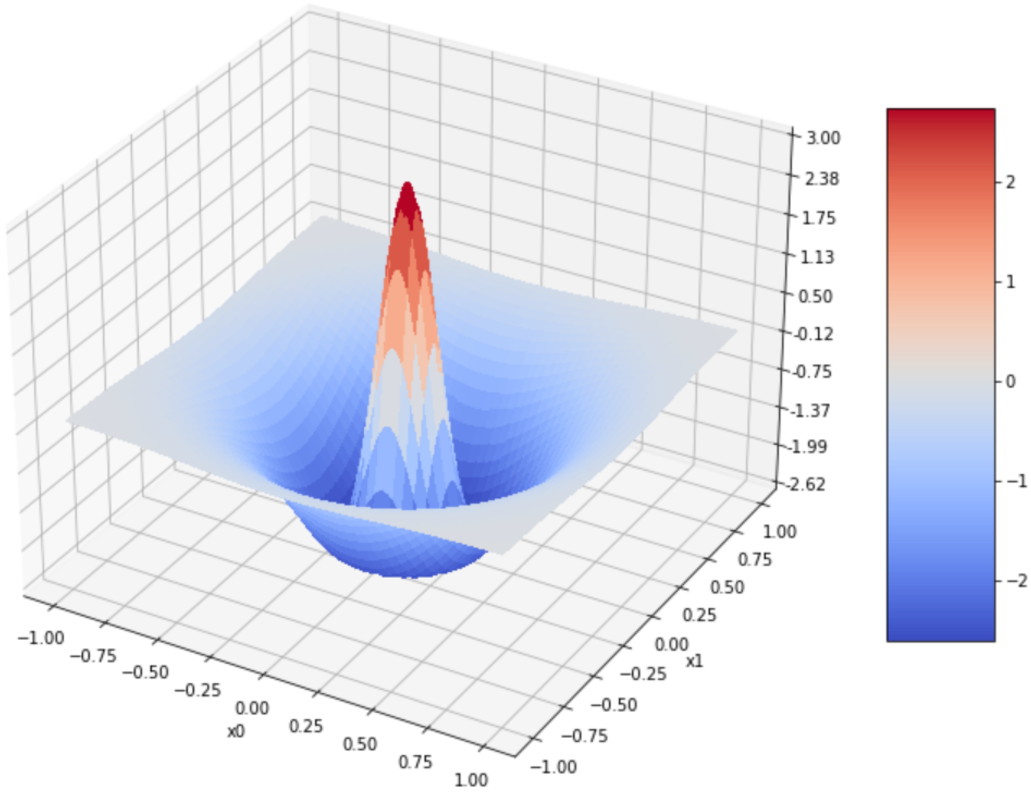
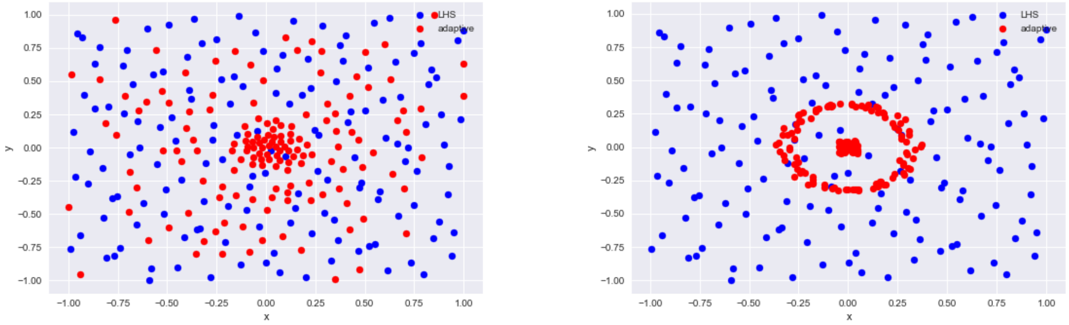


Figure 4.3: Function 4.2

4.5 Advantage over LHS sampling

In order to be able to make reliable statements about which method is better suited for training an ML model, a first preliminary test is performed in which the sampling behavior of the methods is simulated. For this purpose, 150 initial LHS samples are combined with 150 samples that simulate the adaptive sampling behavior. The sampling function used here was the function shown in Equation 4.2 (Figure 4.3). This function is complex enough to show the different performance of ML models and the non-linearities are distributed in such a way that the "adaptive" samples can easily be positioned manually, to simulate the behavior of the adaptive sampling methods, without the need to implement them [14].

$$f(x, y) = -4e^{-\frac{25}{8}(x^2+y^2)} + 7e^{\frac{125}{4}(x^2+y^2)} \tag{4.2}$$



(a) RBF like sampling

(b) Kriging like sampling

Figure 4.4: Sampling positions

The respective distribution of the samples in this experiment is shown in Figure 4.4. The values of the MSE achieved by the ML models on the respective data set are shown in Table 4.1. To be able to evaluate the values, the result of a 300 LHS data set is also entered here. In order to show the statistical mean value that the ML models achieve with the respective data set, the MSE value is the mean value of 10 random forests, each consisting of 10 decision trees.

The result of this experiment is that both methods, the Kirging-based and the RBF-based, are able to improve the MSE of a ML model. However, the distribution of the samples that is determined by means of the combined criterion is by far the best method.

Method	MSE
RBF, combined criterion	0.05801
Kriging, acquisition function	0.1417
LHS	0.1619

Table 4.1: Comparison of the MSE

4.6 Applicability to high-dimensional problems

All possible topologies for surrogate models are subject to the course of dimensionality. This describes the problem that a model with limited data and computation time is not able to approximate the behavior of a system sufficiently accurately for the application. However, the extent of the impact that the high dimensionality of a problem has on the

model of a method differs.
Bayesian Optimization is an optimization method that is best suited for low dimensional applications. Depending on the source, this involves up to 20 dimensions. With this scale, it is already possible to use the method for many of the surrogate models based on simulation data. However, in some cases, systems with a larger number of dimensions may have to be considered as well. [9] [17]
RBF and Kriging models can both be used for higher dimensional systems. In a direct comparison, however, an RBF model is always superior to a Kriging model as the number of dimensions increases. [22]
To counteract the course of dimensionality, there are methods for various model topologies, such as model order reduction.[14] [13]
However, since it is not clear how these affect the ability to determine adaptive samples, they are out of the scope for this thesis.

4.7 Cost reduction

Since all methods are of comparable complexity and it can be assumed that they can be implemented without incurring licencing costs, they are to be assessed as equal with regard to this point.

4.8 Balance of the sample distribution

As already described in Section 4.5, it can be assumed that the methods based on acquisition function have a worse distribution of the samples than the approach using the combined criterion, when it is about reducing the MSE of an ML model. This is due to the fact that the combined criterion is able to distribute each sample optimally in terms of on the one hand position in the design space and on the othe hand distance to the known samples, and it does not switch back and forth between these two properties.

4.9 Summary

Based on the preceding discussion of the advantages and disadvantages of the methods, the method based on the RBF model is implemented with the adaptive samples using the combined criterion.
The summary of the arguments underlying this decision are summarized in Table 4.2.

	Bayesian Optimization	RBF, combined criterion	Kriging, acquisition function
Stability	0	+	0
General applicability	+	+	+
Calculate more than on sample position at once	-	0	-
Short execution time	0	+	0
Advantage over LHS sampling	+	++	+
Applicability th high-dimensional problems	0	++	+
Cost reduction	+	+	+
Balance of the sample distribution	0	++	+

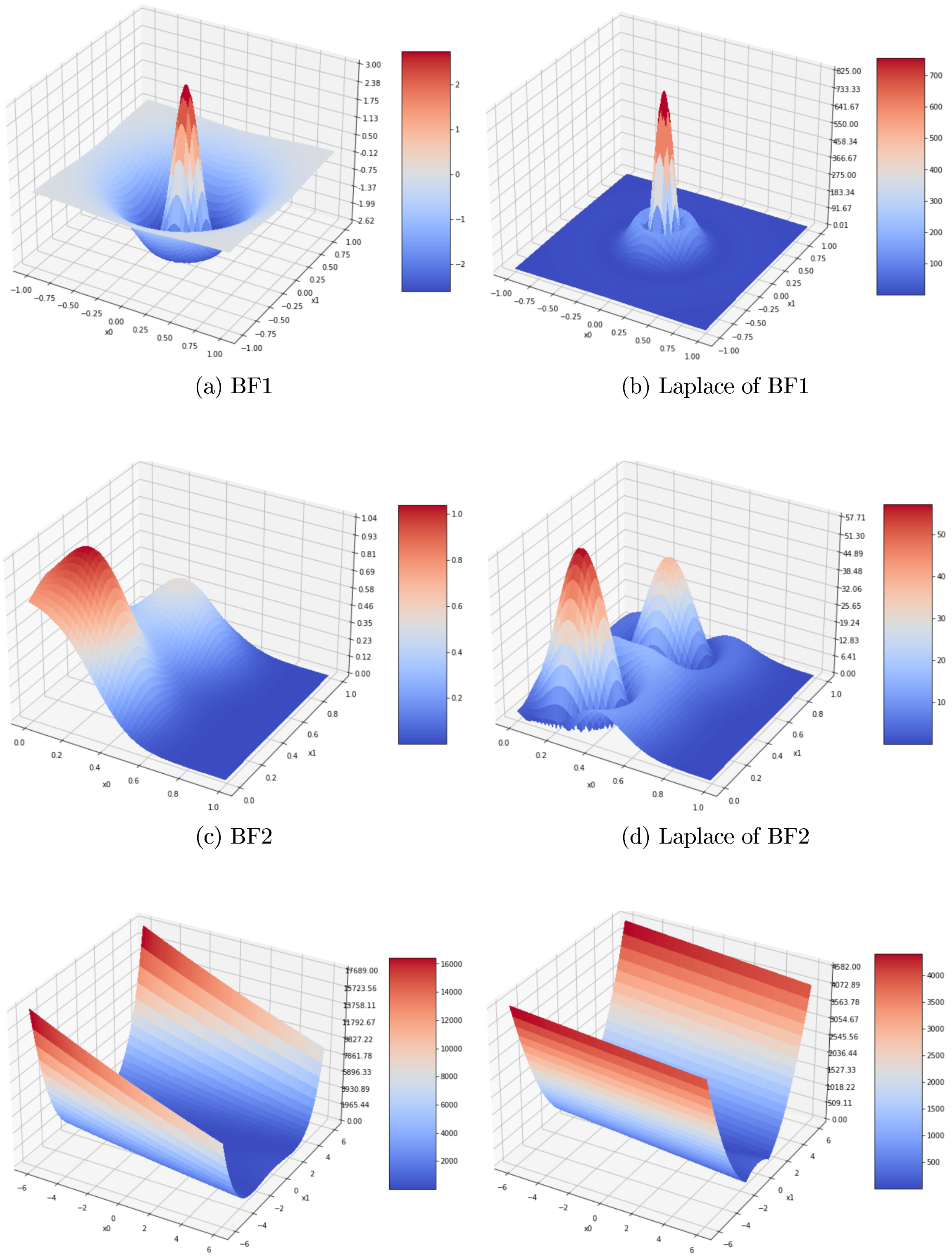
Table 4.2: Comparison of the methods

5 Development of an adaptive DoE method

The following chapter describes the development and implementation of the previously defined method. In order to explore the properties of the method a system behavior is simulated by means of a synthetic functions. This has the advantage over a black box function that the true function is known. Through this, arbitrarily large test data sets can be used, which leads to the most accurate analysis of the behavior of the method and the resulting ML model in the entire design space. This makes it possible to investigate the mechanism of action between the synthetic function and adaptive sampling as well as between adaptive sampling and the ML models. In addition, data points can be generated so quickly that the time required for this is no longer significant in relation to the time used for adaptive sampling. This means that if the method is sufficiently optimized, a large number of tests can be performed, which is important in order to gain as much knowledge as possible about the processes and applicability of the method. Based on this knowledge, It is possible to optimize the procedures and the hyperparameter of the method. For the first tests, 2D functions are used, as they have the advantage, that they can be visualized in an easily understandable way.
The used Benchmark Functions (BF) are shown in Table 5.1. The surface of the functions and the surface after applying the Laplace operator are shown in Figure 5.1. These are common test functions for the use case of adaptive DoE. [14] [10] [12]

$f_1(x_0, x_1) = -4e^{-\frac{25}{8}(x_0^2+x_1^2)} + 7e^{\frac{125}{4}(x_0^2+x_1^2)}$	$x_0 := [-1, 1], x_1 := [-1, 1]$
$f_2(x_0, x_1) = \frac{3}{4}e^{-\frac{1}{4}((9x_0-2)^2+(9x_1-2)^2)} + \frac{3}{4}e^{-\frac{1}{49}(9x_0+1)^2-\frac{1}{10}(9x_1+1)^2} + \frac{1}{2}e^{-\frac{1}{4}((9x_0-7)^2+(9x_1-3)^2)} - \frac{1}{5}e^{-(9x_0-4)^2-(9x_1-7)^2}$	$x_0 := [0, 1], x_1 := [0, 1]$
$f_3(x_0, x_1) = 10(x_1 - x_0^2)^2 + (x_0 - 1)^2$	$x_0 := [-10, 10], x_1 := [-6, 6]$
$f_4(x_0, x_1) = 20 + (x_0^2 - 10 * \cos(2\pi x_0)) + (x_1^2 - 10\cos(2\pi x_1))$	$x_0 := [-2, 2], x_1 := [-2, 2]$
$f_5(x_0, x_1) = 2 + x_0^2 + x_1^2 - \cos(2\pi x_0) - \cos(2\pi x_1)$	$x_0 := [0, 2], x_1 := [0, 2]$

Table 5.1: Used benchmark functions



(a) BF1

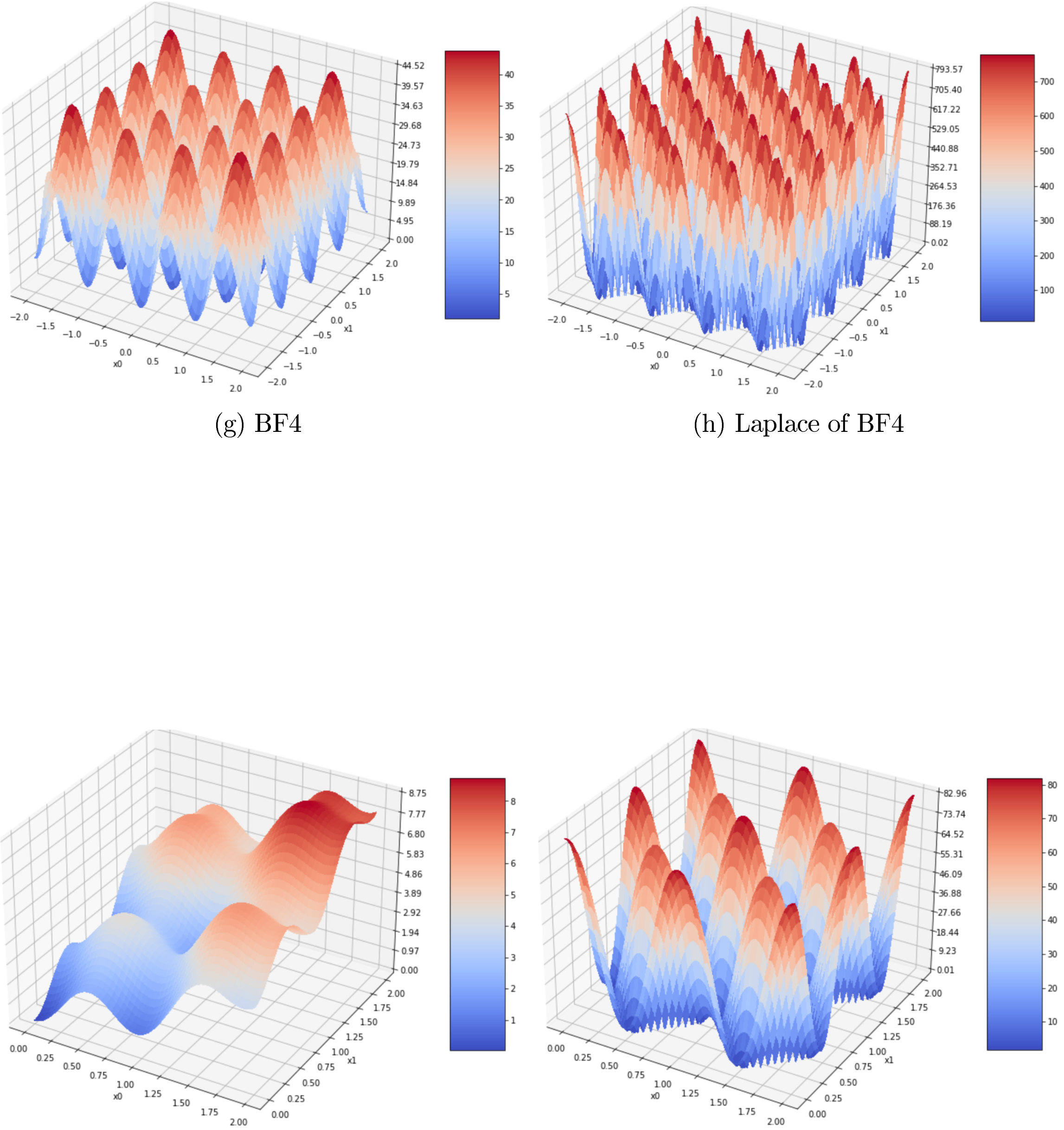
(b) Laplace of BF1

(c) BF2

(d) Laplace of BF2

(e) BF3

(f) Laplace of BF3



(g) BF4

(h) Laplace of BF4

(i) BF5

(j) Laplace of BF5

Figure 5.1: Used benchmark functions

In order to investigate which ML methodology can determine which data set set contains more information with as little variance as possible, RF and NN are compared with regard to the BF used in the Figure 5.2. In order to vary the amount of information, 100 data sets logarithmically distributed in the range of 10^2 to 10^4 have been generated. The sample positions were determined using the LHS algorithm, which ensures that the distribution is the same regardless of the amount of data. Accordingly, the amount of information correlates with the amount of data.

