Structural optimization in engineering design with a focus on process automation.

Fuerle, Fabian

Fuerle, Fabian (2010) *Structural optimization in engineering design with a focus on process automation..* thesis, Swansea University.
http://cronfa.swan.ac.uk/Record/cronfa42567

Structural Optimization in Engineering Design with a Focus on Process Automation

Swansea University
Prifysgol Abertawe

Dipl. Ing. Fabian Fuerle

15th Oct 2010

Thesis submitted to Swansea University
in candidature for the degree of
Doctor of Philosophy

School of Engineering, Swansea University, Swansea

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

Summary

The present work is concerned with the advancement of the knowledge of structural optimization in engineering design while focusing on efficient and easy to use ways of setting-up the required automated processes as well as the problems arising from it. Three industry examples are considered.

In the first example a software tool that serves as a hands-on decision guidance for many occurring design situations for structured wall PE pipes is developed. In order to avoid licensing fees only public domain software or in-house code are used. It offers the efficient and automated simulation of the ringstiffness test as well as the most common pipe installation scenarios. In addition, an optimization feature is implemented for the design of optimum pipe profiles with regards to the ringstiffness test.

In the second example a framework for the optimum design of carbon fibre mountain bike frames is developed. An extensively parameterized and automated simulation model is created that allows for varying tube shapes, paths and laminate ply thicknesses as well as joint locations. For improved efficiency a decomposition approach has been employed that decomposes the original optimization problem into a size optimization sub problem and a shape optimization top level problem. The former is solved by the built-in optimization tool in OptiStruct and the latter by means of surrogate based optimization where each experiment in the DoE is a full size optimization.

The third example is concerned with the optimum design of a blade for a novel vertical axis wind turbine. A design approach similar to those with horizontal axes is chosen. The altered design requirements are accounted for by creating a parameterized simulation model and performing size optimization runs for 32 models with different material settings and shear web locations where the model creation process has been automated.

Declarations

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

15/10/2010

Date *Candidate*

STATEMENT 1

This thesis is the result of my own investigations, except where otherwise stated. Where correction services have been used, the extent and nature of the correction is clearly marked in a footnote(s).

Other sources are acknowledged by footnotes giving explicit references. A bibliography is appended.

15/10/2010

Date *Candidate*

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loans after expiry of a bar on access approved by the Swansea University.

15/10/2010

Date *Candidate*

Contents

Acknowledgements

I would like to express my sincere thanks to Professor Johann Sienz who gave me the opportunity to work as his PhD student, who was always there to help and encourage me. I would specifically like to mention the numerous conferences I was able to attend, the excellent technical equipment I was provided with and the various interesting projects I was able to work on. I would also like to thank Dr. John Pittman for his encouragement and for always being willing to help me.

I would like to thank Dr. Dominik Zimmermann and Professor Christian Miehe for introducing me to the field of Computational Mechanics and to help me build the foundation of the knowledge that I benefitted from every day during the three years.

I would like to thank my parents whose support was an immense help during this PhD and particularly during the studies that made this PhD possible.

I would like to thank Asset Intl. Ltd. and KWH Pipe Ltd. for funding this PhD project. In particular I would like to thank Dr. Vasilios Samaras for the fruitful discussions we had and for all the support with the software. I also would like to thank Dr. Cris Arnold for the help with the material tests.

I would like to thank Chocolate Bikes Ltd. for the interesting project I was able to work on. I would like to thank Dr. Marcus Heuberger, Dr. James Holness and Dr. Bruce Philip for their constant technical support.

I would like to thank C-FEC Ltd. for giving me the chance to work on a very interesting project.

Finally, many thanks to Radek Panczak for his help on the Kriging models, to Helen Packer and Mohammed Taheri for being a constant source of inspiration, to Maartje Joosten for listening whenever I was stuck and to Mauro Innocente for the help during the consultancy work.

Nomenclature

Abbreviations

ACO	Ant Colony Optimization
ANN	Artificial Neural Networks
APME	Association of Plastics Manufacturers in Europe
CAD	Computer Aided Design
CAE	Computer-Aided Engineering
CFD	Computational Fluid Dynamics
CRP	Carbon-fibre Reinforced Polymer
DACE	Design and Analysis of Computer Experiments
DoE	Design of Experiments
EA	Evolutionary Algorithm
EP	Evolutionary Programming
ES	Evolution Strategies
EV	Eigenvalue
FE	Finite Element
FI	Failure Index
FRP	Fibre Reinforced Polymers
GA	Genetic Algorithm
GL	Germanischer Lloyd
GP	Genetic Programming
GRP	Glass–fibre Reinforced Polymer
GUI	Graphical User Interface
HDPE	High Density Polyethylene
IMSE	Integrated Mean Square Error
LHS	Latin Hypercube Sampling
MARS	Multipoint Approximation method based on Response Surface fitting
MFDA	Modified Feasible Directions Algorithm
NC	Non-collapsingness Criterion
NURBS	Non-uniform Rational B-splines
OAS	Orthogonal Array Sampling
OLH	Optimum Latin Hypercube
PA	Polynomial Approximations

PE	Polyethylene
PSO	Particle Swarm Optimization
RBF	Radial Basis Functions
RSM	Response Surface Methodology
SA	Simulated Annealing
SCVR	Standardized Cross-Validation Residual
SQP	Sequential Quadratic Programming
TEPPFA	The European Plastics Pipe and Fitting Association
UD	Uni-directional

Roman Symbols

Scalars

a_i, a_{ij}	unknown coefficients
A_{web}	web area
b_i	control value *i*
b_i^{str}	control value *i* in B-spline describing a straight line
d_i	internal pipe diameter
d_n	nominal pipe diameter
d_o	outer pipe diameter
E	E-modulus
E'	storage modulus
E''	loss modulus
E_{FE}	secant modulus
$E_{Tension}$	E-modulus obtained in tension test
EI	flexural rigidity
f	frequency
$f(\underline{x})$	objective function for design variable set $\underline{x}$
F	force
F_{crit}	failure force for twist buckling
f_{DMTA}	DMTA frequency
F_i	observed response *i*
$\overline{F}_i(\underline{x})$	approximation of observed response *i*
$\overline{F}_{lin}(\underline{x})$	linear approximation function
$\overline{F}_{quad}(\underline{x})$	quadratic approximation function

$f_{straight}$	function indicating the deviation from the straight line
f_{smooth}	function indicating the smoothness of the paths
fac_{visco}	factor taking into account the visco-elastic damping
fac_{level}	factor taking into account the strain level
FI_{Hill}	Hill failure index
FI_{Thres}	maximum allowable failure index
G	bulk modulus
$g_k(\underline{x})$	inequality constraint for design variable set $\underline{x}$
h	height or interpolation parameter
h_i	radially symmetric function
h_i	B-spline knot *i*
$h_k(\underline{x})$	equality constraint for design variable set $\underline{x}$
I_y, I_z	Moment of inertia around the *y*-axis and *z*-axis
l	length
L	pipe sample length
L_i^{obstr}	horizontal intrusion length of node *i*
L_{pq}	distance between points *p* and *q*
L_{pq}^*	amended distance between points *p* and *q*
lps	lateral pipe stiffness
M	mass
$M_{i,p}(h)$	B-spline basis function
M_{Thres}	maximum allowable mass
n_c	number of constraints
n_{ctrl}	number of control values
n_{cycle}	number of cycles
n_{Dim}	number of dimensions
n_{Div}^{max}	maximum number of space divisions during initialization
n_{DV}	number of design variables
n_{DV}^{size}	number of size design variables
n_{DV}^{shape}	number of shape design variables
n_{Eval}	number of B-spline evaluations
n_{gene}	number of bits in a gene of a chromosome
n_{inf}	number of infeasible points
n_{keep}	number of chromosomes in selection for mating

n_P	number of points
n_{spring}	number of springs
OI	obstruction index
OSR	Out-of-saddle-rigidity
p_h	unknown parameter in Kriging interpolation
p_n	gene *n*
P_n	ranking weight
pen	constant penalization term
r	penalization coefficient
R	radius or stress ratio
$r(h)$	B-spline interpolation function
$\hat{r}_{end}$	B-spline end point
$\hat{r}_{start}$	B-spline start point
R_{Thres}	minimum sum of rigidities
S	allowable in-plane shear stress
$S^2(\underline{x}^*)$	mean squared error
S_{spring}	spring stiffness
SF	general safety factor
$SF_{Fatigue}$	fatigue safety factor
SF_{LE}	load safety factor for extreme conditions
SF_{LS}	load safety factor for service conditions
SF_{Mextr}	global material safety factor for extreme loading
SF_{Mserv}	global material safety factor for service conditions
SF_{MA}	ageing material safety factor
SF_{MC}	curing material safety factor
SF_{MG}	general material safety factor
SF_{MM}	manufacturing material safety factor
SF_{MT}	high temperature material safety factor
SGI	ratio between sum of *OSR* and *THR* and frame's weight
SN	ringstiffness
SN_{des}	desired ringstiffness
SN_{exp}, SN_{FE}	ringstiffness from FE simulation and from experiment
t	time
T	temperature
t_1, t_2, t_3	pipe profile wall thicknesses (top, web, bottom)

T_{avg},	average between ambient and pipe temperature
T_{amb}, T_{pipe}	ambient and pipe temperature
t_{test}	duration of the ringstiffness test
THR	Track-holding-rigidity
U, U^*	Audze-Eglais and amended Audze-Eglais potential energy
v	velocity
v_i	slack variable
w	width
w_i	weight *i*
X	allowable stress in ply material direction *1*
x_k^l, x_k^u	lower and upper bound of design variable x_k
y	vertical pipe deflection
Y	allowable stress in ply material direction *2*
$y_{-1}\left(\underline{x}^{(i)}\right)$	prediction of $\underline{x}^{(i)}$ where approximation model is built without $\underline{x}^{(i)}$

Vectors

$\underline{\mathcal{H}}$	B-spline knot vector
$\underline{M}$	midpoint
$\underline{n}_{Div}$	vector of space divisions
$\underline{n}_{Div}^{best}$	currently best vector of space divisions in initialization
$r_i\left(\underline{x}^*\right)$	vector of correlations for location $\underline{x}^*$
$\underline{S}^q$	search direction in iteration *q*
$\underline{x}$	design variable set
$\underline{x}_C$	centre of radial basis function
$\underline{x}_g$	global design variables
$\underline{X}^p$	integer valued coordinates of point *p*
$\underline{y}$	vector of observed responses

Matrices

$\underline{B}$	approximation of the Hessian Matrix of the Lagrange function
$\underline{R}$	correlation matrix

Greek Symbols

Scalars

α	move parameter
β	random number between 0 and 1
δ	radius of radial basis function
δ	out of phase angle
δ_{max}	maximum tip deflection
ϵ_0	amplitude
$\varepsilon_I, \varepsilon_{II}, \varepsilon_{III}$	elastic, retarded elastic, viscous strain
$\bar{\varepsilon}$	average strain
$\dot{\bar{\varepsilon}}$	average strain rate
ε_i	independent errors
ε_{mem}^{max}	maximum membrane strain
ε_{pipe}	representative pipe strain level
$\dot{\varepsilon}_{pipe}$	representative pipe strain rate
$\dot{\bar{\varepsilon}}_{Tension}$	tension test strain rate
θ_h	unknown parameter in Kriging interpolation
λ_i	Lagrangian multiplier
υ	Poisson's ratio
ξ	perturbation factor
ξ_i	point to point distance factor for dimension *i*
σ^2	variance
σ_b	maximum allowable bending stress
σ_y	yield stress
σ_b	bending stress
$\sigma_{max}^{Fatigue}$	fatigue strength
σ_{max}^{Static}	static strength
σ_{limit}	maximum allowable stress
τ	shear stress
$\varphi(\underline{x})$	transfer function
Φ	penalty function

1. Introduction

Summary: This chapter serves as an introduction to the work carried out in this thesis. The motivation and the objectives of the work are presented. Furthermore, the overall layout of the thesis is described and can be summarized as follows: Chapter 2, 3 and 4 provide the necessary background information for the methodologies used within and developed for the three industry examples described in Chapter 5, 6 and 7. Chapter 8 concludes this thesis by listing achievements, giving conclusions and discussing possible future work.

1.1. Motivation

The times when engineers developed their products without any help of Computer-Aided Engineering (CAE) tools are long gone. On the contrary, in virtually any engineering discipline the application of some software tools has become standard procedure to shorten and in many cases also to simplify the design process. Especially in industries where the production of prototypes is an expensive and time consuming task, e.g. the automotive or the aerospace sector, this process has almost been entirely transferred into the virtual world. Only for validation purposes, the final design is tested in the real world prior to production. The main ingredient to the virtual design process is the computer based numerical simulation of the product under the required testing conditions. Most commonly, three CAE tools are used to conduct such a simulation: A pre-processing tool for the creation of the simulation model of the product including the boundary conditions, a solver that solves the underlying physical problem and returns the results and finally a post-processor for the visualization of these results. Interpretation of the results helps the engineer to make the necessary amendments to the current design. The updated design is tested again in the above way until the final design, satisfying all the requirements, is ready for real world validation.

This manual procedure is time consuming and ineffective for two reasons. Firstly, the model has to be updated to some extend when the geometry or properties have changed. Using the Graphical User Interface (GUI) of the pre-processor makes this a tedious job that is prone to errors. Secondly, the engineer's decision on improving the last design is a decision based on interpretation of the results and the engineer's experience. Thus, the effectiveness of the improvement is heavily dependent on the engineer's skills. Furthermore, the review of the results is a time consuming task in itself.

