

Surrogate Modelling for Airfoil Shape optimization

K. van Velden

Delft University of Technology

Surrogate Modelling for Airfoil Shape Optimization

By

K. van Velden

in partial fulfilment of the requirements for the degree of

Master of Science

in Flight Performance and Propulsion

at the Delft University of Technology,
to be defended publicly on Wednesday August 16, 2017 at 9:30 AM.

Supervisor:	Dr. A. Elham	
Thesis committee:	Ir. J.A. Melkert,	TU Delft
	Dr. Ir. R. Vos,	TU Delft
	Dr. Ir. A.H. van Zuijlen,	TU Delft

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.



Acknowledgements

This thesis is performed for the flight performance and propulsion department in order to graduate from aerospace engineering. The motivation for this subject personally stems from interest in optimization methods and how broadly throughout different disciplines optimization can be applied. The reader is expected to have a base level knowledge regarding airfoil aerodynamics and optimization methods.

I'd like to thank my supervisor Ali for guiding me through the thesis research and the Thesis committee for evaluating the work performed. I would like to thank Koen van den Kieboom for helping me understand, implement and work with MSES.

Thanks to Juan Carlos for being readily available to answer any of my questions regarding surrogate models with his experience in the field.

And thanks to the SUMO team from Gent University for answering questions regarding the SUMO toolbox.

Besides the support on the technical front I've received much support and energy from close friends and Family which I thank them for. A final shoutout goes to my brother over in Australia. May we see each other more often.

*K. van Velden
Delft, August 2017*

Summary

Design optimization is paramount for aerospace designs. High fidelity aerodynamic solvers are computationally expensive and should be summoned as little as possible with the best results as possible. Gradient descent optimization techniques are generally fast to find local minimum, but only explore a limited design space. Ideally the global design optimum is found with the least amount of aerodynamic solver calls.

Surrogate model based optimization can be used to outperform gradient based optimizations for high dimensional and high fidelity complex problems. The quality of the surrogate model differs based on the parameterization, initial and sequential sampling plans and the surrogate model chosen. The goal of this thesis is to research the applicability of surrogate model based optimization for CST, Bezier-Parsec and NURBS parameterization for transonic airfoil optimization. The surrogate models converged to better airfoil designs for all parameterization except Bezier-Parsec. Bezier-Parsec parameterization contains equations that are unfeasible for evaluating sampling plans. Higher dimensionality parameterization required more time, but generally created better airfoils.

Five different sequential samplings have been researched. The LOLA-Voronoi and knowledge gradient of feasibility methods seemed most consistent for providing accurate surrogate models. Three different surrogate models have been researched. Ordinary Kriging generally outperformed GPML and RBF. Only when the sample size increased GPML starts to become competitive with Kriging.

The surrogate model based optimizations were able to find significantly different design spaces and as a result were able to achieve better optima than SQP descent optimization could. The surrogate model based optimizations mostly required more time to converge to an optimum than the gradient descent methods. NURBS without weights provided more accurate surrogate models than without weights. The CST parameterization proved most robust. Bezier-Parsec was unable to evaluate an initial sampling plan due to poor interaction with airfoil design space surrounding a flat leading edge camber curve.

Table of Contents

Preface/Acknowledgements.....	v
Summary.....	vii
List of figures.....	xiii
Nomenclature.....	xv
Greek Symbol	xv
Latin Symbol.....	xv
Acronyms.....	xvi
1 Introduction.....	1
2 Airfoil optimizations problem.....	4
3 Airfoil Parameterizations	5
3.1 CST	5
3.2 Bezier-PARSEC.....	6
3.3 NURBS	8
4 Surrogate model building.....	9
4.1 General surrogate model building	9
4.2 Initial Sampling.....	10
4.3 Sequential Sampling.....	12
4.4 Surrogate Models	13
4.5 Model Validation.....	15
5 Test Case application	16
5.1 Comparison of different parameterization.....	16
5.1.1 Fitting results	17
5.2 Aerodynamic solver	20
5.2.1 MSES setting 1	20
5.2.2 MSES setting 2	21
5.3 Gradient descent optimization:.....	22
5.4 Surrogate models based optimization.....	23
5.4.1 Surrogate models	23
5.4.2 BP3434	25
5.4.3 Model selection criterion	26
5.4.4 Validation error analysis.....	28
5.4.5 Optimizing the surrogate model	34
5.5 Results.....	35
5.5.1 Gradient descent optimization results.....	35
5.5.2 CST10 Surrogate model optimization results	37
5.5.3 CST14 Surrogate model optimization Results.....	39
5.5.4 uNURBS Surrogate model optimization results	41
5.5.5 NURBS Surrogate model optimization results	43
5.6 Comparison of optimization methods	45
5.6.1 CST10.....	45
5.6.2 CST14.....	46
5.6.3 uNURBS.....	48

5.6.4	NURBS.....	49
5.7	Comparison of parameterization results.....	51
6	Discussion.....	53
6.1	Parameter range.....	53
6.1.1	Parameter range influence on SQP optimization.....	53
6.1.2	Parameter range influence on SM based optimization.....	53
6.1.3	Parameter range used for this research.....	54
6.2	Surrogate model optimization versus gradient descent optimization.....	55
6.2.1	Effectiveness.....	55
6.2.2	Efficiency.....	55
6.2.3	Robustness.....	55
6.2.4	Feasibility.....	56
6.2.5	Influence of initial design.....	56
6.3	Sequential Sampling Strategy.....	57
6.3.1	Exploitation versus Exploration.....	57
6.4	Parameterization method.....	59
6.4.1	BP3434 failure.....	59
6.4.2	Dimensionality.....	59
6.4.3	CST versus NURBS.....	59
6.5	Sample Size.....	61
6.5.1	Error score.....	61
6.5.2	Surrogate Models and sample size.....	61
6.5.3	Sequential sampling and sample size.....	61
6.5.4	Sample size trade-off.....	61
7	Conclusions and Recommendations.....	63
Conclusions.....	63	63
CST10.....	63	63
CST14.....	64	64
BP3434.....	65	65
uNURBS.....	65	65
NURBS.....	66	66
General Conclusions.....	66	66
Recommendations.....	67	67
Recommendations for SM based optimization.....	67	67
Recommendation for continuing the research.....	68	68
Bibliography.....	69	69
Appendix.....	71	71
Appendix A – REA2822 parameter values and bounds.....	71	71
CST10.....	71	71
CST14.....	72	72
uNURBS.....	72	72
NURBS.....	73	73

Appendix B - MSES Aerodynamic Solver	75
Appendix C - SUMO Toolbox Configuration files.....	76
Appendix D – Surrogate model error validation data	82
Appendix E – Exploited surrogate models result.....	87

List of figures

Figure 1 - Bernstein Polynomials.	6
Figure 2 - BP3434 Airfoil geometry and control points	7
Figure 3 - NURBS Airfoil parameterization.....	8
Figure 4 - Surrogate model building flowchart.....	9
Figure 5 - Two different valid Latin-Hypercube Designs.	10
Figure 6 - Stepwise procedure of TPLHD.	11
Figure 7 - REA2822 Airfoil with camber line	16
Figure 8 - CST10 REA2822 Fitted airfoil	17
Figure 9 - CST14 REA2822 fitted airfoil.....	18
Figure 10 - BP3434 REA2822 fitted airfoil.	18
Figure 11 - uNURBS fitted airfoil.	19
Figure 12 - NURBS fitted airfoil.	19
Figure 13 - Gradient Descent optimization flowchart.....	22
Figure 14 - Surrogate model based optimization.	24
Figure 15 - BP3434 failure	25
Figure 16 - RMS pitching constraint against RMS objective function	26
Figure 17 - MAE pitching constraint against MAE objective function.....	27
Figure 18 - r^2 pitching constraint against r^2 objective function.....	27
Figure 19 - CST10 MAE objective against RMS objective.	28
Figure 20 - CST10 r^2 objective against RMS*MAE objective.....	29
Figure 21 - CST14 MAE objective against RMS objective.	30
Figure 22 - CST14 r^2 objective against RMS*MAE objective.....	30
Figure 23 - uNURBS MAE objective against RMS objective.	31
Figure 24 - uNURBS r^2 objective against RMS*MAE objective.	32
Figure 25 - NURBS MAE objective against RMS objective.	33
Figure 26 - NURBS r^2 objective against RMS*MAE objective.....	33
Figure 27 - Gradient descent optimized airfoil pressure distributions.....	36
Figure 28 - Gradient descent optimized airfoils.....	36
Figure 29 - CST10 surrogate model optimized airfoil pressure distributions.	38
Figure 30 - CST10 Surrogate model optimized airfoils.....	38
Figure 31 - CST14 surrogate model optimized airfoil pressure distributions.	40
Figure 32 - CST14 surrogate model optimized airfoils.	40
Figure 33 - uNURBS surrogate model optimized airfoil pressure distributions.	42
Figure 34 - uNURBS surrogate model optimized airfoils.	42
Figure 35 - NURBS surrogate model optimized airfoil pressure distributions.	44
Figure 36 - NURBS surrogate model optimized airfoils.	44
Figure 37 - CST10 optimized airfoil pressure distribution comparison.....	45
Figure 38 - CST10 optimized airfoil comparison.....	46
Figure 39 - CST14 optimized airfoil pressure distribution comparison.....	47
Figure 40 - CST14 optimized airfoils comparison.	47
Figure 41 - uNURBS optimized airfoil pressure distribution comparison.	48
Figure 42 - uNURBS optimized airfoils comparison.	49
Figure 43 - NURBS optimized airfoil pressure distribution comparison.....	50
Figure 44 - NURBS optimized airfoil comparison.....	50
Figure 45 - Surrogate model optimized airfoil pressure distributions.	52
Figure 46 - Surrogate model optimized airfoils.	52

Nomenclature

Greek Symbol

Symbol	Description	Unit
α_{te}	Trailing camber line angle	[-]
β_{te}	Trailing wedge angle	[-]
γ_{le}	Leading edge direction	[-]
θ	Hyperparameters	[-]
μ	Base term	[-]
ψ	Basis function	[-]
Ψ	Correlation matrix, Gram matrix	[-]
ω_j	Weighting factor	[-]

Latin Symbol

Symbol	Description	Unit
$b_{0,2,8,15,17}$	Bezier variables	[-]
C	Class shape function	[-]
c	Non-Dimensional Chord length	[-]
c_1	First constraint, thickness constraint	[-]
c_2	Second constraint, Pitching moment constraint	[-]
Cd	Drag coefficient	[-]
Cd_i	Initial drag coefficient	[-]
Cm	Pitching moment coefficient	[-]
Cm_i	Initial pitching moment coefficient	[-]
dz_{te}	Half thickness of the trailing edge	[-]
J	Objective Function	[-]
k	Number of dimensions	[-]
K	Covariance Matrix	[-]
M	Mach Number	[-]
n	Number of variables	[-]
n_b	Number of blocks	[-]
n_d	Number of Divisions	[-]
$N_{i,D}$	B-Spline basis functions	[-]
n_p	Number of points	[-]
n_s	Seed number	[-]
n_v	Number of variables	[-]
P_i	Control points	[-]
r^2	Root squared error	[-]
S	Shape functions	[-]
t	Non-Dimensional airfoil thickness	[-]
$v_{KG,n}$	Knowledge Gradient	[-]
w	Weighting factors	[-]
Xx	Non-Dimensional horizontal position of airfoil points	[-]
x_c	Non-Dimensional horizontal position of camber crest	[-]
x_t	Non-Dimensional horizontal position of thickness crest	[-]
y	Non-Dimensional vertical position of airfoil points	[-]
y_c	Non-Dimensional vertical position of camber crest	[-]
y_t	Non-Dimensional vertical position of thickness crest	[-]
$Y(x)$	Value of evaluated sample point, Training data	[-]
$\hat{Y}(x)$	Estimated value of sample point	[-]
Z_{te}	Trailing edge vertical displacement	[-]

Acronyms

Acronym	Description	Unit
2D	2 Dimensional	
AoA	Angle of Attack	[°]
BP	Bezier Parsec	
CST	Class-Shape Transformation	
CST10	Class-Shape Transformation with 10 parameters	
CST14	Class-Shape Transformation with 14 parameters	
GA	Genetic Algorithm	
GD	Gradient Descent	
GPML	Gaussian Process for Machine Learning	
Krig	Kriging	
LE	Leading Edge	
LHD	Latin Hypercube Design	
MAE	Maximum absolute error	
MLE	Moving Least Square Error	
NURBS	Non-Uniform Rotation B-Splines	
Obj	Objective function	
RMS	Root mean Squared error	
TE	Trailing edge	
TPLHD	Translational Propagation Latin Hypercube Design	
SM	Surrogate Model	
SQP	Sequential Quadratic Programming	
uNURBS	Unweighted Non-Uniform Rotation B-Splines	

1 Introduction

In today's engineering world conceptual and preliminary designs are often created with the help of optimization strategies and algorithms. The ideal optimization strategy is able to find the global minimum with as little computational cost possible. Finding the global minimum is important as a better design results in a better final product. Reducing computational costs allow the engineers to evaluate more viable interesting designs in shorter time. This thesis focuses on airfoil shape optimization for drag minimization. The aim of the thesis is to increase the understanding of different optimization strategies and their applicability to airfoil optimization.

Airfoil shape optimizations require aerodynamic solvers. Good quality aerodynamic solvers are computationally expensive. Airfoil optimization is subject to a thickness constraint and a pitching moment constraint. The aerodynamic solver obtains an airfoil design as input and calculates an angle of attack such that a set value for the lift-coefficient is achieved. The corresponding drag coefficient and pitching moment constraints are used for the objective function and constraint. The design parameters depend on the parameterization method used.

The parameterization method has large influence on the design landscape and complexity of the problem. Different parameterization have different qualities such as the complex designs that can be created, flawed unfeasible designs that can be created, number of parameters that increase the dimensionality of the optimization problems and intuitiveness. Selecting a fitting optimization strategy is paramount for reaching successful results.

Gradient descent optimization is often used for airfoil optimization. The descent optimization starts at an initial design and calculates gradient information. The gradient information is used to make a step towards a better design. This is repeated until a minimum is found. Whilst being a relatively fast optimization methods there are two main problems with this optimization strategy for high-fidelity complex problems. Gradient descent optimizations are highly reliant on the initial starting design and only explore a relative small part of the design space is explored before finding a minimum.

Surrogate model based optimization is more useful when global optimization is performed. The optimization strategy used for this research works as follows: First an initial sampling plan is created. The design samples selected by the sampling plan are evaluated by the high-fidelity solver. A surrogate model is fitted through the evaluated sample points. The surrogate model estimates the objective function value with the parameters as input. The surrogate model is updated with sequential samples until a stopping criterion is met. The surrogate model is cheap and fast to evaluate. A genetic optimization is applied to the surrogate model that attempts to find the estimated global minimum. This estimated global minimum points towards interesting design space that will be investigated further by applying the same aforementioned gradient descent optimization. The gradient descent optimization is the final step and uses the estimated global minimum as initial starting point and uses the high-fidelity solver to find the estimated global minimum. The surrogate model based optimization strategy attempts to understand the full design space by the user and when done properly can result in fewer high-fidelity solver summons whilst finding a better design than gradient descent optimization can do.

This thesis researches 3 parameterization methods, 5 sequential sampling strategies and 3 different surrogate models. The resulting surrogate models are validated with error analysis. The surrogate models with the most promising validation error scores are further optimized with the genetic optimization and gradient descent search. The final results are compared to a direct gradient descent optimization. The different optimizations are rated on robustness, efficiency, effectiveness and design space they converge to. Surrogate model based optimization has been researched before with regards to airfoils [1][2][3][4]. The research presented in this thesis adds to the field of research as multiple parameterizations are and sequential samples design are compared and researched. The main research question is formulated as: how applicable is surrogate model based optimization for different parameterization methods and what strategy is recommended regarding effectiveness, efficiency and robustness.

The thesis report is structured such that initially the required background information required for airfoil shape optimization is presented in chapter 2. Chapter 3 presents the airfoil parameterization techniques used. Chapter 4 describes how to build a surrogate model and presents the surrogate model methods used. Chapter 5 is the test case application chapter and describes how the research is performed and presents the results. A discussion is presented in chapter 6. Finally the conclusions and recommendations are given in chapter 7.

At the end of the report the reader should have a clear understanding of how surrogate model based optimization works and which trade-offs should be made with regards to select an optimization strategy, airfoil parameterization, sampling plan and surrogate model in order design a successful optimization strategy for airfoil shape optimization.

2 Airfoil optimizations problem

Optimization problems require an objective function to minimize and are often subject to constraints. For airfoil drag optimization the objective function is defined as the normalized drag value shown in equation 1. The airfoil optimization is performed by setting the lift coefficient (C_l) to a specific value and the solver calculates the angle of attack (AoA). Two constraints are used for this optimization problem. Only the objective function and the pitching moment constraint require a high-fidelity solver.

The first constraint is the thickness constraint shown in equation 2, the optimized airfoil should have not have a smaller maximum thickness due to fuel space. The second constraint is the pitching moment constraint shown in equation 3. The optimized airfoil should not have a larger pitching moment than the initial airfoil. The pitching moment should remain below 0 for stability reasons, but not get a larger negative value. Drag minimization optimizations usually have an active pitching moment constraint thus the stability criterion does not have to be forced.

$$J = \frac{Cd}{Cd_i} \quad (1)$$

$$c_1 = \frac{(t/c_{\max_i} - t/c_{\max})}{t/c_{\max_i}} \quad (2)$$

$$c_2 = \frac{(Cm_i - Cm)}{|Cm_i|} \quad (3)$$

Cd_i , t/c_{\max_i} and Cm_i are the initial design drag coefficient, maximum thickness and pitching moment coefficient. These are different for each parameterization. Chapter 5.1 and 5.2 specify how the initial values are computed.

3 Airfoil Parameterizations

Selecting an appropriate airfoil parameterization is important when considering optimization. This subsection initially explains some criterion to consider when selecting an airfoil parameterization. Afterwards the airfoil parameterizations selected for this research are discussed.

There are 5 criteria to consider when selecting a parameterization technique for optimization techniques [5].

- Parsimoniousness. Parsimoniousness depends on the amount of variables that are required to describe the airfoil. More parameters implies a larger dimensionality. Following the curse of dimensionality [6] a lower variable count can be paramount to an efficient optimization.
- Completeness. A parameterization method that can cover a large design space is often able to create better and more complex airfoils. Keep in mind that being able to cover a large design space often goes hand in hand with the ability to create unfeasible uninteresting exotic airfoils. Evaluating these airfoils is computational waste.
- Orthogonality. Orthogonality guarantees that each airfoil shape corresponds to an unique set of parameters. This is specifically important to reduce the computational time used for evaluating identical airfoils.
- Flawlessness. A flawless method does not create faulty airfoil designs. Evaluating faulty airfoils can be considered wasteful. Reducing worthless design space is beneficial for surrogate model based optimization.
- Intuitiveness. An intuitive method contains easy to understand parameters that relate to the airfoil created. Examples of intuitive parameters are leading edge radius and maximum thickness. With regards to optimization intuitiveness allows the user to set appropriate bounds on the parameters easier.

This thesis researches three different airfoil parameterizations for surrogate model based optimization. The methods are Class-Shape Transformation (CST) [7], Bezier-PARSEC (BP) [8] and Non-Uniform Rotational B-Splines (NURBS) [9].

3.1 CST

The CST method represents a two-dimensional geometry by the product of a class function, $C\left(\frac{x}{c}\right)$, a shape function $S\left(\frac{x}{c}\right)$ and a term that characterizes the trailing edge thickness.

$$\frac{y}{c} = C\left(\frac{x}{c}\right) S\left(\frac{x}{c}\right) + \frac{x \Delta z_{te}}{c} \tag{4}$$

With

$$C\left(\frac{x}{c}\right) \equiv \left(\frac{x}{c}\right)^{N_1} \left[1 - \frac{x}{c}\right]^{N_2} \text{ for } 0 \leq \frac{x}{c} \leq 1 \tag{5}$$

Where for airfoil $N_1 = .5$ and $N_2 = 1$.

To get to the shape function is based on weighted Bernstein binominals shown in figure 1:

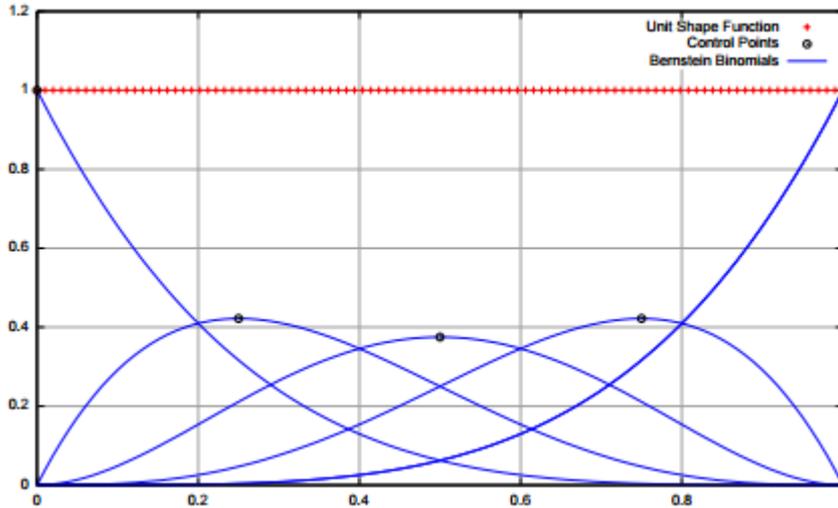


Figure 1 - Bernstein Polynomials. [7]

The values of the weighted factors are the design parameters. The order can be increased to increase the flexibility, completeness and accuracy.

The CST method always produces smooth surfaces and flexible method such that it rarely creates faulty airfoil designs. A downside of the method is that any change made is felt over the entire airfoil, thus not allowing small local changes.

3.2 Bezier-PARSEC

There is a third degree Bezier curve and a fourth degree Bezier curve Bezier-PARSEC method. For this research the fourth degree Bezier curve method is used and is referred to as BP3434. BP3434 uses 10 intuitive parameters and 5 Bezier parameters. The method has 15 parameters in total. Figure 2 presents a visual representation of the parameters. The Bézier-PARSEC method is a relatively new method [8] that has been developed to extend and improve the typical Bezier parameterization. The motivation for the development of this method is a reduction in nonlinear interaction of parameters and to have the parameters more directly linked to the objective function. This will enable the optimizer to converge more quickly. The Bezier-Parsec uses a second order continuity. There are 2 different Bezier-PARSEC methods.

A fourth degree Bezier curve is given by:

$$x(u) = x_0(1 - u)^4 + 4x_1u(1 - u)^3 + 6x_2u^2(1 - u)^2 + 4x_3u^3(1 - u) + x_4u^4 \quad (6)$$

And

$$y(u) = y_0(1 - u)^4 + 4y_1u(1 - u) + 6y_2u^2(1 - u)^2 + 4y_3u^3(1 - u) + y_4u^4 \quad (7)$$

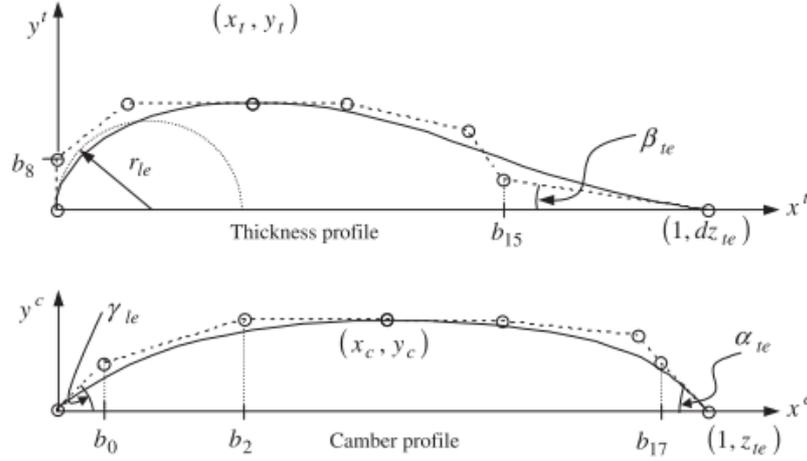


Figure 2 - BP3434 Airfoil geometry and control points [8]

The curves are presented in the following order: leading edge thickness curve, trailing edge thickness curve, leading edge camber curve and trailing edge camber curve respectively.

The leading edge thickness curve control points are given by:

$$\begin{array}{ll}
 x_0 = 0 & y_0 = 0 \\
 x_1 = 0 & y_1 = b_8 \\
 x_2 = -3b_8^2/2r_{le} & y_2 = y_t \\
 x_3 = x_t & y_3 = y_t
 \end{array} \tag{8}$$

With b_8 limited to:

$$0 < b_8 < \min(y_t, \sqrt{-2r_{le}x_t/3}) \tag{9}$$

The trailing edge thickness curve control points are given by:

$$\begin{array}{ll}
 x_0 = x_t & y_0 = y_t \\
 x_1 = (7x_t + 9b_8^2/2r_{le})/4 & y_1 = y_t \\
 x_2 = 3x_t + 15b_8^2/4r_{le} & y_2 = (y_t + b_8)/2 \\
 x_3 = b_{15} & y_3 = dz_{te} + (1 - b_{15}) \tan(\beta_{te}) \\
 x_4 = 1 & y_4 = dz_{te}
 \end{array} \tag{10}$$

The leading edge camber curve control points are given by:

$$\begin{array}{ll}
 x_0 = 0 & y_0 = 0 \\
 x_1 = b_0 & y_1 = b_0 \tan(\gamma_{le}) \\
 x_2 = b_2 & y_2 = y_c \\
 x_3 = x_c & y_3 = y_c
 \end{array} \tag{11}$$

The trailing edge camber curve control points are given by equation 12:

$$\begin{array}{ll}
 x_0 = x_c & y_0 = y_c \\
 x_1 = (3x_c - y_c \cot(\gamma_{le}))/2 & y_1 = y_c \\
 x_2 = (-8y_c \cot(\gamma_{le})) & y_2 = 5y_c/6 \\
 x_3 = b_{17} & y_3 = z_{te} + (1 - b_{17}) \tan(\alpha_{te}) \\
 x_4 = 1 & y_4 = z_{te}
 \end{array} \tag{12}$$

3.3 NURBS

The NURBS parameterization creates a spline using B-spline basis functions and weighted control points [9]. A p -th degree of B-spline basis functions and a vector-valued function given by:

$$y(x) = \sum_{i=0}^n R_{i,p}(x) \mathbf{P}_i \quad (13)$$

With

$$R_{i,p}(x) = \frac{N_{i,p}(x)\omega_i}{\sum_{j=0}^n N_{j,p}(x)\omega_j} \quad (14)$$

Where \mathbf{P}_i are the control points, ω_i the weights of the control points, $N_{i,p}$ are the p -th B-spline basis functions and $y(x)$ is the position of a point on the curve. An airfoil created with NURBS geometry is presented in the figure 3.

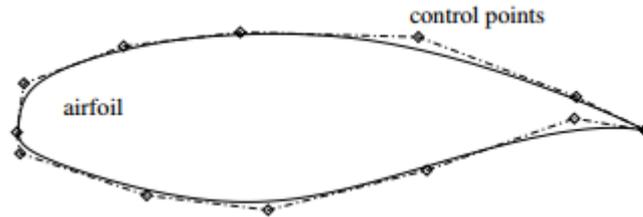


Figure 3 - NURBS Airfoil parameterization. [9]

The basis functions are determined through a knot vector. The knot vector defines the break points of the function. The knot vector is in the form of:

$$\underbrace{\{0, \dots, 0\}}_p, x_{p+1}, \dots, x_{m-p+1}, \underbrace{\{1, \dots, 1\}}_p \quad (15)$$

Increasing the order of the basis functions increases the order of continuity. With basis functions degree of 3 all the curves will be C2 continuous outside of leading and trailing edge.

The NURBS method provides accurate results for optimization for a variety of airfoils with 13 control points or less. For every control point there are 3 design variables namely x, y coordinates and the weighting factor. For typical transonic airfoils at least 13 control points are required. With 13 control points the amount of design variables for an airfoil is equal to 34 including the angle of incidence [9].

4 Surrogate model building

This chapter addresses the theoretical background of building a surrogate model. The theoretical background is limited to the options used for this research. Firstly a surrogate model building scheme is presented. Section 4.2 presents the initial sampling strategy. Section 4.3 presents the sequential sampling plans. Section 4.4 shows the surrogate models used. And finally the validation methods are shown in section 4.5.

4.1 General surrogate model building

To create a surrogate model various steps have to be performed. Figure 4 shows the surrogate model flow chart used by SUMO [10].

Firstly sample points are generated with an initial sampling plan. The initial sample points are evaluated with the high-fidelity solver. A surrogate model is created using the initial training data. The surrogate model attempts to fit the data points accurately and estimates the objective function and constraint behavior at intermediate design space.

The toolbox updates the model until it cannot improve the root relative square error of the model. If no stopping criterion is met an iterative process is started. The sequential sampling process places new points that are evaluated with the high-fidelity solver. The surrogate model is fitted to the new points and validated with root relative square error again until a stopping criterion is met.

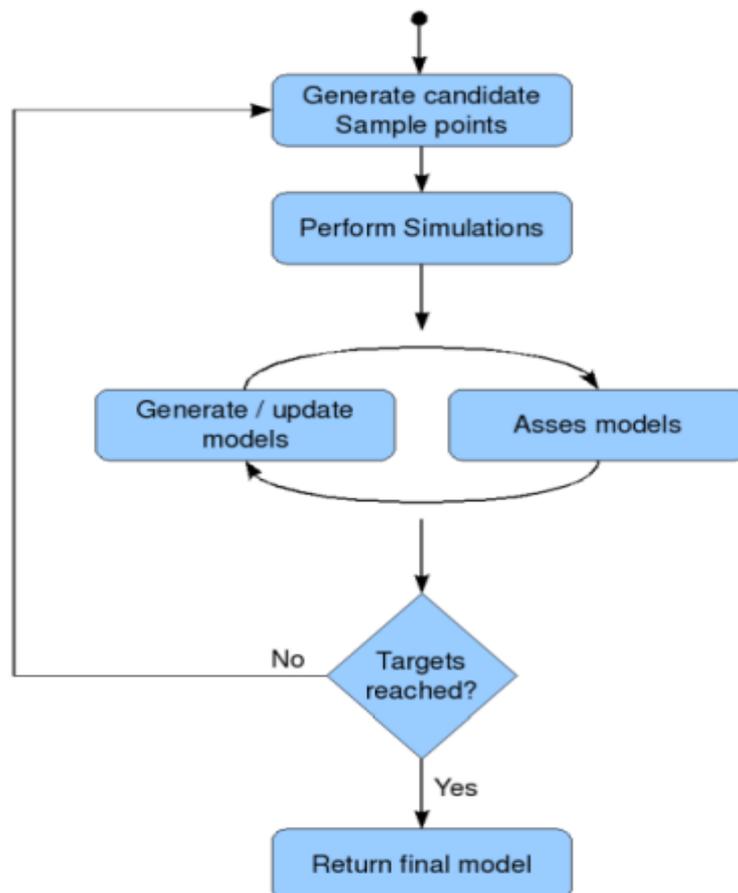


Figure 4 - Surrogate model building flowchart. [10]

4.2 Initial Sampling

For initial sampling plans there are multiple options. Understanding the impact of the initial sampling plan on the rest of the surrogate model building helps with selecting an appropriate sampling plan. The initial knowledge on which the surrogate model and sequential sampling make decisions is based on the initial sampling. A good initial sampling plan gives the sequential sampling and surrogate modeling a good starting direction and provides good coverage over the chosen design space. A poor initial design provides poor information and the sequential sampling and surrogate modeling process will require resources to recover from this. The computational cost for a poor initial sampling plan is thus two-fold. The initial sample and the extra samples that are required to obtain the information the initial sampling plan should have provided. And due to the lack of information it might be troublesome for the sequential sampling plan to understand where to place samples to obtain the required information.

A good spread of data points is preferred for the surrogate model to obtain a better understanding of the interactions between parameters and objective function. Depending on the problem formulation extreme corner points might be avoided as these can be unfeasible designs.

Two different initial sampling strategies are used. The space-filling Latin-Hypercube Design (LHD) [16] and a variation of LHD using a Translational Propagation algorithm (TPLHD) [11] are used. Latin-Hypercube Design divides the design space for each parameter in columns equal to the sample size. LHD sample plans can result in cluster of points and large portions of designs space left unsampled as seen in figure 5. TPLHD divided the design space in extra boxes following equation 16. Figure 6 shows a TPLHD process with seed number n_s set at 1, and the number of points, n_p at 16 for a dimensionality of n_v of 2.

$$n_b = \frac{n_p}{n_s} \quad (16)$$

With each dimension being partitioned into equal number of divisions n_d given by:

$$n_d = (n_b)^{1/n_v} \quad (17)$$

The design then starts with 1 box and for each dimensional adjacent box the new sample point is placed on the next column respective to that parameter. The process is shown in figure 6.

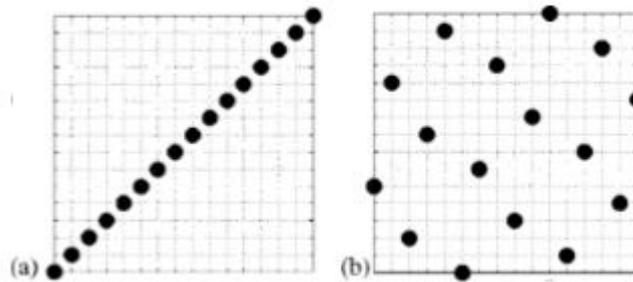


Figure 5 - Two different valid Latin-Hypercube Designs. [11]

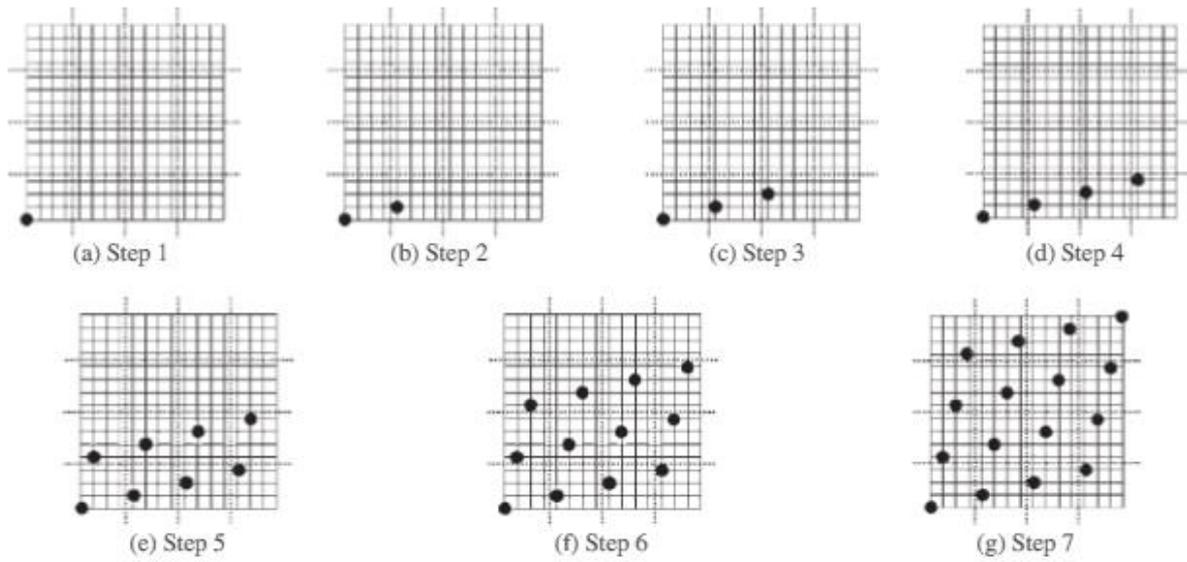


Figure 6 - Stepwise procedure of TPLHD. [11]

TPLHD requires more time to compute the optimal sampling design. Increased dimensionality results in ineffective computational times for the initial sampling plan. LHD is preferred over TPLHD for larger dimensional problems.