The RF model consists of 100 decision trees (DT) and has been created created using scikit-learn with the default set of hyperparameters available in the package for RFs. [19].

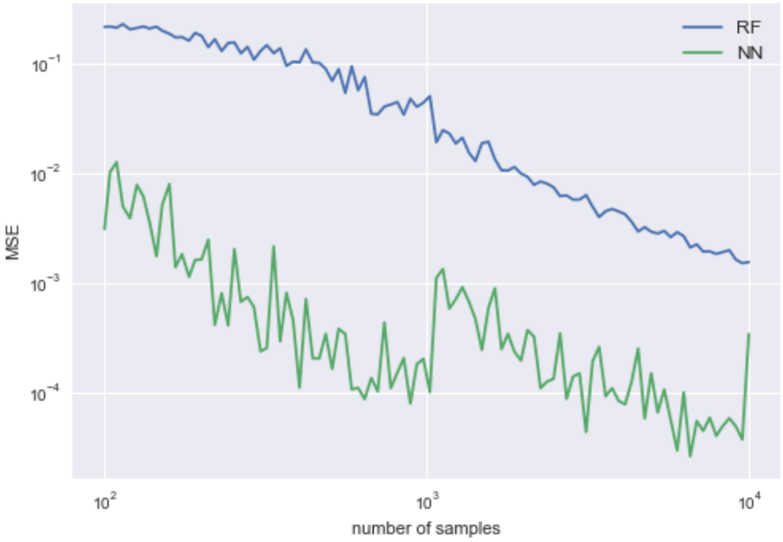
The NN model has been tranied with the hyperparameters described in Table 5.2. The PyTorch library has been used for this purpose [18].

In order to reduce the variance of both models, the mean is evaluated 10 times over the same data set, where each time the train and test split is randomly picked.

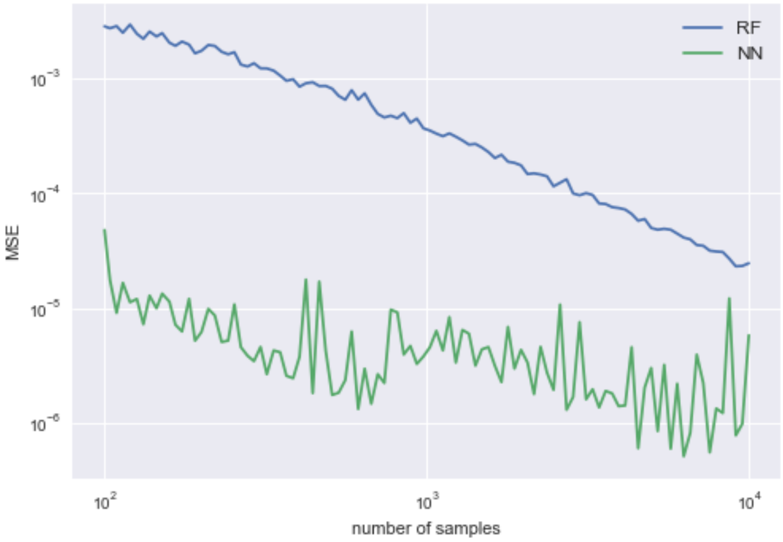
The NN achieve better MSE values for almost all the BF cases, except a few exceptions, but these values are subject to much stronger noise than the RF. Since the aim is to optimise the data set and the ML model is only used for evaluation, the behaviour of the RF is more suitable for the use case. Accordingly, RF are used for the following investigations.

Hyperparameter	Value
neurons_per_layer	40, 40, 40
activations	relu, relu, relu, linear
optimizer	adam
loss	mse
batch_size	256
learning_rate	0.01
epochs	500

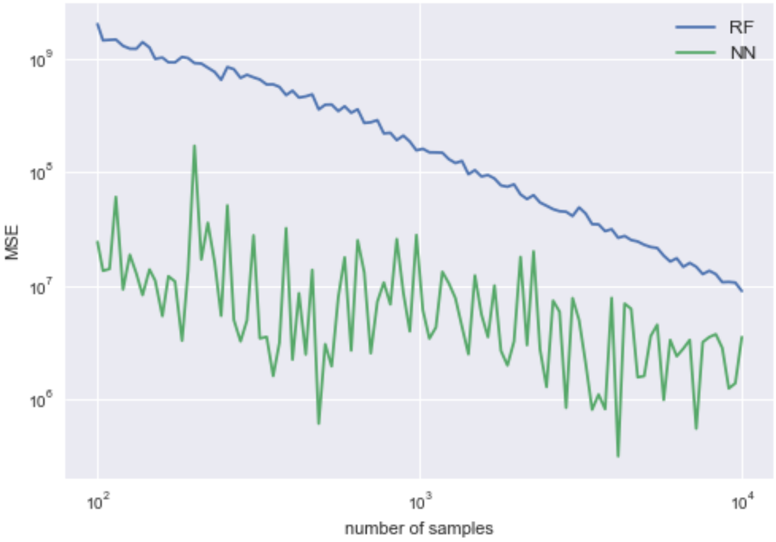
Table 5.2: NN hyperparameter


RF
NN
MSE
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{2}
10^{3}
10^{4}
number of samples

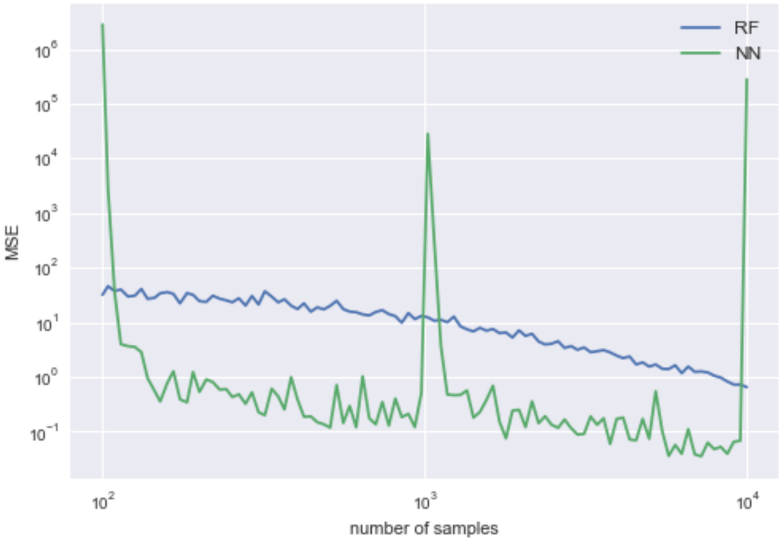
(a) BF1


RF
NN
MSE
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{2}
10^{3}
10^{4}
number of samples

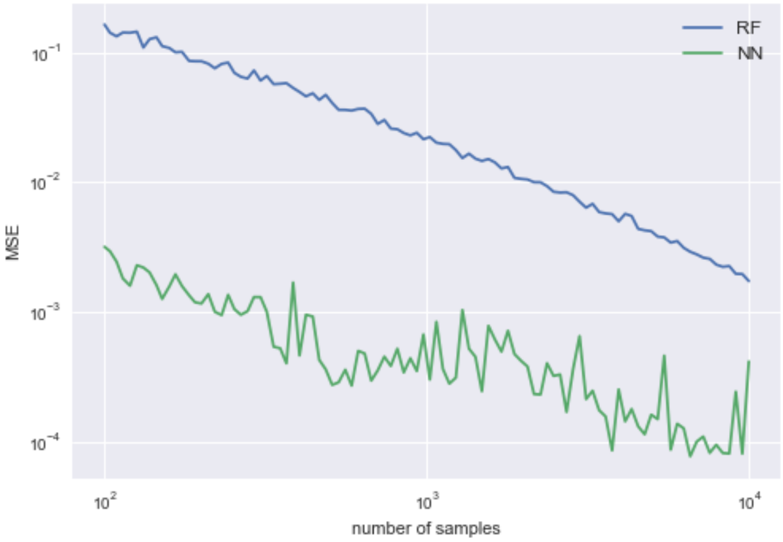
(b) BF2


RF
NN
MSE
10^9
10^8
10^7
10^6
10^2
10^3
10^4
number of samples

(c) BF3


RF
NN
MSE
10^6
10^5
10^4
10^3
10^2
10^1
10^0
10^{-1}
10^2
10^3
10^4
number of samples

(d) BF4



(e) BF5

Figure 5.2: Comparison of ML topologies for the BFs

5.1 First implementation to test feasibility

As already described in Section 5, various synthetic functions are used for testing purposes instead of a black box. The package sciPy is used for this purpose.

To generate an initial data set with which to build the RBF model, a predefined number of LHS sample positions (x_0, x_1) are created using openTURNS [4]. These inputs are then evaluated using the previously defined function so that the samples are complete (x_0, x_1, y_0).

The next step is to determine the reference points for the RBF model. They are also determined with openTURNS using a Halton sequence.

Based on the values of the initial samples and the position of the reference points, the weights are calculated. The procedure described in Section 2.2.1 is used for this.

Once the weights are calculated the position for the the position of the next sample is computed. For this purpose the output of the Laplace operator of the RBF approximation is needed. Hence, the Gauss function shown in Table 2.2 is used to calculate the Laplace of a single RBF is function 5.1 is used for the Laplace approximation.

$$\nabla^2(\alpha, r) = 4\alpha^4 r^2 e^{-(\alpha r)^2} - 2\alpha^2 e^{-(\alpha r)^2}, \tag{5.1}$$

where α is the shaping factor and r is the Eucluide distance between the actual input of the function and the reference point. The total Laplace value that results for the current input corresponds to the sum of all outputs of all Lapace RBF.

The second value needed for the combined criterion is the smallest Eucluide distance between the input of the function and one of the already known collection points. An optimum is then searched for on the combined criterion (equation 2.18, with $\hat{y} = 0$) using PSO. When setting the hyperparameters of PSO, the conditions of the expected surface must be taken into account. In this case it is to be assumed, that the surface of the function is strongly discontinuous. This is due to the fact that it consists of the superposition of lapaceous functions that are heterogeneously distributed in different areas of the design space. Furthermore, the second part of the combined criterion ensures, that the value of the combined criterion is zero at the positions where already known samples lie and that the absolute value of the function increases strongly with increasing Eucluide distances to this samples.

An example of the surface behavior of the combined criterion for a two dimensions case is shown in Figure 5.3. In this example the combined criterion for an RBF approximation with 50 reference points and 150 already known samples is shown.

As it can be seen the surface of the combined criterion is really bumpy, as expected. These conditions make it difficult to find the absolute optimum. The problem is that the particles of the PSO may not be able to overcome local optima. This significantly reduces the probability of finding the global optimum. In order to achieve the best possible results, the hyperparameters of the PSO were set for these conditions and are shown in Table 5.3. These hyperparameter specifications are based on the source [6].

swarm size	30
generations	50
ω	0.9
c_1	0.5
c_2	0.3

Table 5.3: PSO hyperparameter settings

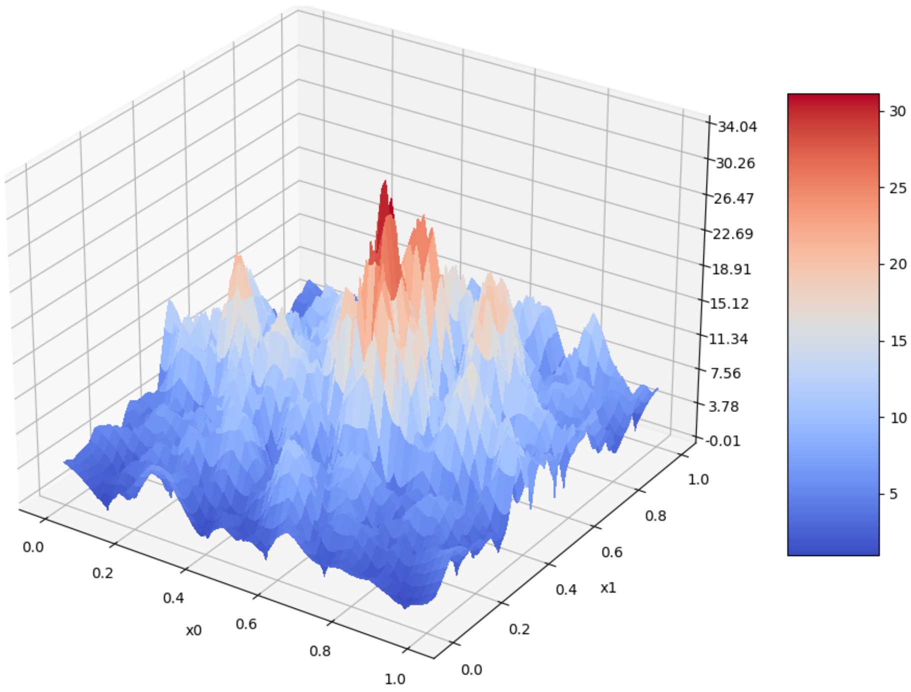


Figure 5.3: Combined criterion 2D surface

As it can be seen a large swarm size has been chosen. This increases the probability that one of the particles, or a group of them are in the area of the global optimum. The cognitive component and social component are chosen so that each particle is significantly more influenced by its own found optimum than by the global known optimum. This serves to ensure that all local optima found are sufficiently close to check if they correspond to the global optimum. Since it can be assumed that the particle that has found the known optimum is close to the actual global optimum, but it is not sure whether it will find this global optimum due to an obvious local optimum, a large number of generations is chosen. This ensures that the particles converge towards this known optimum in the course of the optimisation, despite the rather low social component. Since the particles approach the known optimum from different directions, it is likely that they will find an even better optimum, if one exists. The disadvantage of the aforementioned procedure is that both, a high number of generations, and a large swarm size increase the required computing costs. Especially with the number of generations this is critical because it cannot be parallelised due to the fact that the information of the previous generation is needed. To set up the PSO PySwarms is used.

After the position of the next sample has been determined, it must be evaluated. In the currently applied methodology, this is done by the evaluation of the known synthetic

function at this position. Later, a simulation is performed with these input parameters, which then provides the output values. The new sample is then added to the array of known samples.
With this increased amount of samples, an RBF model is then built again and the next sample is determined. This process is executed iteratively until the specified number of adaptive samples has been reached.
After this process is complete, the samples are saved as pandas data frame, but any other convenient tabular format could be used to store the data and further use it. The structure of the process described can be seen in Figure 5.4.

The basic functionality of the method has been tested on BF1 and some of the results are shown in Figure 5.5.

As can be seen, the method is already able to create new sample positions in the desired areas, compare with the maxima of the Laplace approximation in Figure 5.1b. However, there are some disadvantages of this method that do not yet make it optimal for the objective we want to achieve, i.e. creating as few points as needed and capturing as much information as possible. For example, some clusters of samples were created in areas that are not so relevant. Furthermore, it is noticeable that the largest cluster in Figure 5.5a is not centrically located in the area of non-linearity. In the next chapter, this analysis will be further deepened and the implemented method will be optimized on the basis of these findings.

5.2 Analysis and improvement

In this section a set of improvements designed based on the results and learnings from previous section are presented.