The first problem can be addressed by automating the model creation process. Most CAE tools can be controlled via commands written in a text file. Having established the parameters determining the model's variations, a software can be developed that writes such a command file automatically after finite values for the parameters have been specified. Depending on the complexity of the model, after several automatic model creations, the time required for the development of the software will fall below that that would have been consumed by the manual model creations. Furthermore, provided a thoroughly coded program, the problem of making errors during the model creation is vanquished. Another benefit of such an automated procedure is the fact that the engineer is likely to try more designs as all the work that is required is changing the parameters and pressing a button.

The second problem, i.e. the effectiveness of the design improvement in the design iteration, leads to the field of optimization. Here, the application of mathematical principals both makes the time-consuming manual result interpretation redundant and leads to designs that are optimal with regards to a defined objective. The previously described automated model creation is a pre-requisite for optimization. The specified parameters which govern the model span the so called design-space. All the possible designs are lying within. Optimization techniques seek to find those parameter values that lead to the design that exhibits the best value of the objective function.

With the ever increasing power of inexpensive multi-processor computers the optimization problems being feasible to be solved become larger and more widely applicable. Hence the general interest in structural optimization increases. As a result commercial optimization tools and optimization features in sophisticated CAE packages become more common. Nevertheless, optimization in engineering design is still mainly restricted to companies whose products are very expensive to manufacture or are manufactured in very high numbers e.g. in the aerospace or automotive sector. In other sectors the advantages of optimization are not as apparent and engineers are often reluctant to make use of them. The capabilities of these tools are still mainly limited to standardized problems and when the built-in functionality does not fully cover the problem in question an automated simulation needs to be set up to allow for the numerous simulations required for optimization. This process requires a wide range of knowledge such as model parameterization and batch mode software control and this knowledge is yet not widely spread. In some cases financial constraints rule out these sophisticated CAE tools and less advanced programs need to be used instead in which case this process becomes more cumbersome and more difficult.

In fact the automated simulation and optimization are closely connected. According to (Keane & Nair, 2005) one of the greatest benefits of the set-up of an optimization process is often the automated simulation itself. Even if optimization is not performed regularly the simulation process is simplified significantly. Despite the obvious importance and advantages of automated simulations the published work focusing on this subject is relatively sparse. This was also the outcome of FENet which was a European Thematic Network which aimed at 'improving both the quality of industrial applications of finite element technology and the level of confidence that can be placed in the computed results' (FENet, 2005). Its 110 member organizations represented various industry sectors, such as Aerospace, Land Transport and Marine or Offshore. (Bartholomew & Morris, 2008) discuss its findings. In a series of industry meetings, issues regarding engineering analyses were to be identified and prioritized. Within a list of 74 keywords, in terms of priority, 'Automation' was ranked 9th. The high priority of automation was influenced to some extend by the high interest of the automotive industry in automatic meshing, but for instance the Aerospace and the Marine and Offshore sector expressed the need for the advancement of the automation of the structural analysis process.

The advantages of optimization in engineering design are obvious and the need for the provision of hands-on frameworks that address industry problems is out of question. Also, more attention has to be paid to efficient and easy to use ways of setting-up the required automated processes as well as the problems arising from it. This is the overriding motivation of this work.

1.2. Objectives

Three industrial examples will be investigated within this thesis. All three examples are concerned with optimum design. The objective is to solve these problems using existing techniques and to develop new methods where necessary, hence to broaden the existing knowledge in optimum engineering design. Particular attention will be paid to the realization of the required automation of the underlying simulation processes in order to contribute to the knowledge already existing in this field.

1.3. Layout

Three key modules have to be chosen and integrated in an optimization process which results in the three-columns concept (Hinton et al., 2003). These three modules are the optimization technique, the optimization model and the simulation model.

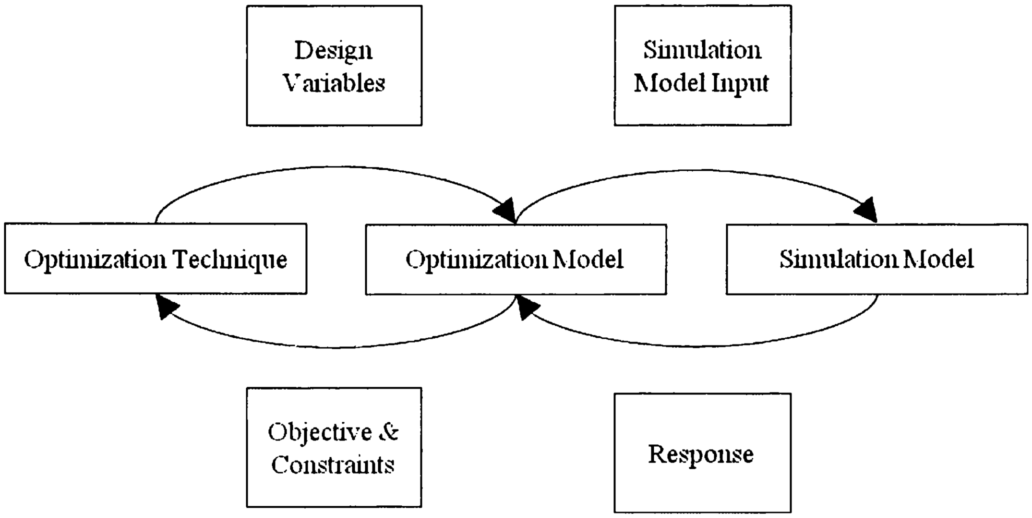


Figure 1.1: Three-columns model for optimization processes.

The third column, i.e. simulation model yields the structural response of the current design. The complexity of the simulation model can vary from a simple analytical expression to a complex finite element (FE) or computational fluid dynamics (CFD) model. This column is treated as a 'black-box' within this thesis and will not be discussed in detail. It shall be noted that the reliability of the optimization results naturally depends on the reliability of the simulation model. Hence, a high confidence in the quality of the employed simulation tools is crucial when performing optimization.

The first column, i.e. the optimization technique, solves the optimization problem. Using the evaluated objective function, constraints and, depending on its type, sensitivity information, the optimization technique improves the current design until convergence occurs. Chapter 2 reviews basic optimization techniques and their theoretical background. The two main categories are mathematical programming methods and evolutionary algorithms. An existing method, that is the permutation Genetic Algorithm (GA), will be amended and used for the creation of Optimum Latin Hypercubes (OLH) for constrained design spaces discussed in Chapter 3.

The second column, i.e. the optimization model, represents the link between the optimization technique and the simulation model. It transforms the current values for the design variables into the necessary input for the simulation model, e.g. the geometry that can be solved by the simulation model. Furthermore, it receives the response information from the simulation and uses it to evaluate the objective function and the constraints and passes them on to the optimizer. This column is the subject of Chapter 4. In particular the proposed approach towards process automation, i.e. automated simulation model creation, will be presented here. Necessary elements, such as available programming language types, ways of remote software control and techniques of model parameterization, are reviewed. Furthermore, a new approach for the decomposition of coupled size and shape optimization problems will be presented. Here, a sophisticated optimization tool is used to efficiently solve the size optimization as a sub problem, while the top level problem, which determines the shape parameters, is solved by means of a surrogate model.

Surrogate based optimization is typically applied when the evaluation of the objective and constraint functions requires the use of computationally expensive simulations. To keep the required time for the optimization acceptable, the number of simulations needs to be kept within limits. In such cases direct application of one of the optimization techniques might not be feasible. One possibility to circumvent this problem is to build a surrogate or metamodel and to use it as an approximation for the expensive simulation code. Here, a small number of function evaluations is performed in a first step. Then the obtained results are used to fit a response surface that interpolates the obtained results and thus approximates the original simulation code. This response surfaces can be evaluated much more efficiently and hence be used for the optimization. Due to its importance surrogate based optimization will be discussed in a separate chapter which is Chapter 3. Starting with the Design of Experiments (DoE), e.g. OLH, a certain number of points in the design space are specified, the models corresponding to each experiment are created and the simulations are conducted. Then the obtained responses are used to create the response surface by using an approximation method. This allows for the efficient evaluation of the approximated functions. One of the optimization techniques from Chapter 2 is then used to find the optimum set of parameters.

Commonly OLHs are created by assuming a complete *n*-dimensional design space even though this might not reflect reality. This assumption causes experiments to be created that might not be feasible or do not have a physical meaning. This means for the creation of the DoE and the response surface, unnecessary computational costs. Simply omitting infeasible points leads to a DoE with fewer experiments than desired and to a set of points that is not optimally distributed. A novel procedure has been developed that allows for the creation of OLHs for arbitrarily constrained design spaces which will be presented in Chapter 3.

A framework for surrogate based optimization will be implemented which contains the developed method for the creation of OLHs as well as an implementation of a continuous GA and a Kriging approximation.

As mentioned before, companies whose products are expensive and difficult to manufacture often take advantage of extensive simulation and sometimes even optimization tools. But when products are relatively cheap and easy to be produced, the use of simulation tools is rarer and an experience based or trial and error product development is a more common approach because knowledge about the necessary simulation tools is missing and the financial and time benefits are not as drastic and obvious. The costs of sophisticated and user-friendly tools is another drawback while the use of public domain programs is often cumbersome and not particularly intuitive. The structured wall pipes studied in Chapter 5 fall into the latter category. In order to introduce CAE, a Java based software tool will be developed that uses license free public domain software and in-house code only. Its use is simplified by automating processes with very few parameters that have to be determined. Furthermore, an optimization procedure is implemented in the tool that allows for the optimum design of the pipe profiles. It uses a simplified simulation model which contains only one box-section of the profile and a horizontal spring support to account for the omitted sections.

With the advent of commercial optimization tools and optimization features within sophisticated CAE programs more and more optimization tasks can be solved in an efficient manner without extensive effort. Nevertheless, many of the features are still limited to standard problems. When design problems go beyond this point the software needs to be extended to meet the desired requirements. Most programs offer the possibility to be controlled via scripting and or commands which simplifies this task. It can be desirable to use the available CAE tool to solve one part of the problem and to solve the other part using a bespoke software tool. For instance sizing of a model meshed with shell elements is standard procedure which can be set up and solved very efficiently by means of commercial tools. But the parameterization for shape optimization is not as straightforward since the parameters govern the geometry outline and a bespoke program can be more suitable. The novel optimization decomposition strategy developed in Chapter 4 is designed for such cases. It is applied in Chapter 6 for the optimum design of carbon-fibre competition mountain bike frames. The implemented framework for surrogate based optimization presented in Chapter 3 will be used for solving the top level shape optimization problem. This industry example will show how surrogate based optimization can help in solving optimization problems efficiently and how a complex structure can be parameterized for the use within automated simulations.

Even when the approach towards the design of an engineering structure is mainly based on experience, optimization and process automation can be helpful tools in improving the final design. This will be shown in Chapter 7 where another engineering design problem is considered. That is the design of a blade in a novel vertical axis wind turbine. Here, the design approach is based on the vast experience on the design of the more common horizontal axis wind turbines. Nevertheless, process automation will be used to test various material settings and shear web locations to account for the differences in loading. Additionally, size and ply stacking optimization are used to improve the final design.

Chapter 8 concludes this thesis by listing achievements, giving conclusions and discussing possible future work.

2. Optimization Techniques

Summary: This chapter gives an overview of existing optimization techniques. The methods used within this thesis, i.e. Sequential Quadratic Programming, the continuous Genetic Algorithm and an extension of the permutation Genetic Algorithm, denoted as selective permutation Genetic Algorithm, are discussed in more detail.

2.1. Introduction

A typical single objective structural optimization problem can be expressed in the following way:

$$\min \qquad f(\underline{x}) \tag{2.1}$$

$$\text{subject to} \qquad g_k(\underline{x}) \leq 0 \qquad k = 1, \dots, m \tag{2.2}$$

$$h_k(\underline{x}) = 0 \qquad k = 1, \dots, n \tag{2.3}$$

$$x_k^l \leq x_k \leq x_k^u \qquad k = 1, \dots, n_{DV} \tag{2.4}$$

Here, the objective $f(\underline{x})$ has to be minimized, while the m inequality constraints g_k and the n equality constraints h_k have to be satisfied. Furthermore, the n_{DV} design variables x_k have to lie within their bounds x_k^l and x_k^u.

Transforming each equality constraint into two equality constraints yields the simplified problem in Equations 2.5 to 2.7.

$$\min \qquad f(\underline{x}) \tag{2.5}$$

$$\text{subject to} \qquad g_k(\underline{x}) \leq 0 \qquad k = 1, \dots, m + 2 \cdot n \tag{2.6}$$

$$x_k^l \leq x_k \leq x_k^u \qquad k = 1, \dots, n_{DV} \tag{2.7}$$

The choice of the method that this problem is solved with governs the efficiency and sometimes whether or not the problem can be solved at all. Parameters that influence this choice are:

- Design variables: discrete, continuous or mixed
- Objective and constraint function: smooth or non-smooth, differentiable, convex

- Design space: constrained or unconstrained, bounded or unbounded, shape, dimension
- Cost of function evaluation
- Availability of gradients
- Existence of local and global minima
- Computability of function values
- Deterministic or stochastic optimization

Generally, all optimization methods can be grouped into two major categories: mathematical programming methods and evolutionary algorithms. The methods in the first group use function and often gradient information and improve the current design in a deterministic hill-climbing manner. They are mainly used for continuous parameters and require few function evaluations. They can converge to a local minimum. Evolutionary methods on the other side usually do not require gradients. Rather than using mathematical principles they mimic natural processes often containing stochastic elements. They are usually more successful in finding global minima but they can require more function evaluations than mathematical methods. They are used for continuous and discrete variables. Methods that do not require gradients are typically referred to as 'zeroth order methods' (Keane & Nair, 2005). They can be in the group of mathematical programming methods or evolutionary algorithms.