4.3 Sequential Sampling

The selected method for sequential sampling is important as this process selects the new data points to fit the model through. An important trade-off when selecting a sequential design is the trade-off between exploration and exploitation. Exploration equates to covering the entire design space and sampling at unsampled design space. A sequential sampling plan that lacks exploration qualities is prone to miss interesting design space. Exploitation implies investigating a smaller design space, especially interesting and feasible design space should be exploited. Sequential sampling techniques without exploitation qualities risk misunderstanding complicated design space.

Five different sequential sampling methods have been selected to research. Take note of the respective letters to the sequential sampling plans as the results will use these letters to refer to the sampling plan.

A. LOLA-Voronoi

LOLA-Voronoi sampling has exploitation and exploration qualities [12]. The exploitation is based on local linear approximations (LOLA). This approximation serves that less points are placed at linear predictable design space and more points are allocated at non-linear design space. The local linear approximation is the gradient of function f given by equation 18.

$$\nabla f = \left(\frac{\delta f}{\delta x_1}, \frac{\delta f}{\delta x_2}, \dots, \frac{\delta f}{\delta x_v} \right) \quad (18)$$

The gradient is estimated using least square regression on neighborhood samples. The difference between the actual location of neighborhood samples and the estimation through the gradient depicts the linearity. Only using exploitation as a sequential sampling method will result in a convergence at the highest initially estimated non-linear area. This is where the Voronoi criterion comes in, as it offers exploration qualities. The Voronoi tessellation estimates the cell size of all the samples. Samples with a large cell size have respectively much unsampled design space surrounding them and this design space is thus selected to be sampled. All points are ranked based on a weighted measure of the LOLA error and Voronoi cell size. The highest points are either in unsampled region, non-linear region or both. New points are then allocated to these regions and computed.

B. Generalized Probability of Feasibility

The generalized probability [13] of feasibility sequential sampling allows the user to specify a range for the objective function in which the selected samples are estimated to lie. This allows the user to exclude regions of exotic unfeasible airfoils. The generalized probability follows the Gaussian distribution using maximum likelihood to estimate the objective function.

C. Expected Improvement

Expected improvement [14] compares possible new points estimation with the current best evaluated point. If the evaluated point is expected to be better the point will be sampled. This method strongly depends on the initial samples location and objective function. The method maximizes equation 19.

$$x = \begin{cases} y_{min}^n - \hat{y}(x), & y_{min}^n - \hat{y}(x) > 0, \\ 0 & , y_{min}^n - \hat{y}(x) \leq 0 \end{cases} \quad (19)$$

Expected improvement contains mainly exploitation qualities as unsampled areas are unlikely to be investigated.

D. Knowledge Gradient

The knowledge-gradient policy uses the computed samples and model information to calculate where the next sample points should be located [15]. The samples are placed at locations that have the estimated largest information value for the surrogate model. The result is that knowledge gradient samples away from earlier sampled points and provides a good coverage in the process. The knowledge gradient is defined as

$$v_{\text{KG},n} = \mathbb{E} \left[\max_{\mu \in \mathbf{x}} \mu^{n+1}(u) | x_n = x \right] - \max_{\mu \in \mathbf{x}} \mu^n(u) \quad (20)$$

The sequential samples x^n are selected as the maximum over the knowledge gradient:

$$x^n \in \underset{x \in \mathbf{x}}{\text{arg max}} v_{\text{KG},n(x)} \quad (21)$$

The Knowledge gradient has exploring and exploiting properties. The exploitation quality stems from the sampling in areas where μ^n is maximized. The exploration qualities are obtained as it does not want to sample in places the method already sampled at.

E. Default

The default sequential sampling strategy is a mix between error based sampling and LOLA-Voronoi (30-70 ratio). The SUMO toolbox allows for a mixture of sequential sampling with weights. The default recommended setting for this is error-based sampling with LOLA-Voronoi. LOLA-voronoi has been explained above. Error sampling tries to sample points where the error is the largest using 5-fold cross validation on a dense grid. The error method samples at location where the model error is estimated to be the largest [16]. The mixture of error and LOLA-Voronoi contains higher exploration qualities than LOLA-Voronoi has.

4.4 Surrogate Models

The surrogate model is used to estimate the objective function and pitching moment constraint based on parameter input. Three different surrogate models have been researched based on what is available in SUMO and recommended in literature. The surrogate models tested are Radial-Basis-Functions (RBF) [17], Kriging [17] and Gaussian-Process for Machine Learning (GPML) [18].

Radial-Basis Functions

RBF consists of a weighted sum of simple functions that estimate the objective function. Examples of the functions used are listed below:

$$\text{Linear:} \quad \psi(r) = r \quad (22)$$

$$\text{Cubic:} \quad \psi(r) = r^3 \quad (23)$$

$$\text{Thin Plate Spline:} \quad \psi(r) = r^2 \ln r \quad (24)$$

$$\text{Gaussian:} \quad \psi(r) = e^{-r^2/2\sigma^2} \quad (25)$$

$$\text{Multi-Quadric:} \quad \psi(r) = (r + \sigma^2)^{1/2} \quad (26)$$

$$\text{Inverse Multi- Quadric:} \quad \psi(r) = (r + \sigma^2)^{-1/2} \quad (27)$$

The RBF is in the form of:

$$\hat{f}(\mathbf{x}) = \mathbf{w}^T \boldsymbol{\psi} = \sum_{i=1}^{n_c} W_i \psi(\|\mathbf{x} - \mathbf{c}^{(i)}\|) \quad (28)$$

$\mathbf{c}^{(i)}$ is the i th of n_c basis function centers, $\boldsymbol{\psi}$ is an n_c vector containing the values of the basis functions evaluated at the Euclidean distance between the prediction site \mathbf{x} and the centers $\mathbf{c}^{(i)}$.

\mathbf{W} still needs to be estimated which can be done through interpolating the following condition:

$$\hat{f}(\mathbf{x}^{(j)}) = \sum_{i=1}^{n_c} W_i \psi(\|\mathbf{x} - \mathbf{c}^{(i)}\|) = y^{(j)}, \quad j = 1, \dots, n \quad (29)$$

One of the conditions to obtain a unique solution of the equation above is that $n_c = n$. If $\mathbf{c}^{(i)} = \mathbf{x}^{(i)}$ than the following equation can be derived.

$$\boldsymbol{\Psi} \mathbf{w} = \mathbf{y} \quad (30)$$

Where $\boldsymbol{\Psi}$ is the Gram matrix defined as:

$$\Psi_{i,j} = \psi(\|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\|), \quad i, j = 1, \dots, n \quad (31)$$

The RBF model used for this research uses Gaussian, Multi-Quadric and an exponential basis functions.

Kriging

The main formula for the Kriging model is the basis-function depicted in equation 32. The Kriging method applied is ordinary Kriging using maximum likelihood to estimate the hyper parameters.

$$\psi^{(i)} = \text{cor}[Y(\mathbf{x}^{(i)}), Y(\mathbf{x})] = \exp\left(-\sum_{j=1}^k \theta_j |x_j^{(i)} - x_j|^{p_j}\right) \quad (32)$$

Where $Y(\mathbf{x})$ are random variables assumed to be the observed responses. The variance of the basis function can be controlled in each dimension in the design space by θ_j , the exponent can be varied in each dimension by p_j . The parameters θ_j and p_j are still unknown. They are chosen to be moving least squares. The natural logarithm of the likelihood without constant terms has to be maximized, this gives the concentrated In-likelihood function:

$$\ln(L) \approx -\frac{n}{2} \ln(\hat{\sigma}^2) - \frac{1}{2} \ln(|\boldsymbol{\Psi}|) \quad (33)$$

With

$$\hat{\sigma}^2 = \frac{(\mathbf{y} - \mathbf{1}\hat{\mu})^T \boldsymbol{\Psi}^{-1} (\mathbf{y} - \mathbf{1}\hat{\mu})}{n} \quad (34)$$

Where $\boldsymbol{\Psi}$ is a $n \times n$ matrix of correlations between the observed data. With each element given by equation 32.

Because of the use of MLE of θ_j and p_j for the parameter estimation some insight of the design landscape is obtained. A low value of p_j indicates low correlation between points. A higher value for p_j indicates correlation and continuity through the sample points.

The parameter θ_j indicates how active and influential a sample point is. A low value indicates all sample points have high correlations. The parameter θ_j also helps determining which design variables are the most important and which variables are not. It is possible to eliminate the variables that have little influence.

Now that θ_j and p_j are estimated the function prediction at unknown \mathbf{x} can be created.

$$\hat{y}(\mathbf{x}) = \hat{\mu} + \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1} (\mathbf{y} - \mathbf{1}\hat{\mu}) \quad (35)$$

Where

$$\hat{\mu} = \frac{\mathbf{1}^T \boldsymbol{\Psi}^{-1} \mathbf{y}}{\mathbf{1}^T \boldsymbol{\Psi}^{-1} \mathbf{1}} \quad (36)$$

Now that the model is known it is important to validate the surrogate model by computing a validation error. The moving squared error for a Kriging model is:

$$s^2(\mathbf{x}) = \sigma^2 \left[1 - \boldsymbol{\psi}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi} + \frac{1 - \mathbf{1}^T \boldsymbol{\Psi}^{-1} \boldsymbol{\psi}}{\mathbf{1}^T \boldsymbol{\Psi}^{-1} \mathbf{1}} \right] \quad (37)$$

Gaussian-Process for Machine Learning

GPML is similar to Kriging, however it uses the conjugate gradient in addition to the maximum likelihood estimate to calculate the hyper parameters, θ , build the surrogate model. The marginal likelihood gradient is computed by equation 38.

$$\begin{aligned} \frac{\delta}{\delta \theta_j} = \log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) &= \frac{1}{2} \mathbf{y}^T K^{-1} \frac{\delta K^{-1}}{\delta \theta_j} K^{-1} \mathbf{y} - \frac{1}{2} \text{tr} \left(K^{-1} \frac{\delta K}{\delta \theta_j} \right) \\ &= \frac{1}{2} \text{tr} \left((\boldsymbol{\alpha} \boldsymbol{\alpha}^T - K^{-1}) \frac{\delta K}{\delta \theta_j} \right) \end{aligned} \quad (38)$$

With $\boldsymbol{\alpha} = K^{-1} \mathbf{y}$, where K is the covariance matrix and \mathbf{y} is the training data.

4.5 Model Validation

After the surrogate model is created three error measurements are taken. These give an indication of the models accuracy. However they are not waterproof, meaning that a model with poor error scores could outperform a model with good error scores. A good model score means it is more likely to have a good model fit.

The three error measurements presented below are typical validation errors [19]:

- Root-mean squared error (RMS)

$$RMS = \sqrt{\frac{1}{q} \sum (y_i - \hat{y})^2} \quad (39)$$

- Maximum absolute error (MAE)

$$MAE = \text{Max}_q (|y_i - \hat{y}_i|) \quad (40)$$

- Root squared error (r^2)

$$r^2 = 1 - \left[\frac{\sum_q (y_i - \hat{y})^2}{\sum_q (y_i - \bar{y})^2} \right] \quad (41)$$

For good models RMS and MAE should be as close to 0 as possible and r^2 should be as close to 1 as possible.

5 Test Case application

This chapter explains how the research is set up, which optimizations have been performed and presents the results. The surrogate model based optimizations are compared to a gradient descent optimization. Initially in section 5.1 a method to compare different parameterization is set up. Secondly in section 5.2 the aerodynamic solver settings are shown. Afterwards in section 5.3 the gradient descent optimization settings are presented. Section 5.4 shows the various surrogate models built for each parameterization, a selection of which surrogate models to further exploit and presents the further exploitation method. The results are presented in section 5.5. Section 5.6 compares the different optimization results for each parameterization. Finally section 5.7 presents a comparison on the airfoil parameterization results.

5.1 Comparison of different parameterization

To be able to compare the different parameterization techniques a common initial starting design is selected. The airfoil design of the REA2822 airfoil presented in figure 7 is selected. The different parameterizations are fitted to the REA2822 and each respective fitting is used as initial design throughout the research. Initially the fitting method is explained and finally the results are given for CST, BP and NURBS.

For CST 5 variants have been fitted to investigate the ability of curve manipulation compared to parameter count. The lowest CST variant has 10 parameters and the largest 18. Two variants will be selected based on their fitting error and corresponding parameter count.

For BP the trailing edge gap is set to 0 such that the total amount of parameters is reduced to 14.

NURBS is fitted with two variants. A weighted and an unweighted version with 7 upper curve control points and 7 lower curve control points. The unweighted version will be referred to as uNURBS. The weighted and unweighted variants have 35 and 21 parameters respectively.

The fitting is performed in Matlab using the `fminsearch` function. The objective function is a squared error equation. This evaluates the distance between the y-values of the REA2822 and the produced airfoil. The equation is shown below:

$$J = \sum_{i=1}^n (y_{RAE} - y_{fit})^2 \quad (42)$$

Where n is equal to the amount of points that create the airfoil. J is the objective function of the fitting optimization. y_{RAE} are the REA2822 airfoil y values and y_{fit} are the y values of the airfoil parameterization. The REA2822 airfoil including camber line is shown in figure 7.

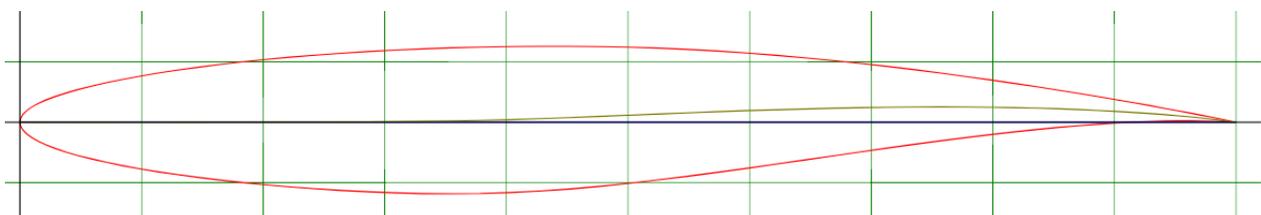


Figure 7 - REA2822 Airfoil with camber line [<http://airfoiltools.com/airfoil/details?airfoil=rae2822-il>]

5.1.1 Fitting results

The results of the fitted airfoils are shown in table 1. Based on the CST fitting errors CST10 and CST14 are chosen for further investigation with regards to surrogate modeling. CST10 as it is the cheapest and CST14 as the fitting error does not further reduce significantly for the added parameters. The REA2822 fitted airfoils are shown in figure 8 to 12.

Table 1 - Fitting errors

Parameterization	Fitting Error RAE2822	Number of parameters
CST10	1.02e-05	10
CST12	8.21e-06	12
CST14	2.76e-06	14
CST16	1.93e-06	16
CST18	1.82e-06	18
BP3434	3.89e-06	14
uNURBS	9.70e-07	21
NURBS	1.07e-06	35

CST10

The CST10 fitted airfoil experiences some issues in following the upper and lower crest curvature. The lower aft concavity also shows to be problematic. This is an indicator for the lack of completeness of using CST method with 10 variables. The results should show if trade-off between parameters is worth the lack of completeness.

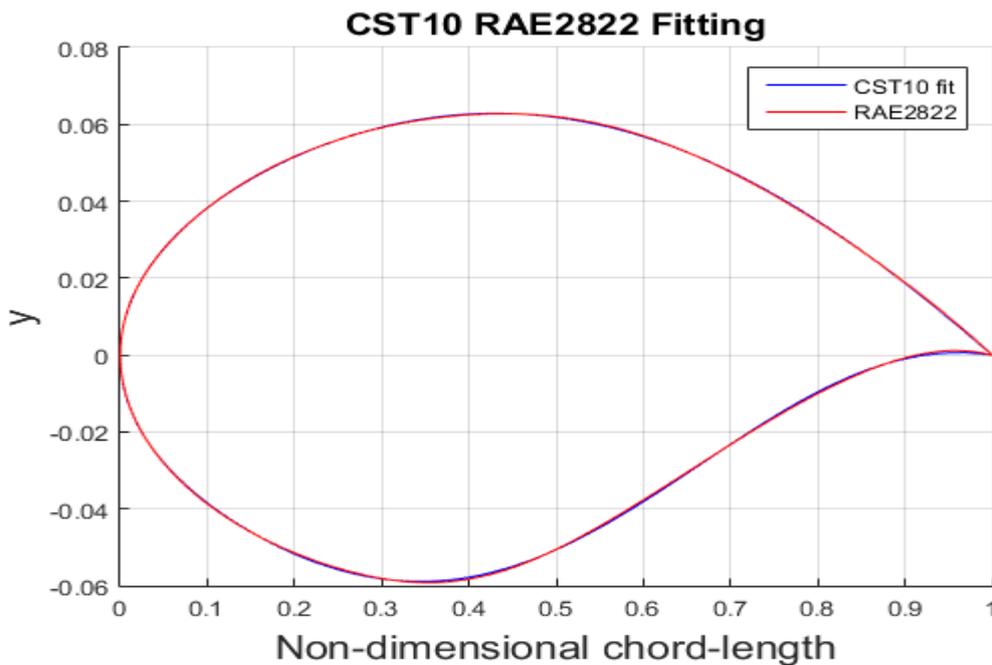


Figure 8 - CST10 RAE2822 Fitted airfoil

CST14

The CST14 method has again a significantly improved fit over CST10. The larger curvature shows less blue indicating fewer errors. The CST14 method error is also reasonably closer to the error of the other methods, surpassing BP3434. The biggest visible blue lines are still at the lower side trailing edge concavity. The increased parameter count results in improved completeness.

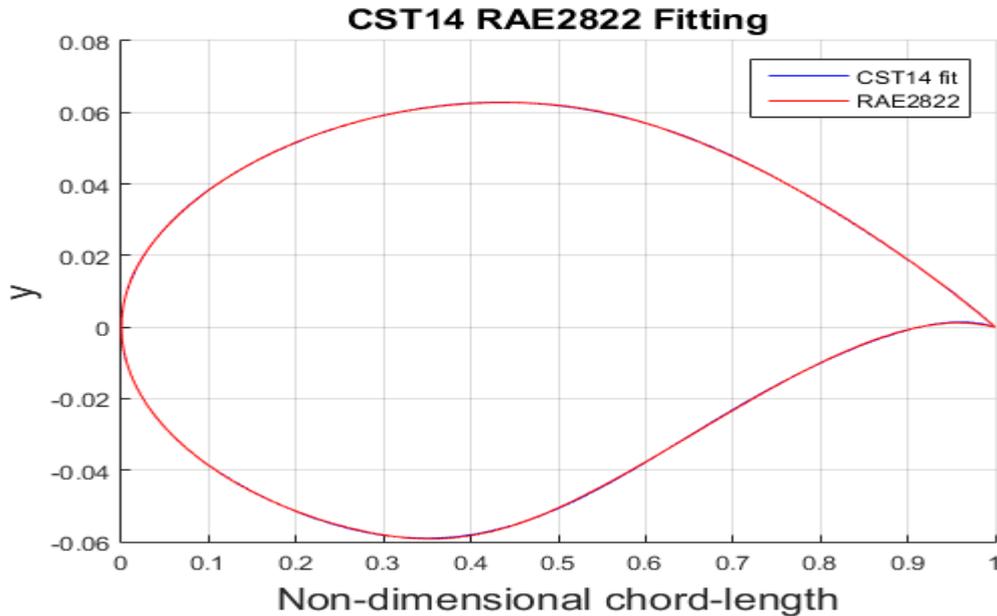


Figure 9 - CST14 RAE2822 fitted airfoil.

BP3434

The BP3434 error is noticeably above NURBS and comparable to the CST14 method. BP3434 has added intuitiveness and a relatively low parameter count. The BP parameterization method creates a front and an aft part for the thickness and camber curve. The connection between these curves can result in a loss of continuity.

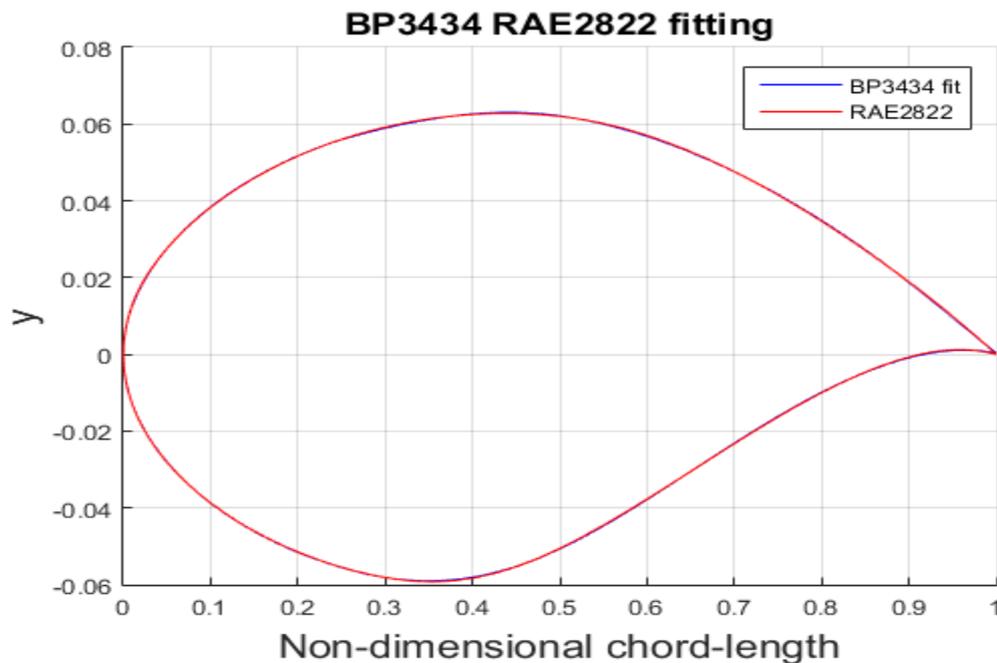


Figure 10 - BP3434 RAE2822 fitted airfoil.

uNURBS

Surprisingly the unweighted NURBS variant outperformed the NURBS method with weights regarding fitting error. With 21 variables it has the best fit of all method presented in the report. There are no areas of the airfoil that seem to be problematic. The capability of the NURBS methods to locally manipulate the curve proves useful for completeness.

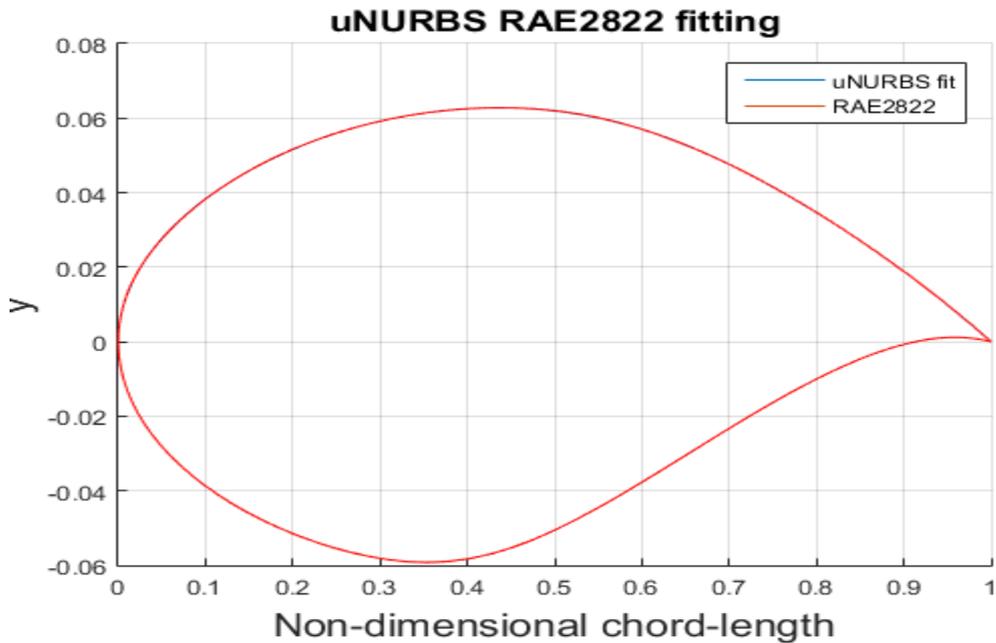


Figure 11 - uNURBS fitted airfoil.

NURBS

NURBS obtained a slightly worse fit than uNURBS. The parameterization method has more local control and flexibility than the CST parameterization method has.

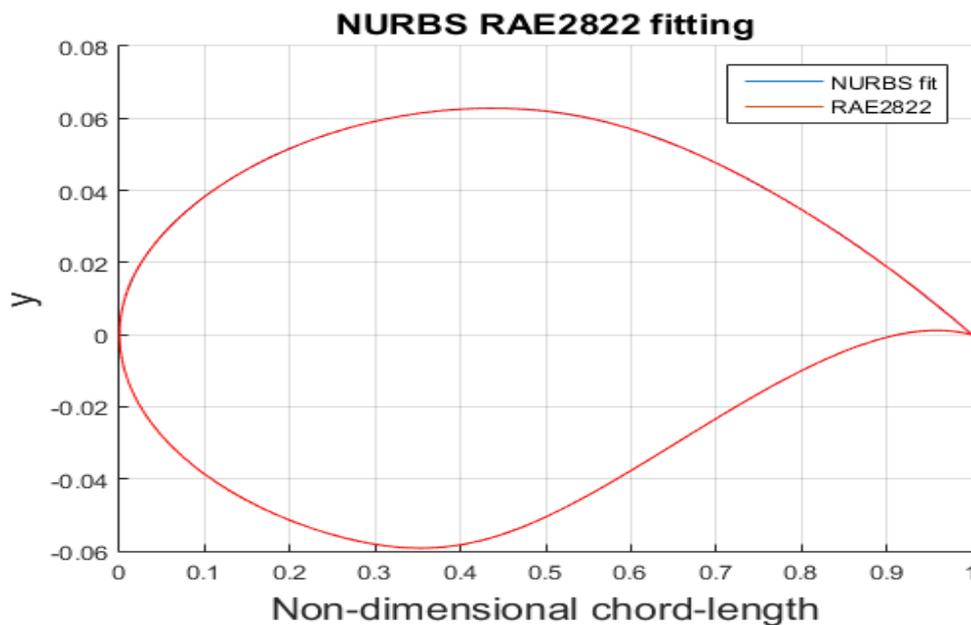


Figure 12 - NURBS fitted airfoil.

5.2 Aerodynamic solver

For aerodynamic optimization problems an aerodynamic solver is run numerous times. The 2D aerodynamic solver used is MSES [20]. The MSES solver has numerous options to analyze airfoils in different flow conditions. With MSES either the lift coefficient is computed for a static angle of attack, or the angle of attack is fitted to a specified lift coefficient.

The solver is used with two different settings for two different reasons. Setting one is used once per parameterization on the best fitted REA2822 airfoils to obtain information on the lift coefficient that is used as baseline lift coefficient for setting two.

Setting two takes the aforementioned lift coefficient as input and is used as high-fidelity solver for the optimizations. The angle of attack is set free in setting two.

The input files used are presented in Appendix B.

5.2.1 MSES setting 1

The goal of setting one is to compute the flow surrounding the REA2822 best fit airfoils at an angle of attack of 2° . Each parameterization obtains slightly different aerodynamic values due to small differences in airfoil geometry. To be able to compare the different parameterizations the same lift coefficient should be achieved throughout the optimizations. The aerodynamic values for each parameterization with these flight settings are shown in table 2. The NURBS lift coefficient is chosen as baseline for the optimization as it the closest to the average.

Table 2 - Best fit airfoil characteristics

Method	Cl	Cd	Cm	t/cmax
CST10	0.5987	0.0113	-0.0765	0.1208
CST14	0.6134	0.0113	-0.0794	0.1209
NURBS	0.6082	0.0113	-0.0786	0.1211
uNURBS	0.6087	0.0112	-0.0784	0.1210
BP3434	0.5971	0.0110	-0.0757	0.1209

MSES Settings 1

- $M = 0.73$
- Cl is set free
- $Re = 2.7e6$
- far-field vortex strength,
- $AoA = 2^\circ$
- LE stagnation.
- LE & TE Kutta condition
- S-momentum equation, with isentropic condition only near leading edge
- vortex+source+doublet airfoil far-field BCs
- Inviscid Calculations
- $N_{crit} = 9$
- Forced transition at 0.01% of chordlength
- Critical Mach number set to 0.99 (weak shocks)
- The artificial dissipation coefficient = -1.0 (negative number disables 2nd order dissipation for better robustness.

5.2.2 MSES setting 2

MSES settings two is used as high-fidelity solver for the optimizations. At each function evaluation the MSES takes the input of the current evaluation airfoil coordinates and the flow conditions Mach, Reynold and Cl into consideration and computes the corresponding angle of attack, drag coefficient, pitching coefficient and pressure distribution. The drag coefficient is used for the objective function, pitching moment coefficient is used for the second constraint and the pressure distribution is used for post analysis.

The REA2822 fitted airfoils using Cl = 0.6082 produces and MSES settings two produce the initial values for the drag coefficient, maximum thickness and pitching moment coefficient used for the objective function and constraints. These values are shown in table 3.

Table 3 – Aerodynamic values of initial design

Parameterization	Cd_i	t/c_{max}	Cm_i	Cl	AoA
CST10	0.0114	0.1208	-0.0763	0.6082	2.0504
CST14	0.0112	0.1209	-0.0793	0.6082	1.9728
BP3434	0.0111	0.1209	-0.0757	0.6082	2.0565
uNURBS	0.0113	0.1210	-0.0785	0.6082	2.0002
NURBS	0.0113	0.1210	-0.0804	0.6082	1.9506

MSES Settings 2

- $M = 0.73$
- $Cl = 0.6082$
- $Re = 2.7e6$
- far-field vortex strength,
- freestream angle of attack,
- LE stagnation.
- LE & TE Kutta condition
- S-momentum equation, with isentropic condition only near leading edge
- vortex+source+doublet airfoil far-field BCs
- Inviscid Calculations
- $N_{crit} = 9$
- Forced transition at 0.01% of chordlength
- Critical Mach number set to 0.99 (weak shocks)
- The artificial dissipation coefficient = -1.0 (negative number disables 2nd order dissipation for better robustness).

5.3 Gradient descent optimization:

To be able to compare the pros and cons of the surrogate model based optimization a gradient descent optimization is performed for each parameterization. The gradient descent optimization scheme is shown in figure 13. The optimization starts at the best fitted REA2822 airfoil. The gradient descent optimization tweaks the design variables and finds a design that has an improvement on the objective function. This process is iterated until a stopping criteria has been met which can be amount of iterations, time, stepsize or improvement increment.

The gradient descent optimization is performed in Matlab using the fmincon function [21]. The settings for the optimization are shown in table 4. The initial starting is the REA2822 best fit. The upper and lower bounds are presented in appendix A.

Table 4 - Gradient Descent optimization settings.

Parameterization	Algorithm	DiffMinChange	DiffMaxChange	TolFun	TolCon	TolX
CST10	SQP	1e-3	1e-2	1e-6	1e-3	1e-6
CST14	SQP	1e-3	1e-2	1e-6	1e-3	1e-6
BP3434	SQP	1e-3	2e-1	1e-6	1e-3	1e-6
uNURBS	SQP	9e-3	6e-2	1e-6	1e-3	1e-6
NURBS	SQP	1e-3	1e-2	1e-6	1e-3	1e-6

The results for the various optimizations are presented in section 5.5.

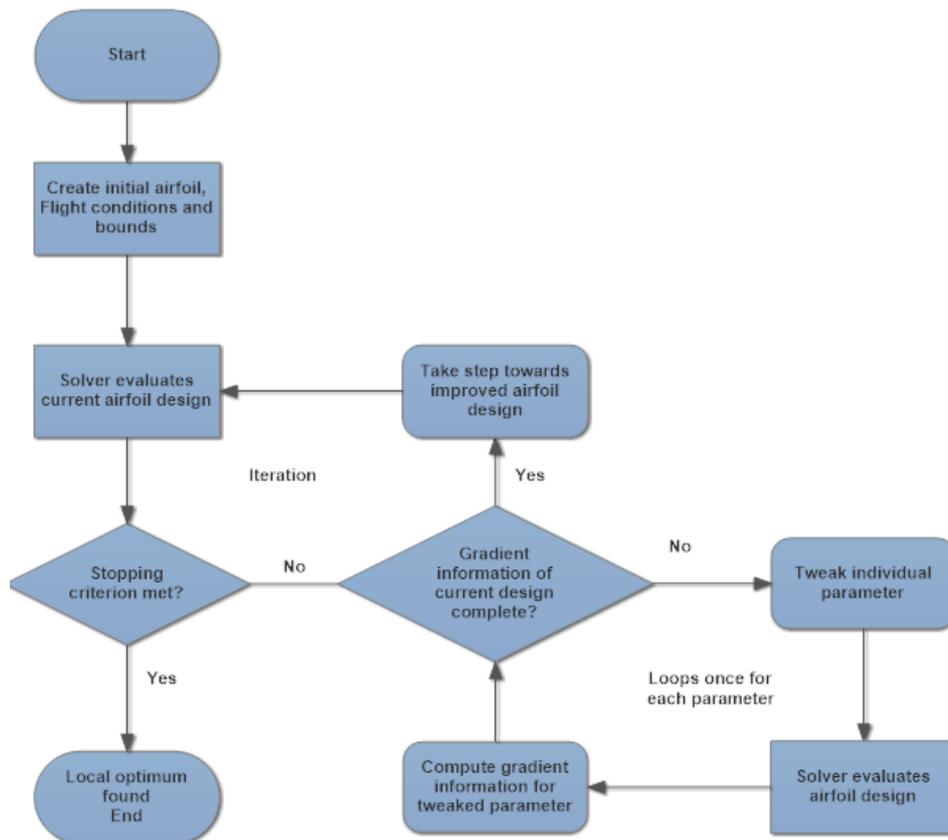


Figure 13 - Gradient Descent optimization flowchart.