5.2.1 Normalisation of the design space

The normalization of the design space has a number of advantages. On the one hand, it corresponds to the standardized procedure recommended to be followed when designing a DoE, see Section 2.1.
In addition, this ensures that an optimal shaping factor can be searched for. With the method implemented so far, the shaping factor must be adapted to the size of the design

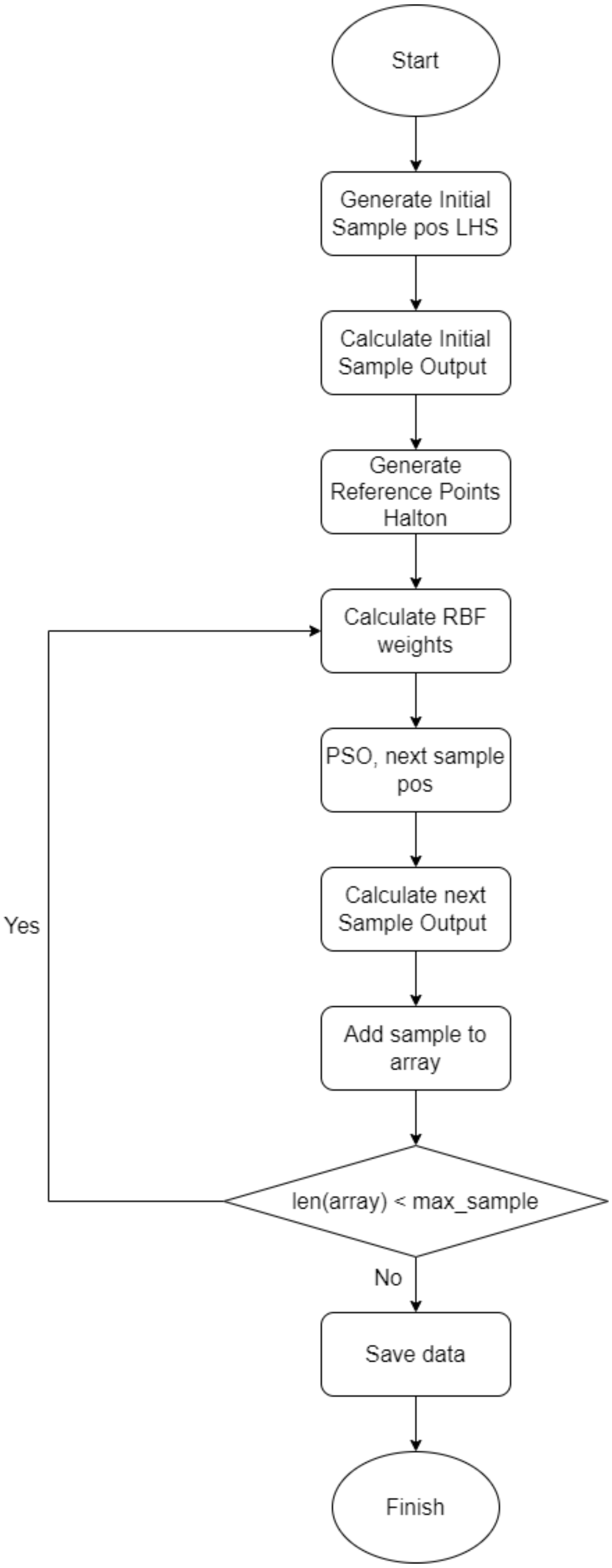
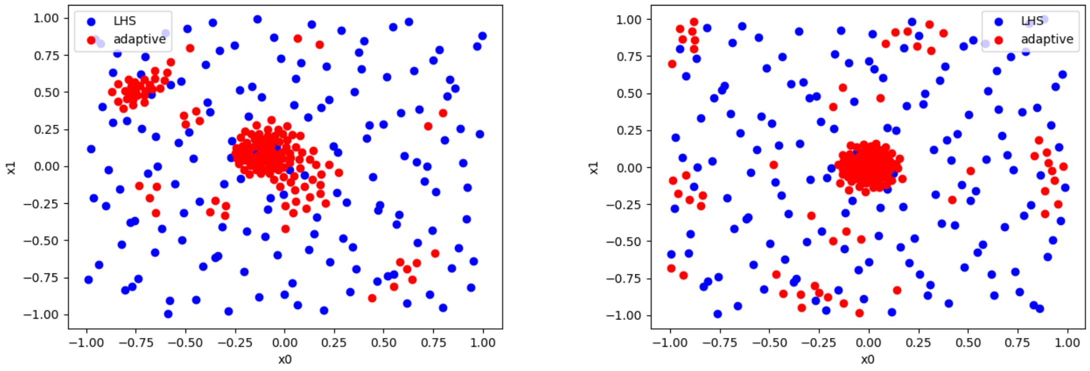

Start
Generate Initial Sample pos LHS
Calculate Initial Sample Output
Generate Reference Points Halton
Calculate RBF weights
PSO, next sample pos
Calculate next Sample Output
Add sample to array
len(array) < max_sample
Yes
No
Save data
Finish

Figure 5.4: Flowchart first implementation



(a) First results benchmark function 1, shaping factor := 6

(b) First results benchmark function 1, shaping factor := 3

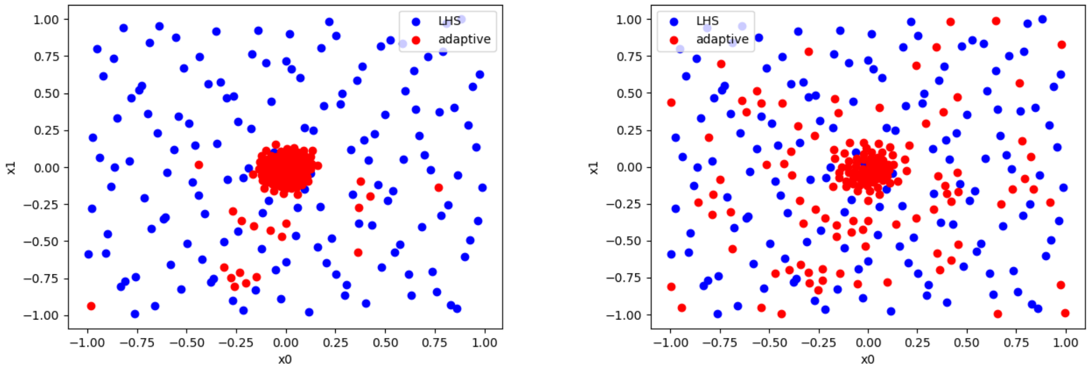
Figure 5.5: First results

space. This results from the fact that narrow RBFs are not able to completely cover a large design space. On the other hand, wide RBFs cannot represent the complexity of the system in a small design space.

Furthermore, it must be assumed that the design space does not always have the geometry of a hypercube, i.e. the individual dimensions do not have the same range of values. This leads to the fact that a good shaping factor cannot be chosen because, as already mentioned, the optimal shaping factor is different for different domains of definition. Since the unbiased data should be available for the creation of the surrogate models, only the inputs used for the RBF are normalized. The stored samples still correspond to the values of the synthetic function, or later to the values of the simulations. The domain of definition after the normalisazion for every dimension is [0, 1].

5.2.2 Offset

As seen in the figures presented in Figure 5.5 all samples are in compact clusters. To have a possibility to adjust the density of these clusters an offset parameter is introduced. This offset parameter is summed up on all values of particles of a given PSO run. The value corresponds to the product of the given factor and the value of the Laplace operator with the highest magnitude in the current PSO run. If, for example, an offset of 1 is selected, this means that in the extreme case where the Laplace operator at the position of a particle becomes 0, the value of the Laplace component is no longer 100% smaller than that of the maximum particle, but 50%. The order of evaluation of the particle



(a) Offset := 0.01

(b) Offset := 0.5

Figure 5.6: Effect of the offset parameter

positions of the Laplace operator remains constant. However, the evaluations are closer together in their amount. This concept leads to the fact that the density of the clusters can be adjusted with the offset parameter. The reason for this is that when the difference between the output of the Laplace operator is smaller, the density to the next sample becomes more important. Accordingly, small offsets lead to compact clusters and large offsets sample a larger area of the design space.
The values of the offset should always be greater than 0 to prevent the output of the Lapace operator from becoming 0 [14]. A visualization of the effect of the offset parameter is shown in Figure 5.6

5.2.3 Variable LHS reference points

As can be seen in the figures presented in Figure 5.5, it occurs that the clusters of the samples are principally in the correct area, but the position of the non-linearity is detected slightly displaced. This is due to the fact that the reference points of the RBF are always in the same place. The approximation becomes progressively better with an increasing number of samples, but the degree of non-linearity changes only slightly because for a specific point only the value of the RBF approximation is changed and the distance to the surrounding reference points as well as the shaping factor remains the same. Accordingly, for a fixed constellation of reference points, there are areas that tend to produce more and areas that tend to produce less of a large value with respect to the Laplace operator, regardless of the approximated behaviour.

The target is to distribute reference points in such a way, that they do not have such an influence on the Laplace operator. However, this is not possible because geometric constellations always form between the reference points regardless of which methodology is used to generate the DoE. However, it is conceivable that this behaviour differs for different stacks of reference points with respect to the position at which it occurs. The selection of a single different positioning of the reference points will therefore not lead to an improvement in this respect but only change the orientation of the offset of the resulting clusters. However, it is possible to change the positions of the reference points after each sample. This means that the effect of the geometric relationships of the reference points will be different for each sample. Assuming that the selection of each stack of reference points is so random that the effects of the geometry of the reference points result in mean-free noise, it can be assumed that the clusters form predominantly at the positions of non-linearity, since their influence on the Laplace operator is constant.
Since it is not possible to create such different reference point positions with the Halton sequence, the LHS algorithm is used. This makes it possible to create an almost unlimited number of different reference point distributions with good design space coverage with the same number of reference points.
Since a new RBF model is calculated after each sample anyway, the additional calculation time required is negligible.
The result of this procedure is shown in Figure 5.7. All the steps used to create this DoE are the same as those used for the DoE in Figure 5.5a. The only difference is that after each adaptive sample a new stack of reference points is chosen based on the LHS algorithm. As can be seen, the largest cluster created using this procedure is centrally located at the position of non-linearity.

5.2.4 Reference points outside the design space

As seen in Figure 5.5b, for example, the clusters that do not belong to the area of non-linearities are predominantly formed at the edges of the design space. This effect is particularly noticeable in the corners. This leads to the assumption that the RBF approximation is not sufficiently accurate in these areas. Considering that all reference points are within the design space, this is also conclusive. Inside the design space, each sample is always surrounded by several reference points that are located in different geometric constellations. This is no longer the case at the edge of the design space. A position that lies within the design space but is closer to its edge than the last reference points is presented to the approximation almost exclusively by these single RBF. Since

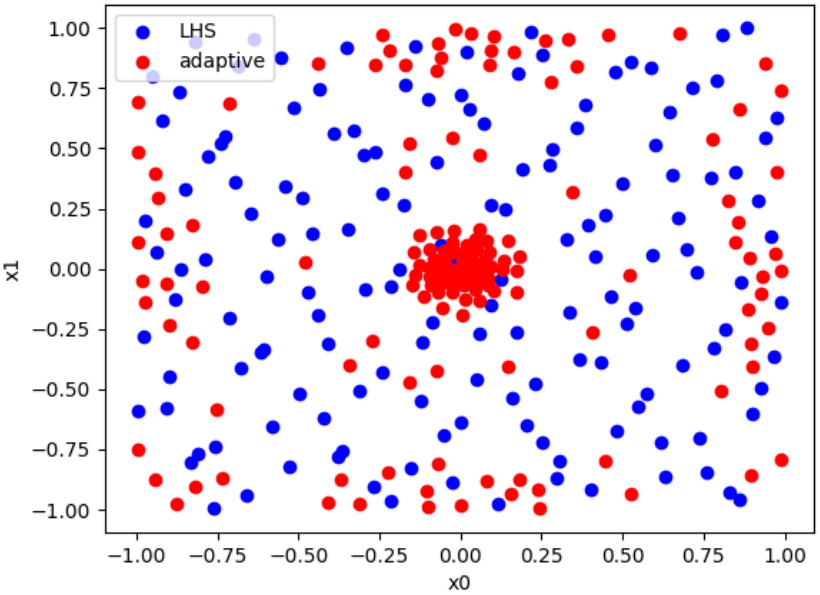


Figure 5.7: Function 1 with LHS reference points

almost only the RBF function of this single reference point has an effect on the approximation in this area, the representation of the curvature is also strongly limited.
To confirm this assumption an RBF approximation is shown in Figure 5.8. For this RBF approximation the same parameter and set of initial samples that lead to the results in Figure 5.5b are used. As it can be seen, the approximation is much more continuous in the middle of the design space than at the edges. In the corners of the design space the value of the approximation is highly different from the real function value. Furthermore it can be seen that the overall behavior of the approximation is different to the synthetic function in terms of value. This is not optimal but also not critical. For this method the most important thing is that the position of the non-linear areas of the approximation matches the non-linear areas of the investigated system.

In Figure 5.9a the Laplace operator for the RBF approximation in Figure 5.8 is shown. As it can bee seen the biggest value of the Laplace operator is close to the center of the design space. This is as well the position of the strongest non-linearity of the sampled function, so this behavior is as desired. There are a lot of areas with high values. For the moment the scope is on the areas at the edges of the design space, where no non-linearities

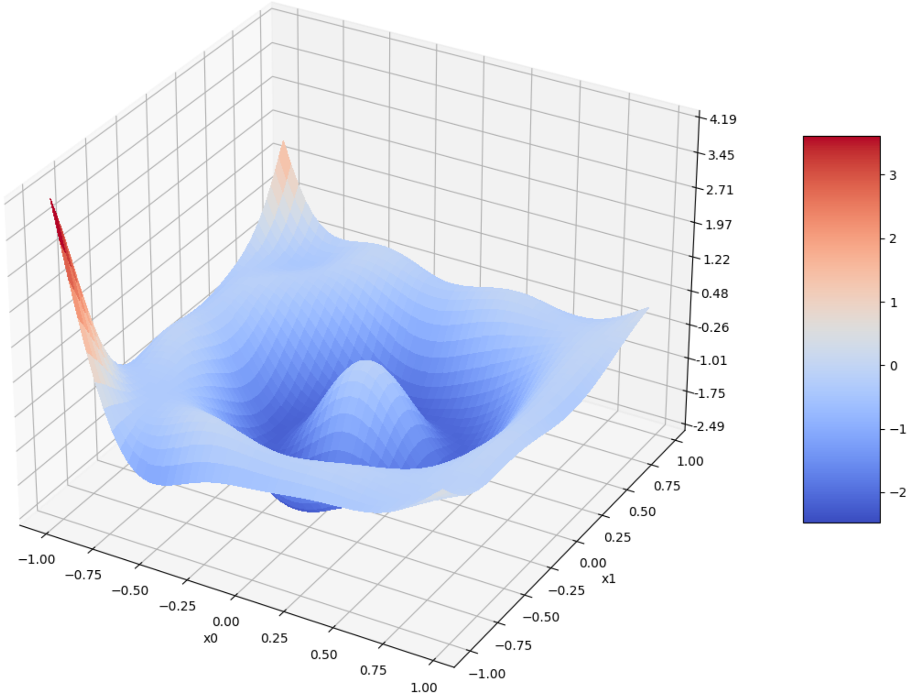
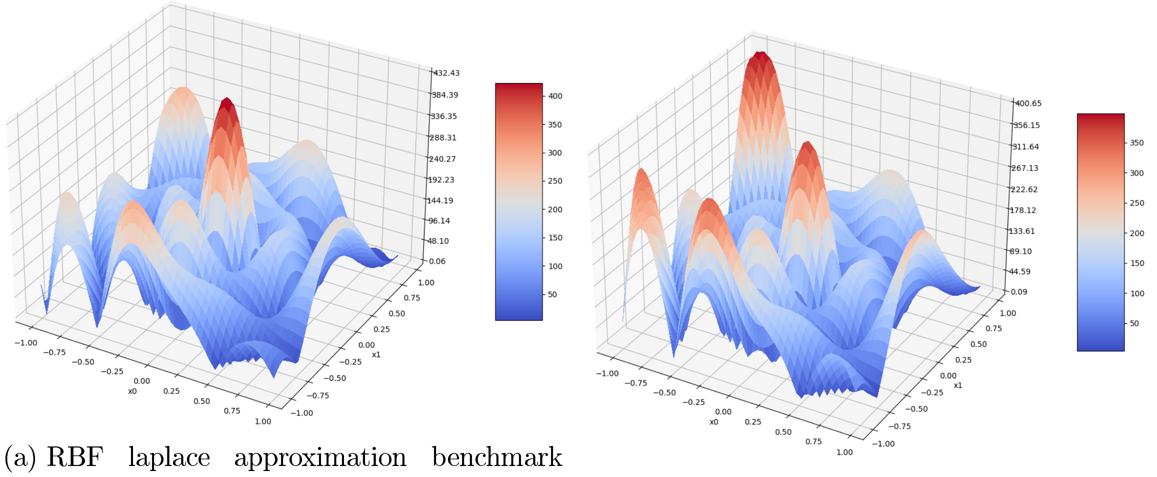


Figure 5.8: RBF approximation benchmark function 1

should be detected. Some of those areas are, $[x_0 \approx -1, x_1 \approx 1]$, $[x_0 \approx -0.25, x_1 \approx -1]$, $[x_0 \approx 1, x_1 \approx 0]$. All of the listed positions lead to clusters in undesirable places as shown in Figure 5.5b.

The results above might lead to the misleading conclusion that generating points that are slightly outside of the design space might improve the mathematical coverage of the DoE. Although creating those points prevents the effect that the outer samples are only significantly influenced by one RBF, the weights of the RBFs whose reference points lie outside the design space are now mainly determined by the outer samples. Accordingly, these RBFs no longer take the average of several samples as the ones further inside the design space do, but instead provide a precise approximation of the outer samples without generalising the function well. Applying this would lead to the Laplace value of the false maxima at the edge of the design space increasing even more, as seen in Figure 5.9b.