In this section, general aspects of both categories are presented and an overview of common methods is given. Furthermore, the methods used within this work will be discussed in more detail. These are Sequential Quadratic Programming (SQP) as a representative of the mathematical programming methods and different types of Genetic Algorithms (GA) representing the evolutionary algorithms (EA).

2.2. Mathematical Programming Methods

The following overview and categorization of mathematical programming methods follows (Hinton et al., 2003). A general overview is given in Figure 2.1.

The group of mathematical programming methods can be split into four main categories: Linear, nonlinear, integer and dynamic programming. The problems that can be optimized by means of linear programming methods are those that are characterized by linear objective and constraint functions. Furthermore, the objective and the constraints have to form a convex set. Nonlinear programming methods on the other

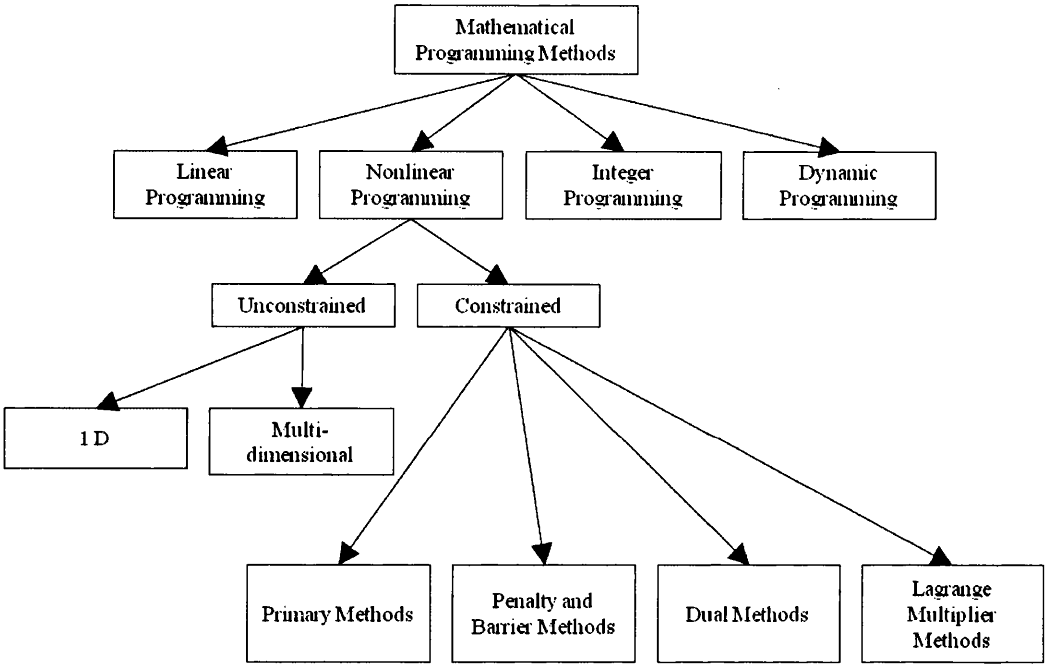


Figure 2.1: Categorization of mathematical programming methods.

hand can handle both linear and nonlinear functions. While the first two categories can only be applied to problems depending on continuous variables, the third one, integer programming, can handle discrete and mixed (discrete and continuous) parameters. The last category, dynamic programming describes the strategy of decomposing an optimization problem into several sub problems which are solved successively by means of another method.

Nonlinear programming methods form the group of optimization methods that is most widely used in structural optimization. They are subdivided into constrained and unconstrained methods. The latter can be split further into 1D and multi-dimensional methods. 1D methods play a crucial role within other methods where a sub-optimization occurs within a line search.

The constrained methods contain four subcategories: The primary methods, the penalty and barrier methods, the dual methods and the Lagrange multiplier methods. The primary methods work in the feasible design space only and never leave it throughout the optimization process. This means, the initial set of parameters needs to lie within a feasible region. An example for this group is the CONMIN algorithm (Vanderplaats & Moses, 1973). The penalty and barrier methods use a penalty term in the objective function penalizing violated constraints with an increase in the function value. Thus, an initially constrained problem becomes unconstrained. Penalty methods approach the optimum from outside and barrier methods from inside the feasible space. The third category within the constrained nonlinear programming methods contains the dual methods. They solve a problem indirectly by stating a dual problem where the unknowns are the Lagrange multipliers. Once computed they are reinserted to obtain the design variables. The Lagrange multiplier methods use quadratic approximations for the objective and linear ones for the constraints.

2.2.1. Kuhn-Tucker Optimality Condition

Using the Lagrangian multiplier method, the objective function can be reformulated as

$$\mathcal{L}(\underline{x},\underline{v},\underline{\lambda}) = f(\underline{x}) + \sum_{i=1}^{m+2\cdot n} \lambda_i \cdot \left(v_i^2 + g_i(\underline{x})\right), \qquad 2.8$$

where v_i is a slack variable and λ_i a Lagrangian multiplier. The necessary condition for a candidate $\underline{x}$ to be an extremum of Equation 2.8 is that its gradient vanishes.

$$\frac{\partial \mathcal{L}}{\partial x_k} = \frac{\partial f(\underline{x})}{\partial x_k} + \sum_{i=1}^{m+2\cdot n} \lambda_i \cdot \frac{\partial g_i(\underline{x})}{\partial x_k} = 0 \qquad k = 1, \dots, n_{DV} \qquad 2.9$$

$$\frac{\partial \mathcal{L}}{\partial v_k} = 2 \cdot \lambda_k \cdot v_k = 0 \qquad k = 1, \dots, m + 2 \cdot n \qquad 2.10$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_k} = v_k^2 + g_k = 0 \qquad k = 1, \dots, m + 2 \cdot n \qquad 2.11$$

These conditions only apply at a regular point, i.e. a point where the gradients of the active constraints are linearly independent.

A constraint is considered to be active if it is zero (within a numerical tolerance). Equation 2.10 determines whether or not a constraint g_k is active. In case of inactivity the corresponding Lagrangian multiplier is zero. Equation 2.11 ensures that none of the constraints is violated, i.e. greater than zero.

Using the definition of an active set, the above necessary conditions for a regular point to be a local minimum can be reformulated. These are the Kuhn-Tucker conditions:

$$\frac{\partial f(\underline{x})}{\partial x_k} + \sum_{i=1}^{m_a} \lambda_i \cdot \frac{\partial g_i(\underline{x})}{\partial x_k} = 0\,, \qquad k = 1, \dots, n_{DV} \tag{2.12}$$

$$\lambda_k \cdot g_k(\underline{x}) = 0\,, \qquad k = 1, \dots, m_a \tag{2.13}$$

$$\lambda_k \geq 0\,, \qquad k = 1, \dots, m_a \tag{2.14}$$

Here, m_a *is the number of* active constraints. The Kuhn-Tucker conditions are necessary and sufficient when the number of active constraints equals the number of design variables. If this is not the case a sufficient condition is a Hessian matrix of the Lagrangian that is positive definite in the subspace tangent to the active constraints (Haftka & Gürdal, 1992). For a convex problem, i.e. convex objective and constraints, the Kuhn-Tucker constraints are necessary and sufficient for a global optimum.

A design in a 2D design space that satisfies the Kuhn-Tucker conditions is illustrated in Figure 2.2. The vectorial sum of the gradient of the objective and the gradients of the active constraints multiplied by their corresponding non-negative Lagrangian multiplier equals to zero.

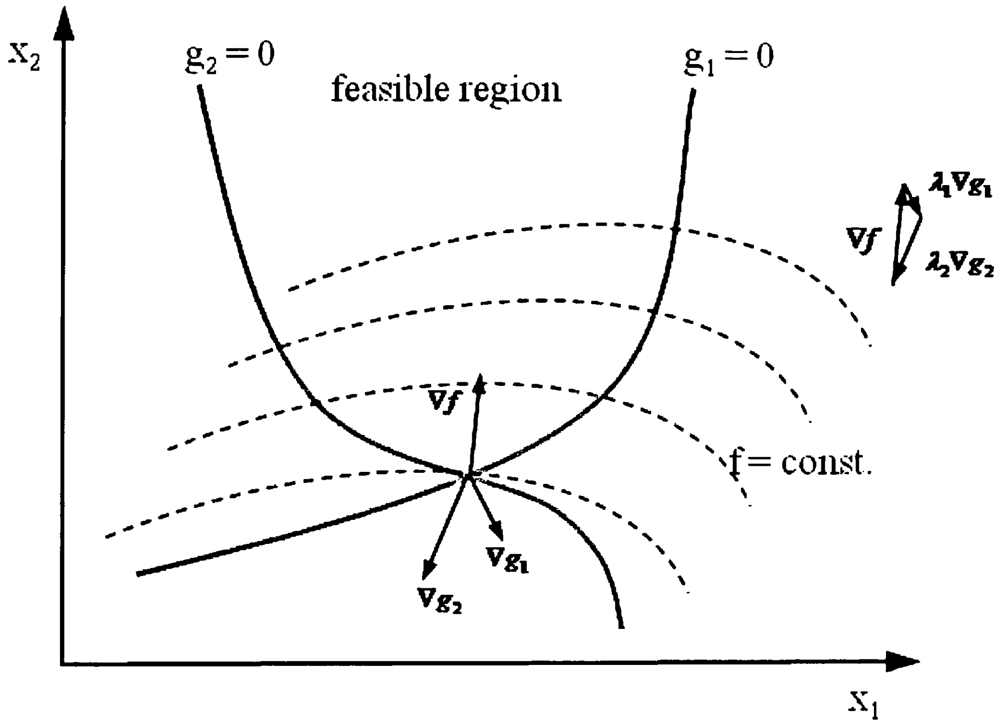


Figure 2.2: Illustration of a design satisfying the Kuhn-Tucker conditions in 2D.

2.2.2. General Optimization Strategy

According to (Hinton et al., 2003), most of the more powerful mathematical programming methods have the following general structure: In each iteration q the previous design $\underline{x}^{q-1}$ is updated according to Equation 2.15.

$$\underline{x}^{q} = \underline{x}^{q-1} + \alpha \cdot \underline{S}^{q} \qquad 2.15$$

Here, $\underline{S}^{q}$ is the actual search direction and α is a scalar move parameter that determines the length of the update vector that is added to the old design. Prior to this step, the direction and the move parameter have to be determined. This is done within two steps: A sub-optimization task and a 1-D search. Depending on the method in question, these two steps are solved in different ways. In this thesis only SQP will be discussed in more detail. For other methods, the reader is referred to (Hinton et al., 2003) and (Haftka & Gürdal, 1992).

After each update the optimizer checks the convergence criteria and carries on with the next iteration if no convergence has occurred.

2.2.3. Sequential Quadratic Programming

SQP is one of the most popular methods among the mathematical programming class (Hinton et al., 2003). The following discussion is done in accordance with (Vanderplaats Research and Development Inc., 1999). In SQP Taylor series approximations of the objective and constraint functions are created. A quadratic approximation of the objective and a linear one for each constraint is used to state the following sub-optimization problem. It is typically solved by means of the Modified Method of Feasible Directions (Hinton et al., 2003) and (Vanderplaats Research and Development Inc., 1999), to obtain the new search direction $\underline{S}^{q}$.

$$\mathit{min} \qquad f\left(\underline{x}^{q-1}\right) + \left(\nabla f\left(\underline{x}^{q-1}\right)\right)^{T} \underline{S}^{q} + \frac{1}{2} \cdot \left(\underline{S}^{q}\right)^{T} \underline{B}^{q-1}\, \underline{S}^{q} \qquad 2.16$$

$$\mathit{subject\ to} \qquad g_{k}\left(\underline{x}^{q-1}\right) + \left(\nabla g_{k}\left(\underline{x}^{q-1}\right)\right)^{T} \underline{S}^{q} \leq 0, \qquad k = 1, \ldots, m_{a} \qquad 2.17$$

Note that the matrix $\underline{B}$ is updated after the one dimensional search and approaches the Hessian Matrix of the Lagrange function. In the first iteration it is assumed to be the identity matrix. Now, the Lagrange multipliers λ_{k} for the active set are calculated.

Using these, the scalar multiplier α is found via the approximate Lagrangian function in Equation 2.18.

$$min \qquad f(\underline{x}^{q-1} + \alpha \cdot \underline{S}^{q}) + \sum_{i=1}^{m_a} u_i^q \cdot max[0, g_i(\underline{x}^{q-1} + \alpha \cdot \underline{S}^{q})] \qquad 2.18$$

$$with \qquad u_i^q = \begin{cases} |\lambda_i| & if\ q = 1 \\ max\left[|\lambda_i|, \frac{1}{2} \cdot (u_i^{q-1} + |\lambda_i|)\right] & if\ q \neq 1 \end{cases} \qquad 2.19$$

With the search direction and the move parameter at hand, $\underline{x}$ and $\underline{B}$ can be updated. The latter is updated by means of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) (Broyden, 1970) formula in Equation 2.20.

$$\underline{B}^{q} = \underline{B}^{q-1} - \frac{\underline{B}^{q-1}\underline{p}\,\underline{p}^{T}\underline{B}^{q-1}}{\underline{p}^{T}\,\underline{B}^{q-1}\,\underline{p}} + \frac{\underline{\eta}\,\underline{\eta}^{T}}{\underline{p}^{T}\underline{\eta}} \qquad 2.20$$

$$with \qquad \underline{p} = \underline{x}^{q} - \underline{x}^{q-1} \qquad 2.21$$

$$\underline{\eta} = \theta\underline{y} + (1 - \theta) \cdot \underline{B}^{q-1}\,\underline{p} \qquad 2.22$$

$$\underline{y} = \nabla\Phi^{q} - \nabla\Phi^{q-1} \qquad 2.23$$

$$\Phi = f(\underline{x}^{q}) + \sum_{i=1}^{m_a} \lambda_i g_i\,(\underline{x}^{q}) \qquad 2.24$$

$$\theta = \begin{cases} 1.0 & if\ \underline{p}^{T}\underline{y} \geq 0.2\underline{p}^{T}\underline{B}\,\underline{p} \\ \dfrac{0.8 \cdot \underline{p}^{T}\underline{B}^{q-1}\underline{p}}{\underline{p}^{T}\underline{B}^{q-1}\underline{p}^{T}\underline{y}} & if \underline{p}^{T}\underline{y} < 0.2\underline{p}^{T}\underline{B}\,\underline{p} \end{cases} \qquad 2.25$$

2.2.4. Convergence

(Hinton et al., 2003) proposes several convergence criteria. If one of the criteria is satisfied, the process terminates in the current iteration. These criteria are:

- The specified maximum number of iterations has been reached
- No feasible region has been found in a specified number of iterations
- The Euclidian norm of the objective function $|f^{q} - f^{q-1}|/|f^{q-1}|$ between two subsequent iterations falls below a specified value

- The absolute change of the objective function $|f^q - f^{q-1}|$ between two subsequent iterations falls below a specified value
- The Kuhn-Tucker conditions are satisfied.