5.4 Surrogate models based optimization

Researching surrogate model based optimization requires several intermediate steps. Initially a large amount of surrogate models are created. Based on the validation errors of the created models a subset of models is selected for further optimization. The selected surrogate models are optimized with a genetic algorithm. The resulting estimated global optimum is then used as an initial design for a final exploitation using gradient descent optimization with the high fidelity solver. The entire surrogate modeling and surrogate model exploitation process is depicted in figure 14 on page 24.

5.4.1 Surrogate models

The surrogate model building is performed with the SUMO toolbox [10]. Several surrogate modeling options are explored in this research. These options are the sample size, sequential sampling plan and surrogate model. The options researched per parameterization are portrayed in table 5. The objective function and pitching moment constraint use similar settings for each surrogate model.

The Bezier-PARSEC parameterization method experienced robustness issues during the evaluation of the sample plan. These issues are addressed in subsection 5.4.2. As a result BP3434 is excluded from the surrogate model based optimization part of the thesis.

Table 5 - Surrogate Models build.

Parameterization	CST10	CST14	BP3434	uNURBS	NURBS
Sample Size	5n, 10n, 20n, 30n	5n, 10n, 20n	N/A	5n, 10n, 20n	5n, 10n
Initial Sampling	TPLHD	TPLHD	N/A	LHD	LHD
Sequential Sampling	a, b, c, d, e	a, b, c, d, e	N/A	a, b, c, d, e	a, b, c, d, e
Surrogate model	RBF, Krig, GPML	RBF, Krig, GPML	N/A	RBF, Krig, GPML	RBF, Krig, GPML
Total surrogate models	60	45	N/A	45	30

There is a difference in what sample size is researched per parameterization. This is due larger dimensional problems causing robustness issues with the SUMO solver with larger sample sizes. CST10 and CST14 are able to uses the TPLHD initial sampling strategy effectively. For uNURBS it required too much computational effort to find the optimal LHD through TPLHD. For 21 parameters 22 hours are required compared to the 5 minutes that are required for 14 parameters. For NURBS TPLHD is also unfeasible.

All the presented sequential sampling methods and surrogate model options have been explored. By creating a large amount of surrogate models trends should appear more apparent than with smaller data sets. The sequential sampling methods are referenced by the letter they got appointed in section 4.3.

The surrogate models are referenced in the following manner: parameterization method, sample size, sequential sampling and surrogate model. I.E. CST10 5n_a_Krig is a surrogate model build for CST10 using 5n samples, LOLA-Voronoi as sequential sampling and Kriging to compute the surrogate model.

5.4.2 BP3434

The intuitive Bezier-Parsec method created problematic airfoils throughout the sampling procedure. The third trailing edge camber control point created by equation 12 in chapter 3.2 shows the parameterization equation that leads up to the failure.

Due to the interaction between the relative random sampling and the leading edge camber direction (γ_{LE}) of the RAE2822 the range for this value is too close to 0. The equation uses the cotangent of the leading edge camber direction. This causes a large sensitivity for the control point itself. The problematic control point often ends up in front or aft of the airfoil. This results in oscillations in the camber curve and thus the airfoil. The oscillation causes MSES to crashes which interrupt the surrogate modeling process.

Figure 15 shows the thickness curve, camber curve and resulting airfoil of such a problematic instance respectively. The red lines depict the position and order of the control points and the blue lines are the resulting curve. BP3434 performed worse than the other methods for the gradient descent search and the method could not handle the randomness in the sampling. For BP3434 the costs of intuitiveness seem to be disproportionate. For the method to work certain equation would require a rework or a different design space further away from flat leading edge camber curve needs to be explored.

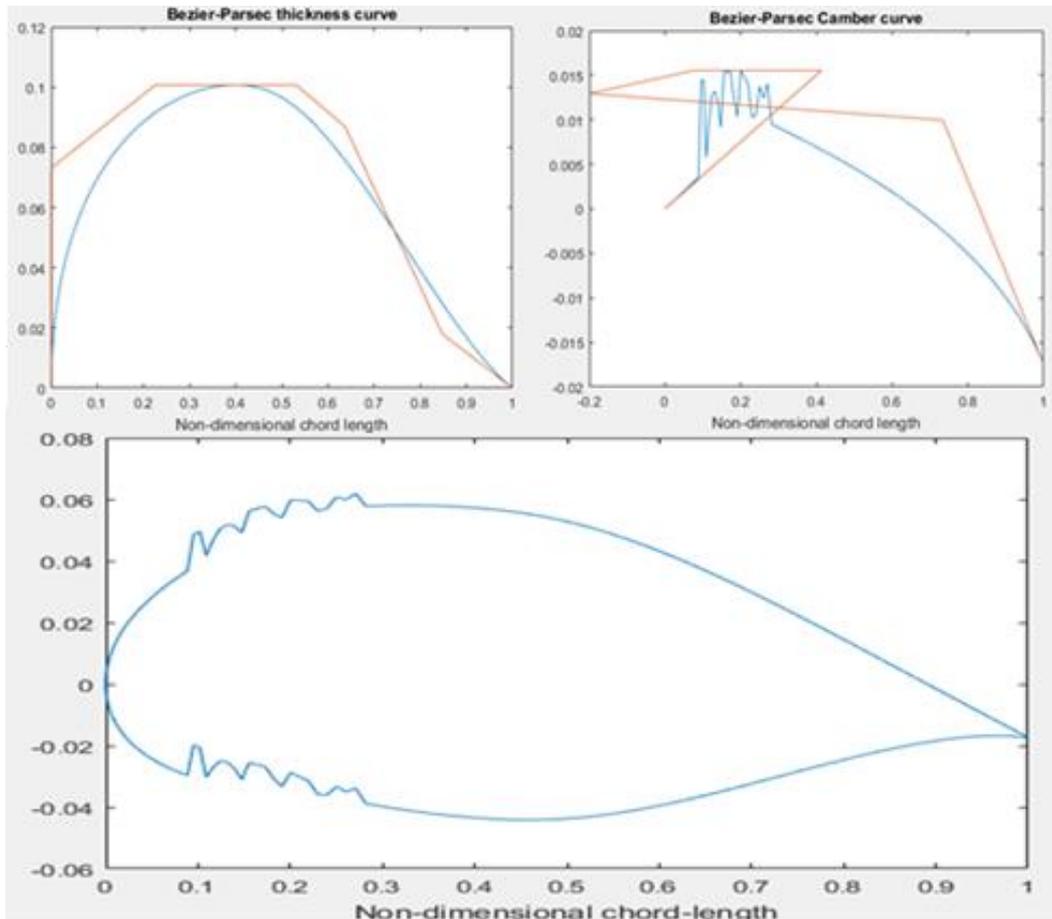


Figure 15 - BP3434 failure

5.4.3 Model selection criterion

For each parameterization surrogate models are created with the different sample settings, sequential sampling plans and surrogate models. As a result 180 different surrogate models have been created for the objective function and the pitching moment constraint. Only a subset of these surrogate models can be further exploited. Some criteria has to be established based on which the eligible surrogate models are selected. This criterion is established in this section.

Each different setting produces 6 validation error scores, three for the objective function model and three for the constraint model. The three different validation errors are the RMS, MAE and r^2 errors as presented in section 4.5. All resulting error scores are available in appendix D.

Comparing 6 data points per surrogate model setting is difficult. To reduce the amount of data required to select the surrogate models for further exploitation a comparison is made how well the objective function is understood compared to the pitching moment constraint. In figure 16 the RMS error for the objective function is plotted against the constraint validation errors. The x-axis shows the error scores for the objective function and the y-axis shows the error scores for the pitching moment constraint.

By plotting the RMS errors against each other it becomes clear that for each tested setting the pitching moment is better understood by the surrogate models than the objective function. This can be seen from the magnitude of the error score, a lower score is better for RMS.

The same trend continues in figure 17 and 18 where the MAE and r^2 errors are plotted against each other.

This comparison shows us that the objective function surrogate models are the determining factor for a successful genetic optimization. Regardless of the surrogate models chosen the pitching moment constraint is better understood than the objective function. Thus the following subchapters will only compare the objective function validation errors to make a selection for further exploitation.

Figure 16, 17 and 18 all show an extra trend. The surrogate models created with lower dimensional parameterizations are better understood. The extra parameters add design space and increase the complexity of the design space, which becomes more challenging for the surrogate models to understand.

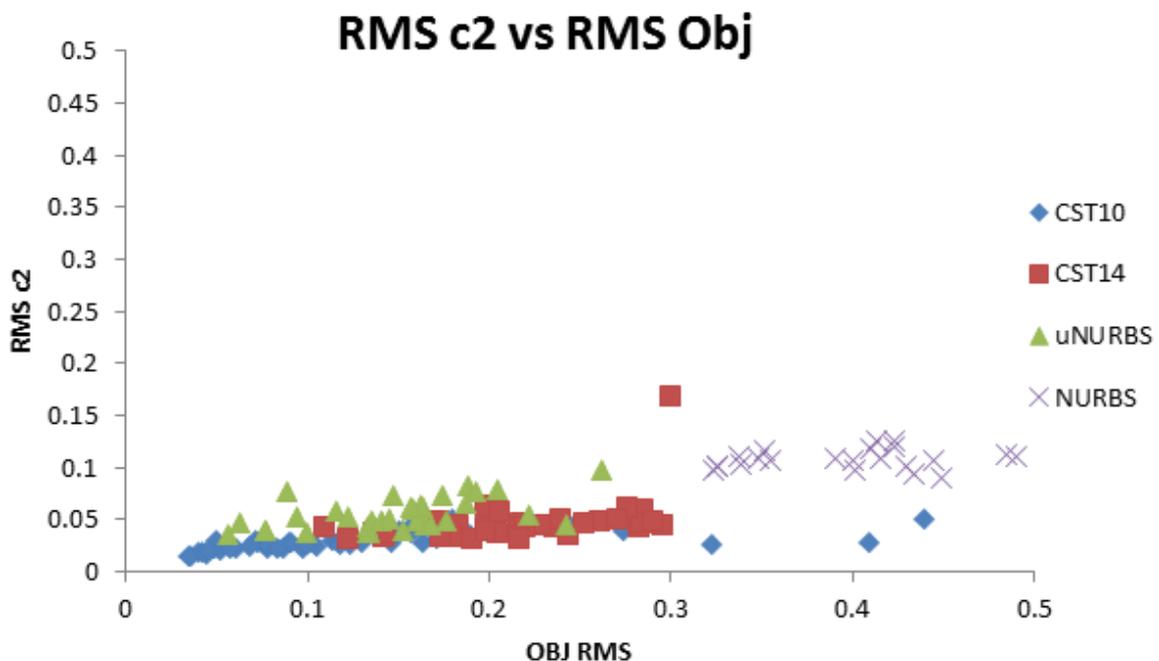


Figure 16 - RMS pitching constraint against RMS objective function

MAE c2 vs MAE Obj

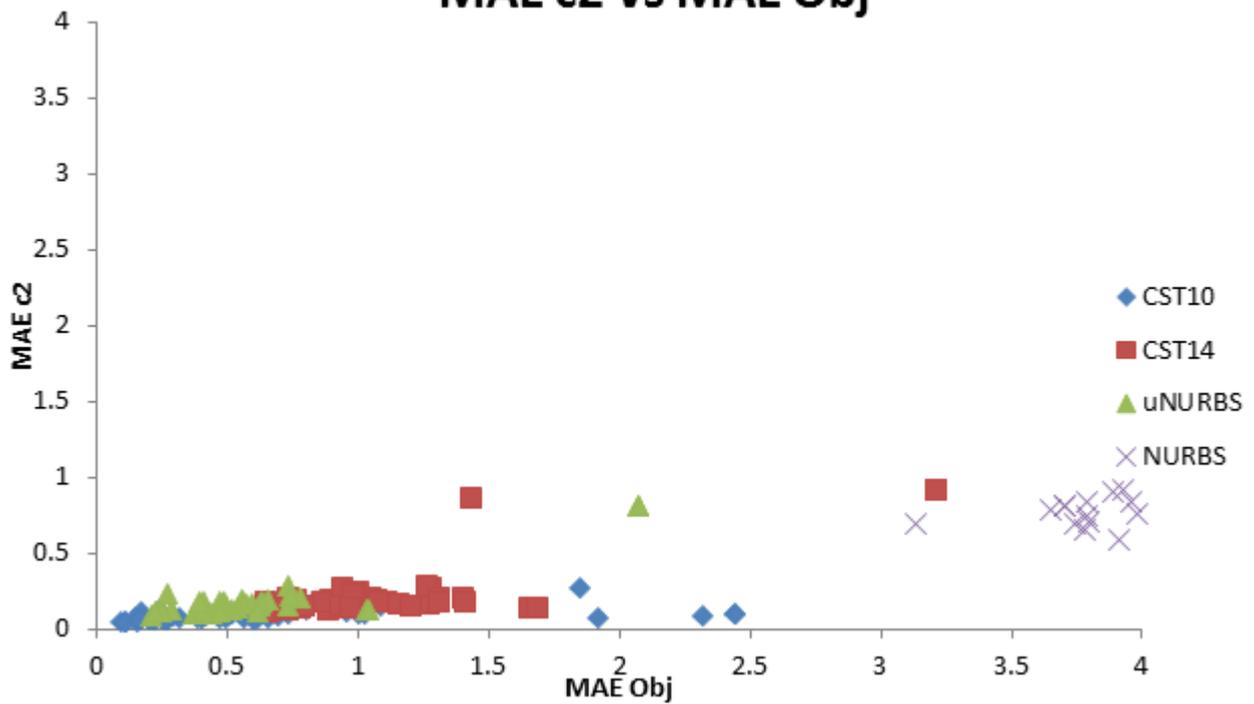


Figure 17 - MAE pitching constraint against MAE objective function

r² c2 vs r² Obj

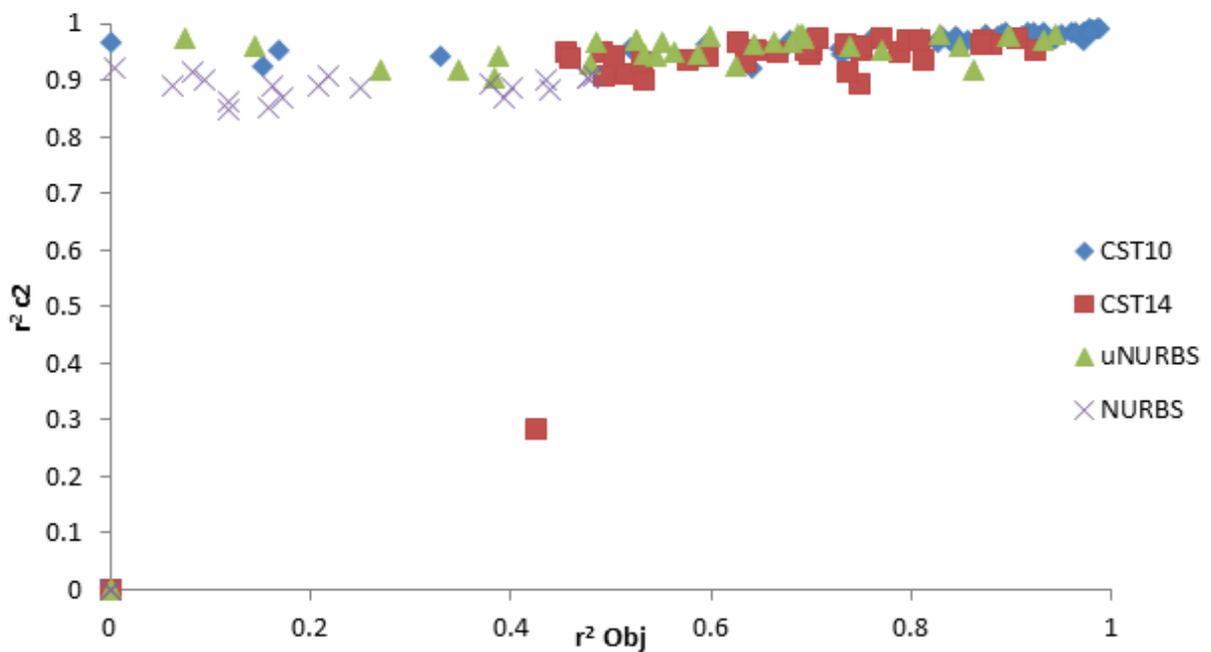


Figure 18 - r² pitching constraint against r² objective function

5.4.4 Validation error analysis

In this section a selection of surrogate models is made for each parameterization to further optimize. Per parameterization a selection is made for different sample size setting. The models with the best objective function validation errors for their respective sample size and parameterizations are eligible for further optimization. At each parameterization a table is given with their respective most promising surrogate models.

For each parameterization 2 figures are shown. The first figure shows the MAE vs RMS validation errors. Both MAE and RMS scores are better towards 0, thus the better models are in the bottom left area. The second figure shows r^2 versus RMS*MAE for the surrogate models. The RMS*MAE value can be used as both individual errors aim for lower scores. Hence the lowest score on RMS*MAE are the models that perform well for both RMS and MAE errors. Models with a r^2 as close as possible to 1 generally perform the best. In figure 20 the upper left corner contains the better model. Note that for readability not all error scores are contained in figures 19 to 26.

CST 10

Figure 19 and 20 show the objective function validation errors for the CST10 parameterization. The models with favorable error scores generally seem to be using either LOLA-Voronoi (a), Generalized probability of Feasibility (b) or default (e) sequential sampling. These models generally used Kriging or GPML. Specifically higher sample size surrogate models seem to be more accurate with GPML. Lower sample size models obtain better scores with Kriging. The selected surrogate models are portrayed in table 6.

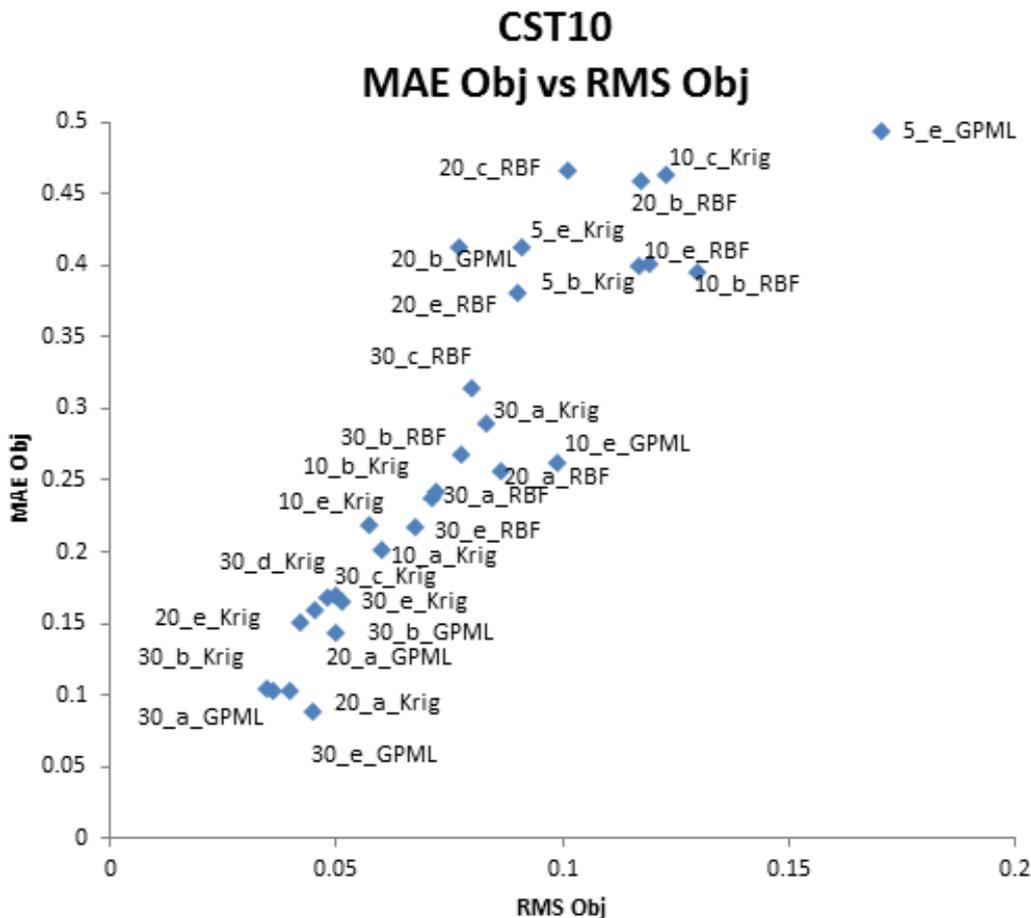


Figure 19 - CST10 MAE objective against RMS objective.

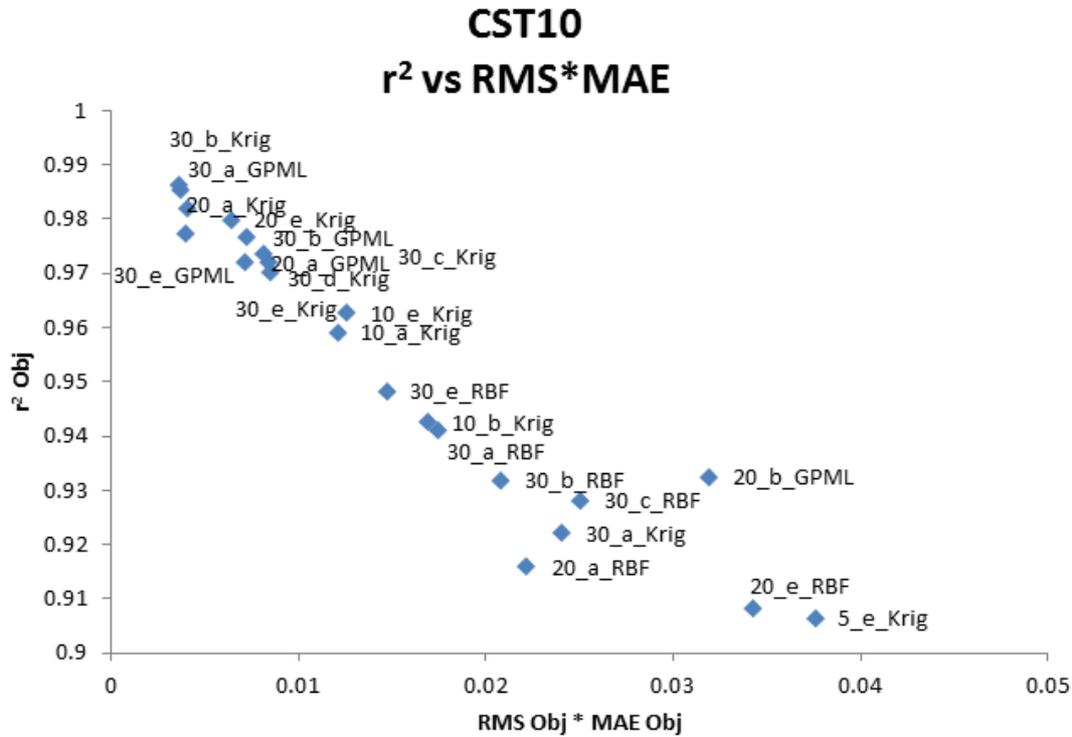


Figure 20 - CST10 r² objective against RMS*MAE objective.

The table below shows the models selected based on the error scores. In figure 19 the better models are in the bottom left corner and in figure 20 the better models are in the top left corner.

Table 6 - Selected CST10 surrogate models.

CST10 Surrogate model				
5n	5n_a_Krig	5n_b_Krig	5n_e_Krig	
10n	10n_a_Krig	10n_b_Krig	10n_e_GPML	
20n	20n_a_Krig	20n_e_Krig		
30n	30n_b_Krig	30n_a_GPML	30n_b_GPML	30n_e_GPML

CST14

The error plots are shown in figure 21 and 22. The results of the exploited surrogate models are shown in table 7. Surrogate models using RBF show more feasible for CST14 than CST10. Again mainly the sequential sampling plans of LOLA-Voronoi (a), generalized probability of Feasibility (b) and default (e) sequential sampling dominate the better error scores. In contrast to CST10 certain 5n sample size surrogate models are in comparable error wise to 10n and 20n models. One RBF surrogate model is selected as well.

CST14 MAE vs RMS

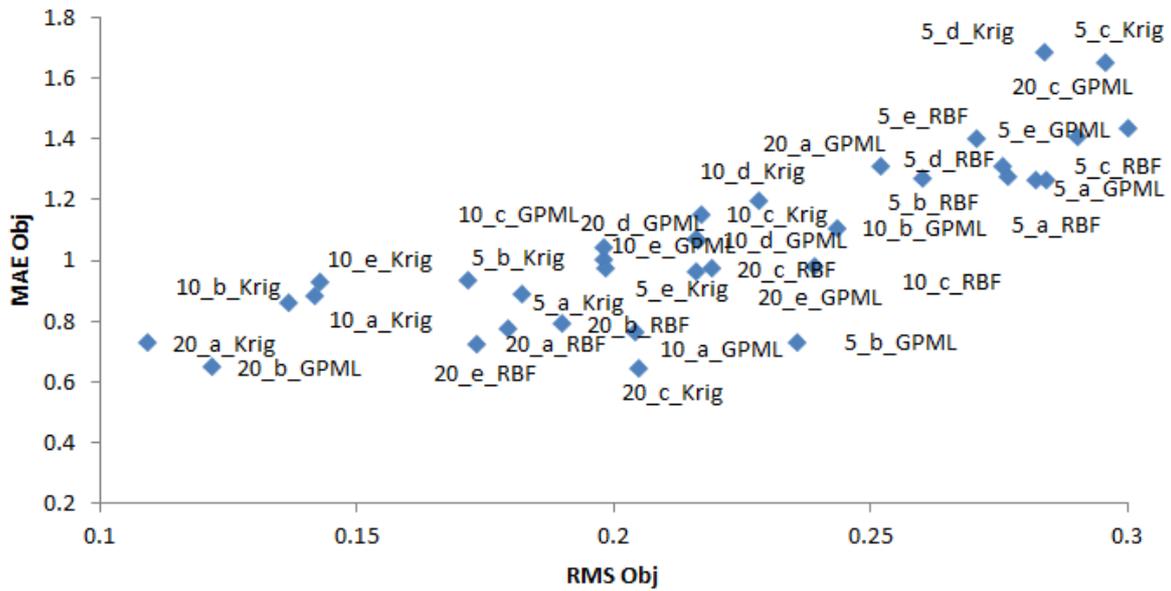


Figure 21 - CST14 MAE objective against RMS objective.

CST14 r^2 vs RMS*MAE

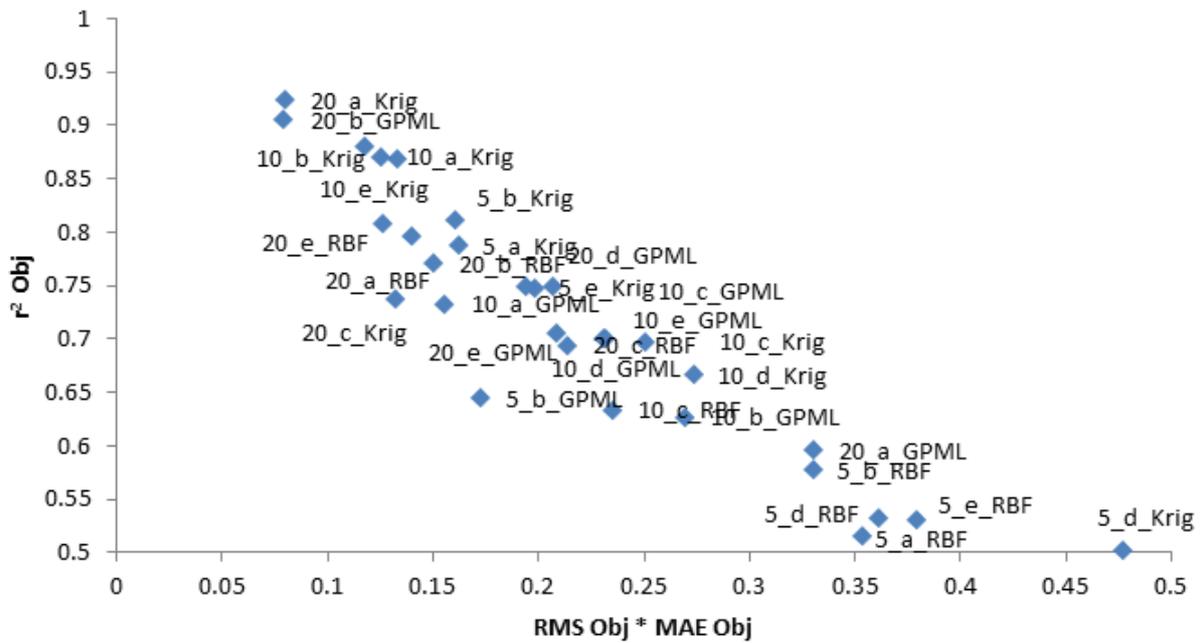


Figure 22 - CST14 r^2 objective against RMS*MAE objective.

Table 7 - CST14 selected surrogate models.

CST14 Surrogate model				
5n	5n_a_Krig	5n_b_Krig	5n_e_Krig	5n_b_GPML
10n	10n_a_Krig	10n_b_Krig	10n_e_Krig	
20n	20n_a_Krig	20n_b_GPML	20n_e_RBF	

uNURBS

Likewise with CST10 and CST14 the better models for sample sizes 5n, 10n and 20n shown in figure 23 and 24 have been selected for further exploitation. For uNURBS the combination of LOLA-Voronoi (a), generalized probability of Feasibility (b) or default (e) combined with Kriging seem to be the better models. The selected models for further optimization are shown in table 8. One model created with expected improvement sequential sampling is selected.

Table 8 - uNURBS selected surrogate models.

uNURBS Surrogate model				
5n	5n_a_Krig	5n_b_Krig	5n_d_Krig	5n_e_Krig
10n	10n_a_Krig	10n_b_Krig	10n_c_Krig	
20n	20n_a_Krig	20n_b_Krig	20n_e_Krig	

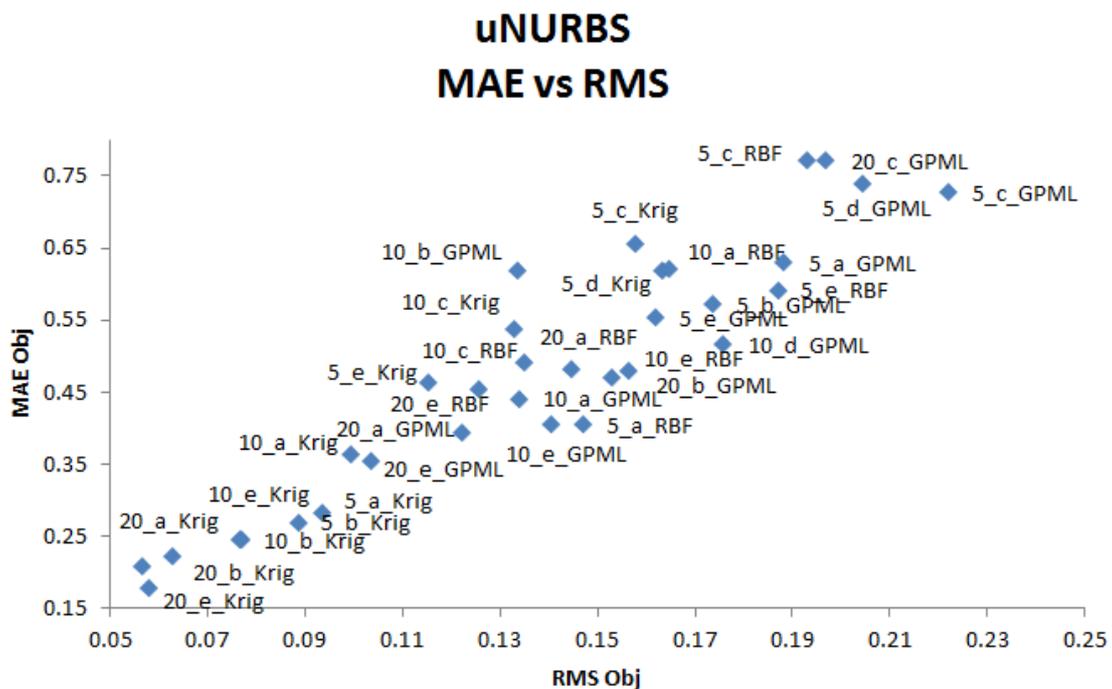


Figure 23 - uNURBS MAE objective against RMS objective.

uNURBS r^2 vs RMS*MAE

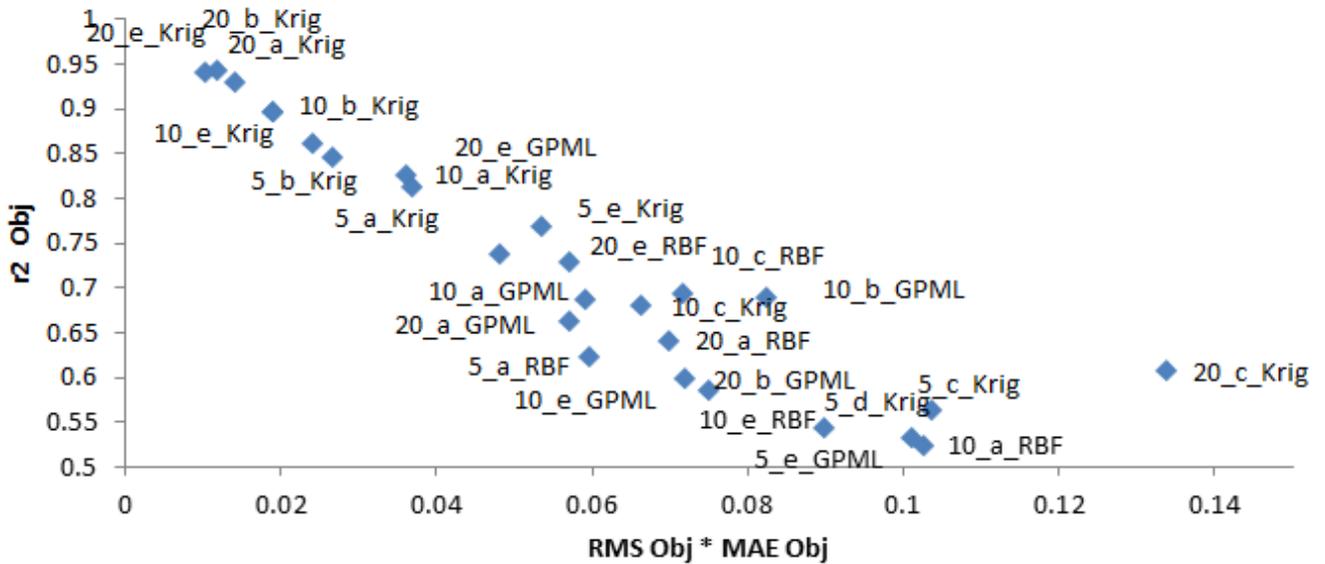


Figure 24 - uNURBS r^2 objective against RMS*MAE objective.