A functional solution to prevent the values of the Laplace operator from increasing so much at the edge of the design space is to choose the shaping factor in such a way that the RBF functions get a wider shape. This leads to the fact that the RBFs that are close to the edge of the design space do not change their value too much between their



(a) RBF laplace approximation benchmark function 1

(b) Reference points 10% out of design space

Figure 5.9: Effect of reference points out of design space

location and the edge. In addition, this also leads to more samples having a significant influence when calculating the weight of each RBF.

5.3 First test

To test the effectiveness of the implemented improvements and their effect on the capabilities of an ML model, tests are performed.

5.3.1 Setup

In the following, the resulting findings are presented. The content of this test run are all possible variations of the parameters in Table 5.4 with all functions from Table 5.1.

Offset	1%	20%	50%	-
Shaping factor	3	6	9	-
Number of reference points	10	20	30	40

Table 5.4: Test 1 parameters

The goal for the offset and shaping is to determine the value that leads to the largest improvement for the ML model. Hence, the design space is normed it can be assumed that there is an optimal shaping factor that is able to describe all functions sufficiently accurately. For the number of reference points the goal is to detect the minimum amount of reference points required to get stable results. The time needed to build the model correlates with the number of reference points to the power of 3 (for the used hardware $(0.0226 * n)^3$ sec), using as few reference points as possible results in the most time eficiente setting.

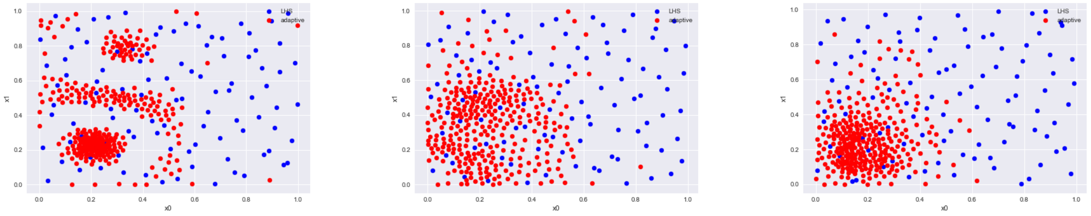
5.3.2 Results

The described test run leads to the conclusion that further optimisations must be performed. The behavior of the developed method as well as that of the ML models differs from the expected behavior.

Shaping factor

The most important information learned from the experiment is that it is not possible to search for a shaping parameter that is able to build a sufficiently accurate approximation for every function. The shaping parameter has to be fitted to the underlying behavior. It is not only that a badly chosen shaping factor leads to a worse result, but it is possible that a DoE fails completely. An example for this based on BF2 is shown in Figure 5.10, which displays the results from an offset of 1% and 40 reference points. The only difference is that the shaping factor is varied. It can be seen that a shaping factor of 3 leads to the best results in terms of the exploring of all the non-linearities. As the shaping factor values increase, the results get worse, as not all of the non-linearities are getting covered.

Another example, based on the results for BF1 is shown in Figure 5.11. Here the visual best result is achieved by the shaping factor 6. For a smaller shaping factor there are a lot of the adaptative samples outside of the circular non-linearity. For a larger shaping factor there is just a big cluster in the middle of the design space. For this BF a badly chosen shaping factor does not lead to an non functional DoE, but it still has a high impact on the results.

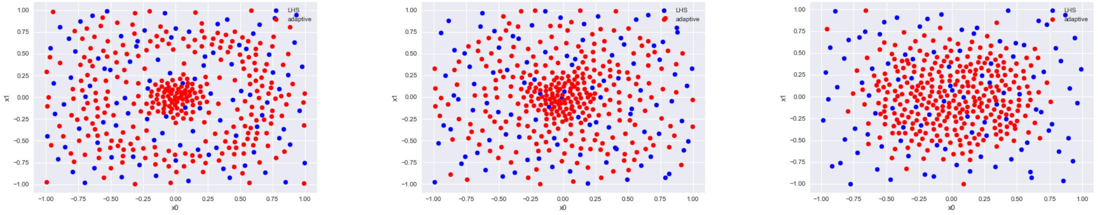


(a) Shaping factor 3

(b) Shaping factor 6

(c) Shaping factor 9

Figure 5.10: Effect of the shaping factor on benchmark function 2



(a) Shaping factor 3

(b) Shaping factor 6

(c) Shaping factor 9

Figure 5.11: Effect of the shaping factor on benchmark function 1

The fact that in most of the test runs of this experiment an inappropriate shaping factor has been used causes that many of the data regarding the other varied parameters are not very informative.

ML MSE

As explained in Section 2.2.3 and Section 5, for a good comparability of the results RF models are used. For all evaluations an RF model consisting of 100 decision trees has been created using the scikit-learn package with the default hyperparameter. [19]

In order to reduce the variance of the model, the mean is evaluated 10 times over the same data set, where each time the train and test split is randomly picked. Hence the behavior of the real function that is approximated by the surrogate model is known and samples can be generated quickly, 10,000 samples distributed by the LHS algorithm are used to test the model and calculate the MSE.

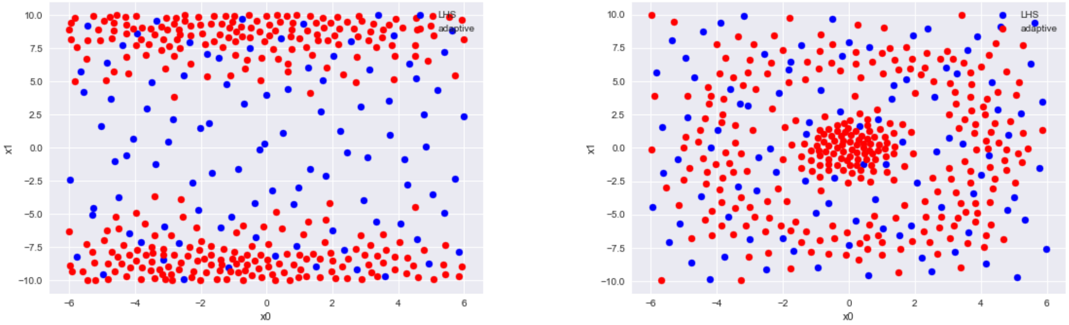
A brief summary of the results can be found in the Table 5.5. The evaluation of all data can be found in the appendix.

	BF1	BF2	BF3	BF4	BF5
LHS MSE	$1.556e^{-1}$	$1.575e^{-3}$	$1111e^{+9}$	$3.325e^{+1}$	$9.626e^{-2}$
Minimal adaptive MSE	$3.260e^{-2}$	$9.532e^{-4}$	$3.858e^{+8}$	$2.218e^{+1}$	$6.002e^{-2}$
Maximum adaptive MSE	$1.183e^{-1}$	$1.914e^{-3}$	$1.126e^{+11}$	$3.369e^{+1}$	$1.255e^{-1}$
Mean adaptive MSE	$5.939e^{-2}$	$1.352e^{-3}$	$1.925e^{+10}$	$2.668e^{+1}$	$9.131e^{-2}$
Coefficient of variation adaptive	43.55%	17.56%	216.9%	9.515%	16.28%
Mean adaptive MSE / LHS MSE	38.18%	85.87%	1733%	80.23%	94.86%

Table 5.5: Test 1 ML results

As it can be seen, the method developed for all BFs except BF3 shows an average improvement in MSE compared to the surrogate model trained with the LHS samples. It is noticeable that BF1 shows by far the most significant improvement. BF1 is the only function for which an improvement is always achieved, regardless of the parameterisation, albeit with a high specific variance with regard to this parameterisation.
For BF5, the average improvement of the MSE is negligible.
The developed method shows on average an extreme deterioration with in terms of the MSE for BF3. Nevertheless, there are parameterisations that lead to the creation of a data set that enables training a surrogate model with an improved MSE value. Figure 5.12 shows the data sets that have led to the greatest improvement and the greatest deterioration. The data set that led to the greatest improvement is parameterised with an offset of 1%, a shaping factor of 9 and 30 reference points (Figure 5.12a). The data set that led to the greatest deterioration is parameterised with an offset of 20%, a shaping factor of 3 and 20 reference points (Figure 5.12b).

As it can be seen in Figure 5.1f, the non-linear areas of BF3 are located at the edges of the design space where x_1 is maximum or minimum. These areas are well covered by the parameterisation that was applied to the data set shown in Figure 5.12a. This is not the case in the data set shown in Figure 5.12b, where the adaptive samples almost do not cover the extreme values of x_1 at all. Since BF3 not only has the non-linearity in this area, but also the highest values in terms of magnitude, a poor approximation in this area leads to a strongly deteriorated MSE. This confirms the assumption that ML-based surrogate models also benefit from a higher sample density in the non-linear domains.
The reason why the results are so poor on average is therefore not linked to the ML approach described in Section 3.1.2, but the instability of the developed adaptive DoE



(a) Best BF3 dataset Test 1

(b) Worst BF3 dataset Test 1

Figure 5.12: Data set comparison BF3 Test 1

method itself described in Section 3.1.1. This instability is caused by an inappropriate parameterisation.

The 6 worst results are significantly worse than the rest of the results. It is particularly interesting to observe that each of these six results form clusters of adaptive sampling that are in the middle of the design space. Furthermore, these 6 data sets are all generated using an offset of 20% and a shaping factor of 3 or 9. A more detailed presentation of the results regarding the shaping factor is shown in the Table 5.6, regarding the offset in the Table 5.7.

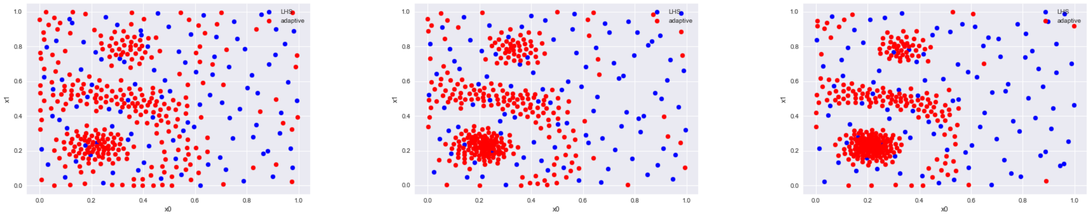
Shaping factor:	3	6	9
Minimal adaptive MSE	$4.954e^{+8}$	$4.686e^{+8}$	$3.858e^{+8}$
Maximum adaptive MSE	$1.126e^{+11}$	$6.183e^{+8}$	$1.126e^{+11}$
Mean adaptive MSE	$1.937e^{+10}$	$5.529e^{+8}$	$3.783e^{+10}$
Coefficient of variation adaptive	215.3%	8.227%	140.0%
Mean adaptive MSE / LHS MSE	1744%	49.77%	3406%

Table 5.6: Test 1 ML shaping factor

The results clearly show that there are also parameterisations for the BF3 with regard to offset and shaping parameters for which the method is stable and leads to an improvement. The possibility of determining an optimal shaping parameter is to be preferred here because, as can be seen in the Table 5.6, this leads to the lowest coefficient of vari-

Offset:	1%	20%	50%
Minimal adaptive MSE	$3.858e^{+8}$	$4.954e^{+8}$	$4.201e^{+8}$
Maximum adaptive MSE	$9.447e^{+8}$	$1.126e^{+11}$	$7.909e^{+8}$
Mean adaptive MSE	$5.780e^{+8}$	$5.661e^{+10}$	$5.654e^{+8}$
Coefficient of variation adaptive	28.75%	98.96%	19.65%
Mean adaptive MSE / LHS MSE	52.03%	5096%	50.91%

Table 5.7: Test 1 ML offset



(a) Offset 50

(b) Offset 20

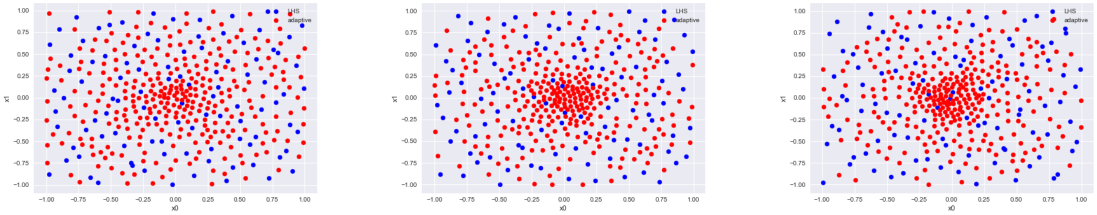
(c) Offset 1

Figure 5.13: Effect of the offset on BF2

ation in the results when varying the other parameters.

Another remarkable feature is that, for example, there are very visually very similar BF2 datasets that however lead to strongly differing MSE values in the surrogate models. This can be observed in the datasets shown in Figure 5.13. Here the shaping factor is always 3 and 40 reference points were always used. Only the offset is changed. Accordingly, the clusters of the adaptive samples are in the same areas, only the density of the clusters is different. Intuitively, it could be assumed that these data sets lead to surrogate models with similar MSE, but this is not the case. The surrogate model trained on the dataset in Figure 5.13a achieves an MSE of $9.532e^{-4}$, which is a significant improvement over the LHS. The surrogate model trained on the dataset in Figure 5.13b achieves an MSE of $1.506e^{-3}$ which is close to that of the LHS in terms of magnitude. The surrogate model trained on the dataset in Figure 5.13c achieves an MSE of $1.888e^{-3}$, which is a significant deterioration compared to the LHS.

This effect seems to be strongly dependent on the used BF, as this behaviour is almost not noticeable for BF1. Figure 5.14 also shows data sets that differ only in their



(a) Offset 50

(b) Offset 20

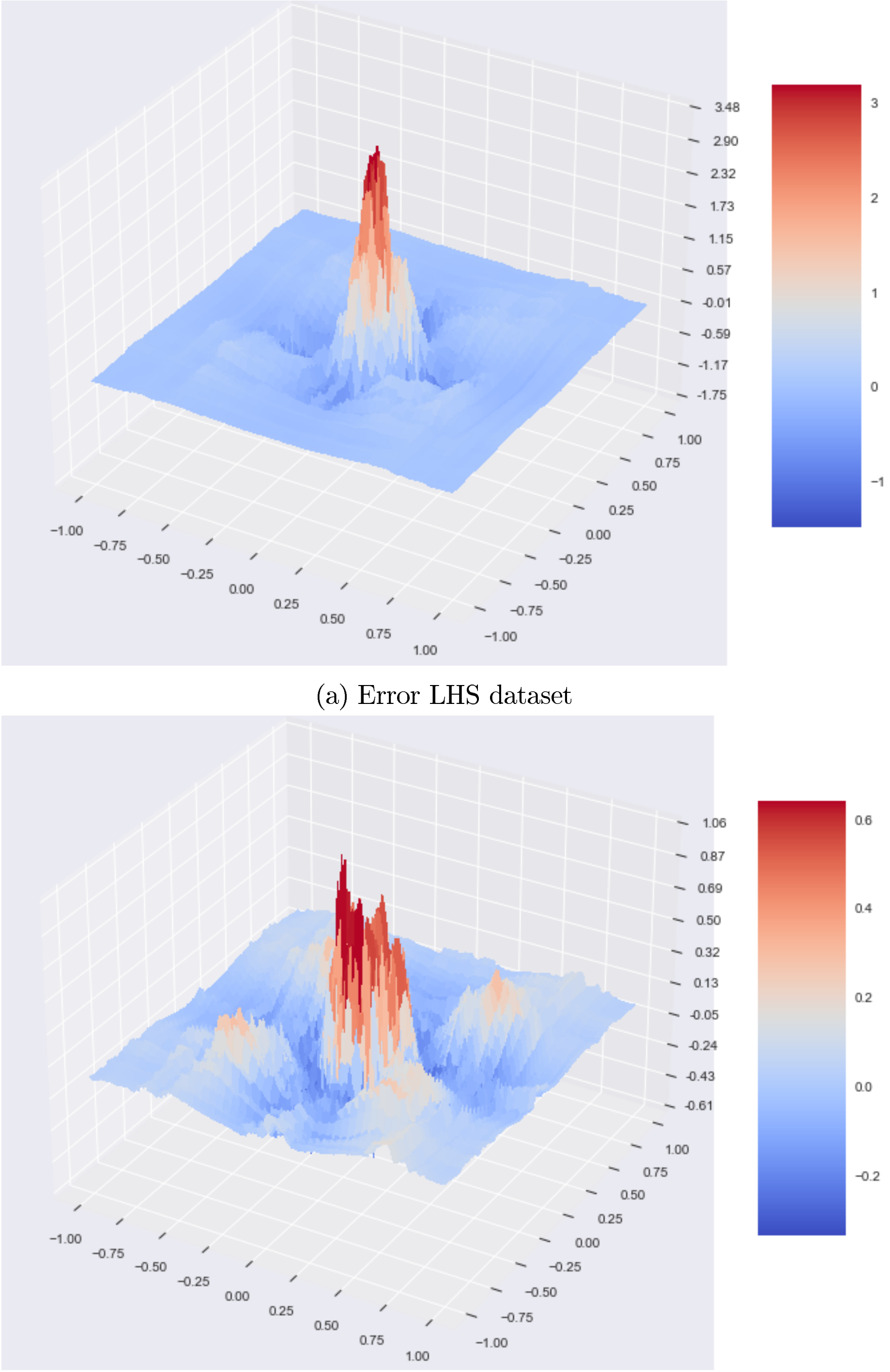
(c) Offset 1

Figure 5.14: Effect of the offset on BF1

offsets. Here the shaping parameter is set to 6 and the number of reference points to 20. The MSE of the surrogate model trained on this basis does not differ at all and is significantly better than that of the LHS dataset. The surrogate model trained on the dataset in Figure 5.14a achieves an MSE of $3.847e^{-2}$, on the dataset in Figure 5.14b achieves an MSE of $3.260e^{-2}$, on the dataset in Figure 5.14c achieves an MSE of $3.861e^{-2}$.