2.2.5. Sensitivity Calculation

In the context of structural optimization the notion sensitivity describes the effect of a parameter variation in a model on a certain characteristic. In other words, a sensitivity is a derivative of a function describing this characteristic with respect to the parameter in question. Sensitivity analyses are not exclusively a mere tool for the use of gradient-based optimization methods. It has become a research topic on its own (Hinton et al., 2003) which is used in reliability analysis, inverse problems and parameter identification problems. Also, it can be used within manual design processes where the engineer redesigns the current design based on parameter sensitivities.

Several different ways for the calculation of sensitivities exist. They can greatly differ in the required implementation effort, the corresponding computational cost and the reliability of the obtained values. Basically, two main categories for sensitivity calculation exist: On the one side the discrete methods and on the other the variational or continuum methods. The former are easy to implement but, depending on the cost for a single simulation, they can lead to very high computational effort. The latter are more reliable and can be very efficient while their implementation can be difficult. In addition to discrete and continuous methods, automatic differentiation exists where variables in the source code are differentiated in an automated manner.

2.2.5.1. Discrete Methods

In discrete methods the sensitivities are obtained using the discretized structure. Three different subclasses can be specified: The Global Finite Difference Method, the Analytical Method and the Semi-Analytical Method.

Global Finite Difference Method

This method is very easy to set up but the computational effort can be very high, because a full simulation has to be performed for every single design variable. The sensitivity of a certain function F, such as the objective or a constraint function, with respect to a specific design variable x_i is calculated by performing an FE simulation. Now the design variable in question is perturbed by a small factor ξ and another simulation is run. The resulting function values of the two runs are compared according to Equation 2.26 to obtain the sensitivity. The sensitivity of any result from a simulation can be calculated in that manner.

$$\frac{\partial F}{\partial x_i} = \frac{F(x_i + x_i \cdot \xi) - F(\underline{x})}{x_i \cdot \xi} \tag{2.26}$$

The source code of the simulation tool does not need to be accessible but can be used in a 'black-box' way.

Analytical Method

In the analytical method the derivatives of the elements in the stiffness matrix are generated analytically. A prerequisite for that is that the source code of the used software is accessible. Also, the implementation can be more complex and derivatives have to be generated for every element subroutine individually. Especially for shape parameters this can be a very difficult task. Two approaches exist in this category: The direct differentiation and the adjoint variable method.

Semi-Analytical Method

In the semi-analytical method, those sensitivities are too complex to be derived analytically are computed using finite differences. This is done on element level. Thus, access to the source code is necessary. To improve efficiency and accuracy, wherever possible analytical methods are used.

2.2.5.2. Variational or Continuum Methods

Next to the discrete methods the second main sensitivity calculation method is the variational or continuum method. Here, not the discretized but the continuous structure is used. The sensitivities are obtained by applying variational mathematical theorems to the functions and operators describing the structure and the optimization problem. Thus, the derivatives are formulated prior to discretization.

2.2.5.3. Automatic Differentiation

In automatic differentiation, the equations are differentiated using the computer source code. Every variable in a computer program is computed by executing a sequence of functions. Knowing the derivative of the standard functions and exploiting the chain rule, the derivatives are built automatically. A second program is created which computes the analytical derivatives at the same time as the original part of the program is executed.

2.3. Evolutionary Algorithms

EAs are used for problems with discrete and continuous design variables. The general idea is to apply mechanisms of evolution, such as selection, mating, and mutation to a population of parameters. During the search, the population evolves according to the principle 'survival of the fittest' and moves towards the optimum solution. A main advantage of EAs is that the population searches the entire design space and thus, in theory, is capable of finding the global optimum at the presence of many local minima.

Some of the most popular methods in the group of evolutionary methods are GAs, evolutionary programming (EP), evolution strategies (ES), simulated annealing (SA) particle swarm optimization (PSO) and ant colony optimization (ACO).

2.3.1. Simulated Annealing

SA was introduced by (Kirkpatrick et al., 1983). It simulates the annealing process of a substance which was heated above its melting temperature and gradually cooled down. During a perfectly controlled cooling process a crystalline lattice is formed which minimizes its energy probability distribution. The algorithm starts with a random set of design variables. They are randomly changed with a higher current heat leading to greater probability of change. Based on the evaluated objective and the current temperature the algorithm determines whether the new set of variables is accepted or rejected. When the objective stops improving the temperature is reduced until it reaches a certain minimum.

2.3.2. Particle Swarm Optimization

PSO has been formulated by (Kennedy & Eberhart, 1995). The idea this algorithm is based on originates from social behaviour of animals such as bird flocking or fish schooling. The individuals in the population matrix are called particles which move in the design space at a certain velocity. This velocity is updated based on the best solution that occurred throughout the optimization process and that in the current iteration. The advantages of PSO are the relatively easy implementation and the small number of parameters that need to be adjusted.

2.3.3. Ant Colony Optimization

ACO has been developed by (Dorigo et al., 1996). Ants that search for food leave a trail of pheromones which attracts other ants to choose the same path which increases the pheromone concentration more. Also, the faster an ant returns from the food location, the more the pheromone level increases. This mechanism is mimicked within an ACO. To avoid premature convergence pheromone evaporation can be implemented. A typical application problem for ant colony optimization is the travelling salesperson problem.

2.3.4. Evolutionary Programming and Evolutionary Strategies

EP and ES methods mimic evolutionary processes. Each individual in the population creates one child by mutation of itself with minor changes being more probable than substantial ones. Furthermore, this probability decreases as the process approaches the optimum. The design variable values are used directly which makes encoding and decoding unnecessary. For more information the reader is referred to (Bäck, 1996).

2.3.5. Genetic Algorithms

GAs are the most common methods within the group of EAs. They were proposed by (Holland, 1975). They adapt evolutionary principles such as selection and survival of the fittest. Depending on the variable representation they can handle discrete, continuous and permutation problems. Three different forms of GAs are presented in more detail in the following. Additionally, a variation of the permutation GA has been developed and will be discussed.

2.3.5.1. Terminology

A central part of a GA is a population of possible solutions to the optimization problem in question. This population is an array containing n_{pop} individuals where each one is represented by a chromosome. Each chromosome is a string of n_{DV} genes. Thus, it is a $n_{pop} \times n_{DV}$ matrix, where n_{DV} is the number of design variables to be determined in the optimization problem. The two dominating ideas behind a GA are the principles of 'survival of the fittest' and 'genetic recombination'. Their counterparts within the GA are selection and mating. The fitness of each chromosome is evaluated via the fitness function, in optimization contexts usually referred to as the objective function. Because the objective function is typically minimized the fitness function is often denoted as cost function. Despite this contradiction, within this thesis, the term fitness will be used to stress the analogy to evolutionary processes.

A GA starts by initializing the population. This is typically done randomly. In the next step the initial population enters the loop of evolution, where its chromosomes are reorganized according to their fitness with the fittest at the top of the list and the least fit one at the end. Depending on the form of the GA in question the chromosomes have to be decoded prior to function evaluation.

The next step is the natural selection, where only a fraction of the population survives (top part of the list) while the rest is discarded (bottom part of the list). The various forms of selection are presented in Section 2.3.5.2 in more detail. Now, the gap that occurred due to the ‘died’ individuals is filled with offspring. To this end, the survived individuals begin to mate and their genes are passed on to their offspring. As in nature some of the genetic information originates from the father and some from the mother, denoted as genetic recombination.

After the mating, mutation of the individuals can occur. To this end, random changes of the genes are performed. For the optimization this step has several advantages. Firstly, new information is added to the pool of genes which allows the exploration of other regions of the variable space. Secondly, it prevents premature convergence, as constantly new genetic information is proposed. Naturally, the fittest individual in the population is excluded from mutation to prevent a possible optimum candidate to be altered.

In a final step, the algorithm checks for convergence. In case of convergence, the algorithm stops. Otherwise, the loop starts again and a new generation is produced. The flow of a GA is illustrated in Figure 2.3.

2.3.5.2. Selection

Selection describes the process of determining which individuals will be paired to produce offspring. Preceding this step, the population has been reordered according to their fitness with the fittest being the first. Several approaches to selection exist, such as pairing from top to bottom, random pairing, tournament selection and biased roulette wheel selection (also denoted as weighted random pairing).

Pairing from top to bottom is the simplest of the mentioned methods. There, going through the list from top to bottom, the two adjacent chromosomes are paired.

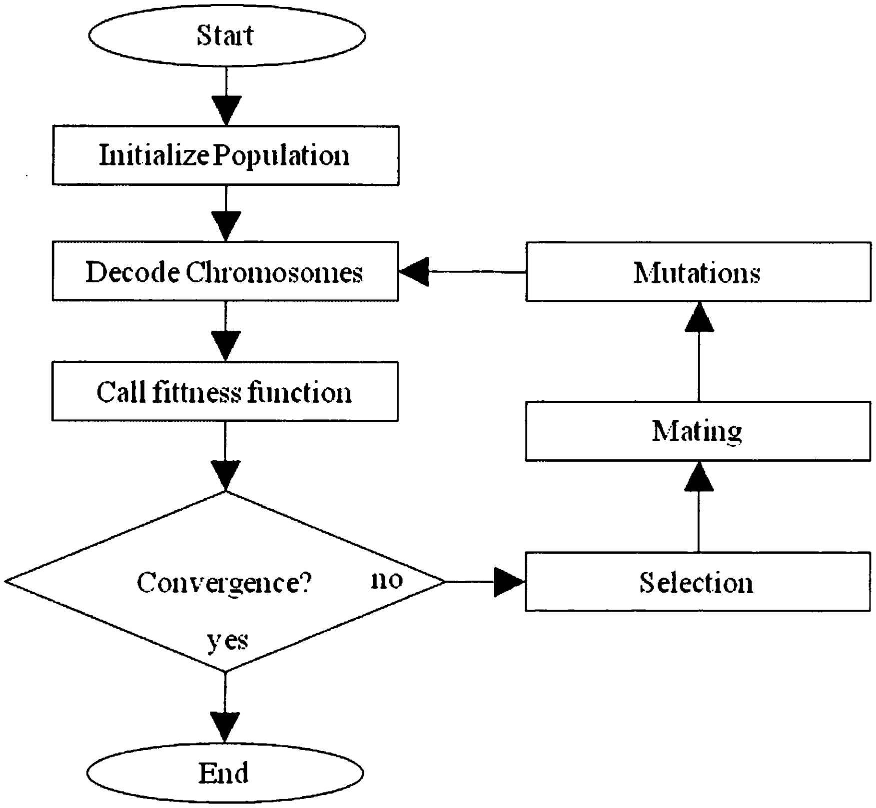


Figure 2.3: Flow of a typical genetic algorithm.

Less deterministic but comparably simple is the random pairing, where the mates are chosen randomly. The remaining two methods both mimic the natural process of selection more closely, by giving fitter chromosomes a higher probability of being selected for mating. The tournament selection randomly picks a certain number of chromosomes from the mating pool, from which the fittest is selected as a parent. The other parent is selected in the same way. In the biased roulette wheel selection, each chromosome is assigned a number P_n according to Equation 2.27 which is dependent on its rank n. n_{keep} is the number of chromosomes in the population that are possible candidates for mating. Then, a random number between 0 and 1 is generated and the chromosome whose P_n is the first to exceed it is chosen.

$$P_n = \sum_{i=1}^{n} \frac{n_{keep} - i + 1}{\sum_{j=1}^{n_{keep}} j} \qquad 2.27$$

Using the fitness rather than the rank leads to a P_n calculated according to Equations 2.28 and 2.29, where $f_{n_{keep}+1}$ is the fittest of discarded chromosomes.

$$P_n = \sum_{i=1}^{n} \left| \frac{F_i}{\sum_{j=1}^{n_{keep}} F_j} \right| \qquad 2.28$$

$$with \qquad F_i = f_i - f_{n_{keep}+1} \qquad 2.29$$

2.3.5.3. Forms of Genetic Algorithms

The format of the genes distinguishes the different forms of GAs and entails differences in the mating and mutation methods. Three different forms of GAs will be discussed in the following: The binary the continuous and the permutation GA. Additionally, a variation of the permutation GA has been developed and will be discussed as well. The best choice of the form of GAs depends on the optimization problem in question.

Binary Genetic Algorithm

The binary GA uses binary numbers to represent the values for the design variables. It is applicable to problems with any form of design variables. This form of GA will not be used within the thesis but since GAs originated in this representation it will be discussed in the following.

Encoding: In a binary genetic algorithm the chromosomes are encoded in binary numbers. Every gene in a chromosome is a binary number made from n_{gene} bits, representing one variable in the optimization. Thus, one variable can take $2^{n_{gene}}$ different values. A typical chromosome in binary format is illustrated in Figure 2.4.