NURBS

The error plots for NURBS are shown in figure 25 and 26. The selected surrogate models for further optimization are portrayed in table 9. NURBS with weights showed the most struggle to find successful surrogate models that managed to estimate the complex design space. The method also required significant amount of time to compute the hyper parameters and the surrogate model. The added time and inaccuracy of the models are unwanted side effects of the extra flexibility provided by the weight parameters.

Table 9 - NURBS selected surrogate models.

NURBS Surrogate model					
5n	5n_a_Krig	5n_b_Krig	5n_d_Krig	5n_e_Krig	
10n	10n_a_Krig	10n_b_Krig	10n_d_Krig	10n_e_Krig	10n_b_RBF

NURBS MAE vs RMS

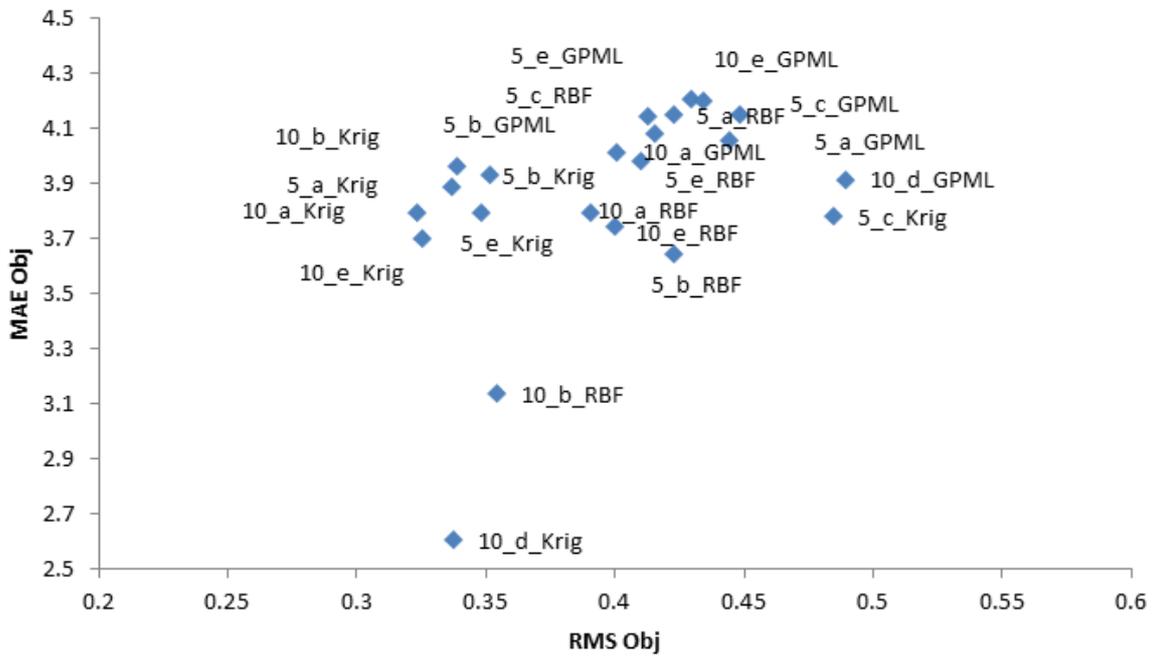


Figure 25 - NURBS MAE objective against RMS objective.

NURBS r^2 vs RMS*MAE

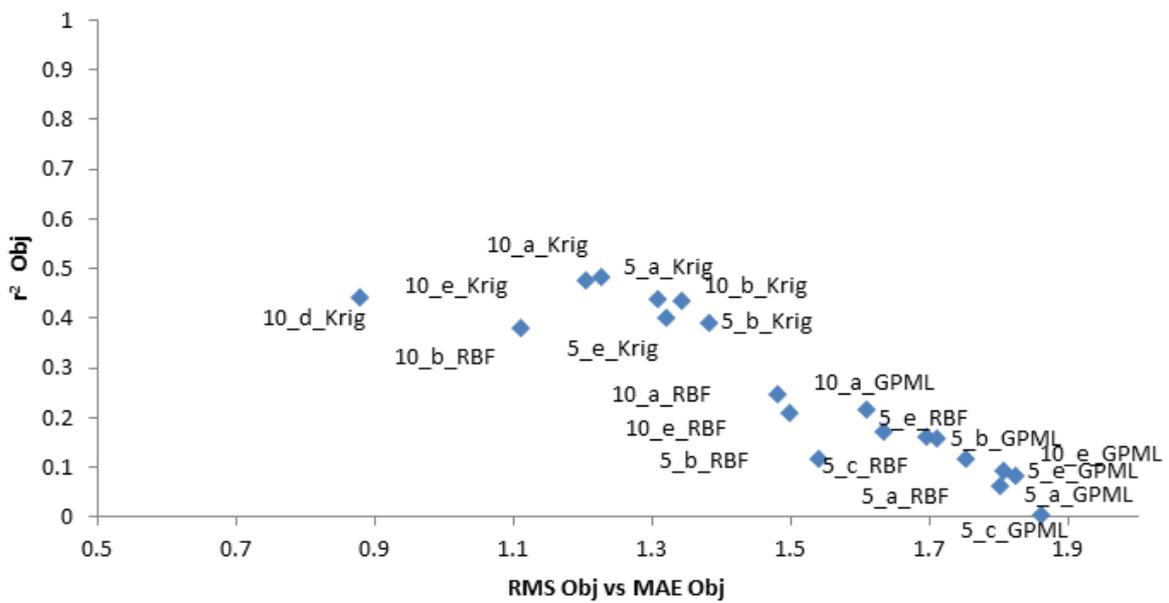


Figure 26 - NURBS r^2 objective against RMS*MAE objective.

5.4.5 Optimizing the surrogate model

Optimizing the surrogate model is the final step to be taken to obtain an optimum design. The selected surrogate models are optimized with a genetic optimization. This results in an estimated global optimum. The estimated global optimum achieved is used as initial starting point for the final exploitation. The final exploitation performs a gradient descent optimization with the high-fidelity solver to find the true optimum in the appointed design space.

Genetic optimization

The selected surrogate models are further exploited. The goal of this step is to find design space that contains a possible global minimum. The resulting estimated global minimum is then used as initial input for the final exploitation.

The surrogate model contains knowledge over the sampled design space and can estimate the objective function and pitching moment constraint in this design space. As the surrogate models are cheap to evaluate a genetic algorithm optimization can be performed.

The genetic optimization starts with an initial population of $10n$ using parameter values matching the successful samples used to build the surrogate model. The algorithm uses the surrogate model to evaluate the fitness of objective function and pitching moment constraint values. The thickness constraint is evaluated as it does not require the aerodynamic solver and is not as computationally expensive.

Genetic optimization uses crossover, mutation and selection to select a new population [21]. Crossover uses fit samples and the new generation maintains shared parameter values between fit samples. Mutation uses a fit sample and alters parameter values such that the next generation can be sampled in unsampled design space. And selection allows for successful points to be retained to the next generation. There is quite some randomness involved with selecting next generations. When unsatisfactory results emerge the genetic optimization can be rerun or the surrogate model can be rebuilt.

The genetic optimization algorithm runs until $100n$ points are evaluated. After this stopping criteria is met the genetic optimization stops and the surrogate model is optimized to a possible global minimum. The optimum found from this step points towards design space that obtains an interesting minimum. The surrogate model genetic optimized minimum is used as input for the final exploitation step.

Final Exploitation

The final exploitation is performed with a gradient descent optimization using the high-fidelity solver. The estimated global optimum from the previous step is used as initial input. The goal of this step is to find the local minima in the design space pointed out by the previous step. The method used for this final exploitation is a gradient descent optimization using SQP as algorithm. The optimization setup is the same as the optimization described in section 5.3 with a different initial design input. The initial design is based on the result of the genetic optimization of the surrogate model. The converged optimum found by the final exploitation is the final result of the surrogate model based optimization.

Sometimes surrogate model based optimization results in poor or unfeasible designs. This is likely caused by a poor design space found by the genetic optimization. The reason for a poor genetic optimization can be due to overfitting [22] or a poor surrogate model. Rerunning the genetic optimization can yield different results as it contains a lot of randomness. The other option is to salvage the surrogate model by adding more sequential sampling steps. The final option is to rebuild the surrogate model from scratch with different settings.

5.5 Results

This section contains the results. Initially the results of the gradient descent optimization are shown. Afterwards the surrogate model based optimization results are shown per parameterization. The results shown are the objective function, drag coefficient, number of solver calls, angle of attack, computational time and plots of the pressure distribution and airfoils.

5.5.1 Gradient descent optimization results

The gradient descent optimization results are shown in table 10. Figure 27 and 28 show the resulting pressure distributions and airfoils.

The objective function value and drag coefficient optimum show the effectiveness. The solver calls and total time in combination with the objective function show the efficiency. The angle of attack tells something about the design space that the solution has been found in.

Note that some parameterizations have a lower drag coefficient but a higher objective function. This is due to some slight differences in the REA2822 fitted airfoils. As a result the different initial designs operated at different angles of attack when computing the initial drag and pitching moment coefficients.

Table 10 - Gradient descent optimization results.

Parameterization	Obj	CD optimum	# Solver	AOA	Time [min]
CST10	0.9354	0.010664	233	2.47°	45.62
CST14	0.9496	0.010635	204	2.47°	51,50
BP3434	0.9603	0.010667	216	2.23°	40.29
uNURBS	0.9404	0.010666	373	1.98°	75.71
NURBS	0.9307	0.010541	1118	1.32°	174.16

The largest difference between the CST and B-spline methods is found in how the lower curvature curves after the crest. The resulting pressure difference experience large difference at the lower side. The BP3434 upper side pressure distribution shows unusual behavior at the maximum thickness.

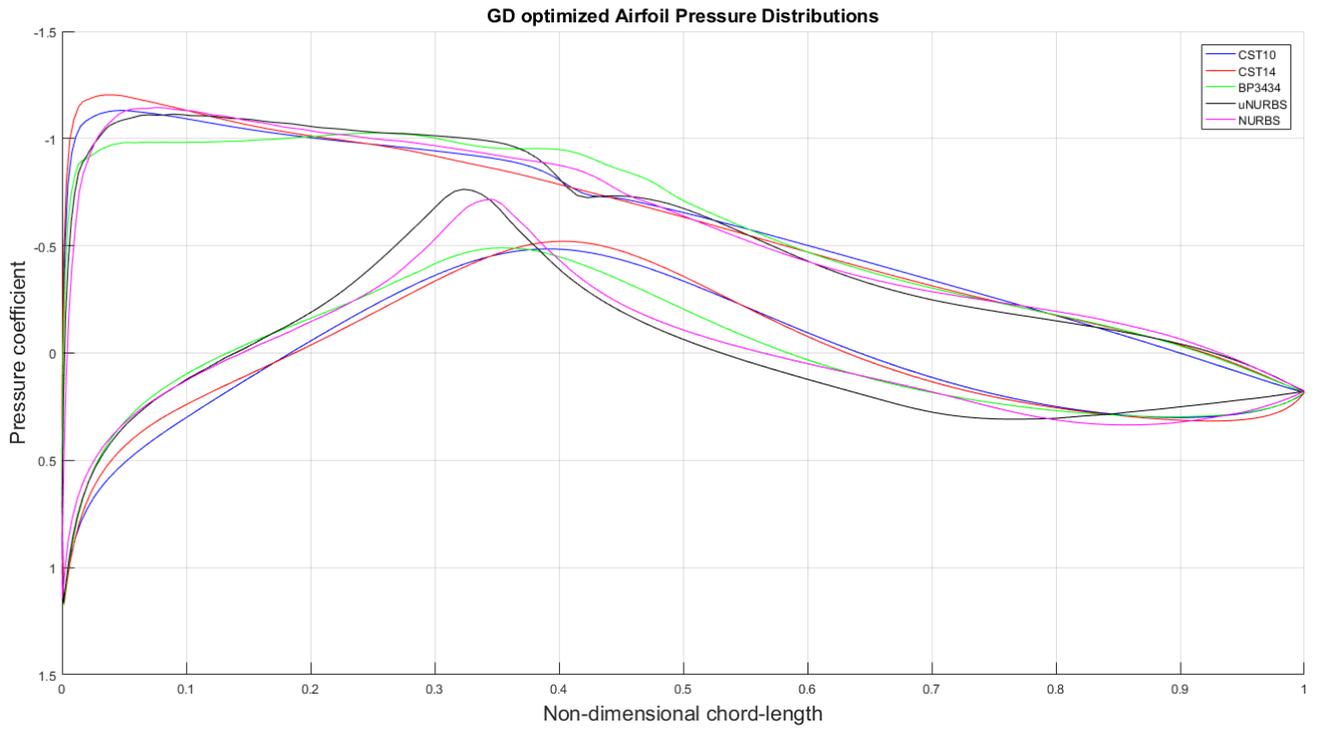


Figure 27 - Gradient descent optimized airfoil pressure distributions.

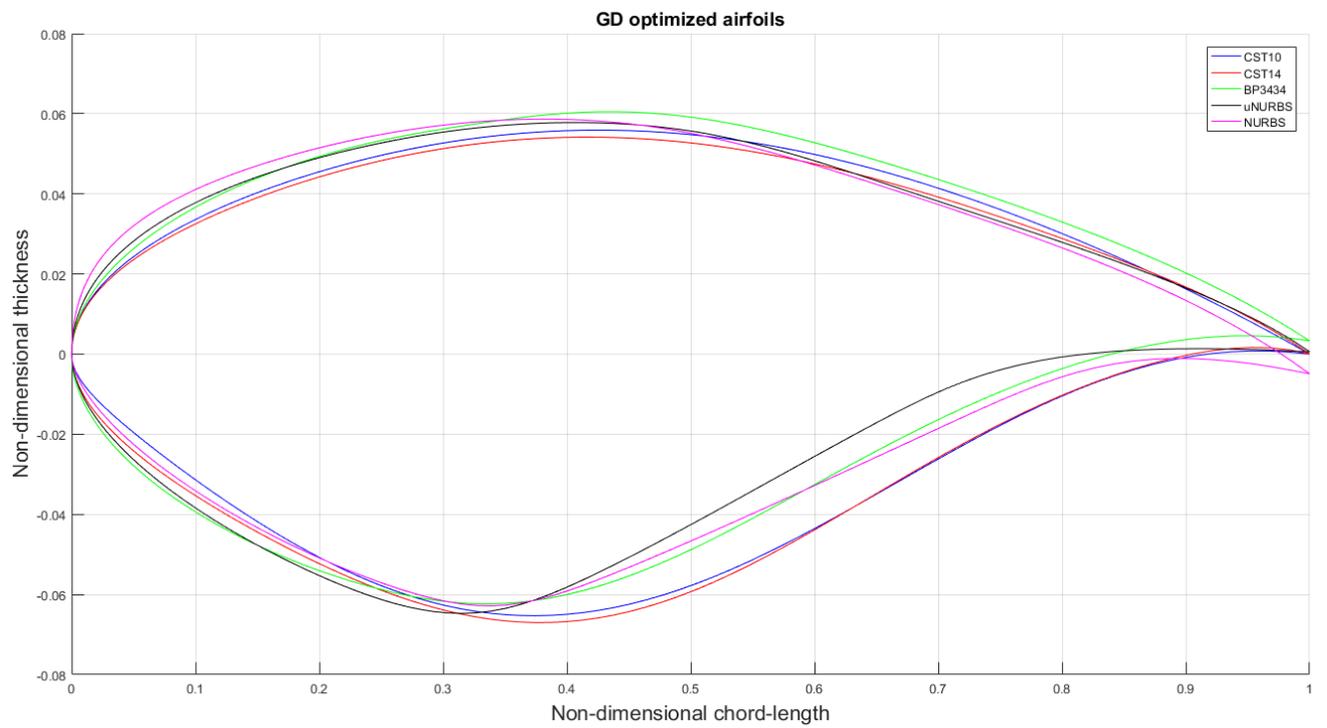


Figure 28 - Gradient descent optimized airfoils.

5.5.2 CST10 Surrogate model optimization results

The surrogate models selected in section 5.4.4 have been further optimized. Table 11 shows the results, figure 29 shows the pressure distribution and figure 30 shows the optimized airfoils. The objective function and drag coefficient show the effectiveness. The effectiveness combined with the number of solver calls and total time depicts the efficiency. The design space found is depicted by the figures and the angle of attack.

Table 11 - CST10 surrogate model based optimized results.

CST10 Surrogate model	Obj	CD	# Solver calls	AOA	Total Time [min]
5n_a_Krig	0.9418	0.010737	225	2.54°	47.00
5n_b_Krig	0.9302	0.010604	219	2.57°	40.19
5n_e_Krig	0.9293	0.010594	223	2.60°	39.31
10n_a_Krig	0.9413	0.010731	228	2.32°	40.83
10n_b_Krig	0.9382	0.010695	332	2.52°	86.43
10n_e_GPML	0.9378	0.010691	333	2.48°	80.59
20n_a_Krig	0.9342	0.010650	400	2.59°	62.36
20n_e_Krig	0.9377	0.010690	495	2.56°	84.08
30n_b_Krig	0.9305	0.010608	536	2.45°	94.88
30n_a_GPML	0.9368	0.010679	519	2.40°	135.73
30n_b_GPML	0.9417	0.010735	479	2.66°	153.19
30n_e_GPML	0.9388	0.010702	488	2.45°	85.56

Interestingly all exploitation converged to relatively high angle of attacks. The results also shows that models with 30n samples do not necessarily provide better design spaces to exploit than models build with lower sample sizes.

The best performing surrogate model is 5n_e_Kriging. With respect to objective function and time required. From figure 29 and 30 it shows that most of the optimized surrogate models found comparable design spaces.

Figure 29 and 30 show that the different surrogate models converged to roughly the same design space. The differences in the upper side peak pressure, lower curve first halve and upper aft curve are noticeable. All CST10 airfoils experience pressure increment on the upper surface which looks like a separation bubble, which is interesting as a transition is already forced on the leading edge. The possibility of a weak shock could not be researched due to an unfortunate combination of time constraints and interface bug with the used windows MSES version.

SM optimized CST10 Airfoil Pressure Distributions

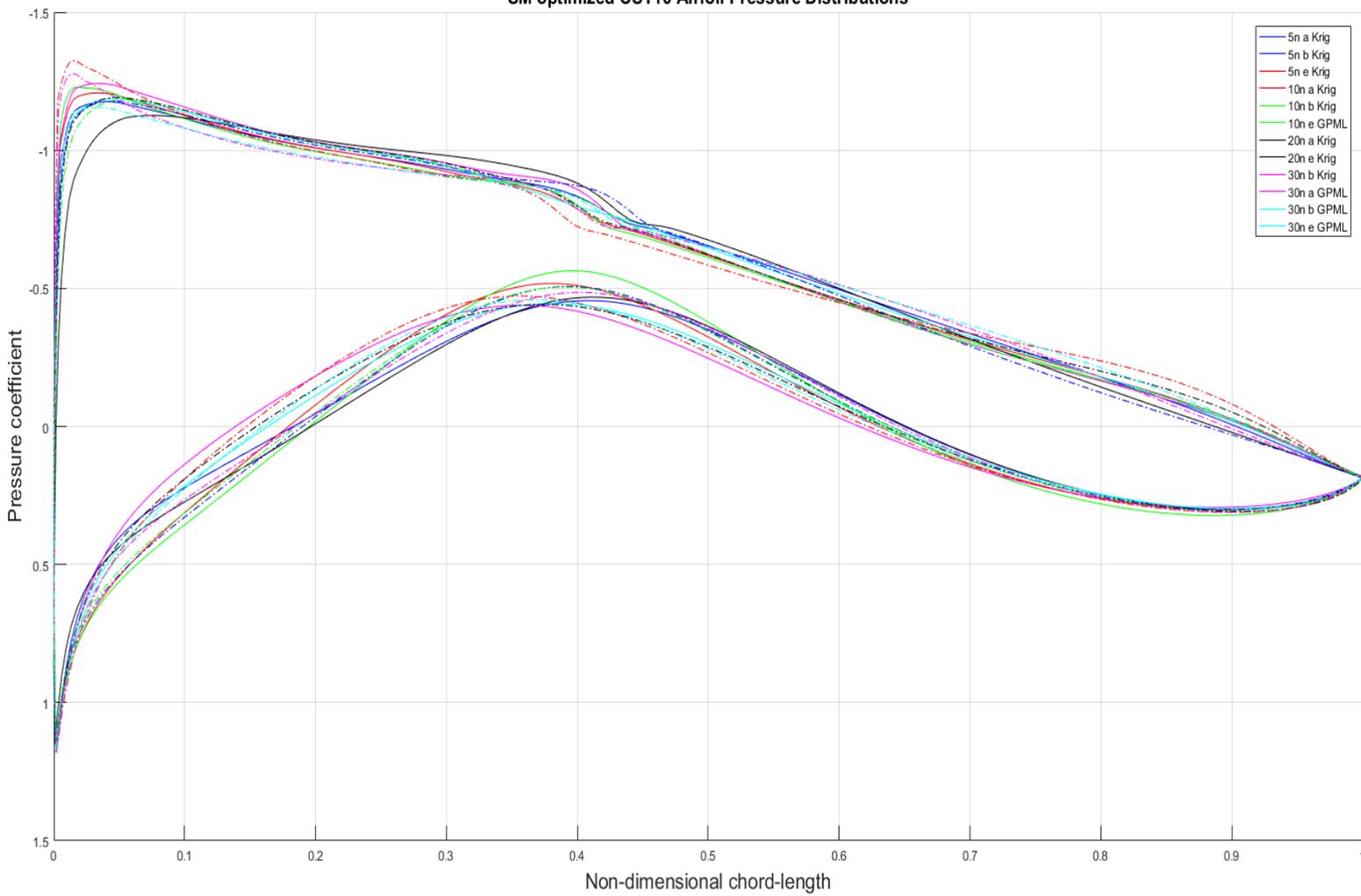


Figure 29 - CST10 surrogate model optimized airfoil pressure distributions.

SM optimized CST10 airfoils

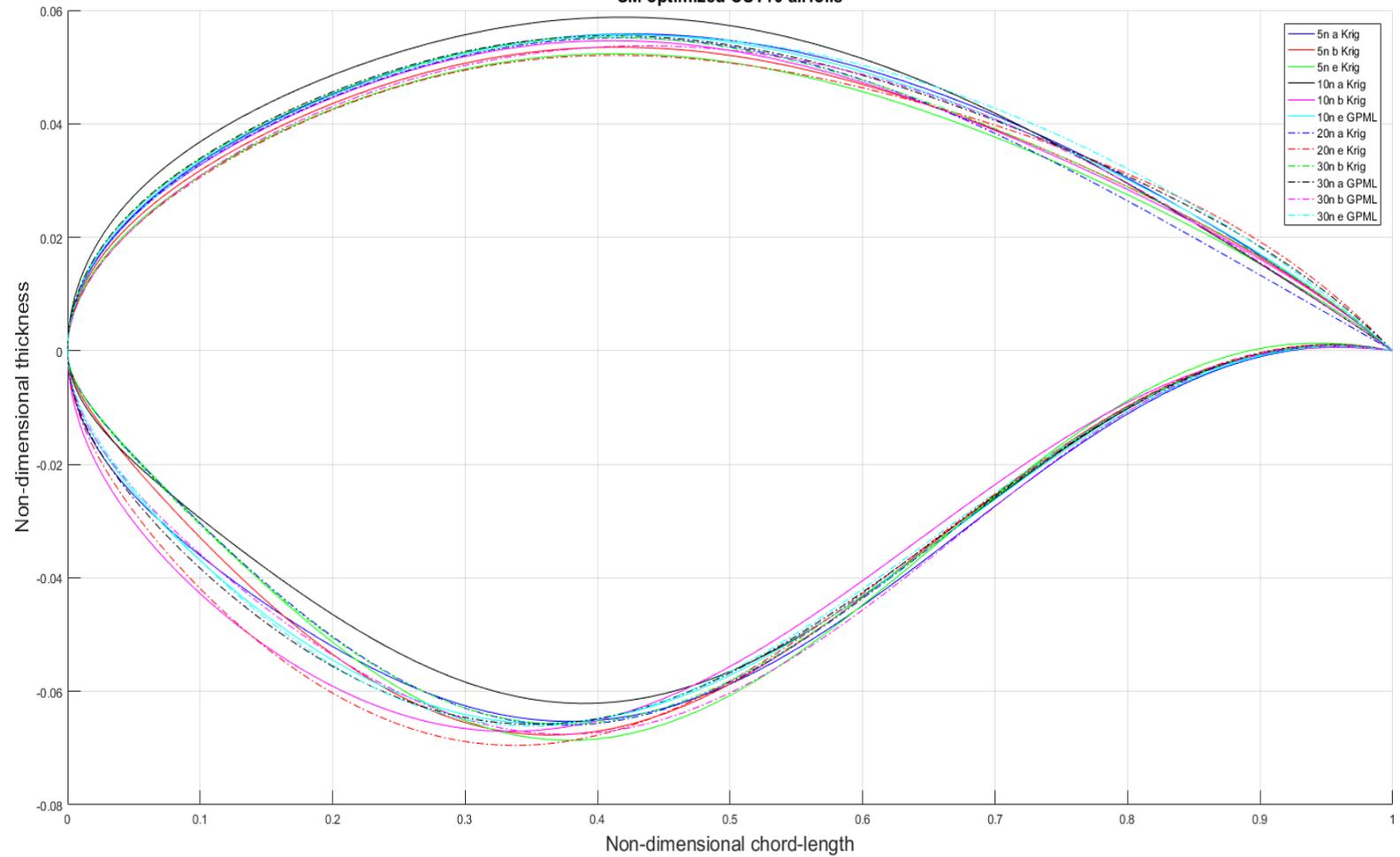


Figure 30 - CST10 Surrogate model optimized airfoils.

5.5.3 CST14 Surrogate model optimization Results

The surrogate models selected in section 5.4.4 have been further optimized. Table 12 shows the results, figure 31 shows the pressure distribution and figure 32 shows the optimized airfoils.

Table 12 – CST14 surrogate model optimized results.

CST14 Surrogate model	Obj	CD	# Solver calls	AOA	Total Time [min]
5n_a_Krig	0.9459	0.010594	336	2.20°	73.60
5n_b_Krig	0.9537	0.010681	334	2.42°	96.69
5n_e_Krig	0.9486	0.010624	266	2.22°	76.05
5n_b_GPML	0.9613	0.010766	221	2.40°	187.07
10n_a_Krig	0.9517	0.010659	361	2.35°	183.85
10n_b_Krig	0.9533	0.010677	332	2.15°	80.15
10n_e_Krig	0.9279	0.010393	415	2.22°	69.58
20n_a_Krig	0.9482	0.010620	525	2.07°	164.10
20n_b_GPML	0.9425	0.010556	579	2.32°	100.55
20n_e_RBF	0.9455	0.010590	473	2.32°	159.03

All the optimization seem to have converged to a lower angle of attack than CST10, but still higher than the initial 2.0°. Specifically the surrogate model created with 10n, default sampling and Kriging performed much better than its peers. Especially considering the time required. The design space found by this model is similar to the converged design space by the gradient descent of the NURBS method considering the lower side pressure distribution.

Appendix E shows that the genetic optimization of 10n_e_Krig required significant less time than the other CST14 surrogate models. The genetic optimization of CST14_10n_e_Krig stopped on solution convergence rather than time.

Figure 31 and 32 show that the different surrogate models also found more varying design spaces compared to the CST10 method. The pressure distribution from the GPML surrogate model with 5n samples shows the largest deviation.

SM optimized CST14 Airfoil Pressure Distributions

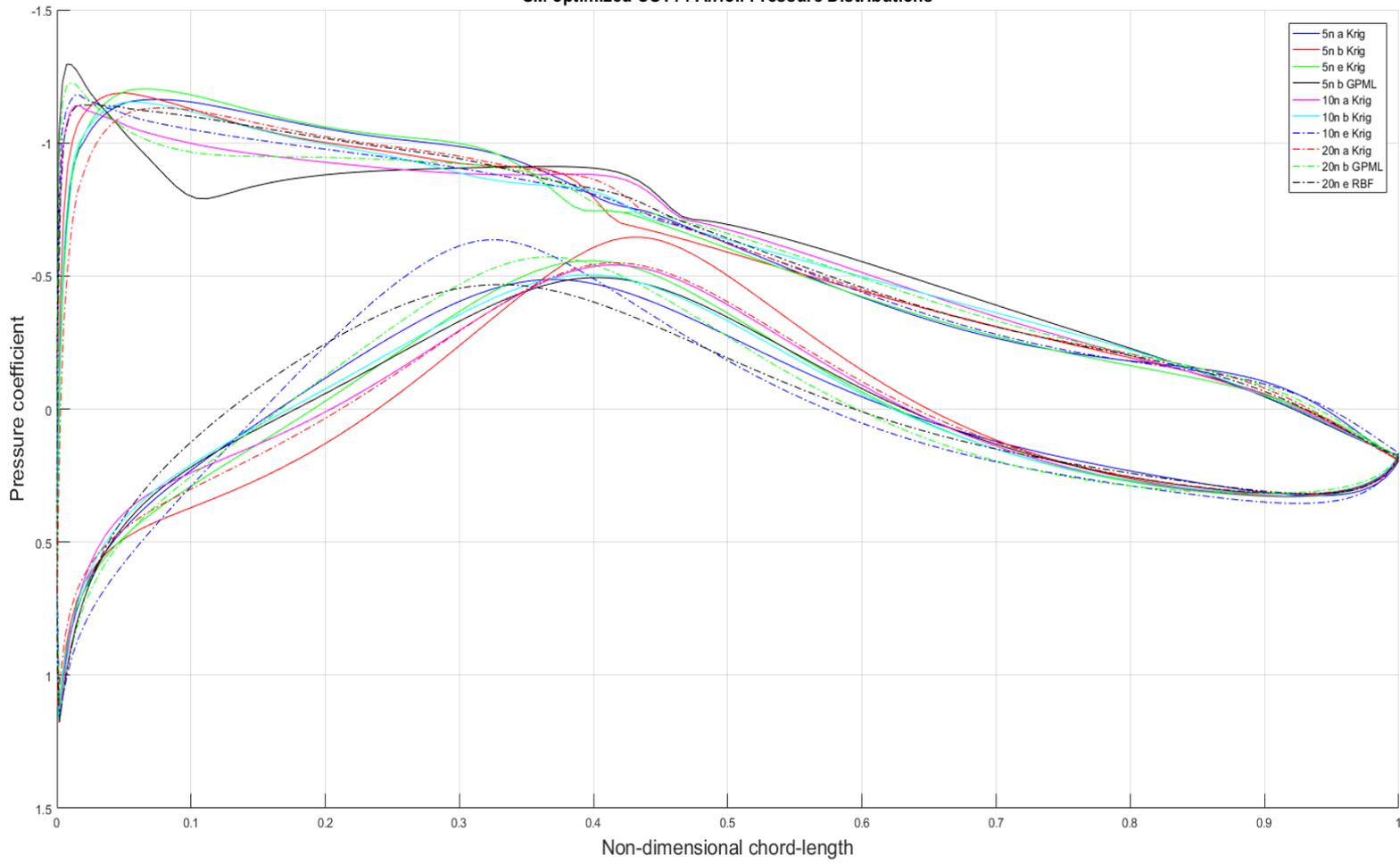


Figure 32 - CST14 surrogate model optimized airfoil pressure distributions.

SM optimized CST14 airfoils

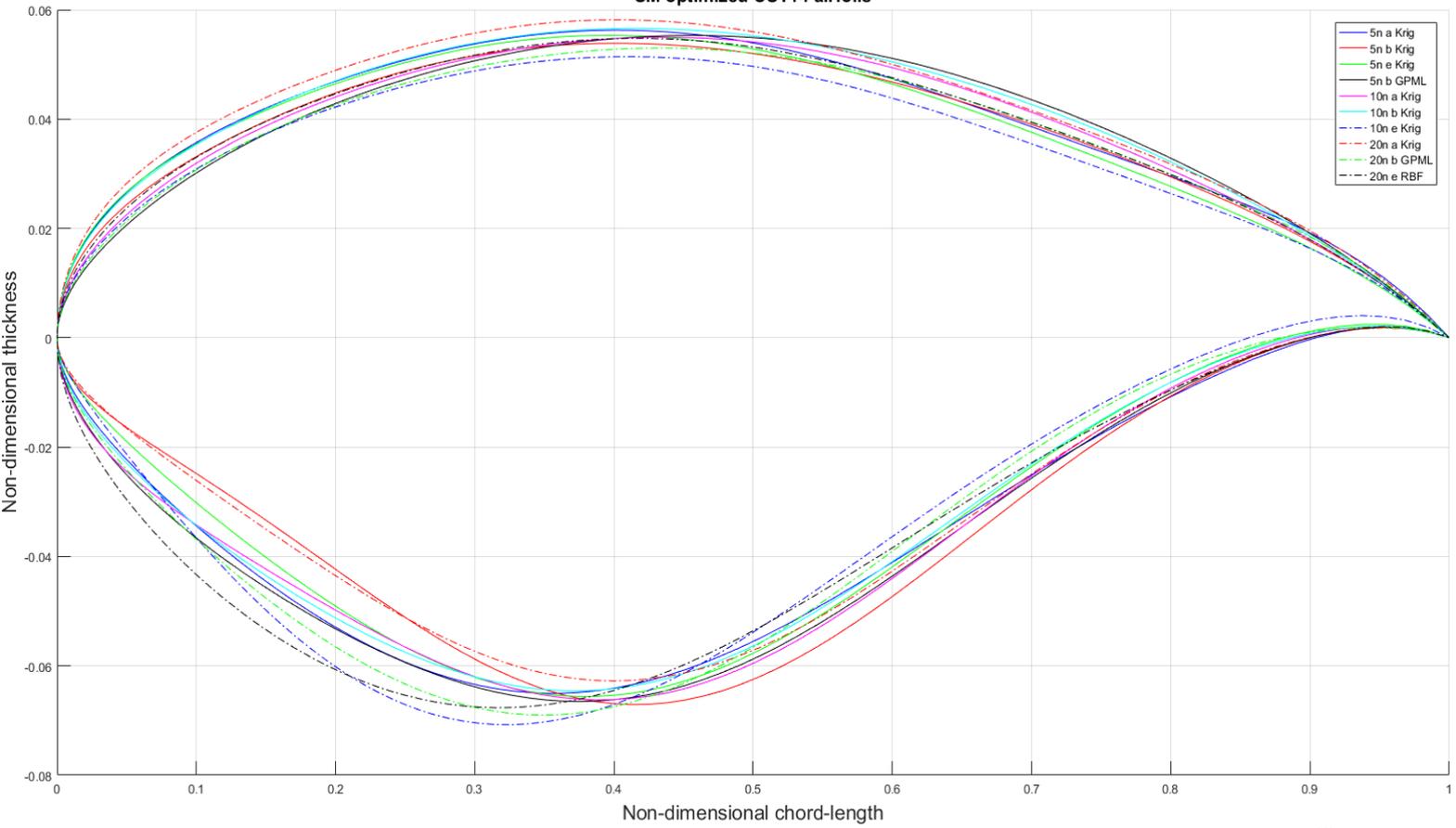


Figure 31 - CST14 surrogate model optimized airfoils.