In order to analyse the observations presented on the previous paragraph, the error of the surrogate models with respect to BF1 and BF2 is shown in Figures 5.15 and 5.16, respectively. In both of those cases, a surrogate model trained with an LHS data set and one trained with an adaptively generated data set are compared.
In each case, the data set from Figure 5.14a and Figure 5.13a with the offset 50% is used, to train the models resulting in the errors of Figure 5.15b and Figure 5.16b. This data set led to an improvement in the MSE for the respective BF. This makes it possible to compare how the effect of the error optimisation differs.

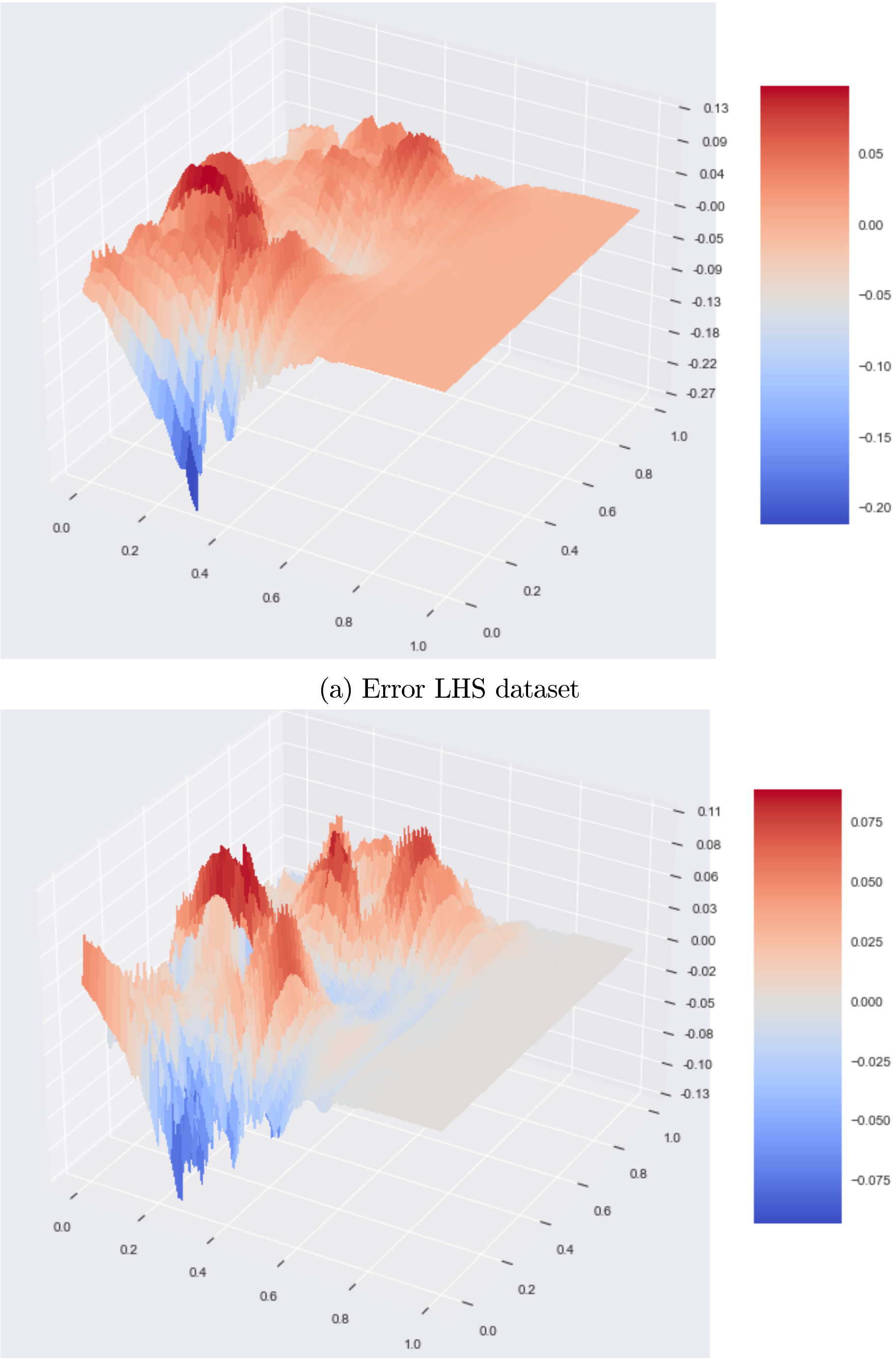
The Figure 5.15a shows that the maximum error of the surrogate model trained on the LHS data makes with respect to BF1 is 3.48. This error is larger in magnitude than the value of BF1 itself, see Figure 5.1a. It can be seen that the surrogate model which was trained with the LHS dataset cannot approximate the real behavior in this area at all. In contrast, Figure 5.15b shows that there is still a significant error at the same position, but the error of the surrogate model trained with the adaptiv dataset it is clearly below the function value. We can conclude then, that adaptative sampling does not bring and improvement of the MSE in BF1 due to an optimised approximatino in the non-linear areas, but a fundamentally correct approximation is made possible at all. This is also the reason why adaptive sampling for BF1 has a great influence on the MSE.
In BF2, the effect of adaptive sampling is different, as shown in the Figure 5.16. The surrogate model trained with the LHS data set is able to approximate the rough shape



(a) Error LHS dataset

(b) Error adaptive dataset

Figure 5.15: Error LHS vs adaptive BF1



(a) Error LHS dataset

(b) Error adaptive dataset

Figure 5.16: Error LHS vs adaptive BF2

of the BF. The error in the entire design space is smaller than the value of the BF. In contrast to BF1, the adaptive sampling does not lead to the basic approximation being made possible, but rather to the approximation being further improved in the areas of the clusters. Since samples that are further apart contain more different information, clusters that are not too compact are suitable in such a case. Since the density of the clusters decreases with increasing offset, a larger offset leads to an improvement if the behavior of the sampled function is just slightly non-linear.

The fact that the offset has no influence on the MSE is a peculiarity that only applies to BF1, see Table 5.8. For the other BFs, a larger offset leads to a lower MSE.

Offset	BF1	BF2	BF3	BF4	BF5	$\varnothing \setminus BF3$	$\varnothing$
1	37.64%	97.52%	52.03%	82.17%	106.21%	80.89%	75.11%
20	38.28%	87.79%	5096.05%	79.94%	94.17%	75.04%	1079.24%
50	38.63%	72.29%	50.91%	78.61%	84.19%	68.43%	64.93%

Table 5.8: Test 1 effect of the offset compared to LHS

Since the goal is to keep the time needed to determine the position of an adaptive sample as short as possible and the factor that significantly influences the time needed is how many reference points are used, the goal is to use as few reference points as possible. However, it is important to ensure that there are enough reference points to determine the position of the non-linearity. To investigate how the number of reference points affects the quality of the ML-based surrogate model, these values are compared in the Table 5.9. The values are averaged over all BFs in order to make a statement that is as general as possible. BF3 was not taken into account because the test cases in which the applied method failed would bias the results too much.

Number of reference points	10	20	30	40
Coefficient of variation adaptive	19.05%	15.91%	21.66%	21.71%
Mean adaptive MSE / LHS MSE	76.91%	74.64%	72.66%	74.93%

Table 5.9: Test 1 effect of number of reference points

As can be seen, the number of reference points used for the RBF function has only a minor influence on the results. This can be interpreted as it being possible to use a small number of reference points in order to minimize the time spent determining the sample positions.

The most important results of this test are summarized below:

- Optimisation of the shaping factor is necessary to ensure that the method is stable.

- A large offset is appropriate when it is known that the behavior of the system is only slightly non-linear.

- Adaptive sampling has the highest potential when only small amounts of data are available for strongly non-linear systems.

5.4 Further improvements

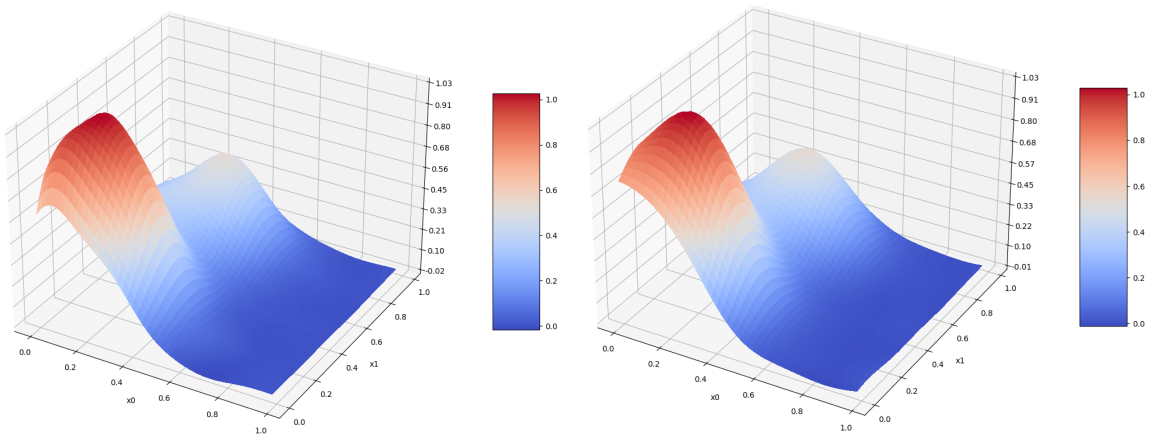
5.4.1 Determine an optimized shaping factor

Based on the knowledge that the shaping factor has a critical influence on the stability of the adaptive sampling method, a method to optimize this shaping factor has to be implemented. Since the optimal shaping factor depends on the sampled system, it makes sense to use the initial data that must be available anyway.

In the best case it would be possible to perform an optimisation so that the approximated Laplacian operator is as similar as possible to the real Laplacian operator of the system, since not the RBF approximation but its Laplacian operator has an influence on the adaptive sampling. However, this is not possible because the samples only contain absolute values, no information about the derivatives. It can be expected that a shaping factor that is optimized for the RBF approximation also leads to an improved Laplace operator.

In order to obtain an error value which can be used as a basis for an optimisation process, the deviation between the value of the RBF approximation and the real output value of the samples is calculated for all initial samples. To reduce this vector of data to a single error measure, the norm value of this vector is calculated.

Based on this error measure, an optimized shaping parameter is determined with the "fmibound" algorithm, which is based on the Nelder-Mead method [21]. An example of



(a) RBF approximation BF2 shaping factor 6

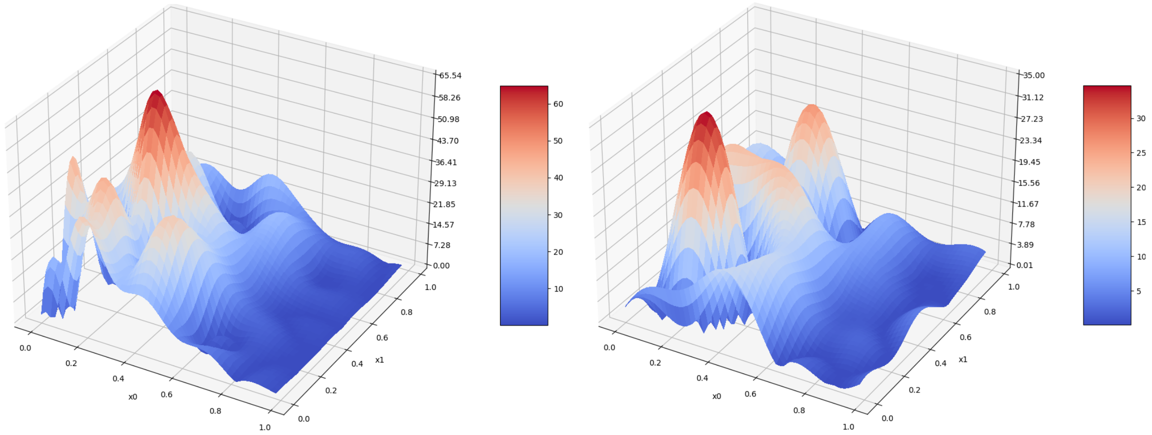
(b) RBF approximation BF2 optimized shaping factor (2.95)

Figure 5.17: Comparison RBF with fixed and optimized shaping factor

the effect of this optimization on the RBF approximation is shown in Figure 5.17, and the effect on the Laplace operator of this RBF approximation is shown in Figure 5.18.

As it can be seen in Figure 5.17, the impact of the optimisation of the shaping factor on the quality of the RBF function is visible but not huge. The error value calculated on the initial samples of the RBF with the shaping factor 6 is $2.566e^{-1}$ while the error value of the RBF with the optimized shaping factor 2.95 is 3.712^{-2}. Therefor, in this example there is a significant improvement in the RBF approximation using the optimized shaping factor.

If onecompares the representation of the approximated Laplace value in Figure 5.18b, is compared with that of the correct Laplace value in Figure 5.1d, it can be noted that they still differ significantly in magnitude. However, the positions of the maximum values are the same. This leads to the fact that the position determined by PSO will also lie in this area. This is a clear difference compared to the Laplace approximation in Figure 5.18a. However, it is not possible to prove that this method used for optimizing the shaping factor always leads to an improvement of the RBF approximation, because the results depend strongly on the randomly selected shaping factor with which the comparison is made. An analysis of whether the method is suitable for stabilizing the adaptive sampling method is given in Section 5.5.



(a) RBF Laplace approximation BF2 shaping factor 6

(b) RBF Laplace approximation BF2 optimized shaping factor (2.95)

Figure 5.18: Comparison RBF Laplace with fixed and optimized shaping factor

5.4.2 Identifying a stack of samples

In order to reduce the time needed to determine the positions of the samples,an extension of the method allowing more than one sample to be determined at once is implemented. Since the RFB approximation does not have to be created for the determination of each sample position, this methodology results in a significant time saving. In addition, the parallelization ensures that when applying the method to a real simulation environment, it is possible to parameterise several of these simulations simultaneously.
To prevent all samples of a stack from being in the same position, the information about the position of the new samples is used, even if the information about their output value is not yet available. This does not improve the RBF approximation, but it prevents an already occupied position from being sampled again. An illustration of the updated process can be found in Figure 5.19.

5.5 Second Test

Since the shaping factor is determined automatically and a new hyperparameter has been added with the stack size, the experiment differs in terms of its parameters from Test 1. The varied parameters are shown in the Table 5.10. All the other settings are the same as in Section 5.3.1.

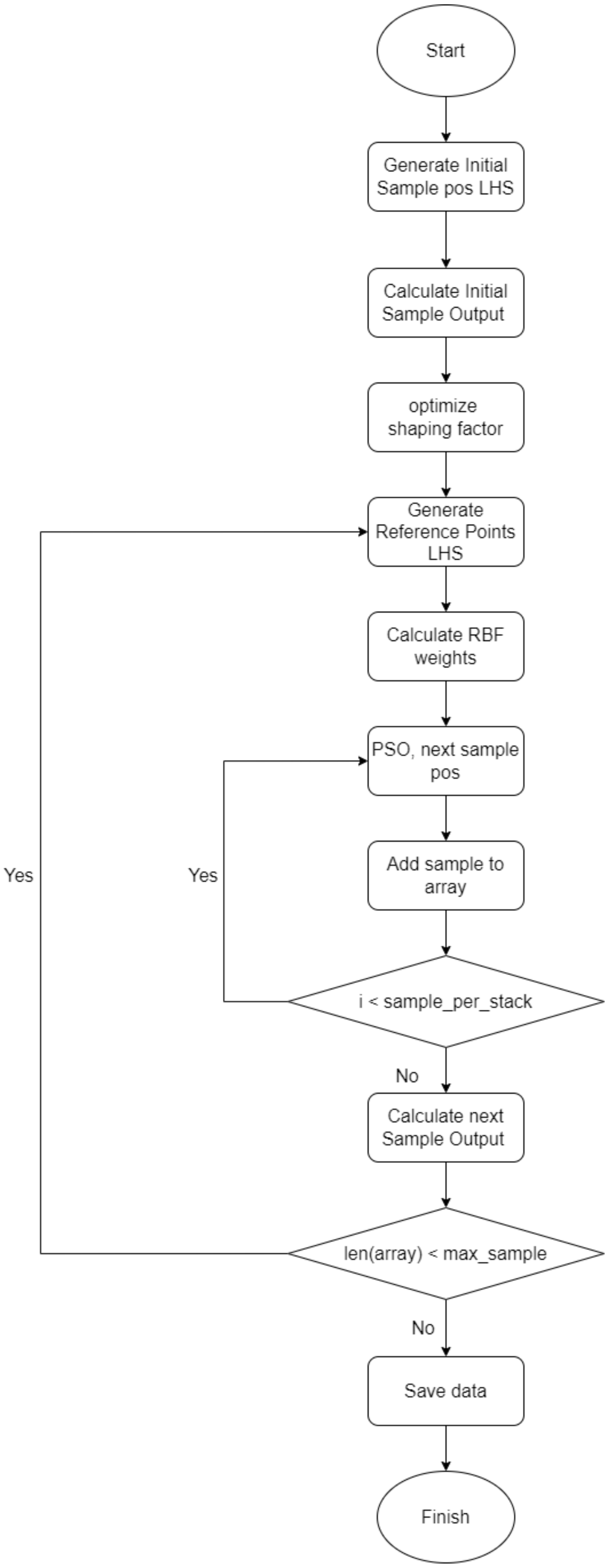


Figure 5.19: Flowchart updated implementation

Offset	1%	20%	50%	-
Number of reference points	10	20	30	40
Sampled stack size	1	5	20	50

Table 5.10: Test 2 parameters

5.5.1 Results

A brief summary of the results can be found in Table 5.11. The evaluation of all data can be found in the appendix.