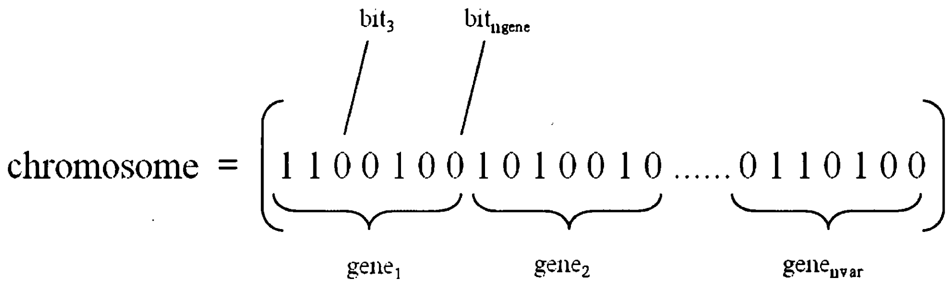


Figure 2.4: Illustration of a chromosome for a genetic algorithm encoded in binary format.

Prior to function evaluation, the chromosome has to be decoded into its original format. This can be any format, such as discrete or continuous numbers, characters, etc. While for discrete values a mere conversion from binary to decimal format is needed, for continuous values, another step, the quantization, is required.

Here, the range of a variable is divided into $2^{n_{gene}}$ intervals. Each discrete value of a gene can be then related to one range. The middle of the range, or sometimes the lower or upper end, is used as the value for that variable. An example of a chromosome made up from four bits is given in Table 2.1. The binary chromosomes are decoded to integer and continuous values. The latter assume a possible range between 0 and 10.

Table 2.1 Example of different values of a binary chromosome with 4 bits. Thus, it can take 16 values. Also, continuous values for a range from 0 to 10 are given for mid, low and high quantized levels.

chromosome	integer value	mid quantized	low quantized	high quantized
0000	*0*	*0.3125*	*0.000*	*0.625*
0001	*1*	*0.9375*	*0.625*	*1.250*
0010	*2*	*1.5625*	*1.250*	*1.875*
0011	*3*	*2.1875*	*1.875*	*2.500*
0100	*4*	*2.8125*	*2.500*	*3.125*
...	...	...	...	...
1111	*15*	*9.6875*	*9.375*	*10.000*

The encoding in ordinary binary numbers can slow the convergence of a GA down. For instance during mating, the cross-over can lead to offspring whose genes are fairly similar to their parent's while their decoded values differ significantly. This problem is magnified with increasing number of bits (Haupt & Haupt, 2004). A possible solution to it is Gray coding where the binary representation is changed in such a way that the cross-over of two similar decoded genes yields offspring with similar values. For more information on Gray coding see (Haupt & Haupt, 2004).

Mating: Many different mating methods exist. Here, three basic ones will be presented. The simplest one is the single cross-over point. It starts by choosing a random number between 1 and the length of the chromosome. The parents' chromosomes are split into two halves at the location of the cross-over point and each of the offspring is made up of one half of $parent_1$ and the other of $parent_2$.

The two point cross-over is similar to the previous method. Only, here two cross-over points are chosen splitting each parent into three sections. The offspring are then made up of two sections from one parent and one of the other. Compared to the first method, the variety of possible offspring is significantly increased. It is also possible to introduce a third parent to create up to 18 different offspring.

Another method is the uniform cross-over where a mask of random zeros and ones is created that has the same length as the chromosomes. The mask determines if a bit in an offspring originates from *parent$_1$* or *parent$_2$*. If the value at location m in the mask is zero, then *offspring$_1$* has the value of bit m of *parent$_1$* and *offspring$_2$* that of *parent$_2$*. The three presented methods are illustrated in Figure 2.5.

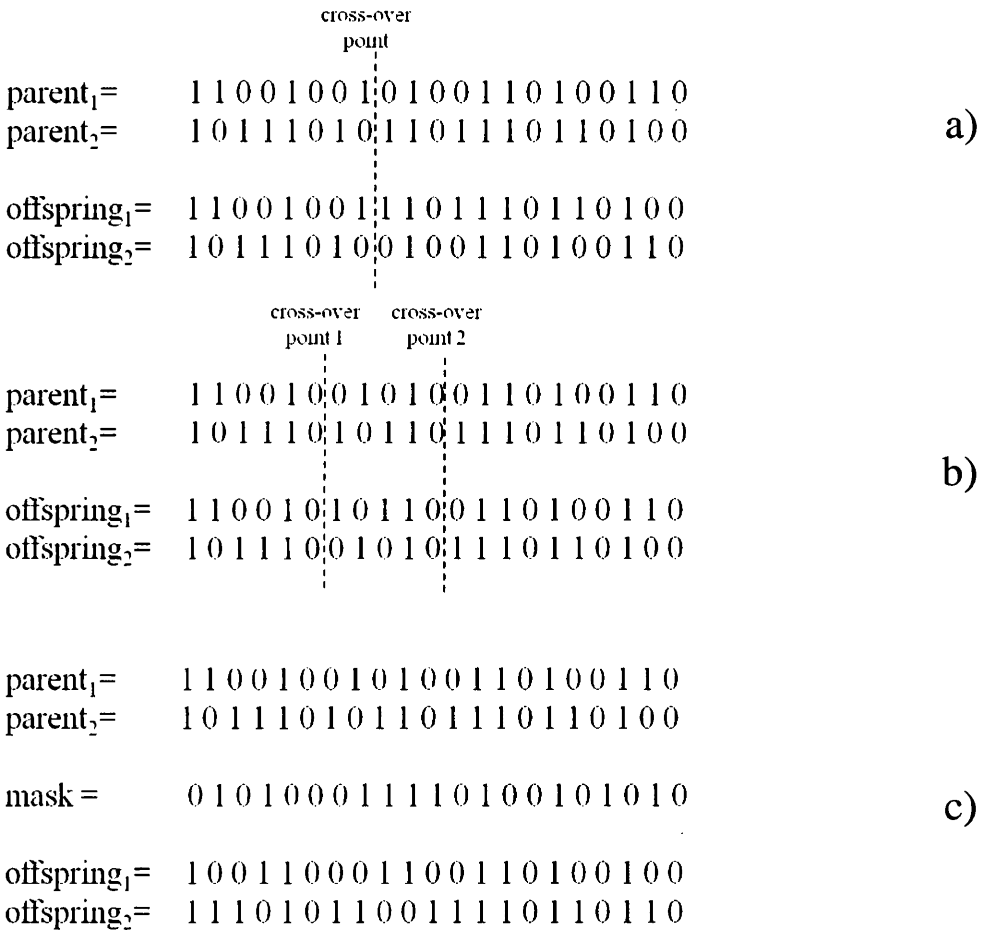


Figure 2.5: Illustration of a three different cross-over methods for binary genetic algorithms: a) the single cross-over point, b) the two point cross-over and c) the uniform cross-over.

<u>Mutation:</u> The mutation in binary genetic algorithms is very straightforward. If bit m in chromosome n was selected for mutation, its value is simply swapped from zero to one and vice versa.

Continuous Genetic Algorithm

Instead of using binary encoding the continuous GA uses floating-point numbers. This naturally confines it to problems which have design variables in continuous format. At the same time no chromosome decoding is necessary which makes the method easier to implement and saves computing time. An implementation of this form of GA is used in the optimization using a Kriging model in Section 3.4 and in Chapter 6.

Encoding: If the variables of the optimization problem are continuous, a binary encoding can be both cumbersome and inaccurate, because the chromosomes have to be decoded prior to function evaluation and for an accurate quantization chromosomes can become very long. The continuous GA avoids this by using floating-point numbers, which are normalized to lie in between 0.0 and 1.0. A typical chromosome is shown in Figure 2.6.

$$\text{chromosome} = \left(p_1 \; p_2 \; p_3 \; p_4 \cdots\cdots p_{nvar} \right)$$

Figure 2.6: Illustration of a chromosome for a genetic algorithm encoded in continuous format.

Mating: Similarly to a binary GA, here too, cross-over points can be selected, that determine the genes of each parent that are propagated to an offspring. Only, here entire genes are swapped, rather than single bits. Because of this, no new information is introduced. A solution to this problem is the blending method. There, the genes of the offspring are combinations of those in the parents. The blending methods can be generally expressed by Equations 2.30 and 2.31.

$$p_n^{off_1} = \beta \cdot p_n^{par_1} + (\beta - 1) \cdot p_n^{par_2} \tag{2.30}$$

$$p_n^{off_2} = (\beta - 1) \cdot p_n^{par_1} + \beta \cdot p_n^{par_2} \tag{2.31}$$

Commonly, β is a random number between 0 and 1, which means the offspring's variable is an interpolation between those of the parents. If values outside the range of the values of the parents shall be created, extrapolation is required. In this case, the newly created values have to be checked if they are still within their range.

(Haupt & Haupt, 2004) suggests a blending combined with a cross-over method. Then, one variable from both designated parents is chosen randomly. These two are combined according to Equations 2.30 and 2.31. The variables to the left of the blended one propagate from *parent*$_1$ to *offspring*$_1$ and to the right to *offspring*$_2$ and vice versa.

Mutation: The Mutation in a binary GA is performed by randomly swapping one bit in a chromosome from zero to one or vice versa. This way, new values can occur. In a continuous GA this procedure of swapping entire variables would not lead to new information. A simple solution is to exchange a randomly chosen gene by a new random number between 0 and 1.

Permutation Genetic Algorithm

Permutation GAs are used whenever problems occur that involve bringing numbers in a list into optimum order, while keeping their content constant. A famous example representing this type of problems is the traveling salesperson, where a set of cities has to be visited in an order that minimizes the distance of the trip. Here, the list of cities can be rearranged in any order while no city can be visited twice and none skipped.

Encoding: The encoding in permutation GAs is arbitrary, i.e. it is irrelevant if integers, characters or floating-point numbers are reordered. Convenient is the use of integer values where each gene represents a problem-specific item in a list. In the traveling salesperson problem, each gene stands for the corresponding city in the list.

Mating: The previously presented mating methods are not appropriate for a permutation GA, since they do not guarantee that each item in the list occurs only once. (Haupt & Haupt, 2004) discusses various methods that consider this requirement. These are the partially matched cross-over, the order cross-over and the cycle cross-over.

In this thesis only the cycle cross-over (Goldberg, 1989) will be discussed in more detail. In the first step, $offspring_1$ is identical to $parent_1$ and $offspring_2$ is identical to $parent_2$. Only the first gene in each chromosome is swapped. Naturally, one item occurs twice in each chromosome, unless the first genes in both parents were identical. In the following steps, the gene in $offspring_1$ that is identical to the first one is found and swapped with the corresponding one in $offspring_2$. This step is repeated until no more doubles occur in the offspring. Figure 2.7 illustrates this procedure with an example.

Mutation: The same problem as for mating, applies to the mutation, too. If a new random item would replace an old gene, as done in continuous GAs, one item would occur twice. This can be avoided by swapping two random genes in the same chromosome. This way the order of the chromosome is mutated while no value is substituted by a new one.

$$\begin{array}{llcll} \text{parent}_1= & [\,7\;3\;1\;5\;6\;2\;4] & \Rightarrow & \text{offspring}_1= & [\,\mathbf{2}\;3\;1\;5\;6\;2\;4] \\ \text{parent}_2= & [\,2\;1\;5\;4\;7\;6\;3] & & \text{offspring}_2= & [\,\mathbf{7}\;1\;5\;4\;7\;6\;3] \\ & & & & \\ & & & \text{offspring}_1= & [\,\mathbf{2}\;3\;1\;5\;6\;\mathbf{6}\;4] \\ & & & \text{offspring}_2= & [\,\mathbf{7}\;1\;5\;4\;7\;\mathbf{2}\;3] \\ & & & & \\ & & & \text{offspring}_1= & [\,\mathbf{2}\;3\;1\;5\;\mathbf{7}\;\mathbf{6}\;4] \\ & & & \text{offspring}_2= & [\,\mathbf{7}\;1\;5\;4\;\mathbf{6}\;\mathbf{2}\;3] \end{array}$$

Figure 2.7: Illustration of the cycle cross-over. Three steps are necessary to obtain two offspring without any duplicates. The bold genes have been altered with respect to their corresponding parent.

Selective Permutation Genetic Algorithm

A variation of the traveling salesperson problem is the case where he or she is supposed to visit only a certain number of cities on the list, i.e. contrary to the original version a certain number is skipped. The problem in question is still of a permutation nature since the current list of cities has to be changed to find the optimum order. At the same time though, the optimum content of the list needs to be determined as well, or in other words which items are best to be left out.

Naturally, the permutation GA discussed in Section 2.3.5.3 is not designed for solving this kind of problem as the content in the list is supposed to remain constant. The cycle cross-over is capable of introducing new information to a chromosome if the content in both parents is not identical but experience showed that often premature and suboptimal convergence occurs if no additional measures are taken. To this end, the employed algorithm has to be extended in a way that it is capable of replacing items in the current list by new items from the list of left-outs in a more designated fashion.

The proposed amendment is based on the assumption that the number of left-out items is small in comparison to the items of choice, i.e. in general the problem in question is a permutation problem with only a selective aspect to it. The selective feature is realized by the introduction of a new mutation scheme, while the mating remains unchanged.

The mutation in the original formulation randomly swaps two items in the list; hence no currently left out item will be introduced to the chromosome. This mutation scheme will be denoted as 'Exchange Mutation' in the context of the selective GA. In order to introduce new information a second mutation scheme, denoted as 'Substitution Mutation', is included in the algorithm. Here, a random item from the current list is replaced by a random item from the list of left-outs. That way no two items will occur twice in the list and its content will change in the course of the optimization.