5.5.4 uNURBS Surrogate model optimization results

The surrogate models selected in section 5.4.4 have been further optimized. Table 13 shows the results, figure 33 shows the pressure distribution and figure 34 shows the optimized airfoils.

Table 13 - uNURBS surrogate model optimized results.

uNURBS Surrogate model	Obj	CD	# Solver calls	AOA	Total Time [min]
5n_a_Krig	0.9376	0.010595	379	1.93°	76.73
5n_b_Krig	0.9361	0.010578	362	1.63°	110.61
5n_d_Krig	0.9336	0.010550	310	1.64°	71.05
5n_e_Krig	0.9426	0.010651	261	1.53°	76.73
10n_a_Krig	0.9693	0.010953	332	1.97°	370.23
10n_b_Krig	0.9354	0.010570	698	1.57°	136.53
10n_c_Krig	0.9295	0.010503	629	1.67°	370.22
20n_a_Krig	0.9327	0.010540	796	1.81°	260.58
20n_b_Krig	0.9327	0.010540	811	1.81°	240.45
20n_e_Krig	0.9325	0.010537	835	1.52°	285.52

The uNURBS surrogate model based optimization converged to lower angles of attack. Overall the results reached with uNURBS performed quite consistent in convergence to good performing airfoils. The outlier in the figures is 10n_a_Krig. It is also the only model with a significantly worse objective function compared to the other uNURBS optimizations.

SM optimized uNURBS Airfoil Pressure Distributions

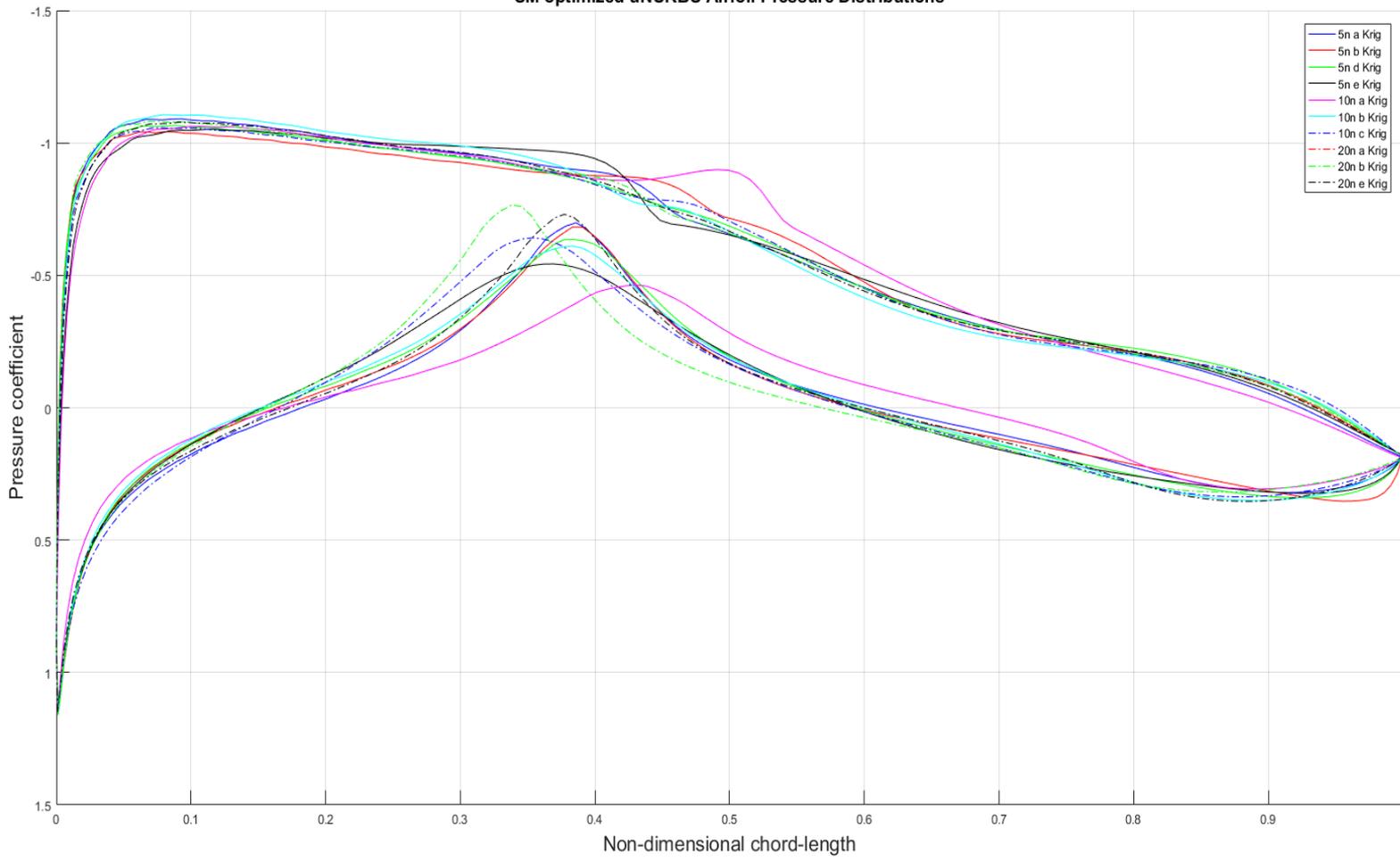


Figure 34 - uNURBS surrogate model optimized airfoil pressure distributions.

SM optimized uNURBS airfoils

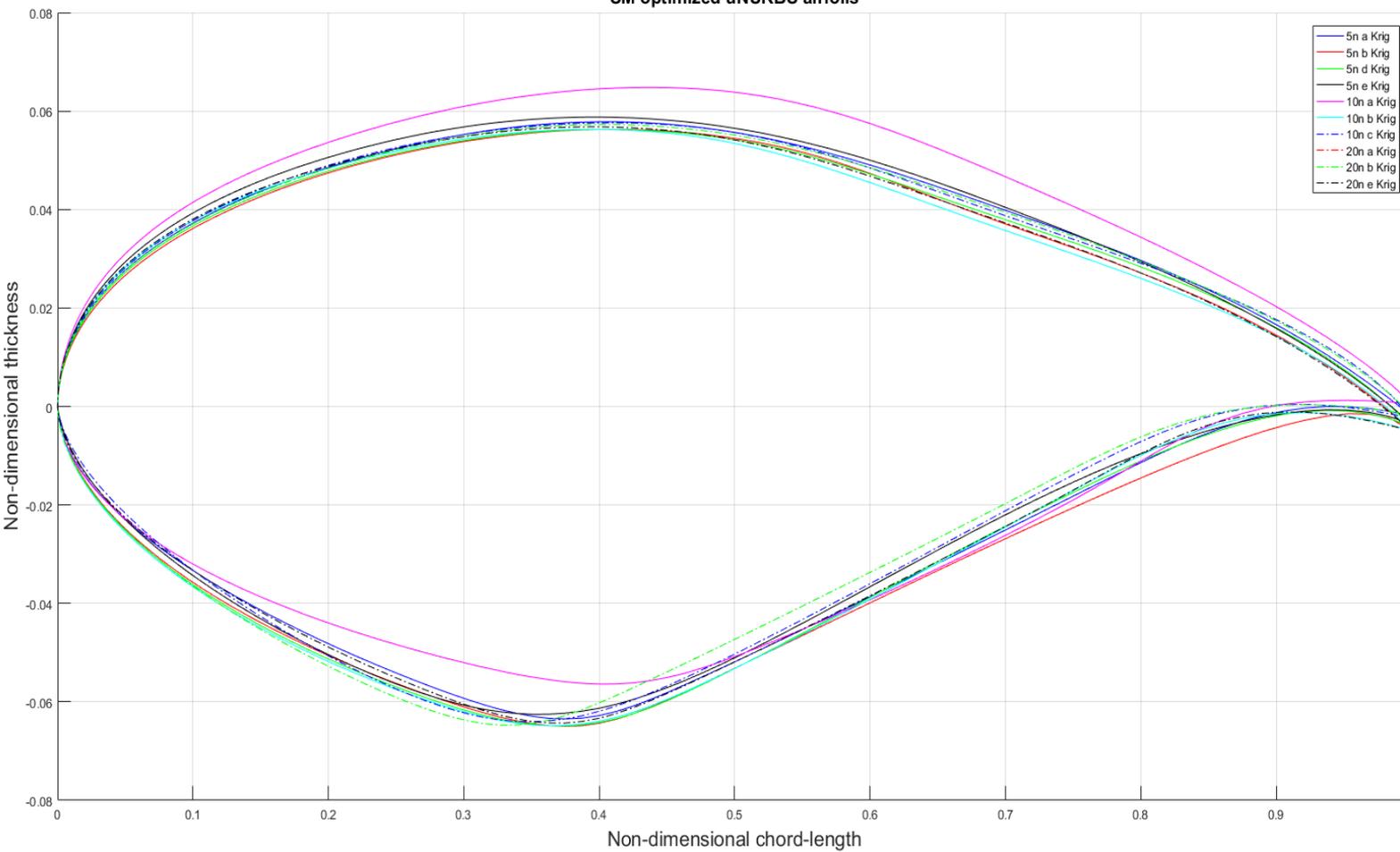


Figure 33 - uNURBS surrogate model optimized airfoils.

5.5.5 NURBS Surrogate model optimization results

The results for the NURBS parameterization are shown in table 14. Figure 35 and 36 show the respective pressure distribution and converged airfoils.

Table 14 - NURBS surrogate model optimized results.

NURBS Surrogate model	Obj	CD	# Solver calls	AOA	Total Time [min]
5n_a_Krig	0.9359	0.010576	707	1.38°	268.84
5n_b_Krig	1.0963	0.012388	327	2.02°	184.92
5n_d_Krig	0.9288	0.010495	857	1.64°	334.62
5n_e_Krig	0.9280	0.010486	816	1.77°	246.35
10n_a_Krig	0.9334	0.010547	1124	1.83°	481.78
10n_b_Krig	0.9278	0.010484	1499	1.91°	437.53
10n_d_Krig	0.9327	0.010540	828	1.90°	450.44
10n_e_Krig	0.9755	0.011024	720	1.78°	450.04
10n_b_RBF	0.9319	0.010531	1390	2.11°	479.14

The NURBS method leaned the most on the SQP descent to converge to good scoring objective functions. NURBS found lower drag coefficient designs than the other parameterizations have. The added parameters allowed for extra pathways during the gradient descent optimization. The optimization converged to higher angles of attack than uNURBS.

There is a lot more variance with the NURBS method than the uNURBS method. The increment in complexity caused by the added weights resulted in higher inconsistency in design space found by the genetic optimization of the surrogate models. Besides the larger amount of required training data the added weights resulted in significant longer computational time required for the model building and the genetic optimization.

SM optimized NURBS Airfoil Pressure Distributions

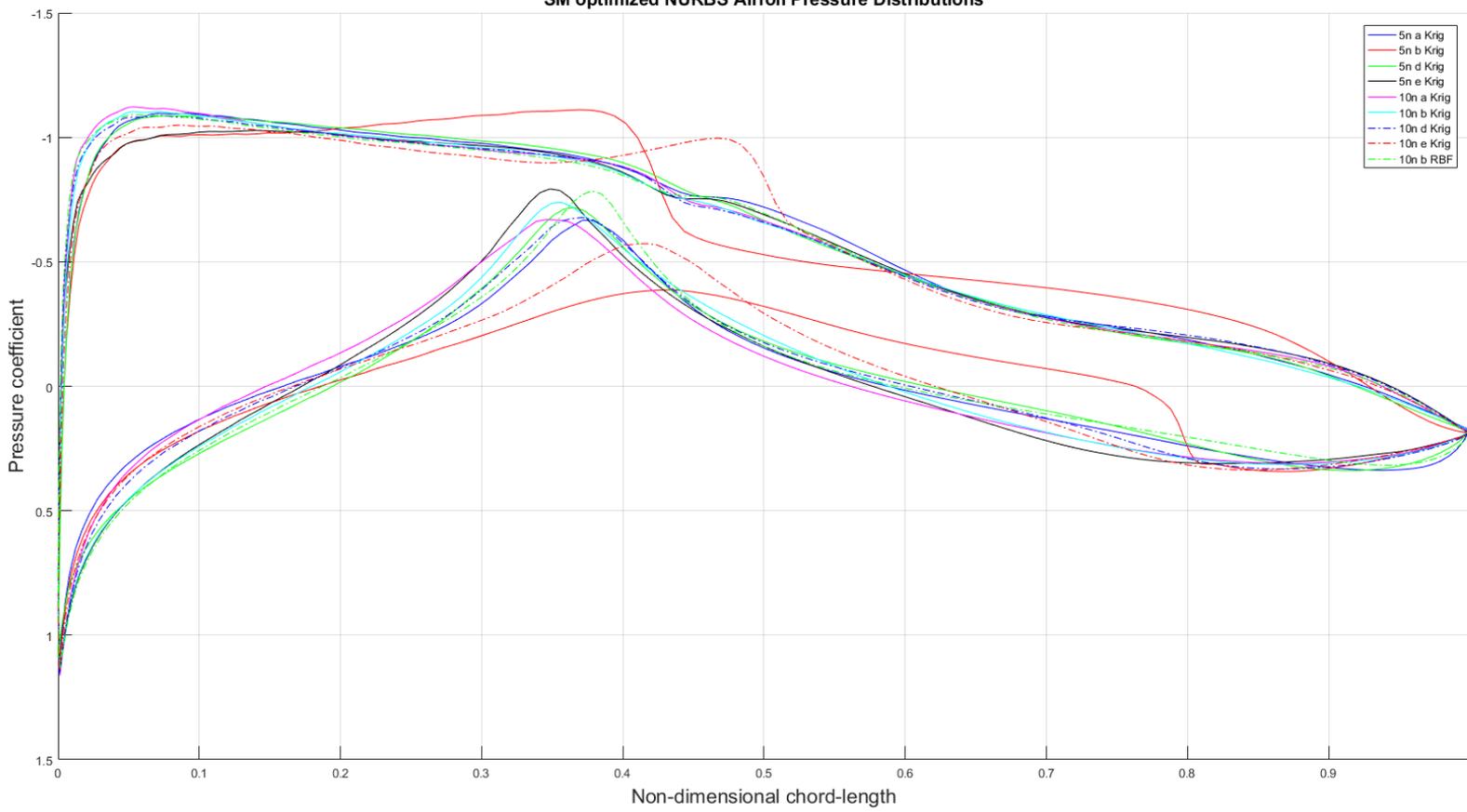


Figure 36 - NURBS surrogate model optimized airfoil pressure distributions.

SM optimized NURBS airfoils

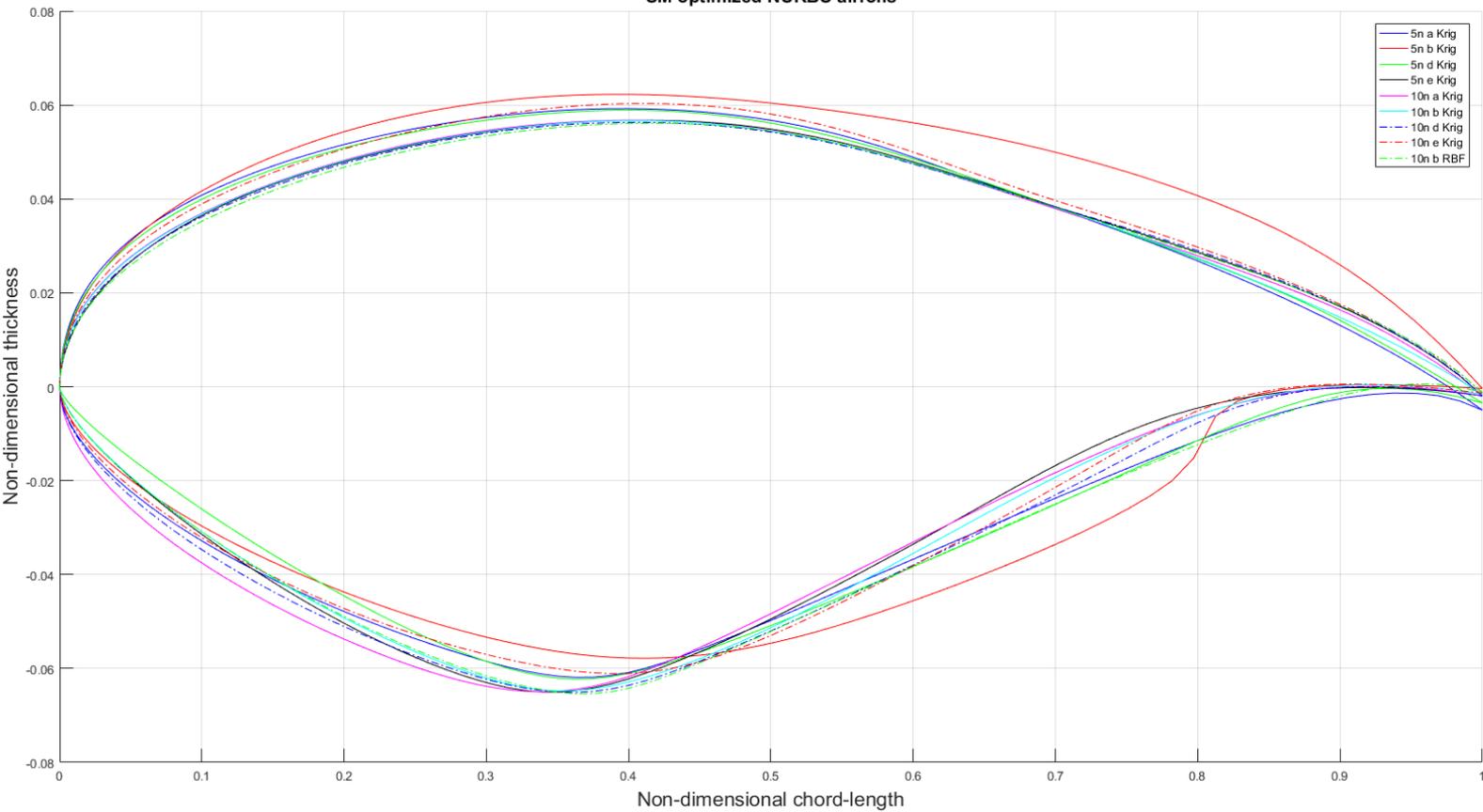


Figure 35 - NURBS surrogate model optimized airfoils.

5.6 Comparison of optimization methods

This section compares the result found for each parameterization with the different optimization strategies. The gradient descent results are compared to the better surrogate model based optimization results. The data as section 5.5 is used.

5.6.1 CST10

The best performing surrogate model for CST10 is the model build using a minimum of 50 samples, 15 samples for the initial TPLHD sampling plan and 39 sequential samples to build a Kriging surrogate model. The sequential sampling has been performed with a mixture of error and LOLA-Voronoi sampling (30-70 ratio respectively). The results are repeated in the table 15, figure 37 and figure 38. Unexpectedly the surrogate model based approach required less computational time than the gradient descent method.

Table 15 - CST10 optimization result comparison

Optimization CST10	Obj	CD	# Solver	AOA	Time [min]
Gradient descent	0.9354	0.010664	233	2.47°	45.62
5n_e_Krig	0.9293	0.010594	223	2.60°	39.31

The results show that the surrogate model approach has outperformed the gradient descent method in terms of effectiveness and efficiency. The surrogate model converged to a different design space than the gradient descent method. A higher peak pressure, smoother upper side pressure increase and a larger curve on the lower side.

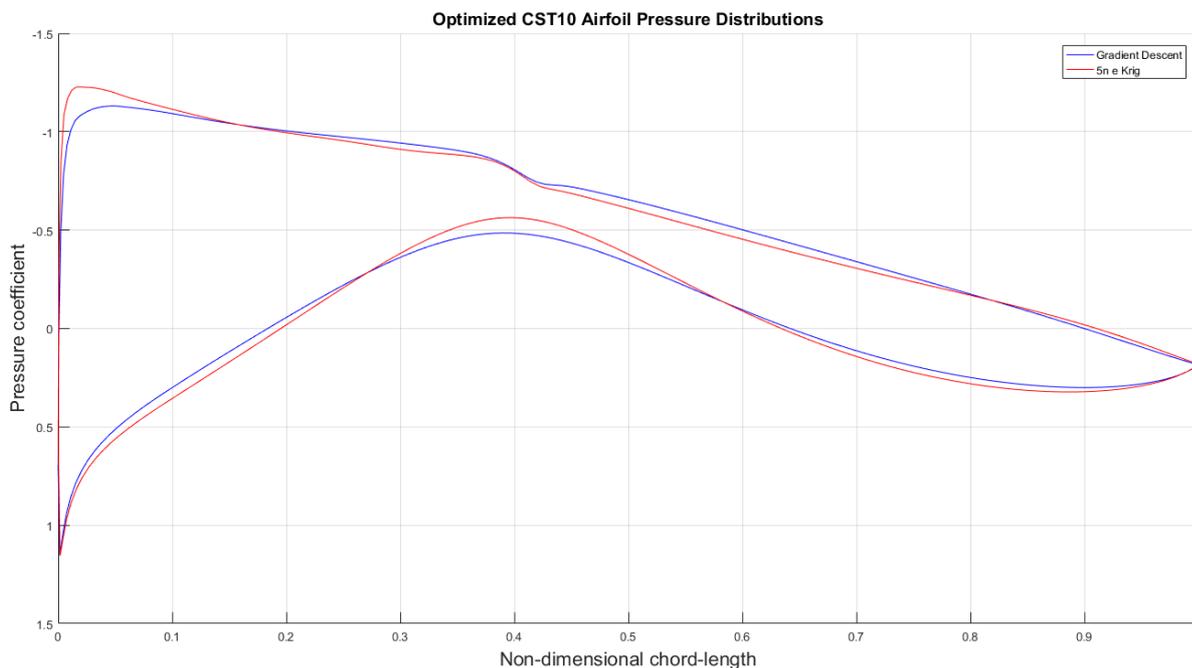


Figure 37 - CST10 optimized airfoil pressure distribution comparison.

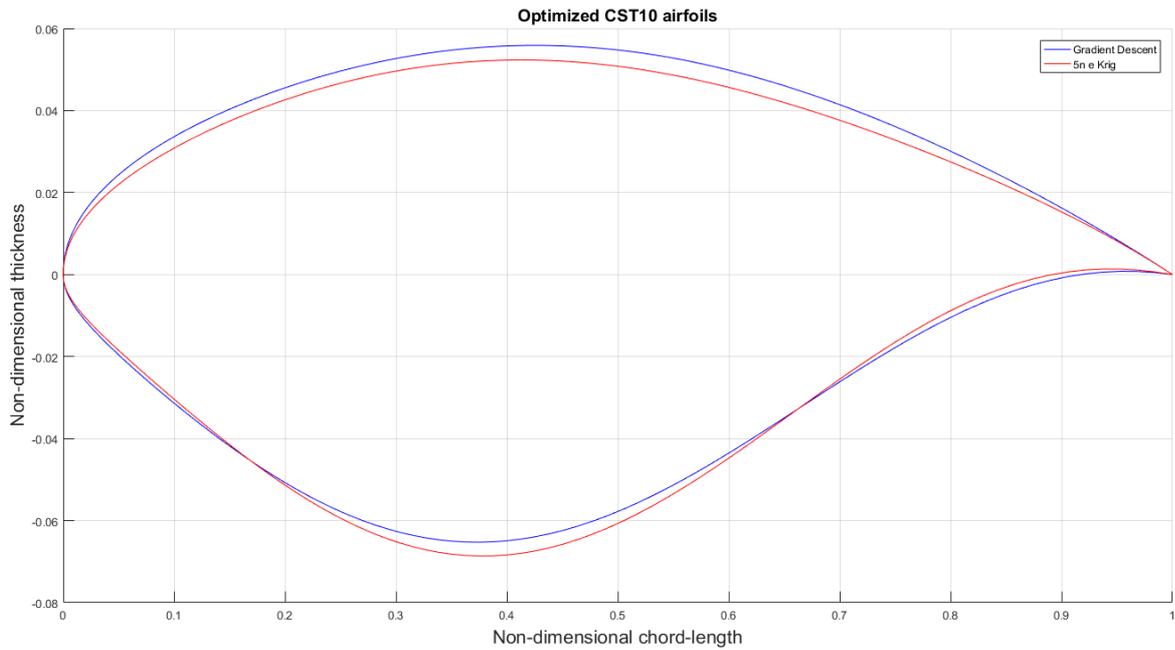


Figure 38 - CST10 optimized airfoil comparison.

The difference in pressure distributions shows that roughly the same design space is found with regards to the shape of pressure distribution. The surrogate model based optimized airfoil (left) finds a steeper peak at the leading edge and slightly more extreme division of front and aft loading. The CST methods parameters effect the full curve. The characteristic lack of local control shows at the thickest point of the airfoils, where an increase in pressure at the upper surface is noticeable.

5.6.2 CST14

CST14 managed to find an impressive minimum through surrogate model based optimization. The 10n sample, default sequential sampling combined with Kriging performance is unparalleled by any of its peers.

Table 16 - CST14 optimization result comparison

Optimization	Obj	CD	# Solver	AOA	Time [min]
CST14					
Gradient descent	0.9496	0.010635	204	2.47°	51,50
10n_e_Krig	0.9279	0.010393	415	2.22°	69.58

Figure 39 and 40 shows the different pressure distributions and airfoils. The surrogate model optimized airfoil shows a steeper peak pressure on the upper side leading edge and a smoother pressure curve on the lower side. The lower crest also moved forward. The difference in upper pressure distribution between CST10 and CST14 is noticeable. CST14 managed to smooth out the upper side pressure distribution kink. The added four parameters seem to be worth the computational cost.

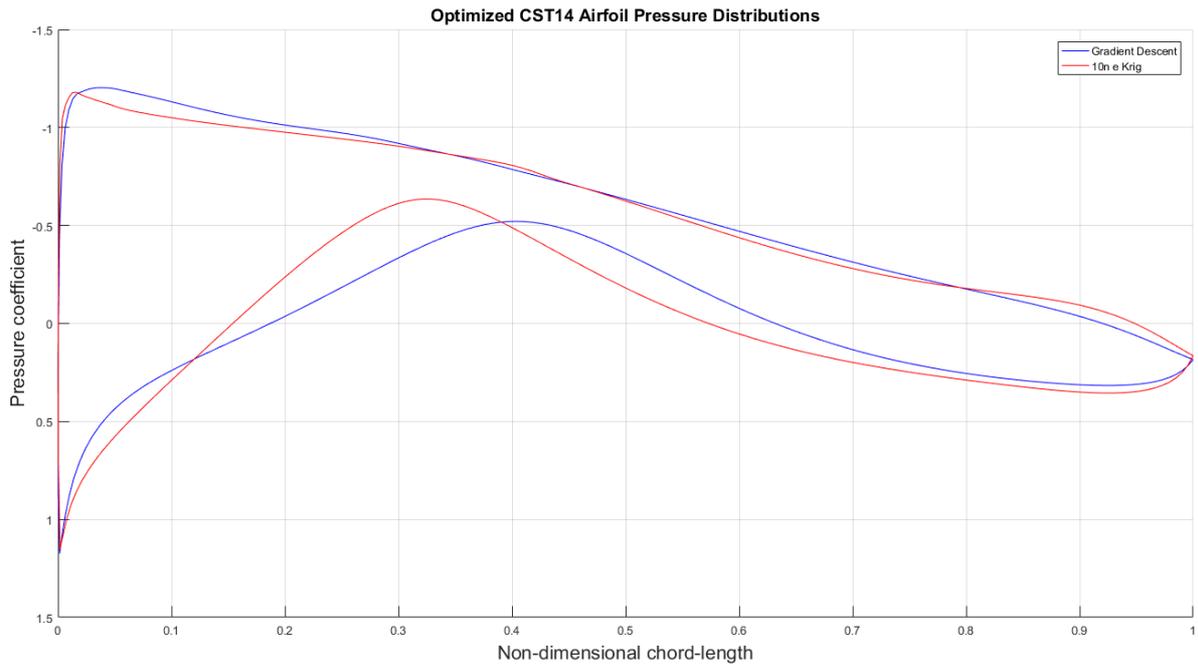


Figure 39 - CST14 optimized airfoil pressure distribution comparison.

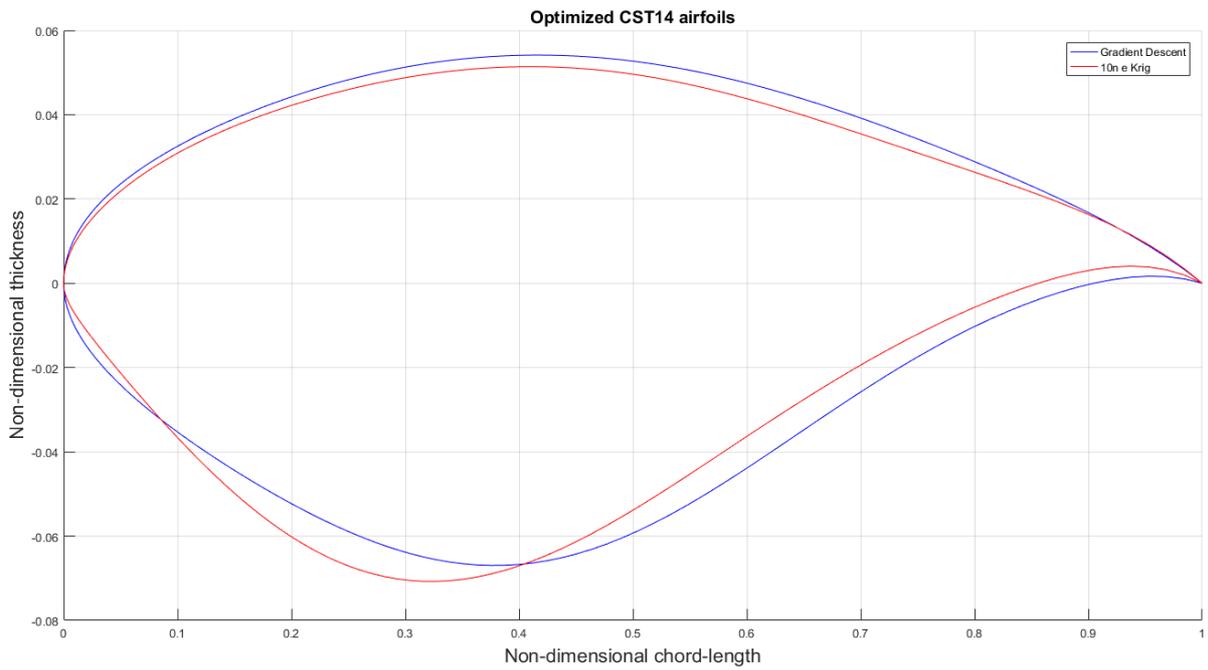


Figure 40 - CST14 optimized airfoils comparison.

5.6.3 uNURBS

The unweighted NURBS surrogate model optimization showed a variety of results. The NURBS methods required significantly more time for the genetic optimization of the surrogate model. The distribution of time for the different steps are presented in appendix E. There are two reasons why this step required significantly more time compared to the respective step for the CST10 and CST14 method. The first reason is a larger parameter count. The second reason is the difference in computational time for creating a NURBS airfoil compared to a CST airfoil. When more computationally expensive aerodynamic solvers are being applied the time difference will shift in favor of the surrogate model based optimizations.

Table 17 - uNURBS optimization result comparison

Optimization uNURBS	Obj	CD	# Solver	AOA	Time [min]
Gradient Descent	0.9404	0.010666	373	1.98°	75.70
5n_d_Krig	0.9336	0.010550	310	1.64°	71.05
10n_c_Krig	0.9295	0.010503	629	1.67°	370.22
20n_e_Krig	0.9325	0.010537	835	1.52°	285.52

The Kriging surrogate model with 10n samples, expected improvement and Kriging found the best optimum. However the computational time is the highest, most of the computation time (222 minutes out of 370 minutes) was used for the genetic algorithm optimization for the surrogate model.

The surrogate models present in table 17 all show significantly improved objective functions at a variety of design spaces that the gradient descent method could not explore.

The uNURBS SM optimized airfoils converged to airfoils that generally have a smoother upper pressure distribution seen in figure 41. The airfoil with best objective function also shows the smoothest lower surface pressure gradient. The uNURBS airfoils have more local control compared to the CST methods. This allows the method to obtain a smoother upper surface pressure gradient as it can locally manipulate the surface to improve the objective function.

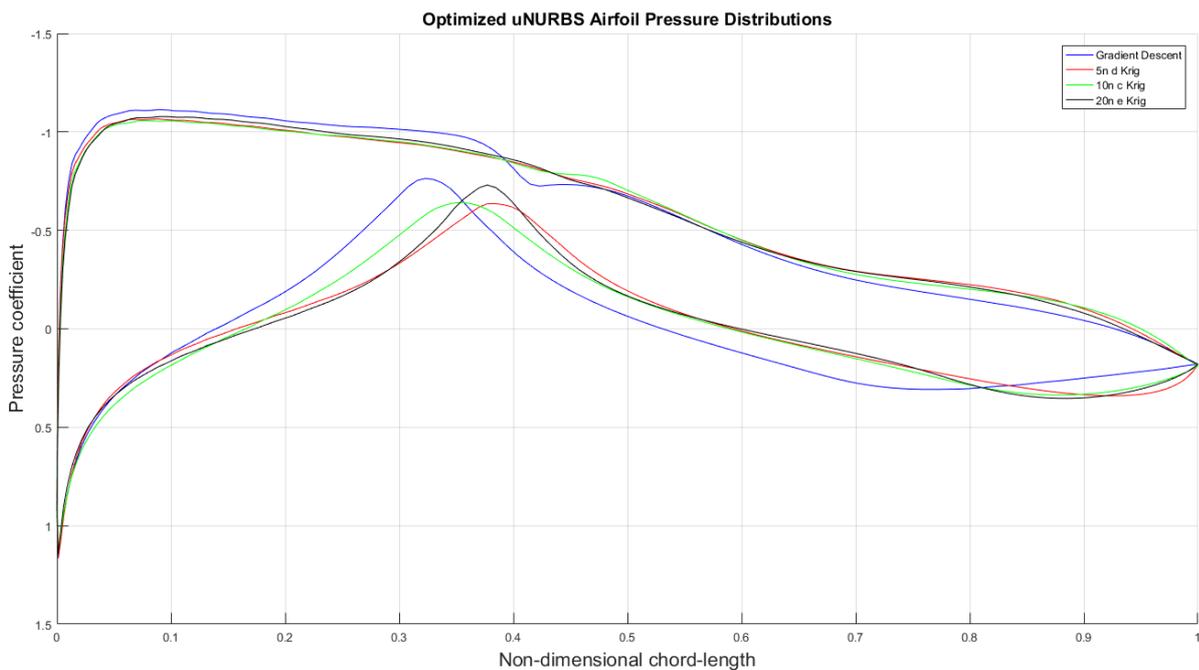


Figure 41 - uNURBS optimized airfoil pressure distribution comparison.