	BF1	BF2	BF3	BF4	BF5
LHS MSE	$1.556e^{-1}$	$1.575e^{-3}$	$1111e^{+9}$	$3.325e^{+1}$	$9.626e^{-2}$
Minimal adaptive MSE	$3.871e^{-2}$	$1.119e^{-3}$	$3.189e^{+8}$	$2.424e^{+1}$	$5.527e^{-2}$
Maximum adaptive MSE	$1.167e^{-1}$	$2.022e^{-3}$	$9.582e^{+8}$	$4.016e^{+1}$	$1.445e^{-1}$
Mean adaptive MSE	$6.079e^{-2}$	$1.525e^{-3}$	$4.832e^{+8}$	$3.006e^{+1}$	$8.985e^{-2}$
Coefficient of variation adaptive	33.69%	16.52%	25.64%	12.10%	24.54%
Mean adaptive MSE / LHS MSE	39.08%	96.83%	43.50%	90.41%	93.34%

Table 5.11: Test 2 ML results

As it can be seen, the average MSE achieved for BF3 is no longer worse than that of the LHS data set. It is, however, the case that all variations of the parameters with respect to BF3 lead to an improvement compared to the LHS data set.

The average MSE values of BF1 and BF5 are almost unchanged. The average MSE values of BF2 and BF4 have worsened slightly, but are still below the comparative value of the LHS data set. The evaluation of the effect of the number of reference points shows that this deterioration affects those tests with a low number of reference points. While in the first test, as shown in Table 5.9, the number of reference points have only a small effect on the MSE, in test 2 it is the case that there is a critical number of reference points that should not be undercut. In order to compare the results with test 1, Table 5.12 shows the effect of the number of reference points on the MSE without considering BF3. Table 5.13 shows the effect of the number of reference points on all BFs.

As it can be seen, the results regarding the MSE for 30 and 40 reference points are comparable to those from Test 1. If fewer reference points are used, the quality of the sample positions decreases proportionally to the number of reference points, which leads

Number of reference points	10	20	30	40
Coefficient of variation adaptive	17.58%	16.65%	21.28%	18.31%
Mean adaptive MSE / LHS MSE	89.18%	80.66%	74.96%	74.85%

Table 5.12: Test 2 effect of number of reference points without BF3

Number of reference points	10	20	30	40
Coefficient of variation adaptive	17.06%	15.48%	19.00%	16.21%
Mean adaptive MSE / LHS MSE	83.38%	72.28%	67.17%	67.69%

Table 5.13: Test 2 effect of number of reference points with BF3

to poorer results of the surrogate models with regard to the MSE.

The finding that a larger offset leads to a lower MSE from Test 1, Table 5.8, is confirmed in Test 2, see Table 5.14. However, in this case and in contrast to the observations made in Test 1, a larger offset also leads to a lower MSE in the case of BF1.

Offset	BF1	BF2	BF3	BF4	BF5	$\varnothing \setminus BF3$	$\varnothing$
1	48.19%	110.61%	44.83%	93.65%	104.43%	89.22%	80.34%
20	39.05%	96.98%	45.02%	94.28%	96.79%	81.78%	74.42%
50	29.99%	82.89%	40.66%	83.32%	78.81%	68.75%	63.13%

Table 5.14: Test 2 effect of the offset compared to LHS

The results of the impact of the stack size on the MSE are shown in Table 5.15. As expected, small stack sizes are preferable because they provide the most information for determining the next sample position. However, even for large stack sizes, adaptive sampling on average improves the MSE in relation to LHS.

Stack size	BF1	BF2	BF3	BF4	BF5	$\varnothing$
1	31.32%	88.95%	42.48%	90.06%	87.64%	68.09%
5	36.43%	98.23%	42.40%	88.22%	95.08%	72.07%
20	41.25%	96.86%	44.42%	86.46%	96.39%	73.08%
50	47.31%	103.26%	44.71%	96.91%	94.25%	77.29%

Table 5.15: Test 2 effect of the stack size compared to LHS

The most important results of this test are summarized below:

- The application of the optimized shaping factor leads to a more stable method with results that are better than the LHS benchmark.

- An insufficient number of reference points has a negative effect on the MSE. However, increasing the number of reference points only improves the MSE up to a limited amount. In the case of the 2D BFs used, this quantity is 30 reference points.

- Determining stacks of samples is possible, but has a negative effect on the MSE as the stack size increases.

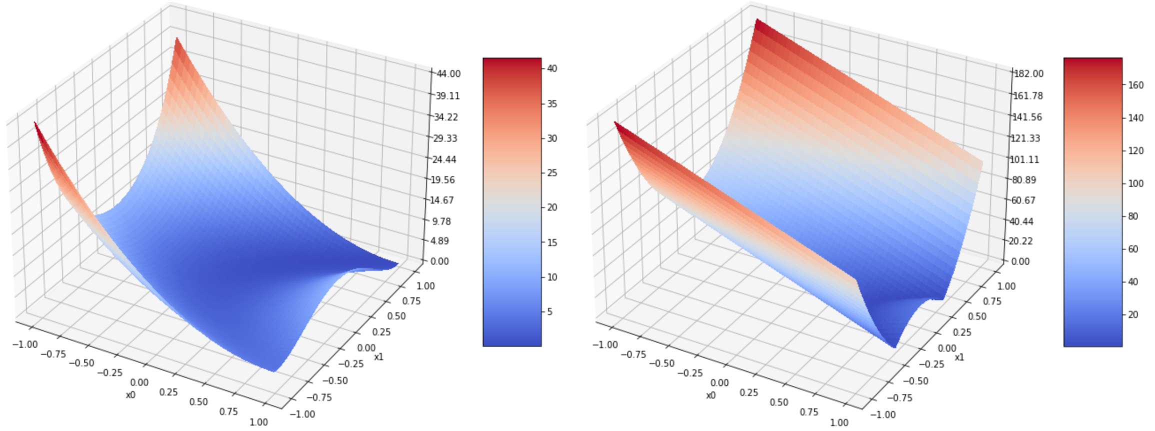
- A high magnitude offset that decreases the density of the resulting clusters of samples leads to an improved MSE.

5.6 Test on 4 dimensional BF

In order to check whether the findings from the previous sections can also be transferred to higher-dimensional problems, they are to be tested in the following on a 4-dimensional BF. The Function 5.2 is used for this purpose.

$$\begin{gathered} BF1\_3(x_0, x_1, x_2, x_3) = 10(-4e^{-\frac{25}{8}(x_0^2+x_1^2)} + 7e^{\frac{125}{4}(x_0^2+x_1^2)}) \cdot 10(x_3 - x_2^2)^2 + (x_2 - 1)^2 \\ x_0 := [-1, 1], x_1 := [-1, 1], x_2 := [-1, 1], x_3 := [-1, 1] \end{gathered} \tag{5.2}$$

Although the function BF1_3 is based on the BF1 and BF3 functions, slight changes have been applied. BF1 has been multiplied by a factor of 10 and the design space of the dimensions previously belonging to BF3 has been changed. These slight changes ensure that the influence on the magnitude of the complete function at the position of the maxima of BF1 and BF3 is approximately the same. This makes it possible to check whether the clusters are in the expected regions by plotting the dimensions.
Since the shape of BF3 has changed somewhat due to the reduction of the design space, it is shown in Figure 5.20. The shape of BF1 is unchanged and can be seen in the Figure 5.1a and Figure 5.1b.



(a) BF3

(b) LBF3

Figure 5.20: BF3 in reduced design space

5.6.1 Setup

In order to gain information about the amount of data for which adaptive sampling is suitable for the creation of ML-based surrogate models, 50 data sets for BF1_3 are created in this experiment. These data sets differ in the amount of samples. They are distributed logarithmically in the range between 100 and 1000 samples. In each data set, 80% of the samples are initially determined using the LHS algorithm and 20% of the data are adaptively sampled. The 20% of the adaptive samples are divided into 10 stacks. This ensures that the stack size is always the same in percentage terms in relation to the size of the specific data set.

Based on the results from Section 5.5, an offset of 50% is chosen and 30 reference points are used for the RBF approximation.

5.6.2 Evaluation

Consistently with the previously done experiments, for all evaluations a RF model consisting of 100 decision trees is created using the corresponding scikit-learn library with the default hyperparameter set [19].

In order to reduce the variance of the model the mean is evaluated 10 times over the same data set, where each time the train and test split is randomly picked. The results are shown in Figure 5.21.

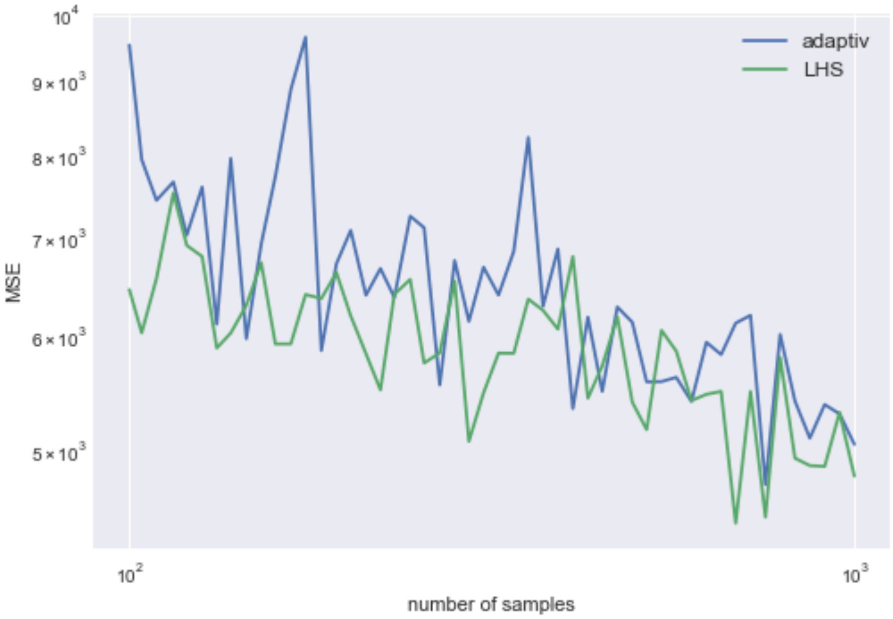
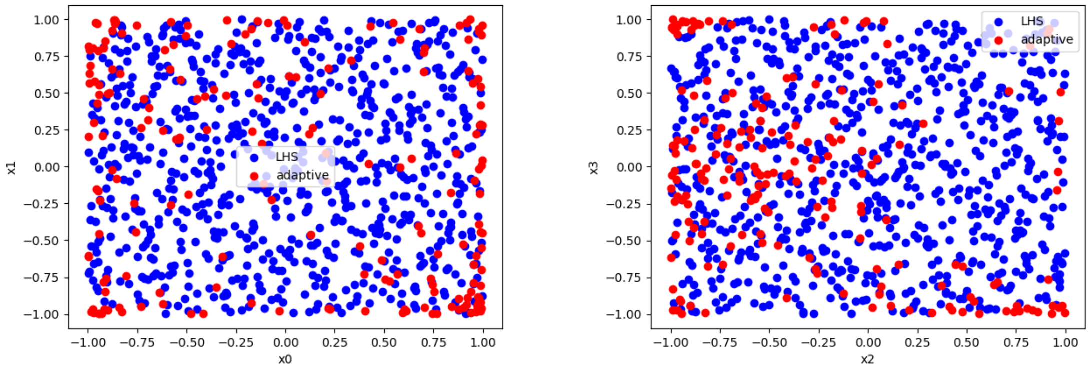


Figure 5.21: Results BF1_3 adaptively sampled and LHS

As expected, regardless of the method used for sampling, larger datasets deliver smaller MSE values. However, it is not the case that the MSE can be reduced by the usage of adaptive sampling compared to the LHS sampling. On the contrary, the results of adaptive sampling for small data sets are subject to strong fluctuations. As the amount of data increases, this effect decreases, but again no lower values of the MSE are obtained compared to the LHS data set.
When looking at the clusters of samples formed, it is noticeable that they are predominantly not in the areas where they are to be expected, as shown in Figure 5.22 as an example. The results of all data sets can be found in the appendix. The positions of the clusters as expected are the maximum values from Figure 5.1b and Figure 5.1f.

This leads to the hypothesis that the RBF approximation cannot represent the sampled BF1_3 sufficiently accurately. To test this hypothesis, the experiment is repeated, but for this example, 100 reference points are used instead in order to increase the quality of the RBF approximation. The results of this experiment are shown in the Figure 5.23.



(a) BF1 part

(b) BF3 part

Figure 5.22: BF1_3 sample distribution for 954 samples, 30 reference points

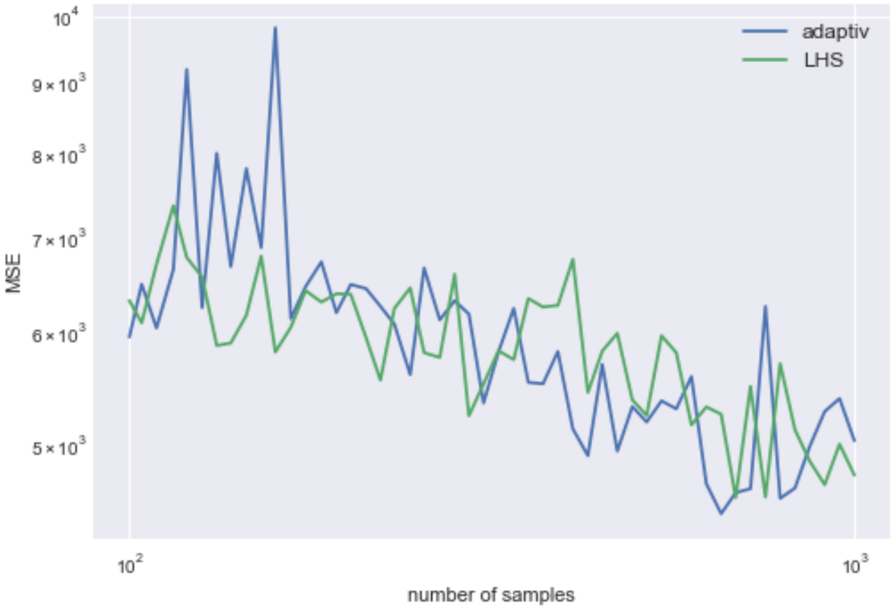
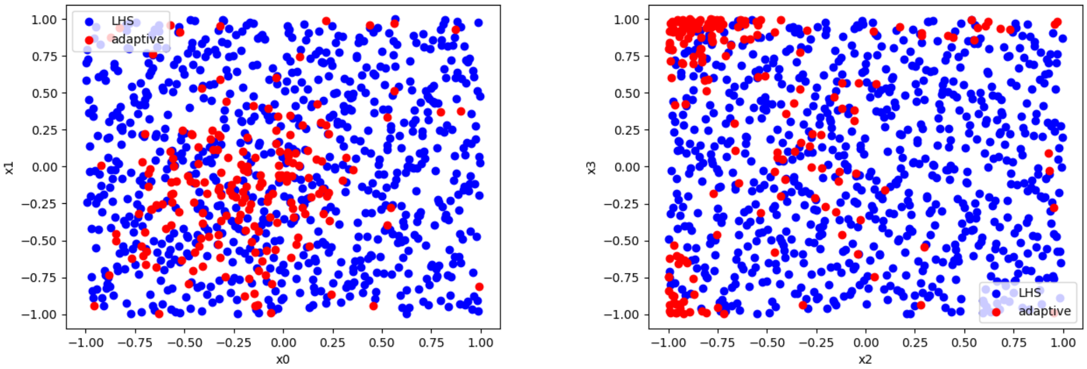


Figure 5.23: Results BF1_3 adaptively sampled and LHS, 100 reference points



(a) BF1 part

(b) BF3 part

Figure 5.24: BF1_3 sample distribution for 954 samples, 100 reference points

As it can be seen, changing the amount of reference points has no effect on the MSE. However, it is possible to observe in some individual cases that the clusters do form within the expected ranges. This is shown as an example in the Figure 5.24, and the results of all data sets can be found in the appendix.