This selective permutation GA is used in Section 3.2.3 for the creation of Optimum Latin Hypercubes for constrained design spaces. Here, a number of points have to be distributed in an n-dimensional design space as space-filling as possible. To this end the vectors for each coordinate direction containing the discrete integer valued coordinates are built and amended to find the points' optimum locations. Since none of the coordinates can occur twice in the list this problem is of a permutation nature. At the same time not all design space divisions might be in use, i.e. there are more divisions than points. Thus, some of the coordinate values have to be left out, which makes the proposed selective permutation GA a suitable optimization technique for this problem.

2.3.5.4. Choice of Parameters

Before a GA can be applied to an optimization problem, some parameters have to be specified. These are the population size, the cross-over and the mutation rate. The best choice is problem dependent and should be handled with care (Hinton et al., 2003). While the population size is highly dependent on the problem in question, e.g. the number of design variables, good working values for the mutation and cross-over rates can be generalized to some extent. For instance (DeJong, 1975) finds a cross-over rate of 60% and a mutation rate of 0.1% to be good working values. Later, (Grefenstette & Fitzpatrick, 1985) proposed 95% cross-over and 1% mutation rate.

2.3.5.5. Constrained Problems

According to (Goldberg, 1989) a practical way of incorporating constraints in a GA is the Penalty method. Here, the objective function is penalized if any of the constraints is violated. This penalization depends on the degree of constraint violation. The optimization problem, specified in Equations 2.5 to 2.7 in Section 2.1 of minimizing the objective $f(\underline{x})$ subject to m inequality constraints $g_i(x) \geq 0$ is transformed into an unconstrained problem by minimizing the penalized objective $\bar{f}(\underline{x})$ in Equation 2.32 is minimized, where r is a penalization coefficient and Φ a penalty function.

$$\bar{f}(x) = f(x) + r \cdot \sum_{i=1}^{m} \Phi[g_i(x)] \qquad 2.32$$

(Goldberg, 1989) suggests for Φ to square the violated constraints, while r depends on the fractions between the values for a typical objective evaluation and moderate constraint violation.

2.3.5.6. Convergence

(Hinton et al., 2003) propose the following three criteria to test for convergence:

- If the percentage difference between the average fitness of all chromosomes and the fitness of the fittest falls below a small specified value
- If the percentage improvement in the best fitness within a certain number of successive generations falls below a small specified value
- If the maximum number of generations has been reached

2.4. Concluding Remarks

In this chapter an overview of existing optimization techniques has been given. Particular attention has been paid to the three methods used in this thesis: SQP, the continuous GA and the selective permutation GA. An SQP type optimization tool is used in Section 5.5.3 for the optimization of structured wall pipe profiles. The continuous GA will be used for the optimization using surrogate models. In Section 3.4 it will be used for the optimization of three benchmark problems and in Chapter 6 for the optimization of a carbon-fibre bicycle frame. The selective permutation GA has been developed to specifically face the problem of creating OLHs for constrained design spaces. Example problems are presented in Section 3.2.3. Furthermore, it is also applied for the creation of the OLH in Chapter 6. The remaining methods have been discussed for completeness only.

3. Surrogate Based Optimization

Summary: The main ingredients of surrogate based optimization are Design of Experiments and design space approximation. This chapter gives an overview over the most important existing methods. A new method for the creation of Optimum Latin Hypercube Design of Experiments in constrained design spaces has been developed and will be presented in detail. Numerous example problems show the validity of the chosen approach. A framework for surrogate based optimization is implemented which contains the developed Optimum Latin Hypercube creation scheme, a Kriging interpolation and a continuous Genetic Algorithm. It is applied to two unconstrained and one constrained benchmark problem.

3.1. Introduction

In many design optimization tasks the evaluation of the objective and constraint functions requires the performance of computationally expensive simulations. To keep the required time for the optimization acceptable, the number of simulations needs to be kept within limits. In such cases direct application of one of the previously presented optimization techniques might not be feasible. One possibility to circumvent this problem is building a surrogate or metamodel and using it as an approximation for the expensive simulation code. Here, a small number of function evaluations is performed in a first step. Then, the obtained results are used to fit curves that interpolate the obtained results and thus approximate the original simulation code. These curves can be evaluated much more efficiently and hence be used for the optimization with one of the previously discussed optimization techniques.

In the following, methods will be presented that allow for the creation of sample plans, often referred to as Design of Experiments (DoE). They determine at which locations in the design space, i.e. for which parameter values, the simulations will be performed. Subsequently, different approximation methods will be discussed that are utilized for the interpolation of the obtained results. Finally, a set of tools will be implemented and applied to three benchmark problems in order to test and validate them.

3.2. Design of Experiments

The accuracy of a surrogate model depends significantly on the choice of the design variable value combinations for which the simulations are run (Keane & Nair, 2005). The sampling plan in the design space that determines these combinations is called DoE. This subject was first studied by (Fisher, 1935) in the context of physical experiments. A key difference between physical and computer experiments is the existence of random errors. Traditional methods for designing experiments take a random error into account. They create DoEs that are specifically made to fit regression models. These traditional techniques include central composite design, Box-Behnken design and full- or fractional-factorial designs. All these methods tend to locate the experiments at the extremes of the design space (Keane & Nair, 2005). The random error which occurs in physical experiments does not occur in computer simulations. Here, numerical noise occurs due to incomplete convergence, round-off errors or the simple fact that continuous physical phenomena are replaced by discretized ones (Alvarez, 2000). For computer simulations, space-filling designs such as Latin Hypercube Sampling (LHS) or Orthogonal Array sampling (OAS) are more desirable (Keane & Nair, 2005) (Simpson et al., 2004).

Several techniques have emerged which aim at an improvement of the uniformity of the LHS by optimizing a particular spreading measure (Queipo et al., 2005) for instance by maximizing the minimum point distances (Johnson et al., 1990). For small sample sizes this approach tends to distribute the points at the extremes of the design space (Keane & Nair, 2005). An alternative approach is presented in (Bates et al., 2003) and (Bates et al., 2004). Here, the OLH is found by minimizing the Audze-Eglais potential energy of the points using a permutation GA.

In practice, design spaces are not necessarily box-shaped but highly irregular in which case a typical approach is to create a DoE for the full design space and omit those experiments that lie outside the feasible space. This approach leads to a DoE with fewer experiments than desired and to a set of points that is not optimally distributed (Keane & Nair, 2005). A strategy for the creation of OLHs for constrained design spaces will be discussed in Section 3.2.3.

Ideally, a surrogate model is created in an iterative way. That is to say the initial DoE is fairly coarse. Then, locations of insufficient accuracy in the model are identified and new experiments added there until the model's accuracy is satisfactory (Keane & Nair, 2005). Adding new points to an existing DoE in a specified sub-domain as homogenously as possible is the objective of Van Keulen's approach (van Keulen & Toropov, 1997). A similar approach has been discussed by (Jones et al., 1998).

Following (Keane & Nair, 2005) and (Queipo et al., 2005), within this thesis a selection of those DoE techniques are discussed in more detail that assume noise-free experiments. These are OAs, LHS and OLHs. In addition to that, the developed formulation for the creation of OLHs in constrained design spaces will be presented.

3.2.1. Orthogonal Arrays

The following review of OA is done in accordance with (Keane & Nair, 2005). An OA for n_P experiments in n_{DV} dimensions is defined by an $n_P \times n_{DV}$ matrix containing elements between 0 and $q-1$. An OA has a strength t, if all the possible q^t rows appear the same γ number of times in every submatrix $n_P \times t$. An OA can be compactly written as $O_a(n_P, n_{DV}, k, t)$. k specifies the number of discretization levels for each design variable. By definition any OA of the form $O_a(n_P, n_{DV}, n_P, 1)$ is an LHS. The values for the design variables can be produced from the matrix via

$$x_j{}^{(i)} = \frac{\pi_j\left(A_j{}^{(i)}\right) + U_j{}^{(i)}}{k}, \qquad 1 \le j \le n_{DV}, \qquad 1 \le i \le n_P \qquad 3.1$$

Here, $\pi_j\left(A_j{}^{(i)}\right)$ is the *ij*-th element of O_a and $U_j^{(i)}$ is a random number between 0 and 1.

(Queipo et al., 2005) note two disadvantages of OAs. Firstly, for a particular combination of rows, columns, levels and strength, it is possible that no OA exists. Secondly, the projection of the points onto one dimension might contain duplicates. The latter is an undesirable characteristic when used with deterministic computer simulations since the same value for one parameter is used multiple times which might reduce the degree of new information that is added to the approximation model.

3.2.2. Optimum Latin Hypercube Sampling

LHS was initially proposed by (McKay et al., 1979) and (Iman & Conover, 1980). Each variable is divided into n_P equidistant levels. For each level, only one point is allowed. (Queipo et al., 2005) call methods exhibiting this requirement stratified. In (Stinstra et al., 2003) this is denoted as the Non-collapsingness criterion (NC) which will be used henceforth. The importance of this is based on the fact that some parameters have only a minor influence on the result of a simulation. Thus, running two collapsing experiments might lead to similar results, i.e. to no new information and effectively to unnecessary computational costs.

The random generation of LHS leads to DoEs that usually exhibit poor space-filling quality. OLHs are LHS that are characterized by an optimal distribution of the placed experiments in the sense of a chosen criterion. Again, several different approaches arose, for instance the minimization of the integrated mean square error (IMSE) by (Sacks et al., 1989), maximization of entropy by (Shewry & Wynn, 1987), the maximin distance criterion by (Johnson et al., 1990) or the minimization of the Audze-Eglais potential energy by (Bates et al., 2003) and (Bates et al., 2004). (Liefvendahl & Stocki, 2006) studied the performance in terms of efficiency and space-fillingness of the last two approaches and concluded that the approach based on the potential energy minimization outperforms the maximin distance criterion. Publications following this recommendation include (Panda & Manohar, 2009) and (Cook & Skadron, 2007).

(Bates et al., 2003) and (Bates et al., 2004) are using the Audze-Eglais method (Audze & Eglais, 1977) which follows a physical analogy: A system of mass-points exert repulsive forces on each other which leads to potential energy in the system. In case of a minimum of the potential energy the points are in equilibrium. The analogous potential energy U for the creation of the OLH is defined as:

$$U = \sum_{p=1}^{nP} \sum_{q=p+1}^{nP} \frac{1}{L_{pq}^{2}}, \qquad 3.2$$

$$L_{pq} = \sqrt{\sum_{m=1}^{n_{Dim}} \left(X_m^p - X_m^q\right)^2} \qquad 3.3$$

Here, L_{pq} is the distance between the points p and q and $\underline{X}^p$ and $\underline{X}^q$ their integer valued coordinates. n_{Dim} indicates the dimensionality of the design space. The optimization problem for the creation of OLHs requires the minimization of the potential energy. The design variables are the coordinates of the experiments which can take any integer value between 1 and n_P. The optimization problem can be formulated as:

$$\min \rightarrow U \qquad 3.4$$

$$with \quad 1 \leq X_m^n \leq n_P \quad for \quad m = 1, \dots, n_{Dim} \quad and \quad n = 1, \dots, n_P \qquad 3.5$$

$$X_m^p \neq X_m^q \quad for \quad m = 1, \dots, n_{Dim} \qquad 3.6$$

A requirement for an OLH is that only one experiment is placed per level, denoted as Non-collapsingness as stated in Equation 3.6. The permutation GA used in (Bates et al., 2003) and (Bates et al., 2004), a priori satisfies this requirement, as it only exchanges coordinates in the same coordinate direction. Thus, this optimization problem is unconstrained. The GA features a mutation scheme that exchanges single coordinates of two random experiments and a cycle cross-over where the parents are columns of coordinates of one coordinate direction.

3.2.3. Optimum Latin Hypercube Sampling for Constrained Design Spaces

Commonly OLHs are created by assuming a complete n-dimensional design space even though this might not reflect reality. This assumption causes experiments to be created that might not be feasible or do not have a physical meaning. This means for the creation of the DoE and the response surface, unnecessary computational costs. Simply omitting infeasible points leads to a DoE with fewer experiments than desired and to a set of points that is not optimally distributed. (Stinstra et al., 2003) developed a procedure for the creation of DoEs for constrained design spaces but here, the NC is not considered.

A novel procedure has been developed that allows for the creation of OLHs for arbitrarily constrained design spaces. The formulation by (Bates et al., 2003) and (Bates et al., 2004) discussed in Section 3.2.2 has been extended to meet this requirement. The OLH is found by minimizing the potential energy of the points according to (Audze & Eglais, 1977) using the selective permutation GA discussed in Section 2.3.5.3. The validity of the approach is demonstrated using 2D and 3D examples.

3.2.3.1. Formulation Outline

In the case of unconstrained design spaces the number of space divisions equals the number of experiments n_P. When infeasible regions in the design space are defined where no experiments are allowed to be placed, no initial combination of feasible points might be found that satisfies the NC and the constraints at the same time, especially if the ratio between infeasible and feasible points is considerable. That means no combination of n_P feasible experiments exists without at least two points sharing the same level, i.e. without violating the NC. A possible solution to this problem is the increase of the number of space divisions for certain dimensions of the design space while keeping n_P constant. This measure increases the number of possible combinations. A vector $\underline{n}_{Div}$ is introduced that contains the chosen number of space divisions for each design space dimension, i.e. it contains the information into how many levels each parameter is divided. For regular OLHs in the original formulation

$\underline{n}_{Div} = [n_P, n_P, \dots, n_P]$. In the developed procedure this requirement is relaxed. Effectively, this means that $n_{Div_i} - n_P$ levels for parameter i do not contain an experiment.