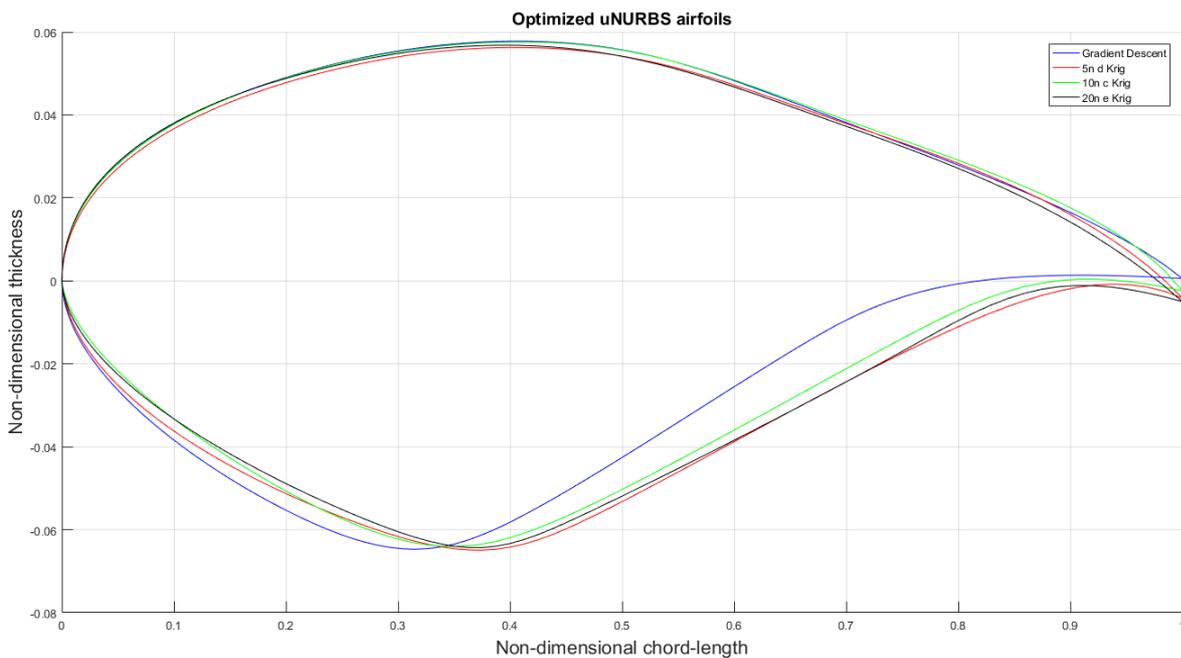


Figure 42 - uNURBS optimized airfoils comparison.

5.6.4 NURBS

The NURBS parameterization has the most flexibility of all the methods researched. The acquired optima have the lowest drag coefficient values. The gradient descent and surrogate model based optimization results are shown in table 18 and the resulting airfoils and pressure distributions are shown in figure 44 and 43.

Table 18- NURBS optimization result comparison

Optimization NURBS	Obj	CD	# Solver	AOA	Time [min]
Gradient Descent	0.9307	0.010541	1118	1.32°	174,16
5n_e_Krig	0.9280	0.010486	816	1.76°	246.35
10n_b_Krig	0.9277	0.010484	1499	1.91°	437.53

The design space found by the gradient descent and the surrogate models is judging from the pressure distribution not wildly different as other parameterization methods experienced. This also shows the strength of the weights for gradient descent optimization. The weights add extra dimensional pathways to better a better optimum. The reduction in orthogonality can work is a pro for gradient descent, but as a con for surrogate modeling.

5n_e_Krig required significantly less solver calls compared to the gradient descent method, but required more computational time. Computing the surrogate model at each sequential sampling stage and the genetic optimization required 38 and 120 minutes respectively.

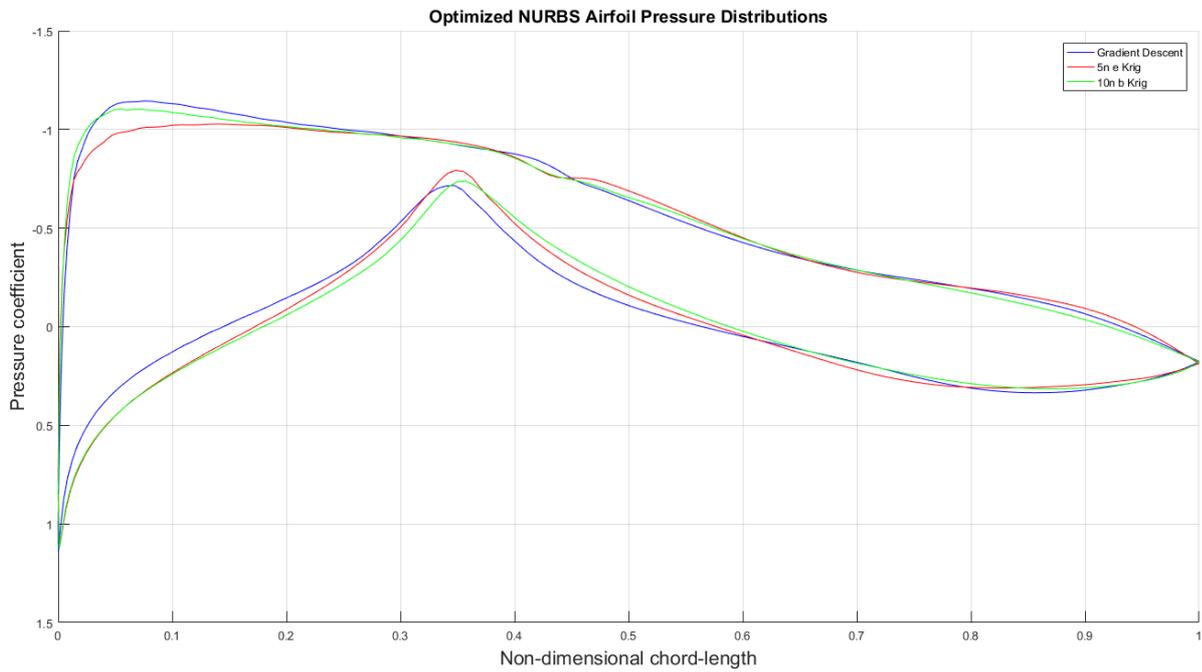


Figure 43 - NURBS optimized airfoil pressure distribution comparison.

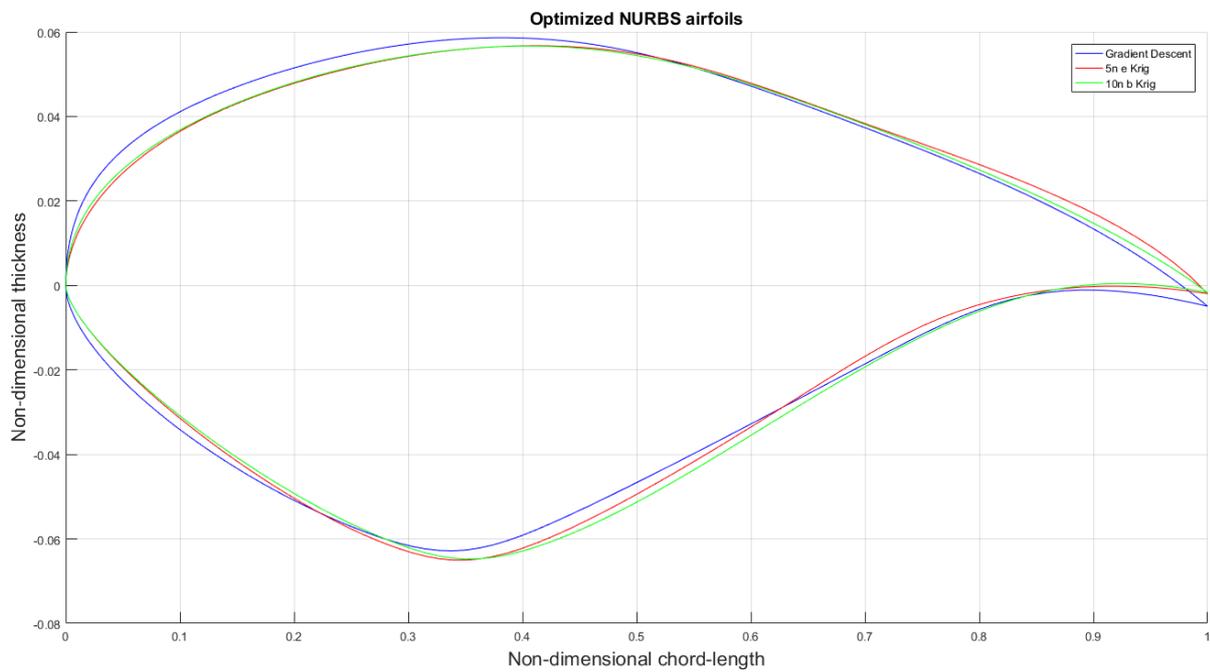


Figure 44 - NURBS optimized airfoil comparison

5.7 Comparison of parameterization results

As a final data comparison the best results from the various optimizations are shown. This comparison showcases the influence of selecting a parameterization method on the results. Table 19 shows the result data and figure 45 and 46 show the pressure distribution and airfoils respectively. The CST14 method produced the airfoil with the least drag.

Table 19 – Surrogate model optimized results.

Parameterization Optimization	Obj	CD	# Solver	AOA	Time [min]
CST10 5n_e_Krig	0.9293	0.010594	223	2.60°	39.31
CST14 10n_e_Krig	0.9279	0.010393	415	2.22°	69.58
uNURBS 10n_c_Krig	0.9295	0.010503	629	1.67°	370.22
NURBS 10n_b_Krig	0.9277	0.010484	1499	1.91°	437.53

The CST parameterization method requires significantly less time to converge to an optimum than the NURBS methods. The increment in variables increase the computational time required in three ways. Firstly more solver calls are required when building the surrogate model and when performing the final gradient descent search. Secondly computing the hyperparameters of the surrogate models requires more time at each sequential sampling stage. And thirdly the genetic optimization requires more time for higher dimensional problems and becomes less likely to find a global optimum.

Figure 46 show that the CST airfoils converged with a deeper lower side crest than the NURBS airfoils did. Due to robustness issues the parameter range on the NURBS airfoil had to be restricted in that area, hence that particular design space might not have been available to the NURBS methods. This can also explain why CST14 managed to find a better performing airfoil than the NURBS parameterization methods. Regardless of the specific bounds the CST methods kept converging to airfoils above the initial 2° angle of attack and the NURBS method kept generally converging to lower angles of attack.

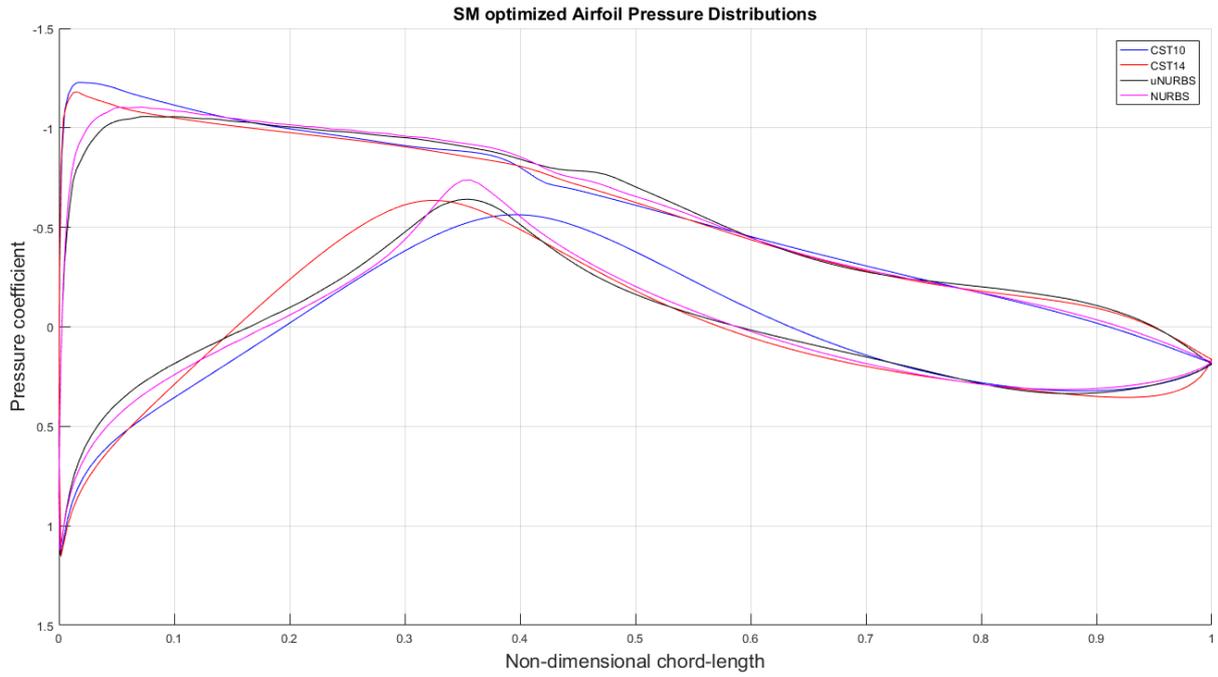


Figure 45 - Surrogate model optimized airfoil pressure distributions.

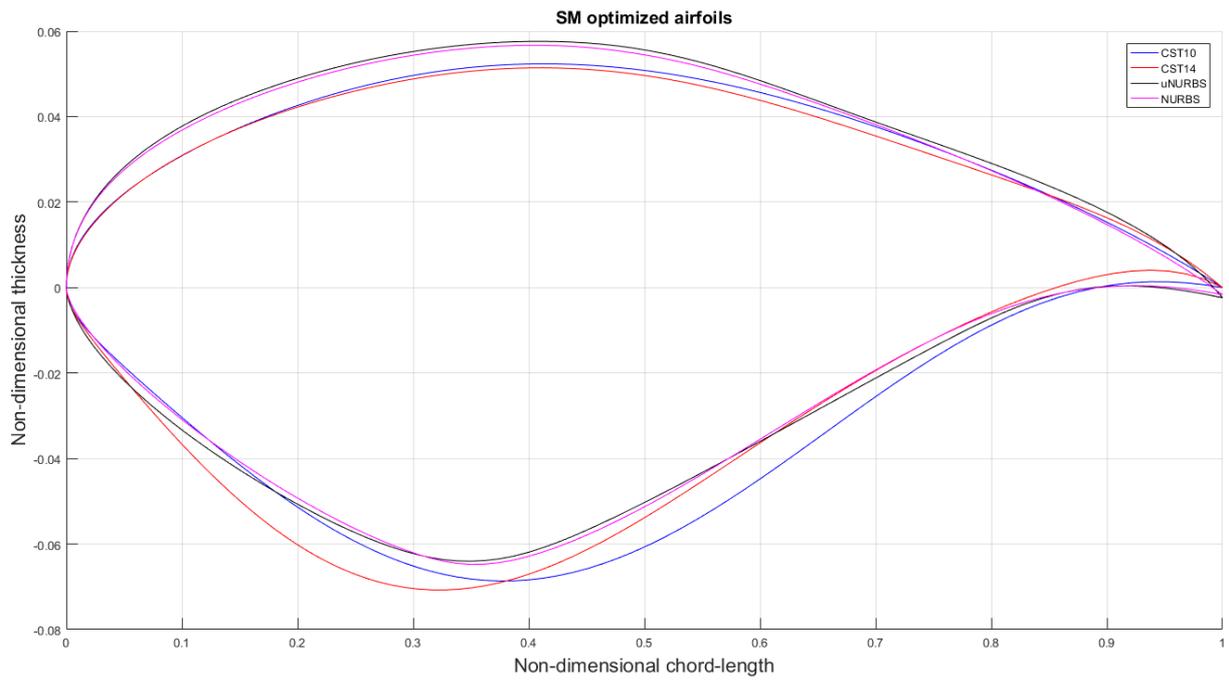


Figure 46 - Surrogate model optimized airfoils.

6 Discussion

This chapter provides a discussion based on the results and experienced gained throughout the research. The discussion starts in section 6.1 with the effect of the parameter range. Afterwards the a discussion is presented in section 6.2 on the different optimization strategies with regards to effectiveness, efficiency, robustness, feasibility and the dependency of the initial design. Section 6.3 provides an in-depth discussion on the different sampling strategies researched. Section 6.4 discussions the pros and cons of the different parameterization methods with regards to optimization. And finally the effect of the sample size is discussed.

6.1 Parameter range

The parameter range set for the parameterization methods is set by an upper and lower bound. This range depicts the design space in which the optimization methods operate. The larger design space influences surrogate model based optimization differently than how it influences gradient descent optimization. These influences are hypothetical as no data proofing these influences has been collected in this research; however the influence is scarcely mentioned in papers but important enough to warrant its own discussion section. First the interaction between parameter range and gradient descent optimization is discussed, afterwards the interaction with surrogate model based optimization and the possible influence on the results are discussed.

6.1.1 Parameter range influence on SQP optimization

When setting bounds for gradient descent optimizations the user can keep the design space quite large without much downside. This does not include bounds that need to be set for manufacturing reasons etc. Generally the gradient descent will not converge to unfeasible designs and limits on bounds are not always reached. Hence a large parameter range just increases the design space in which the optimizer can operate, which may or may not be used without extra costs. Unfeasible design space that is included in the large parameter range is usually not converged towards.

The downside for a wide parameter range for gradient descent optimization happen when the optimizer finds a very good correlation between one parameter and the objective function and this parameter keeps converging to extreme values. This causes the optimizer to delve deep into a design space that might or might not be good. In these cases the user can adjust bounds to prevent this from happening.

6.1.2 Parameter range influence on SM based optimization

The influence of the parameter range is larger for surrogate model based optimization. The user sets a certain design space surrounding the initial airfoil with the parameter range. The surrogate model building process aims to understand the design space set by the user. A larger design space brings to problems for the surrogate model two understand the correlation between parameters and objective function. Not only does the surrogate model contain fewer samples per design space, but a larger design space is more likely to include complex unfeasible design space that the surrogate model is trying to understand.

The impact of fewer samples per design space is fairly straight forward to understand. Larger design space per sample reduces the accuracy of the overall model as there are relatively larger unexplored design spaces between the samples.

The influence of including extra unfeasible design space with high complexity influence can be more detrimental to the surrogate model building process depending on the sampling strategy chosen. For example LOLA-Voronoi sequential sampling places samples at non-linear design space. For airfoils unfeasible complex design space is often non-linear. This can result in an unnecessary large portion of samples being located in unfeasible design space. Besides these samples being computational waste they also influence the surrogate

model. The surrogate model might build itself to understand the unfeasible design space at the cost of accuracy on interesting design space.

Both of these influences are difficult to predict without knowledge on the design space beforehand. The influences are also different for different parameterization and difficult to measure. The parameter range chosen should be a conscious trade-off between computational efficiency, surrogate model quality and the wanted size of the design space.

Higher dimensional problems should experience more problems with large parameter range as the added design space per parameter is increased among all dimensions.

6.1.3 Parameter range used for this research

The parameter range used for this research, presented in appendix A is, in hindsight, on the large side. The bounds used for the gradient descent were quite large to widen the design space. The same bounds were initially applied to the surrogate model building process. Besides fine-tuning to avoid crashes from unfeasible designs the bounds have been largely kept the same.

A smaller parameter range can lead to different results and conclusions due to the aforementioned influences of the parameter range on the surrogate model building process. The effect of this influence depends on the sequential sampling space. Some discussion is presented below on how these effects interact with the results presented in this thesis.

The effect of larger design space per sample should have a negative effect on the quality of the surrogate models. LOLA-Voronoi sequential sampling is equipped to deal with large unsampled design space as the Voronoi cell strategy offers explorative qualities. Hence all the models with an A or an E suffer respectively less from this effect. Surrogate models created with Generalized Probability of Feasibility (B), Expected Improvement (C) and Gradient Knowledge (D) are more dependent on the initial sampling plan and should experience more loss in quality as the limited information per design space is not counteracted by explorative qualities that LOLA-Voronoi offers.

The effect caused by inclusion of unfeasible complex design space implies that the following models can perform better than they did based off the results.

The surrogate models created with LOLA-Voronoi (A), Knowledge Gradient (D), and the mixture between error and LOLA-Voronoi (E) all can suffer in quality from the inclusion of highly non-linear unfeasible design space. The surrogate models with Knowledge Gradient should suffer less than the LOLA-Voronoi models as Knowledge Gradient locate samples at favorable gradients and LOLA-Voronoi would explore the design space in favorable and unfavorable direction surrounding complex unfeasible design space. Generalized Probability of feasibility (B) and Expected Improvement (C) would not exploit sampled complex unfeasible design space further. And experience less loss in surrogate model quality as a result for this effect. Specifically generalized probability of feasibility managed provide good design space coverage by excluding probable unfeasible design space.

Overall it is difficult to pinpoint which models experience larger penalties and in which quantity from the relatively large parameter bound used. Throughout the results LOLA-Voronoi based sampling and generalized probability of feasibility performed the most consistent in creating models with good fitting errors. This could very well be due to the explorative qualities from the LOLA-Voronoi that explore the large unsampled design space between samples that the other sampling methods were struggling with. This in turn makes LOLA-Voronoi (A) a good sequential sampling strategy for when the design landscape is unknown.

6.2 Surrogate model optimization versus gradient descent optimization

From the results it shows that the surrogate model based optimization found better designs than the gradient descent method did. However this is not the only criteria an optimization strategy should be chosen. The best design is the goal of an optimization but it is always a trade-off between costs and results. This subsection discusses effectiveness, efficiency, robustness and feasibility.

6.2.1 Effectiveness

Effectiveness describes how effective the method is for obtaining the best optimum. Since chapter 5 showed that all parameterization techniques obtained lower drag values with the surrogate model based approach. It is safe to say the effectiveness for surrogate model based optimization is higher than the gradient descent optimization.

The main reason why surrogate model based optimization is more effective than gradient descent optimization for airfoil drag optimization is due to the complexity of the problem. Gradient descent optimization only explores a very limited part of the design space before it gets stuck in a local minimum and is heavily reliant on the initial starting design. Table 11 shows various optima found for the CST10 method. The different results show that each different starting point provided a different optimum. The different converged minima show the complexity of the design landscape and how important the initial starting design is for gradient optimization.

The ability to explore larger sections of the design space is paramount to the successful results of the surrogate model based optimizations. Besides better objective function value the added benefit of exploring different designs allows an engineer gain extra insight in the problem and re-adjust constraints/bounds/problem formulation avoiding mistakes that could have gone undetected otherwise.

6.2.2 Efficiency

Finding a good optimum with minimal computational costs equates to an efficient optimization strategy. The surrogate model based method had different results regarding the efficiency criteria. Chapter 5 shows that CST10, CST14, uNURBS and NURBS all found a better optimum with surrogate model based optimization. The gradient descent method showed to be generally faster.

The surrogate models build with 5n training data for CST10 and CST14 performed competitive time-wise with the gradient descent optimization. uNURBS and NURBS required more computational time for evaluating the hyperparameters at each sequential sampling stage and the genetic optimization. The increment in time required for the genetic optimization is due to the larger dimensionality and the increased cost of computing a NURBS airfoil compared to a CST airfoil.

The time efficiency can shift more towards surrogate model based optimization when more expensive solvers are being applied. As the computational effort for the evaluating hyperparameters and genetic optimization become a smaller fraction of the total computational time.

6.2.3 Robustness

A robust method does not create errors or crashes. With regards to robustness the gradient descent optimization is better than the surrogate model based optimization. The randomness involved in the sampling process can create unfeasible/crash worthy designs. uNURBS and NURBS both required tweaking to their bounds to avoid the lower and upper surface to stop crossing near the trailing edge. These problematic airfoils were resulting from unfavorable stacking of parameters, where the control point coordinates and weight all manipulated the curve in the same direction.

BP3434 succeeded to run a gradient descent search, but failed to evaluate the initial sample plan regardless of tweaks made to the bounds. The BP3434 parameterization method failed to evaluate the initial sampling plan due to poor interaction between the methods equations and a flat leading edge camber curve direction.

CST method did not require any tweaking with regards to robustness and are incredibly user friendly.

6.2.4 Feasibility

Feasibility is not as straight forward than the other criteria. Feasibility takes efficiency, effectiveness and robustness into consideration. Depending on the nature of the optimization problem the gradient descent method can still be more feasible than a surrogate model based optimization strategy would be. A surrogate model based optimization requires more time and thought to set up compared to a gradient descent optimization, hence the pay-off should be large enough. This is often the case for high-fidelity problems and problems with complex design landscapes where the area of global minimum is not known yet. The problem should also be capable of dealing with the randomness added during the sampling phase.

This research showed that CST and NURBS methods are feasible for both optimization strategies. BP3434 was not feasible for surrogate model based optimization due to robustness issues.

6.2.5 Influence of initial design

Both optimization strategies use the REA2822 airfoil as starting point which influences the optimized design, however only the gradient descent optimizations result is dictated by the initial design. Surrogate model based optimization is influenced to a lesser extent by the initial design. This subsection explains the importance of the initial design for both strategies.

The marble metaphor depicts this well for the gradient descent optimization. Releasing a marble on a landscape with numerous curves will make the marble roll to the first local minima it finds. Depending on the landscape different initial design will roll to a different minimum.

Surrogate model based optimization uses the initial design in a much different way. The initial design combined with the parameter bounds set the design space the surrogate model needs to understand. The size of the design space surrounding the initial design is set by the bounds. The initial design only decides where the center of that design space is located. Surrogate model based optimization results are dependent on the initial design but not decided by it.

6.3 Sequential Sampling Strategy

The sequential sampling strategy is paramount for a surrogate model to be able to understand the given design space to a useful extent. Multiple factors should be considered when selecting a sequential sampling strategy. This subsection discusses exploitation versus exploration and dependency on initial sampling. The influence on the applied sequential sampling strategies is discussed as well.

6.3.1 Exploitation versus Exploration

A good sequential sampling method should aim for a good balance between exploration and exploitation. The required balance is dependent on the problem. Factors that should be taken into consideration computational sources available compared to solver cost, design space size selected and the complexity of the problem.

Exploration refers to the capability of obtaining a broader view of the whole design space set by the user. Large parts of unsampled design space can be a risk to the models accuracy. Large gaps in information that remain unaddressed can result in missing possible interesting design space that could obtain an global optimum, or could be competitive with the global minimum. Even if the unsampled design space does not entail interesting optima the information obtained from sampling that design space can lead to increased regression qualities for the surrogate model. Good explorative qualities are required when the design space is largely unknown. When the design space is understood and an estimation of in which design space the global optimum may lay explorative qualities become redundant.

Exploitation describes a deeper understanding of a specific design space. Often interesting design spaces should be exploited further. Thus either design spaces that can increase the overall understanding of objective-parameter interaction, or design space that possesses interesting minima. Good exploitative qualities results in the design space surrounding interesting minima being well understood. Good exploitation qualities are nearly always welcome, but are specifically required for small design spaces that could possess a global optimum.

Given the relative large design spaces used for the optimization presented in the thesis it is no surprise that LOLA-Voronoi based sampling proofed most consistent as it has both exploration and exploitation qualities. The danger with LOLA-Voronoi is mostly one of computational waste. Especially problems with high dimensionality and large parameter range. The large design space is likely to have unfeasible design space that show non-linear behavior. LOLA-Voronoi samples at non-linear design space to increase the understanding of that design space. Cutting unfeasible design space by reducing the range on the bounds should increase efficiency of the surrogate models.

Generalized probability of Feasibility also contains a reasonable mixture of exploration and exploitation qualities. The method covers a broad design space whilst reducing possible computational waste by avoiding sampling in estimated unfeasible design space. This results in the surrogate model mostly attempting to understand design spaces that proofed to be feasible. However an inherent risk exists; interesting design space can be skipped when unfeasible samples are evaluated nearby.

Expected improvement performed the least consistent which can be explained by its poor exploration qualities. Expected improvement should perform better with smaller design spaces. For airfoil optimization expected improvement can be used effectively by adding another stage of surrogate model building surrounding the genetically optimized estimated optimum. Another explanation for the poor performance of expected improvement lays within the ability to recover from poor initial sampling. The initial sample size of $3n$ may have been too small for the size of the design space set by the parameter range. The poor explorative qualities from expected improvement make the surrogate model building process highly reliant on the initial sampling in contrast to other sequential sampling methods.

Knowledge Gradient did not perform well. Sampling at positions with that have the largest expected incremental value of information gained seems to not work well with this specific problem. The lack of performance can be explained due to the high dimensionality, large parameter range or a combination of

both. All of these factors result in a large design space that includes many unfeasible areas. However these unfeasible areas are likely to be sampled due to high expected value of information gained. A reason for the error scores of the 10n_d_Krig for NURBS were good is likely due to the amount of randomness involved during the initial and sequential sampling process.

The final sequential sampling design used is the mixture of LOLA-Voronoi and error based sampling. This method performed consistently well. The mixed exploration and exploitation qualities from LOLA-Voronoi are accompanied by the error based sampling which aims to sample at location where the model error is estimated to be the largest. The combination is able to cover a large design space well and seems too able to recover from poor initial designs.

6.4 Parameterization method

The parameterization methods have large influence on the converged airfoil and optimization process. Depending on the parameterization selected a different optimization strategy can be preferred. This section first describes the BP3434 failure. Afterwards the effect of dimensionality is discussed. In section 6.4.3 a discussion on CST versus NURBS is presented.

6.4.1 BP3434 failure

The fact that BP3434 parameterization crashes the solver during the initial sampling process does not invalidate the BP3434 for all optimization use. The specific combination of the REA2822 as initial airfoil accompanied by the randomness in sampling resulted in a cotangent function operating near the vertical asymptote. The BP3434 method can still be used for surrogate modeling with different initial airfoils that have a higher leading edge camber curve direction.

6.4.2 Dimensionality

The parameterizations CST10, CST14, uNURBS and NURBS have 10, 14, 21 and 35 parameters respectively. The amount of parameters equate to the dimensionality to the problem. Each added parameter increases the design space more than the previous one. The higher dimensionality also adds more unfeasible design spaces. This is specifically the case where the curve can be manipulated in the same way by different parameters. NURBS is the largest violator in this case; a curve can be manipulated in a certain direction by moving X, Y or the weights. These effects can stack up unfavorably and can be difficult to contain by lowering bounds. Lowering the bounds to avoid unfavorable stacking has a side effect of removing valid design spaces where these parameters do not cause unfavorable stacking. The larger design space and the added unfeasible design spaces both contribute to increased computational costs.

Another effect of increasing dimensionality is the feasibility of TPLHD as initial sampling instead of LHD. The risk of clustering of sample points is largely avoided with TPLHD sampling as explained in section 4.2. CST10 and CST14 have low enough dimensionality such that the computational time spent on computing the initial sampling plan is justified. uNURBS and NURBS used LHD as a result of the large parameter count. The negative effect of the reduced quality of the initial sampling depends on how well the sequential sampling plans exploration quality. Regardless of the sequential sampling plan LHD is more likely to produce computational waste than TPLHD.

6.4.3 CST versus NURBS

The two main parameterization methods researched are CST and NURBS. The NURBS method is capable of making local changes to the shape of the airfoil curves. The CST method lacks this quality; any change made to a parameter with CST is experienced over the entire curve. The NURBS method is thus more flexible than the CST method in terms of manipulating the airfoil. The increased flexibility correlates with an increased complex design space and increased computational costs.

CST

The lack of local control and reduction in parameters make CST easier to understand for surrogate models. This reduced complexity of the design space increases the accuracy of the surrogate model. Both of these effects help the genetic optimization to converge to design spaces that are likely to contain feasible good designs. The CST methods converged to angles of attack higher than the 2.0° in contrast to the NURBS methods that mostly converged to angles of attack lower than 2.0° .

CST10 vs CST14

The four added parameters proved their worth. The increase in control and flexibility on the curves for the CST14 method allowed for design spaces that the CST10 method could not achieve. The increased computational costs are noticeable but worthwhile.

NURBS

The local control in the NURBS method with increased dimensionality increases complexity and design space for the surrogate model to estimate. This reduces the accuracy of the surrogate model and the design space found by the genetic optimization can vary more in terms of successfulness. The optima's found with the NURBS method are largely thanks to the final exploitation in combination with the flexibility of the weights. NURBS did not manage to achieve a low drag coefficient as CST14 did. One reason for this result can be found in the bounds. The bounds set on NURBS during debugging of the MSES solver were quite strict regarding how deep the lower curve could go. The bounds are shown in appendix A. The CST14 optimum explored the design space with the lower curve dragged down. With NURBS this design space caused some robustness issues with the MSES solver. Hence the design space where CST14 found the best design was as accessible for the NURBS methods.

This shows that higher robustness allows for more completeness.

uNURBS versus NURBS

The difference between weighted and unweighted NURBS is noticeable in the optimum found. The extra flexibility of the weights allows the gradient descent optimization find different pathways to better designs. For surrogate model based optimization the added flexibility can be more of a liability than an asset. The extra flexibility increases the design space, complexity and the added flexibility also allows for the different parameters to stack in a specific ways that creates unfeasible design spaces.

A recommendation for NURBS surrogate model optimization is to reduce the complexity of building the surrogate model and genetically optimize by using the unweighted variant. The found design space can then be optimized with the weighted NURBS variant. This strategy maintains the pros of both methods whilst avoiding the largest cons.

6.5 Sample Size

Based on the different sample sizes used for the surrogate model building some observation can be made on the interaction of sample size with error score, surrogate models, sequential sampling and what kind of trade-off an engineer has to make with regards to selecting an appropriate sample size.

6.5.1 Error score

From chapter 5.4.3 it is evident that a larger sample size benefits the error score for the surrogate models. The difference in error score is generally larger for the best models from 5n to 10n samples. The difference in error score between 20n and 30n shown in figure 18 and 19 is relatively small. The extra computational cost for the added samples do not seem worthwhile compared to the error score benefits from 20n to 30n. For all parameterizations the difference in error score between 10n and 20n seems relatively significant. Specifically the exploited 20n models converged to significantly better designs than the 10n models on average. For uNURBS the difference in quality of the converged airfoils between the 10n and 20n models is too small to warrant the computational costs. The 5n models have the lowest score, but can be surprising in the design space they find. However for consistency a 10n-20n sample size would be better for the model score and the likeliness of finding a good design.

6.5.2 Surrogate Models and sample size

Across all sample sizes Kriging seems to be the most consistent surrogate model in producing good model error scores. For the 5n surrogate models Kriging seems to perform much better than RBF and GPML. For larger sample sizes GPML starts to become a viable alternative to Kriging. The increased sample size helps the machine learning computation for the hyperparameters for the Gaussian process. For larger sample sizes artificial neural networks can also be a viable choice besides GPML and Kriging. RBF had trouble to be a viable alternative to Kriging regardless of sample size. When increasing the sample size further the risk of overfitting [22] can become a problem. A trade-off between overfitting and underfitting needs to be made.