The results presented in this section lead to the conclusion that the RF used in this 4-dimensional application are no longer able to benefit from the distribution of the adaptive samples.
To verify this, the data sets are used again to create a surrogate model. This time NN are used with the parameterisation from Table 5.2. The results are shown in the Figure 5.25.

As it can be seen, it is possible to observe a significant improvement in the MSE of the NN for those data sets with high data volumes in which adaptive sampling is used. For example, for the dataset shown in Figure 5.25, using adaptive sampling reduces the MSE to 34.20 % of the MSE achieved with the LHS data set. However, the results are subject to strong noise, which makes precise evaluation difficult.

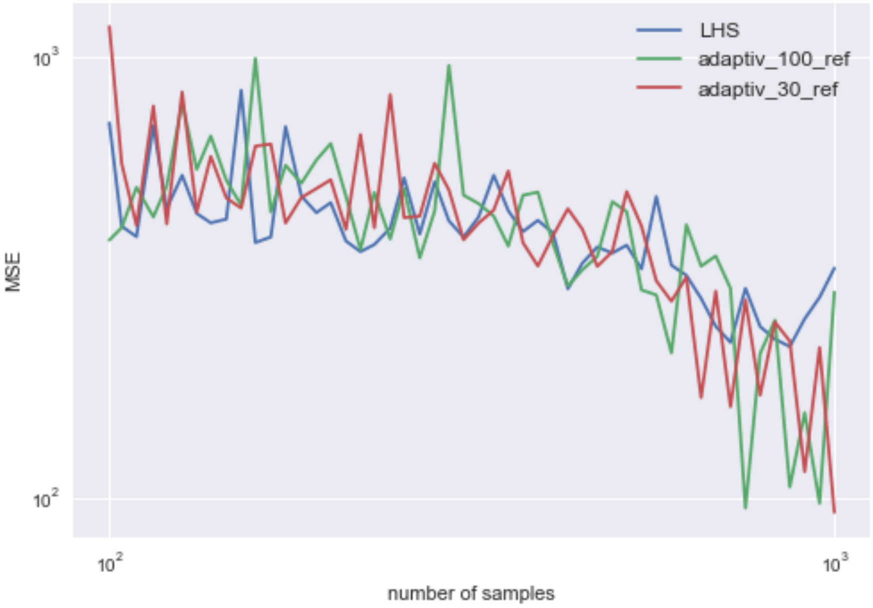


Figure 5.25: Results BF1_3 adaptively sampled and LHS NN

6 Test on real application

In this chapter, the developed method will be tested on a real application. For this purpose, the tool Fast and Advanced Mass Estimation - Wing (FAME-W) is used. The tool will be explained in more detail in Section 6.1.

The aim of this study is to determine whether the results from the Section 5 can be confirmed on a real use case. For this purpose, four different experiments are performed. In the first two experiments, the developed method for adaptive sampling should determine all positions of the adaptive samples at once. The conjoint aim of these two experiments is to determine whether the method can be used if the environment of the simulation cannot be controlled due to technical reasons. While one of the experiments aims at determining how the reduction of the MSE can be impacted by using the same number of samples but a different methodology to determine them, the second experiment keeps the MSE fixed and evaluates how much the number of samples can be reduced without modifying it.

In the following two experiments, the method should then actively start the simulation of the new samples. This makes it possible to continuously improve the RBF approximation used by the method after completion of a simulation run on the basis of the new adaptively sampled data. The final goal of these experiments is to determine whether any improvement can be expected from the use of the aforementioned information and if so, quantify it.

6.1 FAME-W

FAME-W is a software package for structural and load analysis and mass estimation of aircrafts in the preliminary design phase. FAME-W has a range of input and output values that can be used for design. For the results shown in this thesis, the inputs and outputs shown in Table 6.1 are used.

Name	Meaning
Inputs	
Maximum Take Off Weight (MTOW)	Describes the maximum weight of an airplane during take-off.
Maximum Zero Fuel Weight (MZFW)	Describes the maximum weight of an airplane without fuel, cargo and passengers.
Wing span	Spread ot the wing
Chord	Describes the length of a straight line joining the leading edge and trailing edge of an aerofoil.
Thickness-to-Chord ratio (TC ratio)	The ratio of between the maximum wing thickness and the chord.
Sweep	The angle between the airfoil and the airplane transverse axis in plan view.
Outputs	
Wingmass	Describes the mass of the wing without fuel.

Table 6.1: FAME-W used inputs and output

Since the focus of this thesis is on the generation of datasets to train surrogate models, the exact procedure of how outputs are calculated based on inputs is outside the scope of this thesis. As all this data contains sensitive information of the company, it cannot be presented without distortion in the following as well as in the appendix. In order to ensure that all tests are reproducible, this data is only used in standardized form. This ensures that the values of the parameters remain hidden. Since this step is necessary anyway to train ML models, it has no influence on the percentage deviation of the results of the ML models. Accordingly, the effect of adaptive sampling can still be demonstrated. As is the case with other simulations, it can happen with FAME-W that certain parameter constellations lead to unusable design solutions, eg. insufficient fuel volume, i.e. their output value is Not a Number (NaN). Since these values cannot be used to create an RBF approximation, they are ignored in this respect. To prevent this position from being resampled, they are nevertheless taken into account by the square to the power function, which prevents adaptive samples from being set at already sampled positions.

6.2 Setup

Since the results shown in Figure 5.25 indicate that the number of reference points required for the RBF approximation increases as the number of dimensions increases, 100 reference points have also been selected to perform the studies using FAME-W. Since a 50% offset delivers the best results in the previously performed two dimensional experiments, this offset value is also chosen for the case of FAME-W.

6.3 FAME-W asynchron test

As described at the beginning of this chapter, it will first be investigated whether it is possible in principle to use the developed method to meaningfully expand an existing FAME-W data set created using the LHS method. In this context, asynchronous means that after the initial samples have been created, all further adaptive sample positions are determined at once. There is no exchange of information with FAME-W when determining these sample positions. Accordingly, no information about the outputs of the adaptive samples can be used. In the first experiment, a data set with 1000 initial LHS samples is supplemented by 500 adaptive samples. The aim is to determine whether and how much reduction of the MSE can be achieved compared to a 1500 LHS dataset. The positions of these samples are determined all at once on the basis of the initial samples. This has the advantage that the basic functionality of the method can be tested with regard to this use case without having to implement all interfaces. In the second experiment, it will be determined whether it is possible to reduce the number of samples used without deteriorating the MSE of the resulting model. For this purpose, 500 initial LHS samples are supplemented by 400 adaptive samples.

6.3.1 Reduction of the MSE

When creating the 500 adaptive samples using the developed method, clear clusters emerge in the design space. This behavior is exemplified in Figure 6.1, where the positioning of the samples in terms of MTOW and wingspan is shown. A representation of the samples with regard to all parameters of the design space can be found in the appendix. It can be seen that in this example the number of adaptive samples increases with increasing wingspan while the distribution is homogeneous with respect to MTOW.

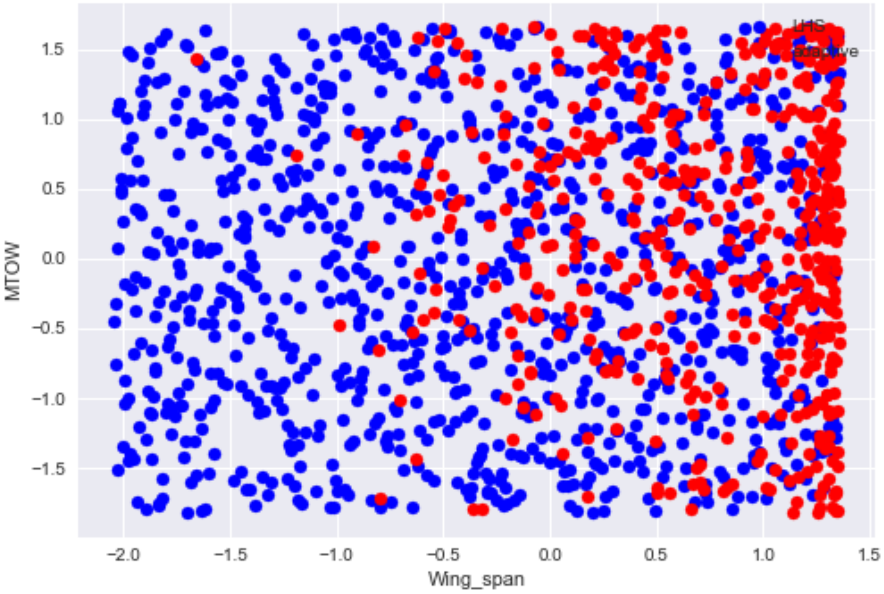


Figure 6.1: Adaptive samples FAME-W MTOW over Wing span

This is logical because an increasing wingspan directly results in a higher material requirement of the wing and the lever in which forces act increases. This also results in a higher material requirement. In contrast, the wing mass increases linearly with increasing MTOW, which should have no influence on the adaptive sampling.

To determine how the use of adaptive samples affects the MSE, the mean value of the MSE of 10 NNs parameterised according to the Table 5.2 is used as in the previous chapters. In order to gain additional information on how the ratio of adaptive samples to initial LHS samples affects the MSE, the data set is evaluated for each added adaptive sample using NNs. This evaluation is visualized in Figure 6.2. Since the resulting graph is noisy, a moving average over 5 samples is applied. An exact visualization and the results as a file can be found in the appendix.

At the beginning of the graph, the achieved MSE of the adaptively generated data set is significantly worse than that of the LHS data set. This is to be expected as a model based on 1000 LHS is compared to one based on 1500 LHS samples and therefore trained with less information. The first added adaptive samples lead to a deterioration, but already

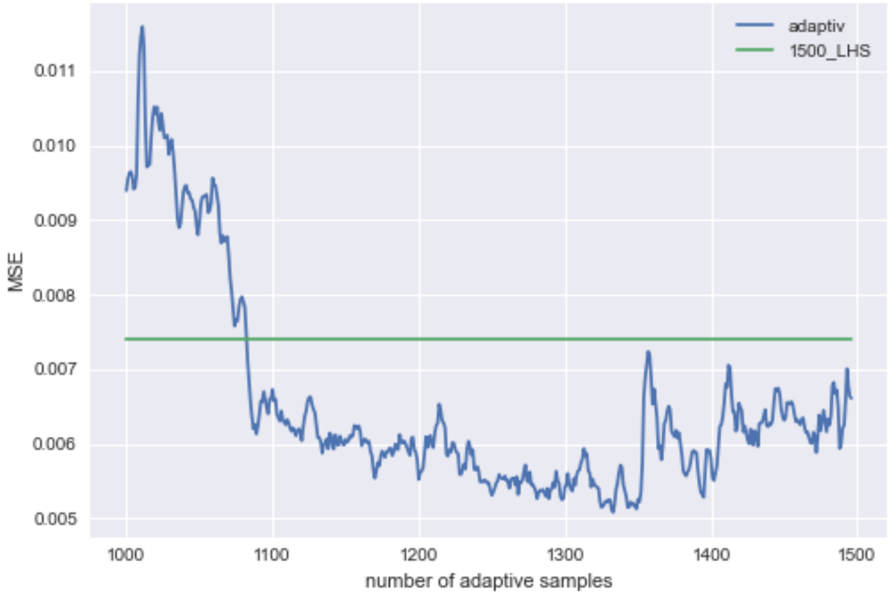


Figure 6.2: FAME-W comparison of 1000+500 adaptive samples and 1500 LHS samples

with the use of 100 adaptive samples an improvement is achieved compared to the 1500 LHS model. The maximum improvement achieved occurs with 293 adaptive samples. The MSE is 62.81% of the MSE achieved with the 1500 LHS data set. Adding more adaptive samples to the data set no longer leads to an improvement in MSE. It can be assumed that this is due to the fact that the data initially given is no longer sufficient after the prediction of so many positions to determine further suitable positions. This will be further investigated in Section 6.4. Nevertheless, even with the complete adaptive data set, a reduction of the MSE to 85.65% of the comparison value is still achieved.

6.3.2 Reduction of the amount of samples

As described at the beginning of this section, in this experiment a data set based on 500 initial LHS samples is enriched with 400 adaptive samples. The positions determined for the adaptive samples are comparable to those of the previous experiment and can be found in the appendix. The results of this experiment are shown in Figure 6.3, where again a moving average is applied over 5 samples.

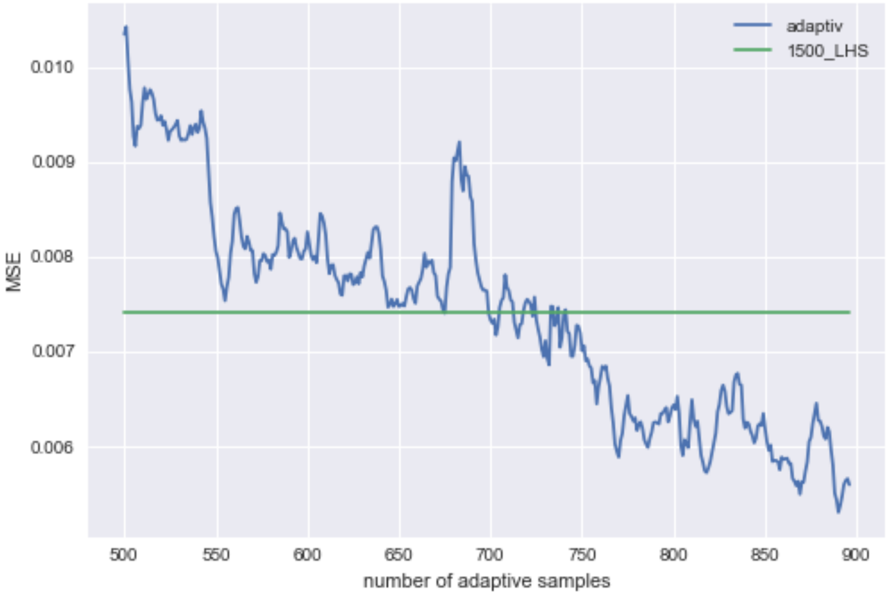


Figure 6.3: FAME-W comparison of 500+400 adaptive samples and 1500 LHS samples

In contrast to the previous experiment, the first adaptive samples do not lead to a deterioration of the MSE. The effect that the MSE of the model deteriorates again with a large proportion of adaptive samples also does not occur here. The MSE achieved with the 1500 LHS data set is already reached with 59 adaptive samples. This would correspond to a reduction in the amount of data required to 37.26% of the amount of data used in the comparison, but this is only a single value that is subject to noise. From 238 adaptive samples onwards, the MSE of the 1500 LHS data set is reached stably, which corresponds to a reduction of the required amount of data to 49.20%. Stable in this context means that the mean MSE over the next 50 samples is below the MSE obtained with the 1500 LHS data set and none of these subsequent values is individually greater than 110% of the MSE obtained with the 1500 LHS data set.

6.4 FAME-W synchron test

As described at the beginning of this chapter, in this section the effect of the developed method for adaptive sampling is investigated when the information of the newly generated adaptive sampling is directly used for the RBF approximation. Since the amount of

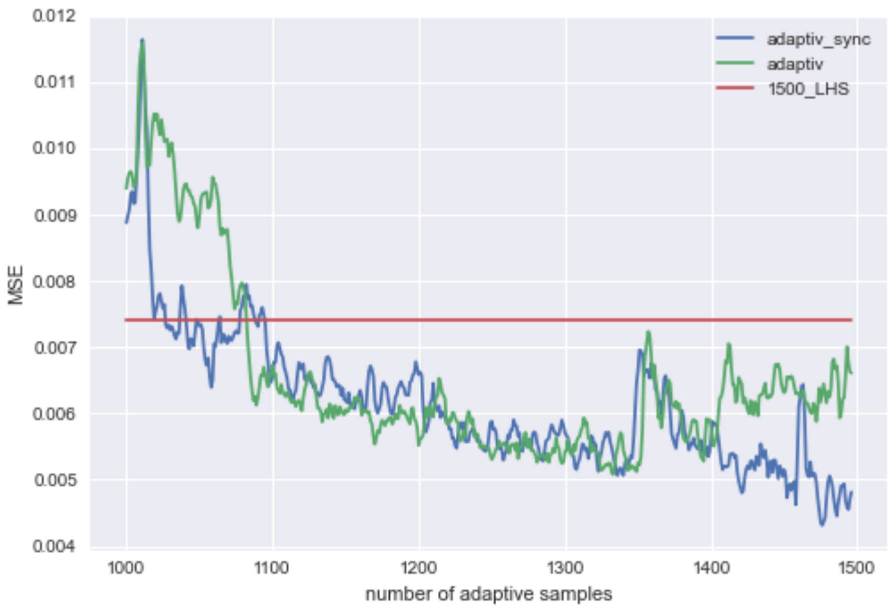


Figure 6.4: Comparison of 1000+500 adaptive samples asynchronous and synchronous

information that can be used for the RBF approximation increases in the course of sampling, it can be assumed that especially for large amounts of adaptive samples better values of the MSE can be achieved. In order to be able to run the simulations of FAME-W in parallel, 5 samples are calculated at a time. The results from Table 5.15 indicate that the optimal results are no longer achieved with this stack size, but for time reasons it is not possible to determine FAME-W samples individually. Since it is common to use one FAME-W instance per available CPU core, a stack size of 5 is a realistic order of magnitude. To enable a comparison with the results from the Section 6.3, the same number of samples is selected in each case.