A common permutation GA, presented in Section 2.3.5.3, as it was used in (Bates et al., 2003) and (Bates et al., 2004) cannot effectively solve a permutation problem where not all available items are selected, but a certain number has to be left out. Thus, the selective permutation GA proposed in the same Section 2.3.5.3 will be employed instead.

Fitness Function

Without a correction of the potential energy in Equation 3.2, similar examples with different $\underline{n}_{Div}$ would exhibit different U even if the results would be identical. This is because point-to-point distances vary for different space divisions. For reasons of comparability the fitness function has been changed in a way that the distances between each coordinate direction are multiplied by a factor ξ that relates the actual ratio of n_P and n_{Div_i} to a ratio of 1 (see Equation 3.9). With this change, a similar result for different ratios yields the same potential energy. The equation for the amended potential energy U^* reads:

$$U^* = \sum_{p=1}^{n_P} \sum_{q=p+1}^{n_P} \frac{1}{{L^*_{pq}}^2}\,, \tag{3.7}$$

with

$$L^*_{pq} = \sqrt{\sum_{i=1}^{n_{Dim}} \left[\left(X_i^p - X_i^q\right) \cdot \xi_i\right]^2} \tag{3.8}$$

$$\xi_i = \frac{n_P - 1}{n_{Div_i} - 1} \tag{3.9}$$

Here, L^*_{pq} is the amended distance between points p and q and ξ_i is the factor that relates the point-to-point distances to those for a ratio of n_P and n_{Div_i} being 1. This is done for each dimension i individually.

Contrary to the problem faced in the original formulation, this one is no longer unconstrained. In accordance to Section 2.3.5.5, a penalty term is added to the fitness function that penalizes a violation of the constraints. For simplicity, a constant penalization term, which does not indicate the degree of constraint violation, is added. The full fitness function reads:

$$f = U^* + \frac{n_{inf}}{n_P} \cdot pen, \tag{3.10}$$

where n_{inf} is the number of experiments that violate the constraints and pen is a penalization factor. Numerical experiments showed that a value of 1000 for pen is a good working value to safely rule out any infeasible experiments.

Instead of minimizing the potential energy directly, as in the original formulation, the fitness function in Equation 3.10 is minimized instead. Furthermore, the optimization problem is now constrained by n_c inequality constraints g_k with which each of the n_P experiments has to be checked for feasibility. An experiment is considered feasible, if it satisfies each one of the n_c constraints. A design variable X_i^n (the i-coordinate of point n) can now take any integer value between 1 and n_{Div_i}. The new optimization problem reads:

$$min \rightarrow f \tag{3.11}$$

$$with \quad 1 \leq X_m^n \leq n_{Div_m} \quad for \quad m = 1, \dots, n_{Dim} \quad and \quad n = 1, \dots, n_P \tag{3.12}$$

$$X_m^p \neq X_m^q \quad for \quad m = 1, \dots, n_{Dim} \tag{3.13}$$

$$g_k(\underline{X}^n) \leq 0 \quad for \quad k = 1, \dots, n_c \tag{3.14}$$

Mating

As discussed in Section 2.3.5.3 the mating scheme in the selective permutation GA is the same as in the common permutation GA. Hence, with respect to the mating the newly developed formulation remains identical to that in (Bates et al., 2003) and (Bates et al., 2004) which is a typical cycle-crossover. Here, the parents are columns of coordinates of the same coordinate direction corresponding to two chromosomes in the population. Practically, two chromosomes are chosen randomly from the first half of the population which is sorted according to their fitness. The probability of being chosen decreases linearly from the first to the last chromosome. Then, for each coordinate direction separately, two vector entrances are exchanged. This means that the parents might temporarily contain a certain entrance twice. If this happens the exchange is repeated for the particular entrance until each value occurs only once.

Mutation

As discussed in Section 2.3.5.3 in the selective permutation GA a new 'Substitution Mutation' is introduced in addition to the existing 'Exchange Mutation'. The latter randomly chooses two experiments in one chromosome and performs a simple exchange of one coordinate in one coordinate direction. Because coordinates are merely exchanged, this scheme will never violate the NC if it has not been violated before. At the same time it means that, in case n_{Div_i} and n_P differ for a dimension i, an unoccupied level will not be filled by means of this mutation scheme.

The 'Substitution Mutation' allows moving an existing experiment onto an unused level. In detail, this scheme, which distinguishes the regular from the selective permutation GA, first randomly chooses an experiment and removes it from the chromosome. Now, for each dimension all unoccupied levels are identified and stored. Finally, a new experiment is created by randomly choosing one of the stored levels for each dimension.

The ‘Exchange Mutation’ is illustrated in Figure 3.1 and the ‘Substitution Mutation’ in Figure 3.2. Both are employed in the developed formulation.

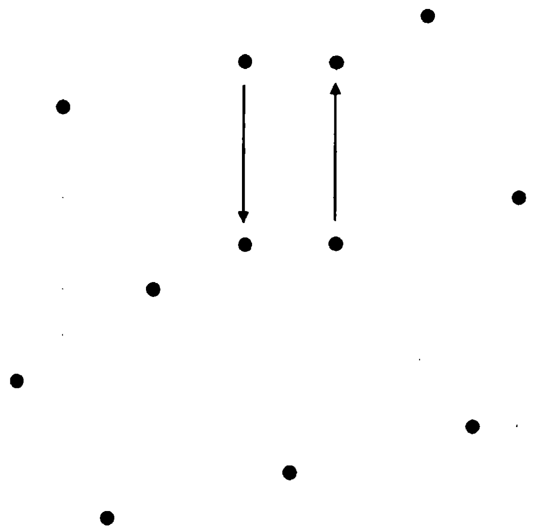


Figure 3.1: Illustration of the ‘Exchange Mutation’ scheme. One coordinate of two random experiments is exchanged. This scheme is used in the original formulation.

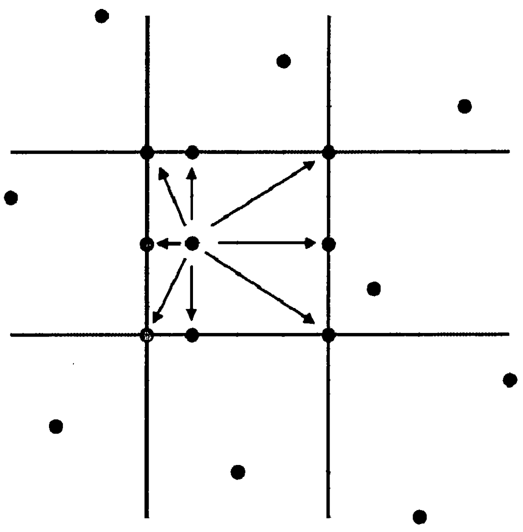


Figure 3.2: Illustration of the ‘Substitution Mutation’ scheme. An existing experiment is substituted by a new one lying on available space divisions.

Formulation Validation

In order to verify that the developed formulation is capable of creating OLHs for constrained design spaces, an unconstrained OLH obtained in (Bates et al., 2004) will be reproduced in a constrained context. The unconstrained OLH contains ten experiments in a 2D design space. For the constrained design space each design variable is defined infeasible in the first third of its interval. The two constraints used to define the infeasible region are:

$$g_1(\underline{X}^n) = 1 + (n_{Div_1} - 1) \cdot \frac{1}{3} - X_1^n \leq 0 \qquad 3.15$$

$$g_2(\underline{X}^n) = 1 + (n_{Div_2} - 1) \cdot \frac{1}{3} - X_2^n \leq 0 \qquad 3.16$$

The number of space divisions for each dimension is increased by 50%, i.e. $n_{Div} = [15,15]$, resulting in a feasible space that again contains ten divisions in both coordinate directions. As expected, the experiments are placed within the feasible space in the same order as in the original one. This shows that the developed formulation works as intended. The unconstrained OLH is shown in Figure 3.3 and its constrained counterpart in Figure 3.4.

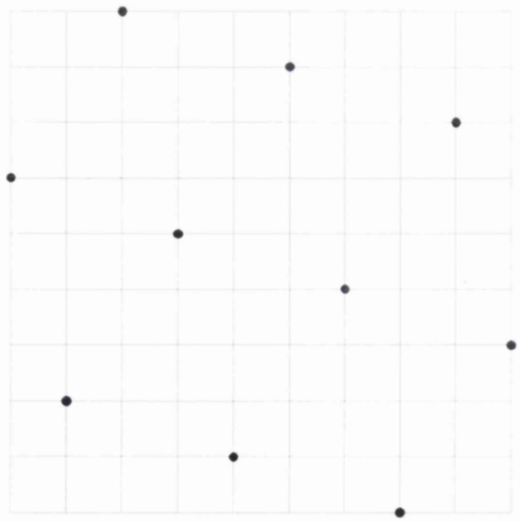


Figure 3.3: Unconstrained OLH for 10 points. Result found in (Bates et al., 2004). The potential energy is 2.07.

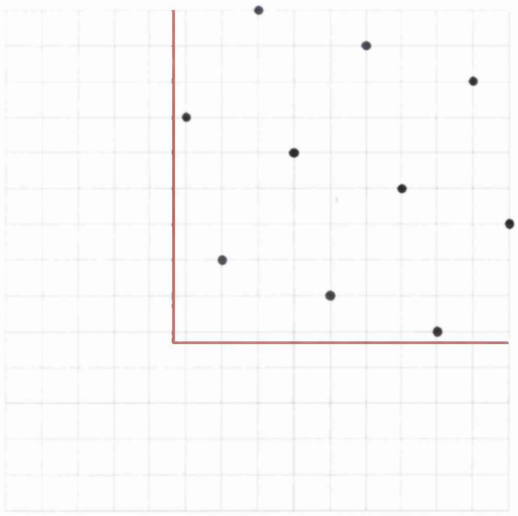


Figure 3.4: Constrained OLH for 10 points and $n_{Div} = [15,15]$. The potential energy is 5.0. The rectangular feasible space contains $n_{Div}^{feas} = [10,10]$.

3.2.3.2. Handling of Space Divisions

The general outline of the developed formulation has been discussed in the previous section. In the following the effect of the design space division into more than n_P levels will be discussed. Firstly, the effect of large infeasible ratios, i.e. a high increase in the entries in $\underline{n}_{Div}$ will be investigated and secondly, an automated procedure for the determination of the required $\underline{n}_{Div}$ presented.

Effect of Large Infeasible Areas

As mentioned before, the entries in $\underline{n}_{Div}$ have to be increased if due to the constraints insufficient possibilities exist to place feasible points without violating the NC. In general one can choose the minimum number of design space divisions $n_{Div\,i}^{min}$ for each dimension for which the algorithm is capable of finding an initial feasible population. In general, the lower the feasibility ratio becomes, i.e. the higher the number of infeasible grid points in the design space becomes, the more the value for $n_{Div\,i}^{min}$ exceeds that for n_P. The effect of that situation can be illustrated by conducting an unconstrained 2D example with $n_P = 25$ and $\underline{n}_{Div} = [1000,1000]$. The resulting LH is illustrated in Figure 3.5. It shows that the fine division of the design space yields a LH that resembles a chessboard. Even though it appears to be space-filling and uniform, it contradicts the basic idea of LH where each dimension stands for a parameter which should be evaluated for as many different values as possible. In this particular example each design variable is only evaluated in five significant locations, rather than 25, basically violating the NC. The same problem occurs for constrained cases with high $n_{Div\,i}^{min}$. Figure 3.6 shows an example where 90% of each design variable is defined infeasible. The constraint requires at least one of the coordinates of an experiment to lie within the first 10% of the design space which can be expressed as:

$$g_1(\underline{X}^n) = min\{X_1^n - 1 - (n_{Div_1} - 1) \cdot 0.1,\ X_2^n - 1 - (n_{Div_2} - 1) \cdot 0.1\} \leq 0 \qquad 3.17$$

The algorithm requires at least 242 space divisions for each dimension for the placement of 50 points. Here too, the space divisions become so close to each other that the NC is practically violated. This shortcoming of the procedure should be kept in mind when applied to problems that require very high $n_{Div\,i}^{min}$.

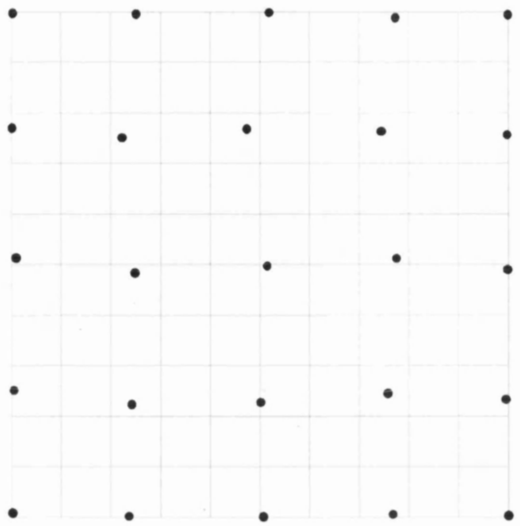


Figure 3.5: Unconstrained LH featuring 25 points and$n_{Div} = [1000{,}1000]$. The potential energy is 2.40.

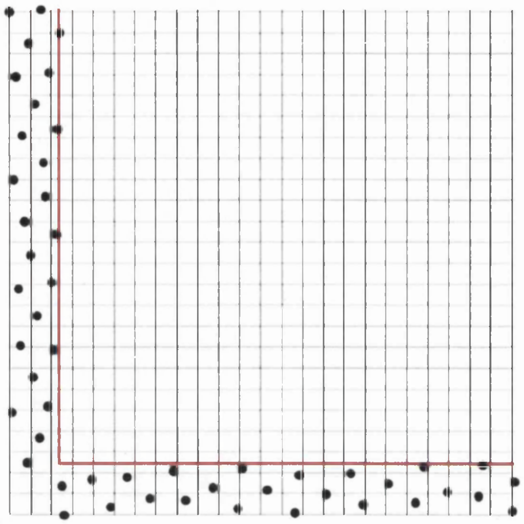


Figure 3.6: Constrained OLH featuring 50 points and $n_{Div} = [242{,}242]$. The potential energy is 12.05.