6.5.3 Sequential sampling and sample size

The error plot results shown in chapter 5.4.3 do not show any noticeable changes in preferred sequential sampling strategy for changing sample sizes. For 5n, 10n and higher the better sequential sampling techniques are LOLA-Voronoi (A), Generalized Probability of Feasibility (B) and the error LOLA-Voronoi mixtures (E). This could be due to these sampling strategies performing well with the relatively large design space per sample point caused by the parameter range used.

6.5.4 Sample size trade-off

When using the sample size as stopping criteria some trade-off should be made between computational costs and model accuracy. The effectiveness of the samples on the surrogate model depends on the size of design space created by the parameter range, initial and sequential sampling and which surrogate model is used. Ideally sufficiently few samples are used as the sample size is responsible a large part of the computational cost. A smaller sample size does result in a weaker model score. Besides adding more samples the model score can be improved by tightening the bounds or selecting different sampling strategy/surrogate models.

7 Conclusions and Recommendations

The conclusions and recommendations for this research are presented in this chapter. Initially the conclusions are shown. The conclusions first show results and conclusions for each parameterization and finally some general conclusions are presented. The recommendations first present recommendations on selecting options for successful surrogate model building. Afterwards recommendations on continuing the research presented are given.

Conclusions

The main research question is how applicable is surrogate model based optimization for different parameterization methods and what strategy is recommended regarding effectiveness, efficiency and robustness. The conclusion shows the results obtained per parameterization method researched and provides conclusions that can be drawn from them. The conclusions are structured per parameterization in the following order: CST10, CST14, BP3434, unweighted NURBS and NURBS. Afterwards a few general conclusions are given.

The result tables listed below refer to surrogate models built with the following settings:

- 5n, 10n, 20n and 30n refer to the amount of samples used with n being the dimensionality.
- CST10 and CST14 initial sampling are done with TPLHD
- uNURBS and NURBS initial sampling are done with LHD
- The second letter refers to the sequential sampling plan used where;
 - A = LOLA-Voronoi
 - B = Generalized Probability of Feasibility
 - C = Expected Improvement
 - D = Knowledge Gradient
 - E = Default (70-30 mixture of Lola-Voronoi and Error sampling)
- Finally RBF, Kriging and GPML refer to their respective surrogate model used.

CST10

The results for gradient descent optimization and surrogate models with best error scores respective to their sample are shown in table 20. The following conclusions can be drawn from the results.

- The surrogate model optimization with 5n and 10n sample points performed most comparable with the gradient descent search in terms of efficiency.
- The surrogate model based optimizations managed to find different designs spaces including optima with better objective functions.
- The CST method is robust for both optimization strategies.
- Surrogate models with larger sample sizes performed more consistent.
- CST10 optimizations converged to relatively high angles of attack.

Table 20 - CST10 Optimization results

Optimization CST10	Obj	CD	# Solver	AOA	Time [min]
Gradient descent	0.9354	0.010664	233	2.47°	45.62
5n_a_Krig	0.9418	0.010737	225	2.54°	47.00
5n_b_Krig	0.9302	0.010604	219	2.57°	40.19
5n_e_Krig	0.9293	0.010594	223	2.60°	39.31
10n_a_Krig	0.9413	0.010731	228	2.32°	40.83
10n_b_Krig	0.9382	0.010695	332	2.52°	86.43
10n_e_GPML	0.9378	0.010691	333	2.48°	80.59
20n_a_Krig	0.9342	0.010650	400	2.59°	62.36
20n_e_Krig	0.9377	0.010690	495	2.56°	84.08
30n_b_Krig	0.9305	0.010608	536	2.45°	94.88
30n_a_GPML	0.9368	0.010679	519	2.40°	135.73
30n_b_GPML	0.9417	0.010735	479	2.66°	153.19
30n_e_GPML	0.9388	0.010702	488	2.45°	85.56

CST14

The second parameterization researched is a CST method with 14 parameters. The results for the gradient and surrogate model based optimization are listed in the table below.

Table 21 - CST14 Optimization results

Optimization CST14	Obj	CD	# Solver	AoA	Time [min]
Gradient descent	0.9496	0.010635	204	2.47°	51,50
5n_a_Krig	0.9459	0.010594	336	2.20°	73.60
5n_b_Krig	0.9537	0.010681	334	2.42°	96.69
5n_e_Krig	0.9486	0.010624	266	2.22°	76.05
5n_b_GPML	0.9613	0.010766	221	2.40°	187.07
10n_a_Krig	0.9517	0.010659	361	2.35°	183.85
10n_b_Krig	0.9533	0.010677	332	2.15°	80.15
10n_e_Krig	0.9279	0.010393	415	2.22°	69.58
20n_a_Krig	0.9482	0.010620	525	2.07°	164.10
20n_b_GPML	0.9425	0.010556	579	2.32°	100.55
20n_e_RBF	0.9455	0.010590	473	2.32°	159.03

- Optimizations performed with CST14 converged to lower angles of attack than CST10, but higher than the NURBS methods.
- 10n_e_Krig obtained the design with the lower drag coefficient of the entire research.
- Especially the sequential sampling plans depicted with a, b and e in combination with Kriging proved successful for CST14.
- CST14 is very robust for both optimization strategies.
- The 4 added parameters over CST10 allowed for extra design space and extra control that showed necessary in obtaining better designs.

BP3434

BP3434 has been tested as an intuitive method. Due to unfortunate interaction between the initially chosen airfoil (RAE2822) and the randomness during the sampling process unfeasible airfoils are created that crash the solver. The Bezier-Parsec method has its merits for gradient descent optimization and could work in surrogate model based optimization, however a different type of base airfoil should be taken, or some of the base equations in the parameterization would have to be revisited.

uNURBS

The unweighted NURBS variant performed adequately as a parameterization technique with intermediate dimensionality compared to CST and NURBS. The final results are shown in the table 22 and are followed up by conclusions regarding the uNURBS method.

- uNURBS performed very consistent with the surrogate model based optimization.
- uNURBS is less robust than CST as it required specific tweaking on bounds to avoid the upper and lower curve intersecting near the trailing edge. uNURBS is more robust than NURBS as the opportunity for unfavorable stacking of parameters is excluded.
- uNURBS converged to lower angles of attack than CST.
- uNURBS has a good trade-off between effectiveness and efficiency regarding surrogate model building.

Table 22 - uNURBS optimization results

uNURBS Surrogate model	Obj	CD	# Solver calls	AoA	Total Time [min]
Gradient Descent	0.9404	0.010666	373	1.98°	75.70
5n_a_Krig	0.9376	0.010595	379	1.93°	76.73
5n_b_Krig	0.9361	0.010578	362	1.63°	110.61
5n_d_Krig	0.9336	0.010550	310	1.64°	71.05
5n_e_Krig	0.9426	0.010651	261	1.53°	76.73
10n_a_Krig	0.9693	0.010953	332	1.97°	370.23
10n_b_Krig	0.9354	0.010570	698	1.57°	136.53
10n_c_Krig	0.9295	0.010503	629	1.67°	370.22
20n_a_Krig	0.9327	0.010540	796	1.81°	260.58
20n_b_Krig	0.9327	0.010540	811	1.81°	240.45
20n_e_Krig	0.9325	0.010537	835	1.52°	285.52

NURBS

With 35 parameters the NURBS method reached the upper dimensionality limits of the NURBS toolbox. This caused the surrogate model building process to become inefficient as the toolbox is not optimized for these problems. Regardless of these issues the NURBS method managed to converge to the best drag coefficient scores. Table 23 shows the results for the NURBS method.

Table 23 - NURBS optimization results

NURBS Surrogate model	Obj	CD	# Solver calls	AOA	Total Time [min]
Gradient Descent	0.9307	0.010541	1118	1.32°	174,16
5n_a_Krig	0.9359	0.010576	707	1.37°	268.84
5n_b_Krig	1.0963	0.012388	327	2.01°	184.92
5n_d_Krig	0.9288	0.010495	857	1.63°	334.62
10n_a_Krig	0.9334	0.010547	1124	1.83°	481.78
10n_b_Krig	0.9277	0.010484	1499	1.91°	437.53
10n_d_Krig	0.9350	0.010565	828	1.90°	450.44
10n_e_Krig	0.9755	0.011024	720	1.78°	450.04
10n_b_RBF	0.9319	0.010531	1390	2.10°	479.14

- NURBS showed to be quite inconsistent in providing efficient and effective surrogate models.
- The added weight parameters increased the design space and the complexity of it. The added weights allowed for unfavorable stacking of parameters which led to inclusion of unfeasible designs space in the surrogate model.
- The unfavorable stacking of parameters caused robustness issues and NURBS required the most tweaking to work.
- The added weights do work well for gradient descent optimization. The weights allow pathways in the design space to other designs that would otherwise not be found.

General Conclusions

- Surrogate model based optimization converged to better designs than gradient descent optimization for all CST and NURBS.
- Sequential sampling proved most effective with LOLA-Voronoi, Generalized Probability of Feasibility or the Default setting.
- 20n and 30n surrogate models did not offer much better design spaces than 10n models did. The risk of overfitting and the increased cost suggest that surrogate models with less than 20n samples are preferred.
- Ordinary Kriging outperformed RBF and GPML consistently.
- CST converges to higher angles of attack than NURBS methods do.
- Surrogate model based optimization provided better optima for CST, NURBS
- Gradient descent optimization requires generally less computational costs than surrogate model based optimization.

Recommendations

Due to the nature of the research question the recommendation will be split into two parts. The first part addresses the recommendations for performed an airfoil optimization using surrogate model based optimization based on the results and conclusions. The second part contains recommendations regarding continuing this research.

Recommendations for SM based optimization

For a successful surrogate model based optimization a parameterization method, initial sampling plan, sequential sampling plan and surrogate model needs to be selected. The resulting strategy should be effective, efficient and robust. A recommendation is given for a NURBS parameterization and a CST parameterization.

NURBS

When using NURBS for SM optimization the use of weights for surrogate model building and genetic optimization of the surrogate model is not recommended. The added weights increase flexibility, but also increase the design space, increase the complexity and add unfeasible design spaces through unfavorable stacking of design parameters. The unweighed variant performed much in consistently finding interesting design spaces with the use of surrogate models.

The weights can be used after the design space is found that will be further exploitation of the interesting design space. The added flexibility of the weights increase the costs of computing gradient information, but allow for extra pathways to better optimums that are not found without weights.

A sample size of between 10n samples should suffice for reasonable accuracy from the surrogate model. More samples increased the error scores, but did not lead to better final optima. If computational resources are limited lower sample sizes can work, but can be less reliable.

For sequential sampling LOLA-Voronoi, Generalized Probability of Feasibility or a mixture of LOLA-Voronoi and error based sampling is recommended. These three sequential sampling strategies showed consistently effective throughout the variety of tested surrogate models.

For the surrogate model Kriging is recommended over GPML and RBF. Kriging is capable of understanding complex design spaces with limited resources.

CST

If the computational resources are available the CST variant with 14 parameters is recommended over the variant with 10 parameters for optimization. The extra flexibility and control provided by the 4 extra parameters proved to be worthwhile throughout this research. With 14 parameters the initial sampling plan can still be efficiently created with TPLHD.

Kriging performed the best overall for the CST methods accompanied by LOLA-Voronoi, generalized probability of feasibility or the mixture of LOLA-Voronoi with error based sequential sampling.

The CST methods performed well with sample ranges of 5n to 10n. Larger sample sizes can improve accuracy of the surrogate model but showed inefficient in this research.

Recommendation for continuing the research

To increase further understanding regarding surrogate model based optimization the following bullet points can be researched.

- Different surrogate model based optimization schemes. After the genetic algorithm optimization is performed a surrogate model can be created with a narrow parameter range surround that optimum. Due to the narrow design space selected different sampling plans might be preferred such as expected improvement. The surrogate model of the narrowed down design space can then be further exploited.
- The effect of increasing and decreasing the parameter range. How does the increased design space effect different surrogate models.
- Different parameterizations
- Different Kriging variants and different optimization strategies to determine the hyper parameters
- Different initial sample sizes and plans
- Different combination of sequential sampling plans. This research had LOLA-Voronoi mixed with error based sampling at a 70-30 ratio.
- Which settings can be recommended for the surrogate model representing the pitching moment constraint.
- Investigation of generalized probability of feasibility versus LOLA-Voronoi for larger parameter bound settings.

Bibliography

1. Shahrokhi, Ava, and Alireza Jahangirian. "A surrogate assisted evolutionary optimization method with application to the transonic airfoil design." *Engineering Optimization* 42.6 (2010): 497-515
2. Kampsolis, Ioannis C., et al. "Multilevel optimization strategies based on metamodel-assisted evolutionary algorithms, for computationally expensive problems." *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on. IEEE, 2007*
3. Emmerich, Michael, and Boris Naujoks. "Metamodel assisted multiobjective optimisation strategies and their application in airfoil design." *Adaptive computing in design and manufacture VI*. Springer London, 2004. 249-260.
4. Jouhaud, J-C., et al. "A surrogate-model based multidisciplinary shape optimization method with application to a 2D subsonic airfoil." *Computers & Fluids* 36.3 (2007): 520-529.
5. Sripawadkul, Vis, Mattia Padulo, and Marin Guenov. "A comparison of airfoil shape parameterization techniques for early design optimization." *13th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Fort Worth, Texas*. 2010
6. Keogh, Eamonn, and Abdullah Mueen. "Curse of dimensionality." *Encyclopedia of Machine Learning*. Springer US, 2011. 257-258.
7. Ceze, Marco, Marcelo Hayashi, and Ernani Volpe. "A study of the CST parameterization characteristics." *27th AIAA Applied Aerodynamics Conference*. 2009
8. Derksen, R. W., and Tim Rogalsky. "Bezier-PARSEC: An optimized aerofoil parameterization for design." *Advances in engineering software* 41.7 (2010): 923-930
9. Painchaud-Ouellet, Simon, et al. "Airfoil Shape Optimization Using a Nonuniform Rational B-Splines Parametrization Under Thickness Constraint." *AIAA journal* 44.10 (2006): 2170-2178.
10. Gorissen, Dirk, et al. "A surrogate modeling and adaptive sampling toolbox for computer based design." *Journal of Machine Learning Research* 11.Jul (2010): 2051-2055.
11. Viana, Felipe AC, Gerhard Venter, and Vladimir Balabanov. "An algorithm for fast optimal Latin hypercube design of experiments." *International journal for numerical methods in engineering* 82.2 (2010): 135-156.
12. Crombecq, Karel, et al. "A novel sequential design strategy for global surrogate modeling." *Simulation Conference (WSC), Proceedings of the 2009 Winter*. IEEE, 2009.
13. Martinez, Alexandre Souto, Rodrigo Silva González, and César Augusto Sangaletti Terçariol. "Generalized probability functions." *Advances in Mathematical Physics* 2009 (2010).
14. Bartz-Beielstein, Thomas. "Research in evolutionary computation." *Experimental Research in Evolutionary Computation: The New Experimentalism* (2006): 3-12.
15. Scott, Warren, Peter Frazier, and Warren Powell. "The correlated knowledge gradient for maximizing expensive continuous functions with noisy observations using gaussian process regression." *Operations Res. Financial Eng., Princeton, NJ* (2010).
16. Koziel, Slawomir, and Leifur Leifsson. "Surrogate-based modeling and optimization." *Applications in Engineering* (2013).
17. Forrester, Alexander IJ, and Andy J. Keane. "Recent advances in surrogate-based optimization." *Progress in Aerospace Sciences* 45.1 (2009): 50-79
18. Rasmussen, Carl Edward. "Gaussian processes for machine learning." (2006).
19. Ahmed, M. Y. M., and N. Qin. "Surrogate-based aerodynamic design optimization: use of surrogates in aerodynamic design optimization." *13th International Conference on Aerospace Science & Aviation Technology, Cairo, Egypt*. 2009.
20. Drela, Mark. "A User's Guide to MSES 3.05." *Massachusetts Institute of Technology (MIT), Cambridge* (2007).
21. The Mathworks, Inc.. "Global Optimization Toolbox: User's Guide (r2017b)." "Retrieved 2017 from https://nl.mathworks.com/help/pdf_doc/optim/optim_tb.pdf

22. Shin, Miyoung, Robert G. Sargent, and Amrit L. Goel. "Optimization and response surfaces: Gaussian radial basis functions for simulation metamodeling." *Proceedings of the 34th conference on Winter simulation: exploring new frontiers*. Winter Simulation Conference, 2002.

Appendix

There are 5 different Appendices.

- Appendix A contains the REA2822 parameter values for different parameterization and the lower and upper bounds used.
- Appendix B contains the MSES aerodynamic Solver settings
- Appendix C contains the SUMO configuration file
- Appendix D contains the collected raw data on the created surrogate models.

Appendix A – REA2822 parameter values and bounds

This appendix presents the REA2822 best fit parameter values, gradient descent and surrogate model lower and upper bounds and the parameter values for the final optimized airfoils.. This presentation is done for each parameterization method. The bounds are based on normalized REA2822 best fit values.

CST10

REA2822 best fit

CST10 Coefficient	1	2	3	4	5
Upper	0.1267	0.1427	0.1843	0.1996	0.1975
Lower	-0.1302	-0.1312	-0.2334	-0.0669	0.0324

Gradient Descent optimization lower & upper bounds.

CST10 Coefficient	1	2	3	4	5
Upper	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5
Lower	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5

Surrogate model optimization lower & upper bounds.

CST10 Coefficient	1	2	3	4	5
Upper	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3
Lower	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3

Gradient descent optima

CST10 Coefficient	1	2	3	4	5
Upper	0.1120	0.1213	0.1782	0.1590	0.1768
Lower	-0.0737	-0.1653	-0.2752	-0.0669	0.0382

5n_e_Krig optimia

CST 10Coefficient	1	2	3	4	5
Upper	0.0995	0.1166	0.1766	0.1242	0.1769
Lower	-0.0680	-0.1583	-0.3257	-0.0347	0.0470

CST14**REA2822 best fit**

CST14 Coefficient	1	2	3	4	5	6	7
Upper	0.1259	0.1389	0.1608	0.1635	0.2226	0.1693	0.2147
Lower	-0.1277	-0.1467	-0.1217	-0.2759	-0.0096	-0.1261	0.0781

Gradient Descent optimization lower & upper bounds.

CST14 Coefficient	1	2	3	4	5	6	7
Upper	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5
Lower	0.5-1.5	0.5-1.5	0.5-2.0	0.5-1.5	0.5-1.5	0.5-1.5	0.5-2.0

Surrogate model optimization lower & upper bounds.

CST14 Coefficient	1	2	3	4	5	6	7
Upper	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3
Lower	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3	0.7-1.3

Gradient descent optima

CST 14Coefficient	1	2	3	4	5	6	7
Upper	0.1099	0.1090	0.1514	0.1440	0.1808	0.1164	0.2169
Lower	-0.1029	-0.1423	-0.1414	-0.3442	-0.0098	-0.1188	0.0851

10n_e_Krig optima

CST 14Coefficient	1	2	3	4	5	6	7
Upper	0.0958	0.1246	0.1082	0.1827	0.1394	0.0893	0.2335
Lower	-0.0668	-0.1868	-0.2428	-0.2101	-0.0096	-0.1205	0.1489

uNURBS**REA2822 best fit**

Control points	1	2	3	4	5	6	7
Upper X	0	0.0005	0.1774	0.5079	0.7523	0.9165	1.0000
Upper Y	0	0.0220	0.0576	0.0679	0.0438	0.0171	0
Lower X	0	0.0018	0.2924	0.5169	0.7529	0.9272	1.0000

Lower Y	0	-0.0297	-0.0683	-0.0522	-0.0134	0.0045	0
----------------	---	---------	---------	---------	---------	--------	---

Gradient Descent optimization lower & upper bounds.

Control points	1	2	3	4	5	6	7
Upper X	-	0.5-10.0	0.5-1.5	0.5-1.5	0.5-1.22	0.5-1.04	-
Upper Y	-	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.95-1.05
Lower X	-	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.33	0.5-1.08	-
Lower Y	-	0.5-1.5	0.5-1.1	0.5-1.1	0.5-1.5	0.5-1.5	0.95-1.05

Surrogate model optimization lower & upper bounds.

Control points	1	2	3	4	5	6	7
Upper X	-	0.8-4.5	0.8-1.2	0.8-1.2	0.8-1.16	0.95-1.02	-
Upper Y	-	0.85-1.2	0.85-1.15	0.85-1.15	0.88-1.15	0.94-1.15	0.98-1.03
Lower X	-	0.5-5.5	0.75-1.5	0.85-1.2	0.85-1.25	0.85-1.03	-
Lower Y	-	0.85-1.2	0.85-1.1	0.85-1.1	0.85-1.1	0.85-1.2	0.98-1.03

Gradient descent optima

Control points	1	2	3	4	5	6	7
Upper X	0	0.0007	0.2136	0.4935	0.6644	0.9371	1.0000
Upper Y	0	0.0270	0.0561	0.0607	0.0410	0.0148	0.0006
Lower X	0	0.0019	0.2629	0.3887	0.6739	0.7904	1.0000
Lower Y	0	-0.0230	-0.0710	-0.0635	-0.0114	0.0040	0.0006

10n_c_Krig optima

Control points	1	2	3	4	5	6	7
Upper X	0	0.0008	0.1817	0.4878	0.6718	0.9506	1.0000
Upper Y	0	0.0248	0.0538	0.0618	0.0401	0.0160	0.0976
Lower X	0	0.0079	0.2880	0.4275	0.7871	0.8892	1.0000
Lower Y	0	-0.0190	-0.0703	-0.0632	-0.0070	0.0043	0.0976

NURBS

REA2822 best fit

Control points	1	2	3	4	5	6	7
Upper X	0	0.0005	0.1888	0.5253	0.7619	0.8988	1.0000
Upper Y	0	0.0228	0.0591	0.0674	0.0417	0.0202	0.0001
Upper weights	0.9999	0.9938	1.0026	1.0071	0.9954	1.0002	1.0010
Lower X	0	0.0017	0.2786	0.4983	0.7544	0.9272	1.0000
Lower Y	0	-0.0294	-0.0663	-0.0553	-0.0129	0.0044	0.0001
Lower Weights	1.0074	0.9885	0.9993	1.0071	0.9967	1.0011	0.9999

Gradient Descent optimization lower & upper bounds.

Control	1	2	3	4	5	6	7
----------------	----------	----------	----------	----------	----------	----------	----------

points							
Upper X	-	0.5-1.5	0.5-1.5	0.5-1.27	0.5-1.29	0.5-1.03	-
Upper Y	-	0.5-1.3	0.8-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.95-1.05
Upper weights	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.1-2.0
Lower X	-	0.1-2.0	0.1-2.0	0.12.0	0.1-1.33	0.1-1.08	-
Lower Y	-	0.1-2.0	0.5-1.05	0.5-1.05	0.5-1.5	0.5-1.1	0.95-1.05
Lower Weights	0.1-20	0.1-20	0.1-20	0.1-20	0.1-2.0	0.1-2.0	0.1-2.0

Surrogate model optimization lower & upper bounds.

Control points	1	2	3	4	5	6	7
Upper X	-	0.8-4.5	0.8-1.2	0.8-1.2	0.9-1.16	0.95-1.02	-
Upper Y	-	0.85-1.2	0.85-1.15	0.85-1.15	0.85-1.15	0.94-1.15	0.98-1.03
Upper weights	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5
Lower X	-	0.5-5.5	0.75-1.5	0.85-1.2	0.85-1.25	0.85-1.03	-
Lower Y	-	0.85-1.2	0.85-1.1	0.85-1.1	0.85-1.2	0.85-1.03	0.98-1.03
Lower Weights	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5	0.5-1.5

Gradient descent optima

Control points	1	2	3	4	5	6	7
Upper X	0	0.0006	0.1705	0.4530	0.6614	0.9121	1.0000
Upper Y	0	0.0279	0.0567	0.0614	0.0419	0.0152	-0.0049
Upper weights	0.8361	0.9732	0.8158	1.0316	0.9545	1.0094	1.0138
Lower X	0	0.0020	0.3049	0.4088	0.7382	0.8617	1.0000
Lower Y	0	-0.0247	-0.0677	-0.0606	-0.0128	0.0041	-0.0049
Lower Weights	1.0463	0.8165	1.2264	0.9961	0.9294	1.0139	1.0077

10n_b_Krig optima

Control points	1	2	3	4	5	6	7
Upper X	0	0.0019	0.2107	0.5080	0.6626	0.9151	1.0000
Upper Y	0	0.0236	0.0623	0.0555	0.0430	0.0157	0.0983
Upper weights	1.0832	1.1250	0.7994	1.3256	0.8333	0.6909	0.6947
Lower X	0	0.0074	0.3220	0.4707	0.7034	0.8920	1.0000
Lower Y	0	-0.0253	-0.0675	-0.0635	-0.0135	0.0042	0.0983
Lower Weights	0.8751	0.5566	1.3408	0.5519	0.7094	1.0651	1.23333

Appendix B - MSES Aerodynamic Solver

This appendix covers the input file for the MSES. There are three input files used as input. The MSES.XXX file contains the solver conditions and flow information. A different MSES.XXX is used for MSES setting 1 and 2 as discussed in section 5.2. The BLADE.XXX file describes the size of the box and the airfoil. GRIDPAR.XXX file contains the information on the grid settings used for the computation. For further information on the workings of MSES the reader is referred to the MSES guide [20].

MSES Setting 1

MSES.XXX

3	4	5	DCIRC	DALFA	DSBLE	
3	4	5	LE Kutta conditions		TE Kutta Conditions	ALFAIN
0.73	0.8	2.0	MACHIN		CLIFIN	ALFAIN
3	2		ISMOM	IFFBC		
2.7e6	9.0		REYNIN	ACRIT		
0.01	0.01		XTRSn	XTRPn		
0.99	-1		MCRIT	MUCON		

MSES Setting 2

MSES.XXX

3	4	5	DCIRC	DALFA	DSBLE	
3	4	6	LE Kutta conditions		TE Kutta Conditions	CLIFIN
0.73	0.6082	2.0	MACHIN		CLIFIN	ALFAIN
3	2		ISMOM	IFFBC		
2.7e6	9.0		REYNIN	ACRIT		
0.01	0.01		XTRSn	XTRPn		
0.99	-1		MCRIT	MUCON		

BLADE.XXX

'-3.0	4.0	-4.0	4.5	XINL	XOUT	YBOT	YTOP
X(1)	Y(1)						
X(2)	Y(2)						
X(.)	Y(.)						
.	.						
.	.						
.	.						
X(end)	Y(end)						

A small trailing edge gap is enforced that stops the solver from crashing. The origin of these crashes is unknown. But the small gap fixes the crashes.

GRIDPAR.XXX

141						Points
0.4						Exponent for point division
37						Vertical lines in front of airfoil
37						Vertical lines aft of airfoil
27						Streamlines above airfoil
18						Streamlines below airfoil
0						Streamlines between elements
1.3						Smoothing Parameter
2.5						Aspect ratio of each cell at stagnation point
0.85						X spacing parameter
AoA						Angle of attack
0.4	0.8	1.0				Spacing parameters element 1
0.03	0.07	1.0	1.0	1.0	0.0	Spacing Refinement

Appendix C - SUMO Toolbox Configuration files

This appendix contains a relevant xml input files for the SUMO toolbox [10]. Due to the large amount of difference xml files used only one example will be given. The example is based on NURBS 10n a Kriging. The toolbox requires a main configuration file and a simulator configuration file.

The main configuration file contains the information about the sampling techniques, surrogate model and stopping criteria used. Most of the configurations in the main configuration file have been left to the default settings. Plotting options are set to false to speed up the surrogate model building. Numbers of samples have been adjusted to the correct values for the research. NURBS is the only method that samples 30 new points at each sequential sampling stage due to the long time required to build a surrogate model in 35 dimensions. The simulator configuration file contains the input, output, parameters, parameter range and the high-fidelity solver used.

Main Configuration file

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<ToolboxConfiguration version="2016a">
  <Plan>
    <ContextConfig>default</ContextConfig>
    <SUMO>default</SUMO>
    <LevelPlot>default</LevelPlot>

    <Simulator>NURBSSM/NURBSSM.xml</Simulator>

    <Run name="" repeat="1">

      <InitialDesign>lhd</InitialDesign>

      <SequentialDesign>lola-voronoi</SequentialDesign>

      <DataSource>matlab</DataSource>

      <ModelBuilder>kriging</ModelBuilder>

      <Measure type="CrossValidation" target="0.01"
errorFcn="rootRelativeSquareError" use="on" />

      <Outputs>
        <Output name="Obj" constraintLB="0.8" constraintUB="1.8">
          </Output>
        <Output name="c2" constraintLB="-1000" constraintUB="0.05">
          </Output>
        </Outputs>
      </Run>
    </Plan>
    <ContextConfig id="default">
      <OutputDirectory>output</OutputDirectory>
      <PlotOptions>
        <Option key="saveBestModel" value="true"/>
        <Option key="saveIntermediateModels" value="false"/>
        <Option key="plotModels" value="false"/>
        <Option key="WindowStyle" value="normal"/>

        <Option key="withContour" value="true"/>
        <Option key="plotContour" value="false" />
      </PlotOptions>
    </ContextConfig>
  </ToolboxConfiguration>
```

```

        <Option key="plotUncertainty" value="false"/>
        <Option key="outputType" value="png"/>
    </PlotOptions>
    <Option key="keepOldModels" value="off"/>
    <Option key="parallelMode" value="off"/>
    <Option key="classificationMode" value="false"/>
    <Option key="numberOfClasses" value="0"/>
    <Option key="preserveOrder" value="true"/>
    <Profiling>
        <Profiler
name=".*SampleMinimum.*|.*Measure.*|.*BestModel.*|.*ElapsedTime.*|.*MemoryUse.*"
enabled="true">
            <Output type="toPanel"/>
            <Output type="toImage"/>
            <Output type="toTable"/>
            <Output type="toFile"/>
        </Profiler>
    </Profiling>

</ContextConfig>

<Logging>
    <RootLogger>
        <Option key="runsInMainLog" value="true"/>
        <Option key="Level" value="INFO"/>
        <Handlers>
            <ConsoleHandler>
                <Option key="Level" value="INFO"/>
            </ConsoleHandler>
            <FileHandler>
                <Option key="Level" value="ALL"/>
                <Option key="Pattern" value="Sumo.log"/>
            </FileHandler>
        </Handlers>
    </RootLogger>
    <Option key="iminds.sumo" value="FINEST"/>
    <Option key="Matlab" value="FINEST"/>
</Logging>
<LevelPlot id="default" type="LevelPlot" combineOutputs="true">
    <Option key="makeLevelPlots" value="off"/>
    <DataSource
type="iminds.sumo.datasources.datasets.ScatteredDatasetDataSource">
        </DataSource>
</LevelPlot>

<SUMO id="default" type="SUMO">
    <Option key="createMovie" value="no"/>
    <Option key="minimumTotalSamples" value="0"/>
    <Option key="maximumTotalSamples" value="350"/>
    <Option key="maximumTime" value="Inf"/>
    <Option key="maxModelingIterations" value="Inf"/>
    <Option key="stopOnError" value="true"/>
    <Option key="minimumInitialSamples" value="100%"/>
    <Option key="minimumSamples" value="1"/>
    <Option key="maximumSamples" value="30"/>
    <Option key="minimumAdaptiveSamples" value="0" />
    <Option key="newSamplesMustSatisfyConstraints" value="yes"/>

    <Option key="adaptiveModelingInitialDesignOnly" value="no"/>
</SUMO>

```

```

    <DataSource id="matlab" type="MatlabDirectDataSource"/>
    <DataSource id="matlabOld"
type="iminds.sumo.datasources.matlab.MatlabDataSource">
        <Option key="maxResubmissions" value="1"/>
        <Option key="sampleTimeout" value="-1"/>
    </DataSource>

    <!-- A highly adaptive sampling algorithm which performs a trade-off between
exploration (filling up the design space as equally as possible)
and exploitation (selecting data points in highly nonlinear regions). lola-
voronoi is the only sample selector which currently supports
multiple outputs, auto-sampled inputs and constraints. -->
    <SequentialDesign id="lola-voronoi" type="LOLAVoronoiSequentialDesign"
combineOutputs="false">
        <!-- Number of frequency values returned for each submitted sample. Only
used with auto-sampled inputs. -->
        <Option key="frequencies" value="11"/>
        <!-- Whether a directed search should be performed while performing
exploitation. Allows for more aggressive exploitation. -->
        <Option key="directedSearch" value="false" />
        <!-- Distance metric to use. Note: distance metrics of the individual
rankers will be overruled! -->
        <Distance type="EuclideanDistance" />

        <ExploitationSampleRanker type="LOLASampleRanker">
            <!-- Integer between 2 and 20 -->
            <Option key="neighbourhoodSize" value="2"/>
        </ExploitationSampleRanker>
        <ExplorationSampleRanker type="VoronoiSampleRanker" />
    </SequentialDesign>

    <!-- Build kriging models using the maximum likelihood to set the thetas -->
    <ModelBuilder id="kriging" type="ModelBuilder" combineOutputs="false">
        <Option key="nBestModels" value="1" />

        <!-- See the documentation for possible regression and correlation
functions -->
        <ModelFactory type="KrigingFactory">
            <Option key="regressionFunction" value="regpoly0"/>
            <Option key="multipleBasisFunctionsAllowed" value="false" />

            <!-- Use slice sampling to estimate the hyperparameters.
Slower but results in a better estimate of the posterior.
Very useful for optimization using expected improvement, etc. -->
            <!-- <Option key="numberOfSlices" value="100"/> -->

            <!--<Option key="hyperparameters0" value="0.5" />-->
            <BasisFunction>corrmatern32</BasisFunction>

            <!-- Alternative Basis Functions (correlation functions) -->
            <!--<BasisFunction>correxp</BasisFunction>-->
            <!--<BasisFunction>corrgaussp</BasisFunction>-->

            <!-- likelihood function to use (default: @marginalLikelihood): -->
            <!-- <Option key="hpLikelihood" value="@pseudoLikelihood" /> -->

            <!-- Enable the next set of options for noisy data (regression) -->

```

```

        <!--<Option key="lambda0" value="-5"/>--> <!-- initial value
(required) -->
        <!--<Option key="lambdaBounds" value="[-15; 0]"/>--> <!-- bounds
(optional) -->

        <Optimizer>fminconWithDerivatives</Optimizer>
    </ModelFactory>

</ModelBuilder>

<BasisFunction id="corrgauss" type="BasisFunction" name="corrgauss" >
    <Parameter name="theta" min="-2" max="2" scale="log" duplicate="true" />
</BasisFunction>

<BasisFunction id="corrmatern32" type="BasisFunction" name="corrmatern32" >
    <Parameter name="ell" min="-2" max="2" scale="log" duplicate="true" />
</BasisFunction>