6.4.1 Reduction of the MSE

The complete setup of this experiment, with the exception of the use of adaptive samples to optimize the RBF approximation, is the same as in Section 6.3.1. To compare the results of these experiments, the results of both asynchronous and synchronous adaptive sampling are shown in Figure 6.4.

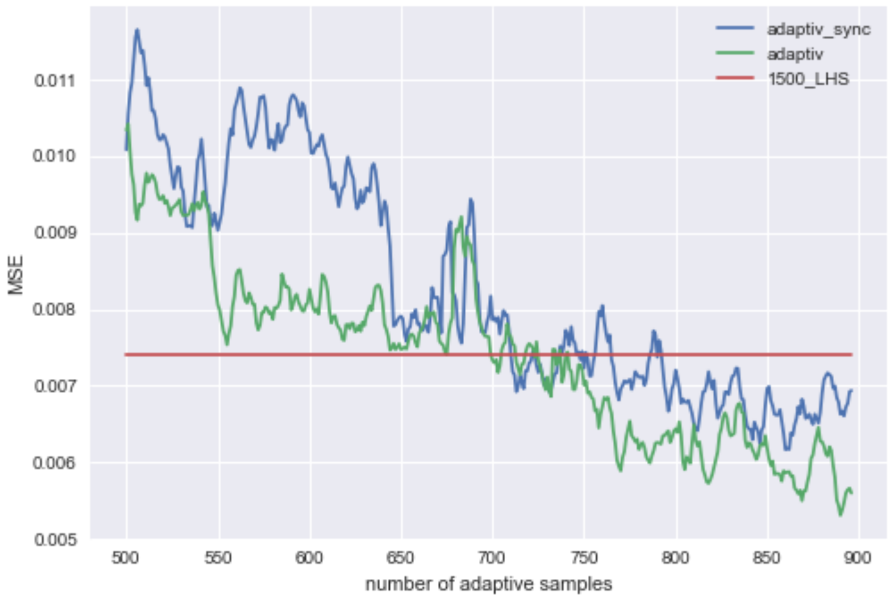


Figure 6.5: Comparison of 500+400 adaptive samples asynchronous and synchronous

It is noticeable that the results of both experiments are very similar. Nevertheless, it can be seen that the MSE of the synchronously sampled experiment is lower, especially when using few or many adaptive samples in relation to this experiment. The greatest improvement in MSE with the synchronous variant of adaptive sampling is greater than with the asynchronous variant. The best MSE achieved is 54.19% in relation to the 1500 LHS data set. It is also noticeable that with the synchronous variant the value of the MSE in relation to the comparative value does not increase again with increasing number of adaptive samples as is the case with the asynchronous variant.

6.4.2 Reduction of the amount of samples

The complete setup of this experiment, with the exception of the use of adaptive samples to optimize the RBF approximation, is the same as in Section 6.3.2. To compare the results of these experiments, the results of both asynchronous and synchronous adaptive sampling are shown in Figure 6.5.

In contrast to the experiments in which the reduction of the MSE is investigated, in this case the the asynchronous variant of the adaptative sampling achieves the best results. Especially in the range of the first 150 adaptive samples, the MSE achieved by means of the synchronously created adaptive samples is higher. As the number of samples increases, the difference in MSE between the two methods decreases, but the asynchronous variant also achieves lower MSE values in this case. Nevertheless, from 296 adaptive samples onwards, the MSE of the 1500 LHS data set is reached stably, which corresponds to a reduction of the required amount of data to 53.07%.

6.5 Assessment of the results

The results of the previous tests are summarized in the Table 6.2. The values are given as a percentage of the values of the 1500 LHS data set used as a benchmark.

	asynchronous	synchronous
Best MSE achieved	62.81%	54.19%
Reduction amount of data	49.20%	53.07%

Table 6.2: FAME-W improvement achieved compared to LHS benchmark

The results of this experiment confirm that the adaptive sampling method is suitable for this application. However, the underlying data are not sufficient to extend this statement to the general case. Although the best values achieved correspond to significant improvements compared to the benchmark, they are single values of a specific series of measurements that are subject to noise. Additionally, the fact that the results are worse in the synchronous variant of the sample-amount reduction study, despite of having a larger amount of information to determine the sample positions, leads to the conclusion that the results are subject to deviations. Hence, more information must be generated in order to be able to make more general statements.

Nevertheless, this result is an indication of the general suitability of this method. All applicable requirements from Chapter 3 are achieved in this test. The method has led to improved values of the MSE of the surrogate models with the parameters determined by the investigations of Chapter 5 stable in all experiments. These improvements have been achieved by using only the information from the samples. The stack size can be varied. The maximum stack size used is 44% of the available sample quantity and although this leads to a smaller amount of information for the RBF approximation than with a smaller stack size, the MSE of the surrogate models could be improved. The proportion of computing time used to determine the sample positions is less than 3% of the total computing time in all cases. The comparison of this value with the improvements shown in Table 6.2 shows, that the rate of information retrieval could be increased. The amount of data needed to generate a surrogate model that achieves the same MSE could be reduced by about 50% compared to the LHS benchmark, as shown in Table 6.2. This leads directly to the possibility of reducing the cost of generating surrogate models by reducing the amount of data and thus the simulation cost.

7 Summary

7.1 Conclusion

In this thesis, various possibilities of optimizing the Design of Experiment (DoE) creation for computer simulations are investigated. In order to generate an advantage over the currently used DoE methods, an adaptive sampling method has been developed. This has the advantage that information about the specific use case can already be used during the creation of the data set. Accordingly, it is possible to generate information at positions where a disproportionately large amount of information is needed for the specific application. A diverse set of tests has been designed to accordingly test the method and optimised for the 2-dimensional case. Based on the results of these tests, a set of procedures and parametrisations that lead to stable data sets that are better suited for training ML-based surrogate models than the currently used Latin Hypercube Sampling (LHS) algorithm has been defined. Both the possibility of reducing the amount of data and the possibility of creating surrogate models of better quality are generated by this. For higher dimensional use cases, the aforementioned improvements have also been achieved compared to the LHS algorithm, although less stable. Finally, the potential of the method observed in the test functions has been successfully tested and confirmed in a real aeronautical use case.

7.2 Outlook

If the results from Table 6.2 can be confirmed, this would correspond to a significant advantage over the current method. However, there is currently not enough information available on the applicability of the method in the high-dimensional use case to guarantee stable use. Further experiments have to be conducted and the disturbance variables have to be reduced. Some necessary improvements are:
- The definition of the parameterisation for high-dimensional applications, especially the

number of reference points to be used.
- The reduction of disturbance variables in the evaluation of test results caused by Neural Networks, e.g. by the Ray Tune Framework.
- The implementation of a more suitable optimisation algorithm for determining the shaping parameter, for example based on the "leave one out cross validation".
- The optimal ratio between initial samples and adaptive samples should be determined.
- The extension of the methodology to more than one variable in order to broaden its applicability scope.

Bibliography

[1] *Sampling Plans*. Kap. 1, S. 1–31. In: *Engineering Design via Surrogate Modelling*, John Wiley Sons, Ltd, 2008. – URL https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470770801.ch1. – ISBN 9780470770801

[2] Anthony A. Giunta, Michael S. E.: OVERVIEW OF MODERN DESIGN OF EXPERIMENTS METHODS FOR COMPUTATIONAL SIMULATIONS. In: *AIAA* (2003)

[3] Baudin, Michaël ; Dutfoy, Anne ; Iooss, Bertrand ; Popelin, Anne-Laure: *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*. S. 1–38. In: Ghanem, Roger (Hrsg.) ; Higdon, David (Hrsg.) ; Owhadi, Houman (Hrsg.): *Handbook of Uncertainty Quantification*. Cham : Springer International Publishing, 2016. – URL https://doi.org/10.1007/978-3-319-11259-6_64-1. – ISBN 978-3-319-11259-6

[4] Baudin, Michaël ; Dutfoy, Anne ; Iooss, Bertrand ; Popelin, Anne-Laure: *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*. S. 1–38. In: Ghanem, Roger (Hrsg.) ; Higdon, David (Hrsg.) ; Owhadi, Houman (Hrsg.): *Handbook of Uncertainty Quantification*. Cham : Springer International Publishing, 2016. – URL https://doi.org/10.1007/978-3-319-11259-6_64-1. – ISBN 978-3-319-11259-6

[5] Danka, Tivadar ; Horvath, Peter: *modAL: A modular active learning framework for Python*. 2018. – URL https://arxiv.org/abs/1805.00979

[6] Engelbrecht, A. P.: *Computational Intelligence: An Introduction*. Second. John Wiley and Sons, 2007

[7] Fahrmeir, Ludwig ; Kneib, Thomas ; Lang, Stefan: *Regression: Modelle, Methoden und Anwendungen*. 2. Berlin : Springer Verlag, 2009

[8] Frazier, Peter I.: *A Tutorial on Bayesian Optimization*. 2018. – URL https://arxiv.org/abs/1807.02811

[9] Frazier, Peter I.: *A Tutorial on Bayesian Optimization*. 2018. – URL https://arxiv.org/abs/1807.02811

[10] Huang, Jiangtao ; Gao, Zhenghong ; Zhou, Zhu ; Zhao, Ke: An improved adaptive sampling and experiment design method for aerodynamic optimization. In: *Chinese Journal of Aeronautics* 28 (2015), Nr. 5, S. 1391–1399. – URL https://www.sciencedirect.com/science/article/pii/S1000936115001570. – ISSN 1000-9361

[11] Kien T. Le, Abdoulaye D.: Randomization in Small Sample Surveys with the Halton Sequence. In: *Survey Research Methods* 12 (2018), Nr. 3, S. 247–257

[12] Lei, Bowen ; Kirk, Tanner ; Bhattacharya, Anirban ; Pati, Debdeep ; Qian, Xiaoning ; Arróyave, Raymundo ; Mallick, Bani: Bayesian optimization with adaptive surrogate models for automated experimental design. In: *npj Computational Materials* 7 (2021), 12

[13] Li, Yaohui ; Shi, Junjun ; Yin, Zhifeng ; Shen, Jingfang ; Wu, Yizhong ; Wang, Shuting: An Improved High-Dimensional Kriging Surrogate Modeling Method through Principal Component Dimension Reduction. In: *Mathematics* 9 (2021), Nr. 16. – URL https://www.mdpi.com/2227-7390/9/16/1985. – ISSN 2227-7390

[14] Mackman, Thomas ; Allen, C. ; Ghoreshi, Mehdi ; Badcock, Ken: Comparison of Adaptive Sampling Methods for Generation of Surrogate Aerodynamic Models. In: *AIAA Journal* 51 (2013), 4, S. 797–808

[15] Majdisova, Zuzana ; Skala, Vaclav: Radial basis function approximations: comparison and applications. In: *Applied Mathematical Modelling* 51 (2017), S. 728–743. – URL https://www.sciencedirect.com/science/article/pii/S0307904X17304717. – ISSN 0307-904X

[16] Majdisova, Zuzana ; Skala, Vaclav ; Smolik, Michal: Algorithm for placement of reference points and choice of an appropriate variable shape parameter for the RBF approximation. In: *Integrated Computer-Aided Engineering* 27 (2019), 09, S. 1–15

[17] Park, Changhwi ; Kim, Seong R. ; Kim, Young-Gu ; Kim, Dae S.: *Generative Evolutionary Strategy For Black-Box Optimizations.* 2022. – URL https://arxiv.org/abs/2205.03056

[18] Paszke, Adam ; Gross, Sam ; Massa, Francisco ; Lerer, Adam ; Bradbury, James ; Chanan, Gregory ; Killeen, Trevor ; Lin, Zeming ; Gimelshein, Natalia ; Antiga, Luca ; Desmaison, Alban ; Kopf, Andreas ; Yang, Edward ; DeVito, Zachary ; Raison, Martin ; Tejani, Alykhan ; Chilamkurthy, Sasank ; Steiner, Benoit ; Fang, Lu ; Bai, Junjie ; Chintala, Soumith: PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Wallach, H. (Hrsg.) ; Larochelle, H. (Hrsg.) ; Beygelzimer, A. (Hrsg.) ; Alché-Buc, F. (Hrsg.): *Advances in Neural Information Processing Systems 32.* Curran Associates, Inc., 2019, S. 8024–8035. – URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[19] Pedregosa, F. ; Varoquaux, G. ; Gramfort, A. ; Michel, V. ; Thirion, B. ; Grisel, O. ; Blondel, M. ; Prettenhofer, P. ; Weiss, R. ; Dubourg, V. ; Vanderplas, J. ; Passos, A. ; Cournapeau, D. ; Brucher, M. ; Perrot, M. ; Duchesnay, E.: Scikit-learn: Machine Learning in Python. In: *Journal of Machine Learning Research* 12 (2011), S. 2825–2830

[20] Skala, Vaclav ; Karim, Samsul Ariffin A. ; Zabran, Marek: Radial Basis Function Approximation Optimal Shape Parameters Estimation. In: Krzhizhanovskaya, Valeria V. (Hrsg.) ; Závodszky, Gábor (Hrsg.) ; Lees, Michael H. (Hrsg.) ; Dongarra, Jack J. (Hrsg.) ; Sloot, Peter M. A. (Hrsg.) ; Brissos, Sérgio (Hrsg.) ; Teixeira, João (Hrsg.): *Computational Science – ICCS 2020.* Cham : Springer International Publishing, 2020, S. 309–317. – ISBN 978-3-030-50433-5

[21] Virtanen, Pauli ; Gommers, Ralf ; Oliphant, Travis E. ; Haberland, Matt ; Reddy, Tyler ; Cournapeau, David ; Burovski, Evgeni ; Peterson, Pearu ; Weckesser, Warren ; Bright, Jonathan ; van der Walt, Stéfan J. ; Brett, Matthew ; Wilson, Joshua ; Millman, K. J. ; Mayorov, Nikolay ; Nelson, Andrew R. J. ; Jones, Eric ; Kern, Robert ; Larson, Eric ; Carey, C J. ; Polat, İlhan ; Feng, Yu ; Moore, Eric W. ; VanderPlas, Jake ; Laxalde, Denis ; Perktold, Josef ; Cimrman, Robert ; Henriksen, Ian ; Quintero, E. A. ; Harris, Charles R. ; Archibald, Anne M. ; Ribeiro, Antônio H. ; Pedregosa,

Fabian ; van Mulbregt, Paul ; SciPy 1.0 Contributors: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. In: *Nature Methods* 17 (2020), S. 261–272

[22] Wang, Yilun ; Shoemaker, Christine A.: *Sensitivity Analysis for Computationally Expensive Models using Optimization and Objective-oriented Surrogate Approximations*. 2014. – URL https://arxiv.org/abs/1410.7291

[23] Williams, Bianca ; Cremaschi, Selen: Novel Tool for Selecting Surrogate Modeling Techniques for Surface Approximation. 50 (2021), S. 451–456. – URL https://www.sciencedirect.com/science/article/pii/B9780323885065500711. – ISSN 1570-7946

[24] Younhee Choi, Sungmin Yoon Junemo K.: Comparison of Factorial and Latin Hypercube Sampling Designs for Meta-Models of Building Heating and Cooling Loads. In: *MDPI energies* (2021)

[25] Zuzana Majdisova, VaclavSkala: Radial Basis Function Approximations: Comparison and Applications. In: *arXiv* (2018)

A Appendix

The attached CD contains the following documents.

A.1 Code

A.2 Data and images investigations and benchmarks

A.3 Data and images FAME-W

Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit „– bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit [(§ 18 Abs. 1 APSO-TI-BM bzw. § 21 Abs. 1 APSO-INGI)] – ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen.“

Quelle: § 16 Abs. 5 APSO-TI-BM bzw. § 15 Abs. 6 APSO-INGI

Dieses Blatt, mit der folgenden Erklärung, ist nach Fertigstellung der Abschlussarbeit durch den Studierenden auszufüllen und jeweils mit Originalunterschrift als letztes Blatt in das Prüfungsexemplar der Abschlussarbeit einzubinden.
Eine unrichtig abgegebene Erklärung kann -auch nachträglich- zur Ungültigkeit des Studienabschlusses führen.

Erklärung zur selbstständigen Bearbeitung der Arbeit

Hiermit versichere ich,

Name: Max

Vorname: Sahlke

dass ich die vorliegende Masterarbeit ▼ bzw. bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit – mit dem Thema:

Development of an optimised design of experiments methodology for the creation of surrogate models

ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

- die folgende Aussage ist bei Gruppenarbeiten auszufüllen und entfällt bei Einzelarbeiten -

Die Kennzeichnung der von mir erstellten und verantworteten Teile der -bitte auswählen- ist erfolgt durch:

Hamburg	30.09.2022	
Ort	Datum	Unterschrift im Original