<BasisFunction id="corrmatern52" type="BasisFunction" name="corrmatern52" >
    <Parameter name="ell" min="-2" max="2" scale="log" duplicate="true" />
</BasisFunction>

<BasisFunction id="correxp" type="BasisFunction" name="correxp" >
    <Parameter name="theta" min="-2" max="2" scale="log" duplicate="true" />
</BasisFunction>

<!-- Latin Hypercube DOE -->
<InitialDesign id="lhd" type="LatinHypercubeDesign">
    <!-- how many points to generate -->
    <Option key="points" value="105"/>
    <!--<Option key="weight" value="0.5"/>-->
    <!--<Option key="coolingFactor" value="0.9"/>-->
    <!--<Option key="p" value="5.0"/>-->
</InitialDesign>

<!-- Matlab fmincon (active-set) using derivative information (used for
models in SUMO-toolbox) -->
<Optimizer id="fminconWithDerivatives" type="MatlabOptimizer">
    <Option key="gradobj" value="on" />
    <Option key="derivativecheck" value="off" />
    <Option key="diagnostics" value="off" />
    <Option key="algorithm" value="active-set" />
</Optimizer>
</ToolboxConfiguration>

```

Simulator Configuration File

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<Simulator>
  <Name>NURBS MSES REA2822</Name>
  <Description>
    NURBS simulator configuration file
  </Description>

  <!-- The input parameters -->
  <InputParameters>
    <Parameter name="uP2X" type="real" value="1.0" minimum="0.8"
maximum="4.5"/>
    <Parameter name="uP3X" type="real" value="1.0" minimum="0.8"
maximum="1.2"/>
    <Parameter name="uP4X" type="real" value="1.0" minimum="0.8"
maximum="1.2"/>
    <Parameter name="uP5X" type="real" value="1.0" minimum="0.9"
maximum="1.16"/>
    <Parameter name="uP6X" type="real" value="1.0" minimum="0.95"
maximum="1.02"/>

    <Parameter name="uP2Y" type="real" value="1.0" minimum="0.85"
maximum="1.2"/>
    <Parameter name="uP3Y" type="real" value="1.0" minimum="0.85"
maximum="1.15"/>
    <Parameter name="uP4Y" type="real" value="1.0" minimum="0.85"
maximum="1.15"/>
    <Parameter name="uP5Y" type="real" value="1.0" minimum="0.85"
maximum="1.15"/>
    <Parameter name="uP6Y" type="real" value="1.0" minimum="0.94"
maximum="1.15"/>
    <Parameter name="TEY" type="real" value="1.0" minimum="0.98"
maximum="1.03"/>

    <Parameter name="uW1" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
    <Parameter name="uW2" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
    <Parameter name="uW3" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
    <Parameter name="uW4" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
    <Parameter name="uW5" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
    <Parameter name="uW6" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
    <Parameter name="uW7" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>

    <Parameter name="lP2X" type="real" value="1.0" minimum="0.5"
maximum="5.5"/>
    <Parameter name="lP3X" type="real" value="1.0" minimum="0.75"
maximum="1.5"/>
    <Parameter name="lP4X" type="real" value="1.0" minimum="0.85"
maximum="1.2"/>
    <Parameter name="lP5X" type="real" value="1.0" minimum="0.85"
maximum="1.25"/>
    <Parameter name="lP6X" type="real" value="1.0" minimum="0.85"
maximum="1.03"/>
  </InputParameters>
</Simulator>
```

```

        <Parameter name="lP2Y" type="real" value="1.0" minimum="0.85"
maximum="1.2"/>
        <Parameter name="lP3Y" type="real" value="1.0" minimum="0.85"
maximum="1.1"/>
        <Parameter name="lP4Y" type="real" value="1.0" minimum="0.85"
maximum="1.1"/>
        <Parameter name="lP5Y" type="real" value="1.0" minimum="0.85"
maximum="1.2"/>
        <Parameter name="lP6Y" type="real" value="1.0" minimum="0.85"
maximum="1.03"/>

        <Parameter name="lW1" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
        <Parameter name="lW2" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
        <Parameter name="lW3" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
        <Parameter name="lW4" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
        <Parameter name="lW5" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
        <Parameter name="lW6" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>
        <Parameter name="lW7" type="real" value="1.0" minimum="0.5"
maximum="1.5"/>

</InputParameters>

<!-- The output parameters -->
<OutputParameters>
    <Parameter name="Obj" type="real"/>
    <Parameter name="c2" type="real"/>
</OutputParameters>

<!-- A simulator may have multiple implementations: as an executable, a
    java main class, a dataset, ...-->
<Implementation>

    <Executables>
        <!-- An arch attribute may make sense here as well -->
        <Executable platform="matlab">NURBS_MSES_Solver</Executable>
    </Executables>

</Implementation>
</Simulator>

```

Appendix D – Surrogate model error validation data

Table 24 - CST10 Error validation data

Surrogate Model CST10	RMS_Obj	MAE_Obj	R ² _Obj	RMS_c2	MAE_c2	R ² _c2	SM building time	Evaluated samples	Failed samples
CST10_05_3_a_GPML	0.1563	0.5133	0.7289	0.0385	0.0941	0.9558	5.81	55	0
CST10_05_3_a_Krig	0.1053	0.5985	0.8744	0.0240	0.0650	0.9828	13.29	57	4
CST10_05_3_a_RBF	0.1718	0.5823	0.6707	0.0400	0.1148	0.9525	5.82	56	2
CST10_05_3_b_GPML	0.1633	0.6164	0.6987	0.0292	0.0718	0.9746	5.93	55	0
CST10_05_3_b_Krig	0.1169	0.3991	0.8451	0.0290	0.0721	0.9749	5.43	55	0
CST10_05_3_b_RBF	0.1508	0.6062	0.7426	0.0405	0.0954	0.9511	5.20	55	0
CST10_05_3_c_GPML	0.1893	0.9554	0.5941	0.0350	0.1107	0.9636	8.16	55	4
CST10_05_3_c_Krig	0.1370	0.7319	0.7880	0.0390	0.1034	0.9549	9.64	63	12
CST10_05_3_c_RBF	0.1791	0.7077	0.6410	0.0516	0.1489	0.9212	8.81	51	0
CST10_05_3_d_GPML	0.2064	1.0249	0.5189	0.0377	0.0998	0.9578	9.99	51	0
CST10_05_3_d_Krig	0.1527	0.6662	0.7419	0.0388	0.1050	0.9551	7.08	51	0
CST10_05_3_d_RBF	0.2747	0.8005	0.1528	0.0501	0.1225	0.9260	8.12	60	10
CST10_05_3_e_GPML	0.1703	0.4935	0.6784	0.0315	0.0736	0.9705	6.98	56	2
CST10_05_3_e_Krig	0.0909	0.4133	0.9065	0.0284	0.0839	0.9759	5.37	56	2
CST10_05_3_e_RBF	0.1555	0.5065	0.7294	0.0418	0.1020	0.9483	8.38	55	1
CST10_1_3_a_GPML	2.2e6	2.5e6	-0.0006	1.1176	1.5101	-0.0010	13.41	110	1
CST10_1_3_a_Krig	0.0600	0.2010	0.9592	0.0231	0.0557	0.9842	11.47	115	10
CST10_1_3_a_RBF	0.1132	0.5020	0.8549	0.0310	0.0845	0.9714	18.72	100	0
CST10_1_3_b_GPML	7.3e6	3.5e7	-12.6865	3.7880	18.1475	-13.3648	18.22	113	6
CST10_1_3_b_Krig	0.0712	0.2378	0.9426	0.0296	0.1207	0.9739	11.55	117	14
CST10_1_3_b_RBF	0.1297	0.3959	0.8096	0.0284	0.0673	0.9759	17.85	111	2
CST10_1_3_c_GPML	0.4394	1.8475	-0.9369	0.0501	0.2674	0.9254	16.21	111	11
CST10_1_3_c_Krig	0.1230	0.4634	0.8292	0.0271	0.0798	0.9781	11.78	103	2
CST10_1_3_c_RBF	0.1485	0.6078	0.7533	0.0362	0.0904	0.9611	11.03	103	2
CST10_1_3_d_GPML	2.4e6	1.1e7	-0.3036	1.2966	5.1994	-0.2166	18.52	117	16
CST10_1_3_d_Krig	0.1456	0.6931	0.7601	0.0291	0.0885	0.9748	11.62	100	0
CST10_1_3_d_RBF	1.3114	2.4375	0.0000	0.0390	0.0968	0.9547	22.13	119	18
CST10_1_3_e_GPML	0.0989	0.2626	0.8893	0.0256	0.0647	0.9805	11.57	112	6
CST10_1_3_e_Krig	0.0573	0.2195	0.9628	0.0236	0.0534	0.9834	12.04	113	6
CST10_1_3_e_RBF	0.1191	0.4009	0.8397	0.0326	0.1005	0.9684	13.06	112	5
CST10_2_3_a_GPML	0.0497	0.1437	0.9720	0.0311	0.0795	0.9712	22.89	212	4
CST10_2_3_a_Krig	0.0398	0.1029	0.9821	0.0189	0.0431	0.9893	32.38	212	5
CST10_2_3_a_RBF	0.0862	0.2570	0.9160	0.0227	0.0553	0.9847	35.23	213	6
CST10_2_3_b_GPML	0.0773	0.4128	0.9326	0.0274	0.1222	0.9776	23.94	212	4
CST10_2_3_b_Krig	1.2e6	1.2e6	0.0000	0.6048	0.9863	0.0001	24.63	219	18
CST10_2_3_b_RBF	0.1175	0.4586	0.8440	0.0273	0.0907	0.9778	38.62	213	6
CST10_2_3_c_GPML	0.1244	0.6547	0.8249	0.0321	0.0749	0.9693	42.70	216	16
CST10_2_3_c_Krig	1.6e7	6.3e7	-12.0144	8.0411	31.8554	-11.6098	32.65	200	0
CST10_2_3_c_RBF	0.1010	0.4657	0.8845	0.0281	0.0794	0.9764	29.49	208	8
CST10_2_3_d_GPML	1.2e6	1.4e6	-0.0013	0.8046	1.1985	-0.0008	42.31	242	42
CST10_2_3_d_Krig	0.0974	0.5565	0.8927	0.0235	0.0668	0.9836	37.18	212	12
CST10_2_3_d_RBF	0.4086	2.3133	-0.8393	0.0291	0.0851	0.9748	53.45	241	41
CST10_2_3_e_GPML	0.3226	1.9192	-0.1278	0.0265	0.0792	0.9791	22.01	214	10
CST10_2_3_e_Krig	0.0421	0.1513	0.9799	0.0182	0.0448	0.9901	22.73	216	12
CST10_2_3_e_RBF	0.0900	0.3803	0.9084	0.0281	0.0680	0.9765	22.37	209	2

Surrogate Model CST10 continued	RMS_Obj	MAE_Obj	R ² _Obj	RMS_c2	MAE_c2	R ² _c2	SM building time	Evaluated samples	Failed samples
CST10_3_3_a_GPML	0.0361	0.1028	0.9853	0.0159	0.0491	0.9925	63.55	314	8
CST10_3_3_a_Krig	0.0829	0.2903	0.9222	0.0227	0.0822	0.9847	51.61	313	6
CST10_3_3_a_RBF	0.0722	0.2413	0.9410	0.0289	0.0785	0.9752	38.30	317	14
CST10_3_3_b_GPML	0.0454	0.1598	0.9766	0.0211	0.0612	0.9868	54.37	320	20
CST10_3_3_b_Krig	0.0347	0.1049	0.9864	0.0152	0.0435	0.9932	51.77	332	24
CST10_3_3_b_RBF	0.0775	0.2684	0.9320	0.0233	0.0639	0.9839	56.08	318	16
CST10_3_3_c_GPML	0.2739	1.0012	0.1690	0.0401	0.1038	0.9521	50.15	343	43
CST10_3_3_c_Krig	0.0498	0.1696	0.9720	0.0244	0.0725	0.9823	111.78	307	7
CST10_3_3_c_RBF	0.0796	0.3138	0.9282	0.0270	0.0696	0.9783	34.87	310	9
CST10_3_3_d_GPML	0.2450	1.0814	0.3288	0.0433	0.1480	0.9442	73.01	324	24
CST10_3_3_d_Krig	0.0482	0.1679	0.9737	0.0231	0.0759	0.9841	63.28	311	10
CST10_3_3_d_RBF	1.3114	2.4375	0.0000	0.0329	0.1004	0.9677	76.26	339	39
CST10_3_3_e_GPML	0.0448	0.0891	0.9773	0.0174	0.0454	0.9910	52.41	316	13
CST10_3_3_e_Krig	0.0513	0.1658	0.9702	0.0217	0.1120	0.9860	54.41	314	9
CST10_3_3_e_RBF	0.0676	0.2173	0.9483	0.0248	0.0623	0.9817	45.36	312	7

Table 25 - CST14 Surrogate model data

Surrogate Model CST14	RMS_Obj	MAE_Obj	R ² _Obj	RMS_c2	MAE_c2	R ² _c2	SM building time	Evaluated samples	Failed samples
CST14_05_3_a_GPML	0.2820	1.2620	0.4899	0.0439	0.1857	0.9509	7.60	71	0
CST14_05_3_a_Krig	0.1822	0.8905	0.7879	0.0445	0.1980	0.9494	10.36	85	8
CST14_05_3_a_RBF	0.2766	1.2780	0.5152	0.0593	0.2656	0.9105	10.23	72	2
CST14_05_3_b_GPML	0.2355	0.7329	0.6445	0.0427	0.1313	0.9534	86.77	84	6
CST14_05_3_b_Krig	0.1717	0.9348	0.8111	0.0496	0.2757	0.9375	33.29	83	4
CST14_05_3_b_RBF	0.2601	1.2682	0.5775	0.0496	0.1715	0.9374	13.31	83	4
CST14_05_3_c_GPML	3.5e6	3.8e6	-0.0007	1.8466	2.2065	-0.0008	13.28	91	20
CST14_05_3_c_Krig	0.2956	1.6539	0.4540	0.0444	0.1413	0.9499	9.44	71	0
CST14_05_3_c_RBF	0.2841	1.2622	0.4929	0.0607	0.2879	0.9065	8.65	71	0
CST14_05_3_d_GPML	3.5e6	3.6e6	-0.0003	1.8282	2.2157	-0.0001	10.73	75	4
CST14_05_3_d_Krig	0.2836	1.6824	0.5020	0.0476	0.1363	0.9422	8.87	75	4
CST14_05_3_d_RBF	0.2756	1.3104	0.5320	0.0626	0.2025	0.9001	10.10	79	8
CST14_05_3_e_GPML	0.3000	1.4347	0.4241	0.1695	0.8604	0.2848	12.39	81	9
CST14_05_3_e_Krig	0.1981	1.0006	0.7480	0.0643	0.2390	0.8950	9.52	72	9
CST14_05_3_e_RBF	0.2706	1.4006	0.5301	0.0509	0.2094	0.9340	11.85	78	8
CST14_1_3_a_GPML	0.2042	0.7621	0.7332	0.0372	0.1891	0.9648	16.44	143	2
CST14_1_3_a_Krig	0.1418	0.8840	0.8709	0.0331	0.1315	0.9720	26.05	155	6
CST14_1_3_a_RBF	1.4060	3.2069	-0.0013	0.1982	0.9149	0.0000	19.44	155	7
CST14_1_3_b_GPML	0.2434	1.1062	0.6268	0.0360	0.1848	0.9670	15.71	155	6
CST14_1_3_b_Krig	0.1366	0.8599	0.8802	0.0368	0.1803	0.9654	16.66	155	7
CST14_1_3_b_RBF	1.4077	3.2069	0.0000	0.1982	0.9149	0.0000	19.86	142	0
CST14_1_3_c_GPML	0.1979	1.0452	0.7495	0.0426	0.2017	0.9536	32.12	180	40
CST14_1_3_c_Krig	0.2171	1.1531	0.6975	0.0462	0.1646	0.9459	21.10	144	4
CST14_1_3_c_RBF	0.2391	0.9816	0.6336	0.0513	0.2450	0.9330	22.03	143	2
CST14_1_3_d_GPML	0.2162	1.0672	0.6999	0.0424	0.1905	0.9542	32.17	166	26
CST14_1_3_d_Krig	0.2281	1.1975	0.6663	0.0445	0.1573	0.9498	25.75	140	0
CST14_1_3_d_RBF	1.4077	3.2069	0.0000	0.1982	0.9149	0.0000	17.57	144	4
CST14_1_3_e_GPML	0.2158	1.0745	0.7011	0.0348	0.1735	0.9691	30.10	158	18
CST14_1_3_e_Krig	0.1427	0.9312	0.8693	0.0378	0.1507	0.9635	31.64	159	15
CST14_1_3_e_RBF	1.4077	3.2069	0.0000	0.1982	0.9149	0.0000	38.04	162	21
CST14_2_3_a_GPML	0.2518	1.3106	0.5962	0.0478	0.1792	0.9418	67.92	287	6
CST14_2_3_a_Krig	0.1094	0.7294	0.9232	0.0432	0.2032	0.9524	64.60	307	26
CST14_2_3_a_RBF	0.1794	0.7775	0.7963	0.0331	0.1389	0.9720	52.04	301	15
CST14_2_3_b_GPML	0.1217	0.6509	0.9051	0.0318	0.1141	0.9741	38.68	299	11
CST14_2_3_b_Krig	2.7e7	1.7e8	-18.8615	12.8132	82.2951	-18.9506	43.86	301	14
CST14_2_3_b_RBF	0.1899	0.7911	0.7708	0.0313	0.1520	0.9751	39.81	287	6
CST14_2_3_c_GPML	0.2902	1.4074	0.4597	0.0490	0.1786	0.9390	49.89	299	18
CST14_2_3_c_Krig	0.2049	0.6457	0.7369	0.0579	0.1814	0.9151	72.68	287	6
CST14_2_3_c_RBF	0.2190	0.9733	0.6943	0.0427	0.1801	0.9536	34.36	284	4
CST14_2_3_d_GPML	0.1982	0.9772	0.7490	0.0400	0.2273	0.9592	41.58	301	20
CST14_2_3_d_Krig	5.4e5	5.4e5	0.0000	0.4303	0.7763	0.0000	39.38	284	4
CST14_2_3_d_RBF	1.4077	3.2069	0.0000	0.1982	0.9149	0.0000	43.24	291	10
CST14_2_3_e_GPML	0.2160	0.9664	0.7057	0.0318	0.1397	0.9741	43.55	318	38
CST14_2_3_e_Krig	1.2e7	6.7e7	-8.3390	13.1002	65.4148	-6.2725	56.49	322	38
CST14_2_3_e_RBF	0.1735	0.7247	0.8086	0.0333	0.1422	0.9718	47.48	325	39

Table 26 - uNURBS Surrogate model data

Surrogate Model CST14	RMS_Obj	MAE_Obj	R ² _Obj	RMS_c2	MAE_c2	R ² _c2	SM building time	Evaluated samples	Failed samples
uNURBS_05_3_a_GPML	0.1882	0.6303	0.3832	0.0831	0.1778	0.9060	13.50	115	6
uNURBS_05_3_a_Krig	0.0935	0.2834	0.8473	0.0533	0.1270	0.9612	12.02	113	2
uNURBS_05_3_a_RBF	0.1469	0.4053	0.6240	0.0743	0.1834	0.9253	13.03	114	4
uNURBS_05_3_b_GPML	0.1735	0.5717	0.4790	0.0728	0.1681	0.9292	10.23	112	0
uNURBS_05_3_b_Krig	0.0886	0.2701	0.8624	0.0772	0.2308	0.9194	11.75	113	2
uNURBS_05_3_b_RBF	0.2614	0.7333	-0.0350	0.0975	0.2821	0.8727	12.03	112	0
uNURBS_05_3_c_GPML	0.2218	0.7273	0.1441	0.0545	0.1543	0.9596	12.64	106	0
uNURBS_05_3_c_Krig	0.1578	0.6558	0.5636	0.0598	0.1901	0.9514	14.58	107	2
uNURBS_05_3_c_RBF	0.1928	0.7723	0.3483	0.0773	0.2047	0.9187	12.19	110	4
uNURBS_05_3_d_GPML	0.2044	0.7400	0.2689	0.0789	0.2155	0.9175	14.38	106	0
uNURBS_05_3_d_Krig	0.1631	0.6190	0.5337	0.0619	0.1661	0.9477	16.00	110	4
uNURBS_05_3_d_RBF	1.3117	2.0737	0.0000	0.3157	0.8079	0.0000	13.60	107	2
uNURBS_05_3_e_GPML	0.1619	0.5540	0.5445	0.0641	0.1899	0.9446	15.10	129	22
uNURBS_05_3_e_Krig	0.1154	0.4643	0.7699	0.0578	0.1734	0.9544	13.50	124	16
uNURBS_05_3_e_RBF	0.1872	0.5920	0.3867	0.0656	0.1611	0.9417	13.23	120	10
uNURBS_1_3_a_GPML	0.1338	0.4408	0.6871	0.0383	0.1042	0.9800	36.03	215	4
uNURBS_1_3_a_Krig	0.0993	0.3640	0.8271	0.0379	0.0985	0.9804	32.58	217	8
uNURBS_1_3_a_RBF	0.1647	0.6222	0.5242	0.0452	0.1514	0.9721	33.86	227	8
uNURBS_1_3_b_GPML	0.1334	0.6177	0.6896	0.0370	0.1116	0.9813	25.73	231	16
uNURBS_1_3_b_Krig	0.0770	0.2456	0.8960	0.0398	0.1248	0.9784	26.81	215	4
uNURBS_1_3_b_RBF	0.1602	0.5550	0.5503	0.0490	0.1624	0.9672	32.41	216	6
uNURBS_1_3_c_GPML	0.2417	1.0295	0.0735	0.0446	0.1269	0.9729	49.87	212	2
uNURBS_1_3_c_Krig	0.1330	0.5375	0.6930	0.0439	0.1379	0.9737	77.53	211	0
uNURBS_1_3_c_RBF	0.1348	0.4910	0.6811	0.0499	0.1560	0.9661	46.40	215	4
uNURBS_1_3_d_GPML	0.1757	0.5162	0.4843	0.0488	0.1219	0.9675	30.77	219	8
uNURBS_1_3_d_Krig	0.1672	1.0383	0.5241	0.0449	0.1263	0.9725	54.66	216	6
uNURBS_1_3_d_RBF	1.3117	2.0737	0.0000	0.3157	0.8079	0.0000	28.69	220	10
uNURBS_1_3_e_GPML	0.1404	0.4062	0.6628	0.0484	0.1271	0.9680	37.64	247	32
uNURBS_1_3_e_Krig	0.0767	0.2454	0.8968	0.0387	0.1115	0.9796	43.58	233	21
uNURBS_1_3_e_RBF	0.1562	0.4800	0.5858	0.0629	0.1727	0.9466	29.53	242	27
uNURBS_2_3_a_GPML	0.1223	0.3933	0.7382	0.0532	0.1788	0.9616	71.45	430	8
uNURBS_2_3_a_Krig	0.0565	0.2084	0.9440	0.0351	0.0895	0.9832	126.95	430	8
uNURBS_2_3_a_RBF	0.1446	0.4824	0.6420	0.0511	0.1214	0.9646	70.66	429	6
uNURBS_2_3_b_GPML	0.1527	0.4703	0.5986	0.0401	0.1189	0.9781	48.55	446	20
uNURBS_2_3_b_Krig	0.0629	0.2216	0.9306	0.0464	0.1131	0.9707	109.02	445	18
uNURBS_2_3_b_RBF	CRASHED								
uNURBS_2_3_c_GPML	0.1968	0.7711	0.4009	0.0428	0.1200	0.9751	77.66	432	12
uNURBS_2_3_c_Krig	0.1502	0.8899	0.6075	0.0409	0.1134	0.9772	349.44	435	15
uNURBS_2_3_c_RBF	1.3117	2.0737	0.0000	0.3157	0.8079	0.0000	62.50	427	6
uNURBS_2_3_d_GPML	0.2264	1.2194	0.1973	0.0598	0.1701	0.9519	110.71	428	7
uNURBS_2_3_d_Krig	0.3384	1.6478	-0.6616	0.0413	0.1203	0.9768	269.25	439	18
uNURBS_2_3_d_RBF	0.4966	1.8345	-1.7048	0.0756	0.2063	0.9230	99.90	470	49
uNURBS_2_3_e_GPML	0.1035	0.3553	0.8124	0.0371	0.0950	0.9813	118.57	467	43
uNURBS_2_3_e_Krig	0.0578	0.1788	0.9413	0.0365	0.0987	0.9818	144.64	469	49
uNURBS_2_3_e_RBF	0.1255	0.4549	0.7289	0.0435	0.1128	0.9742	85.04	459	39

Table 27 - NURBS Surrogate model data

Surrogate Model NURBS	RMS_Obj	MAE_Obj	R ² _Obj	RMS_c2	MAE_c2	R ² _c2	SM building time	Evaluated samples	Failed samples
NURBS_5_3_a_GPML	0.4441	4.0556	0.0624	0.1072	0.6741	0.8911	50.15	212	18
NURBS_5_3_a_Krig	0.3368	3.8850	0.4396	0.1107	0.9011	0.8838	42.80	207	8
NURBS_5_3_a_RBF	0.4226	4.1464	0.1176	0.1200	0.7878	0.8635	37.43	208	11
NURBS_5_3_b_GPML	0.4154	4.0807	0.1620	0.1082	0.7124	0.8890	23.15	209	12
NURBS_5_3_b_Krig	0.3515	3.9294	0.3924	0.1167	0.9213	0.8710	25.27	207	9
NURBS_5_3_b_RBF	0.4225	3.6463	0.1175	0.1267	0.7783	0.8479	21.46	310	14
NURBS_5_3_c_GPML	0.4483	4.1485	0.0054	0.0897	0.5899	0.9237	40.99	189	13
NURBS_5_3_c_Krig	0.4847	3.7797	-0.1224	0.1121	0.6553	0.8808	53.10	189	13
NURBS_5_3_c_RBF	0.4127	4.1430	0.1572	0.1253	0.8124	0.8514	27.79	181	5
NURBS_5_3_d_GPML	2.83E+06	3.78E+06	-0.0091	0.5735	1.5949	0.0036	32.48	194	19
NURBS_5_3_d_Krig	6.1468	34.6426	-7.3032	0.0970	0.5879	0.9108	72.61	213	38
NURBS_5_3_d_RBF	1.4387	5.5274	0.0000	0.3692	2.0377	0.0000	41.98	221	46
NURBS_5_3_e_GPML	0.4297	4.2039	0.0949	0.1023	0.6952	0.9007	49.47	207	26
NURBS_5_3_e_Krig	0.3481	3.7915	0.4003	0.1093	0.8427	0.8869	35.67	204	20
NURBS_5_3_e_RBF	0.4101	3.9806	0.1717	0.1182	0.7631	0.8677	34.51	205	22
NURBS_1_3_a_GPML	0.4008	4.0135	0.2173	0.0983	0.7207	0.9083	214.63	408	36
NURBS_1_3_a_Krig	0.3233	3.7919	0.4827	0.0976	0.7501	0.9097	142.04	388	27
NURBS_1_3_a_RBF	0.3904	3.7927	0.2489	0.1091	0.7042	0.8871	134.59	411	43
NURBS_1_3_b_GPML	6.34E+03	3.29E+04	-2.8638	0.4047	1.6205	0.0000	151.47	410	40
NURBS_1_3_b_Krig	0.3385	3.9618	0.4352	0.1031	0.8314	0.8994	75.02	392	38
NURBS_1_3_b_RBF	0.3544	3.1365	0.3799	0.1062	0.6962	0.8930	180.28	410	40
NURBS_1_3_c_GPML	1.3570	6.7681	-1.2061	0.0925	0.5832	0.9190	76.11	376	21
NURBS_1_3_c_Krig	2.92E+06	2.81E+07	28.7123	4.2666	44.7140	21.9455	312.00	370	20
NURBS_1_3_c_RBF	1.4387	5.5274	0.0000	0.3692	2.0377	0.0000	69.21	425	75
NURBS_1_3_d_GPML	0.4893	3.9153	-0.0365	0.1100	0.5924	0.8854	79.13	398	48
NURBS_1_3_d_Krig	0.3251	3.7004	0.4771	0.1012	0.8086	0.9032	299.39	380	30
NURBS_1_3_d_RBF	1.4387	5.5274	0.0000	0.3692	2.0377	0.0000	78.64	374	24
NURBS_1_3_e_GPML	0.4341	4.1979	0.0830	0.0937	0.5814	0.9167	133.40	404	43
NURBS_1_3_e_Krig	0.3251	3.7004	0.4771	0.1012	0.8086	0.9032	144.69	313	52
NURBS_1_3_e_RBF	0.4000	3.7455	0.2083	0.1068	0.6974	0.8919	112.92	377	27

Appendix E – Exploited surrogate models result

Table 28 - Exploited surrogate models results

Model	SM descent optim	SM GA optim	SM GA + high fidelity SQP Optim	High fidelity calls to build SM	High fidelity calls SQP descent	Time SM building	Time GA optim	Time SQP Descent	AoA final
CST10 5n a Krig	0.9792	0.9802	0.9418	57	168	13.3	12.76	20.95	2.54°
CST10 5n b Krig	0.9617	0.9615	0.9302	55	164	5.43	10.84	23.92	2.57°
CST10 5n e Krig	0.9640	0.9605	0.9293	56	167	5.37	13.54	20.40	2.60°
CST10 10n a Krig	0.9730	0.9727	0.9413	115	113	11.5	13.62	15.71	2.31°
CST10 10n b Krig	1.0074	1.0070	0.9382	117	215	11.5	31.76(-)	43.17	2.52°
CST10 10n e GPML	0.9658	0.9653	0.9378	112	221	11.6	39.82	29.18	2.48°
CST10 10n e Krig	0.9731	0.9726	0.9341	113	166	12.0	11.21	22.04	2.55°
CST10 10n c RBF	0.9751	0.9756	0.9423	103	143	11.0	21.36	17.88	2.20°
CST10 20n a Krig	0.9697	0.9546	0.9342	212	188	32.4	6.629	23.33	2.59°
CST10 20n e Krig	0.9687	0.9689	0.9377	216	279	22.7	21.39	39.98	2.56°
CST10 20n a GPML	0.9874	0.9867	NaN	-	-	-	27.39	-	-
CST10 30n b Krig	0.9778	0.9768	0.9305	332	204	51.8	15.47	27.61	2.45°
CST10 30n a GPML	0.9729	0.9726	0.9368	314	205	63.6	30.68	41.44	2.40°
CST10 30n b GPML	0.9949	0.9908	0.9417	320	159	54.4	78.48(-)	20.31	2.66°
CST10 30n e GPML	0.9756	0.9749	0.9388	316	172	52.4	24.32	18.84	2.45°
CST10 30n d Krig	0.9596	0.9615	0.9392	311	118	63.3	16.64	17.33	2.76°
CST10 30n e Krig	0.9645	0.9654	0.9329	314	159	54.4	19.40	23.38	2.38°
CST14 5n b Krig	0.9845	0.9846	0.9537	83	251	33.3	23.93	39.46	2.42°
CST14 5n a Krig	1.0427	1.0442	0.9459	85	251	10.4	23.74	39.46	2.20°
CST14 5n b GPML	1.0347	1.0158	0.9613	84	137	86.8	86.70	13.58	2.40°
CST14 5n e Krig	0.9680	0.9679	0.9486	72	194	9.52	42.52	24.00	2.22°
CST14 10n b Krig	0.9765	0.9936	0.9533	155	177	16.7	47.27	31.20	2.15°
CST14 10n a Krig	0.9765	0.9683	0.9517	155	206	26.1	138.09	19.66	2.35°
CST14 10n b GPML	0.9589	0.9616	0.9667	155	105	15.7	53.41	23.87	2.32°
CST14 10n e Krig	0.9731	1.0300	0.9279	113	302	12.0	19.99	37.59	2.22°
CST14 20n a Krig	0.9795	0.9882	0.9482	307	218	64.6	54.25	45.25	2.07°
CST14 20n b GPML	0.9729	0.9714	0.9425	299	280	38.7	36.30	25.55	2.32°
CST14 20n a GPML	0.9697	0.9687	0.9544	287	186	67.9	34.29	41.47	2.19°
CST14 20n e RBF	0.9815	0.9858	0.9455	209	264	22.4	103.35	33.28	2.32°
uNURBS5n d Krig	0.9762	0.9641	0.9336	110	200	16.0	33.11	21.94	1.64°
uNURBS 5n b Krig	0.9873	0.9951	0.9361	113	249	11.8	65.60	33.20	1.63°
uNURBS 5n a Krig	0.9968	0.9872	0.9376	113	266	12.0	24.14	40.59	1.93°
uNURBS 5n e Krig	1.0284	1.0193	0.9426	113	148	13.5	66.45	17.52	1.53°
uNURBS 10n a Krig	1.0074	0.9983	0.9693	217	115	32.6	46.60	13.00	1.97°
uNURBS 10n c Krig	0.9833	0.9743	0.9295	211	418	77.5	222.31	70.42	1.67°
uNURBS 10n b Krig	0.9815	0.9698	0.9354	215	483	26.8	25.81	83.92	1.57°
uNURBS 10n e Krig	0.9999	0.9935	NaN	233	NAN	43.6	68.07	NAN	NAN
uNURBS 20n e Krig	0.9621	0.9535	0.9325	469	366	145	88.56	51.96	1.52°
uNURBS 20n b Krig	0.9756	0.9690	0.9327	445	366	109	69.50	61.95	1.81°
uNURBS 20n a Krig	0.9693	0.9592	0.9327	430	366	127	71.72	61.86	1.81°
NURBS 5n a Krig	1.0330	1.0309	0.9359	207	500	42.8	160.89	65.15	1.37°
NURBS 5n b Krig	1.1061	1.1608	1.0963	207	121	25.3	145.36	14.26	2.01°
NURBS 5n d Krig	1.5709	1.3394	0.9288	213	644	72.6	130.16	131.86	1.63°
NURBS 5n e Krig	0.9844	0.9827	0.9280	204	614	37.7	120.4259	88.2270	1.76°
NURBS 10n a Krig	0.9889	0.9921	0.9334	388	736	142	221.89	117.89	1.83°
NURBS 10n b Krig	0.9713	0.9758	0.9277	392	1107	75.0	214.94	147.59	1.90°

Model Continued	SM descent optim	SM GA optim	SM GA + high fidelity SQP Optim	High fidelity calls to build SM	High fidelity calls SQP descent	Time SM building	Time GA optim	Time SQP Descent	AoA final
NURBS 10n d Krig	1.0036	1.0339	0.9350	380	448	299	98.765	52.67	1.90°
NURBS 10n e Krig	1.0081	1.0965	0.9755	361	159	159	267.60	23.44	1.78°
NURBS 10n b RBF	1.0806	1.0806	0.9319	410	980	180	161.30	137.83	2.10°