Automatic Determination of Space Divisions

Before the formulation for the generation of OLHs for constrained design spaces can be applied, the necessary design space divisions have to be determined. This is not always as straightforward as in the validation example in Section 3.2.3.1. When infeasible regions are spread irregularly or when the design space has more than three dimensions, this can become a difficult task. Nevertheless, as explained in Section 3.2.3.2 the number of space divisions plays a crucial role in satisfying the NC. Hence, the minimum required number $n_{Div\ i}^{min}$ should be used for each dimension in all applications. A procedure has been developed that allows for the automatic determination of $\underline{n}_{Div}$. This procedure finds $\underline{n}_{Div}$ by minimizing the following quality criteria:

$$min \qquad \sum_{i=1}^{n_{Dim}} \left(n_{Div_i} - n_P\right) + \sum_{i=1}^{n_{Dim}} \left(n_{Div_i} - n_P - avg\right)^2 \cdot 10^{-5} \qquad 3.18$$

$$with \qquad avg = \frac{1}{n_{Dim}} \cdot \sum_{i=1}^{n_{Dim}} \left(n_{Div_i} - n_P\right) \qquad 3.19$$

The first term in Equation 3.18 states that the sum of the differences between the space divisions and n_P should be as low as possible. The second term was added to distinguish between two options that exhibit the same sum but varying individual divisions. In this case the option where the divisions have a smaller standard deviation is the preferred one. In this sense, for instance, $n_{Div} = [55,55]$ would be preferred to $n_{Div} = [53,57]$.

The flow of the procedure for the minimization of Equation 3.18 is illustrated in Figure 3.7.

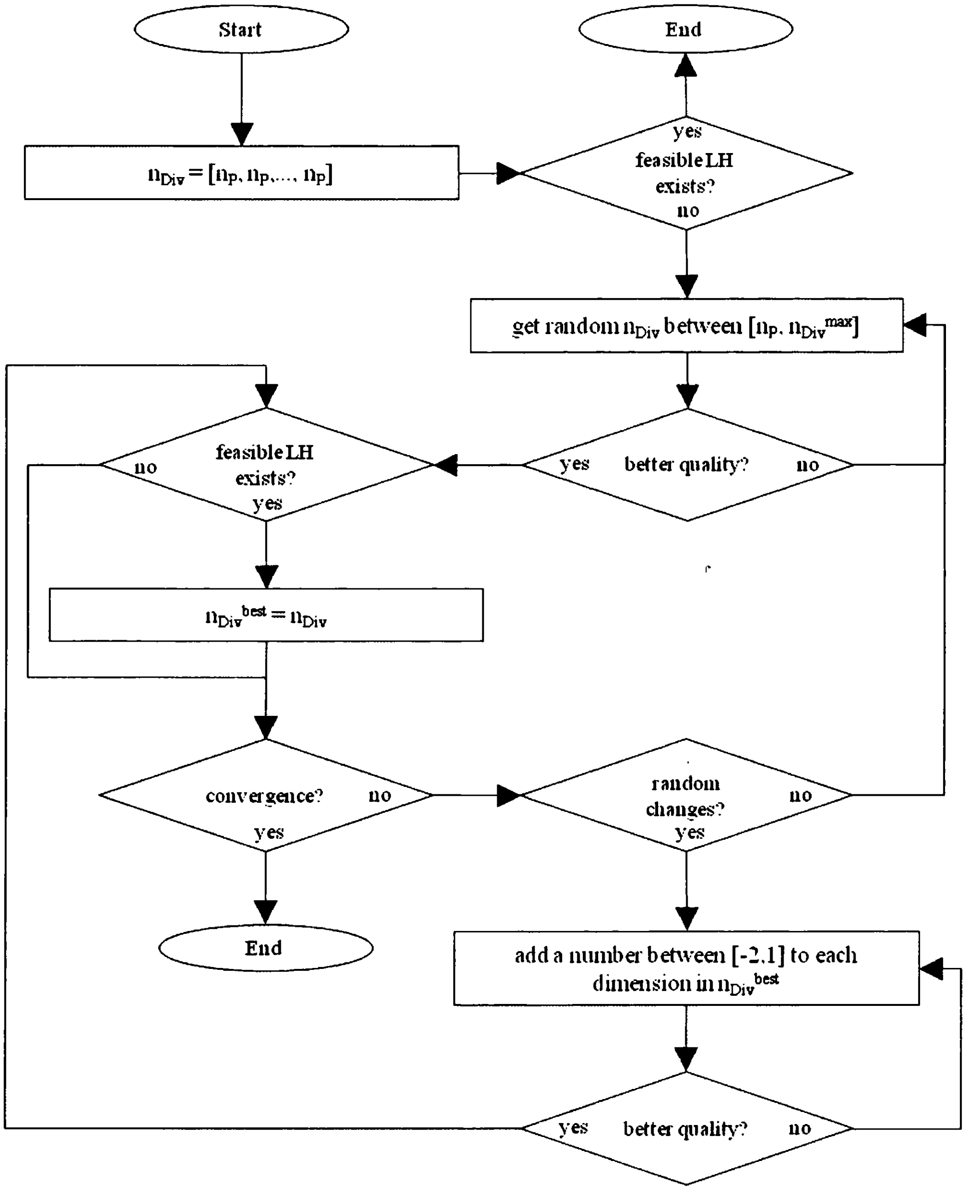


Figure 3.7: Flow of the procedure to automatically determine the necessary space divisions.

It can be described as follows: In a first step $\underline{n}_{Div} = [n_P, n_P, \ldots, n_P]$ is tried. If a feasible set of experiments can be found that satisfies the NC, the algorithm stops. If this is not the case a loop is started where the entries in $\underline{n}_{Div}$ are gradually decreased to find the minimum required set. To this end a maximum value is specified, typically $n_{Div}^{max} = 10 \cdot n_P$ is a safe assumption. Then, a set of n_{Dim} random numbers between n_P and n_{Div}^{max} is generated and assigned to $\underline{n}_{Div}$. This is repeated until this random $\underline{n}_{Div}$ features a better quality than the currently best $\underline{n}_{Div}^{best}$. Once this is the case, it is attempted to find a feasible set of experiments that satisfies the NC for the corresponding $\underline{n}_{Div}$ and, if successful, the currently best $\underline{n}_{Div}^{best}$ is updated accordingly.

In either case a convergence check is performed and the process terminates or progresses depending on its result. To speed up convergence another source of input has been added in the next step. There, the $\underline{n}_{Div}^{best}$ is taken as a starting point and random numbers between $[-2,1]$ are added to each dimension. Again, this is repeated until the quality exceeds that of $\underline{n}_{Div}^{best}$ and jumps to the location where a feasible set of experiments is sought. The probability of this feature being used increases during the process as $\underline{n}_{Div}^{best}$ approaches the best possible combination. If the process determines not to perform these random changes, it carries on by choosing $\underline{n}_{Div}$ randomly between n_P and n_{Div}^{max}. A convergence check is successful once $100 \cdot n_{Dim}$ attempts to find a feasible initial set of experiments failed successively.

3.2.3.3. Results

Several 2D and 3D example problems have been solved to show the capabilities of the developed procedure. The results will be discussed in the following.

2D Examples with Similar Space Divisions

In the following, four 2D examples will be presented where infeasible regions are defined in the design space in a way that both dimensions are affected to the same extent, i.e. the required $\underline{n}_{Div}$ will contain similar entries.

Example 1: This 2D example requires the placement of 28 points in a design space where all experiments are considered to be infeasible if lying outside a quarter circle of radius equal to the length of one coordinate direction. One constraint is used in this example. It describes the quarter circle of radius R and midpoint $\underline{M}$. This constraint can be formulated as:

$$g_1(\underline{X}^n) = \sqrt{\sum_{i=1}^{2}(M_i - X_i^n)^2} - R \leq 0 \qquad 3.20$$

$$M_1 = M_2 = 1, \qquad R = \frac{n_{Div_1} + n_{Div_2} - 2}{2} \tag{3.21}$$

The algorithm found $\underline{n}_{Div} = [29,29]$. The resulting OLH is shown in Figure 3.8. The potential energy equals to 4.08 and was obtained after 3053 generations.

It can be seen that the experiment density is slightly lower near the centre of the quarter circle. This is due to the fact that the number of feasible grid-points is higher in that area and thus, when projected onto one of the two axes, more points need to lie on that line. Since the NC allows only one experiment per division, the point-to-point distances must increase towards the centre. The other extreme can be found near the ends of the axes. Here, only one feasible grid-point is available when projected onto the other dimension, hence the chance of the level being occupied already is lower. This shortcoming of the procedure could be diminished by employing non-equidistant space divisions with larger spacing towards the constraint boundaries and vice versa, but this measure exceeds the scope of the present work. Nevertheless, the uniformity of the placed points can be considered good and naturally better than that of the feasible experiments in an OLH that has been created for the entire design space.

Example 2: This 2D example has a circular infeasible region in the middle of the design space with a radius of a quarter of the length of one coordinate direction.

The used constraint reads:

$$g_1(\underline{X}^n) = R - \sqrt{\sum_{i=1}^{2} (M_i - X_i^n)^2} \leq 0 \tag{3.22}$$

$$M_1 = 1 + \frac{n_{Div_1} - 1}{2}, \qquad M_2 = 1 + \frac{n_{Div_2} - 1}{2}, \qquad R = \frac{n_{Div_1} + n_{Div_2} - 2}{8} \tag{3.23}$$

15 experiments need to be created and $\underline{n}_{Div} = [15,15]$ was found to be sufficient to obtain an initial feasible population. The resulting OLH is shown in Figure 3.9. The potential energy equals to 2.68 and was obtained after 119 generations.

The experiments are uniformly spread around the circle. Their diagonal arrangement resembles that of the validation example in Section 3.2. A small decrease in point-to-point distances can be found towards the constraint boundary. The explanation for this

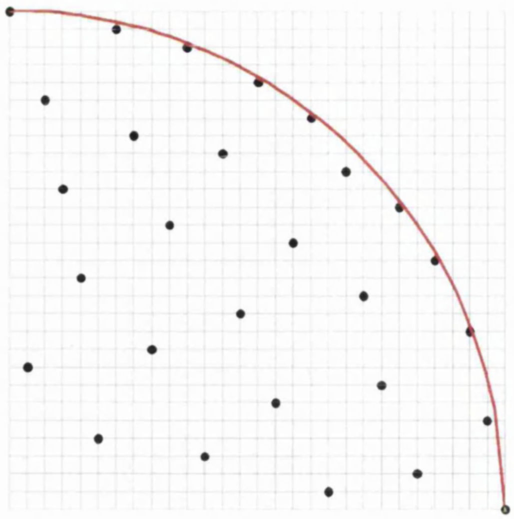


Figure 3.8: Example 1. Constrained OLH for 28 points and $nDiv = [29{,}29]$ divisions. The potential energy is 4.08.

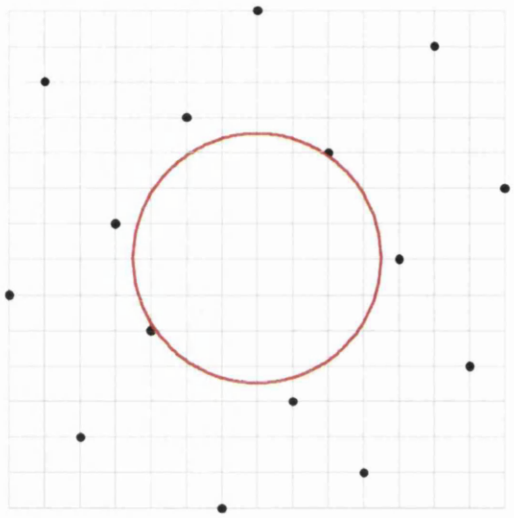


Figure 3.9: Example 2. Constrained OLH for 15 points and $nDiv = [15{,}15]$. The potential energy is 2.68.

is similar to that in Example 1. Due to the lower n_P and the fact that, in no region of the design space, the constraint affects one dimension only, the overall uniformity is better.

Examples 3 and 4: These 2D examples are of similar nature, but they differ in the size of the infeasible region. In both cases this region is rectangular and located in the corner of the design space. Their side lengths are 1/3 and 2/3 of the length of one coordinate direction, respectively. The constraint used in these examples reads:

$$g_1(\underline{X}^n) = min\{X_1^n - 1 - (n_{Div_1} - 1)\cdot\xi,\ X_2^n - 1 - (n_{Div_2} - 1)\cdot\xi\} \leq 0 \qquad 3.24$$

where ξ is 2/3 for Example 3 and 1/3 for Example 4. This difference has a significant impact on $\underline{n}_{Div}$ which is, for the placement of 50 experiments, $\underline{n}_{Div} = [50{,}50]$ in the first and $\underline{n}_{Div} = [77{,}77]$ in the second case. The resulting OLHs are shown in Figure 3.10 and Figure 3.11. The potential energies equal to 4.68 and 6.71 and were obtained after 1353 and 1521 generations, respectively.

It can be seen that in both cases the density of the placed experiments is lower in those regions where the divisions projected onto each dimension are all feasible. This is due to the fact that, despite the existence of more feasible grid-points in this area, the NC prevents the algorithm from placing more experiments. In areas where one of the coordinate directions is affected by the constraints, more space-divisions can be used in the other direction. Example 1 is very similar to this one. The difference is that the influence of the constraint on a dimension does not decrease gradually as one progresses

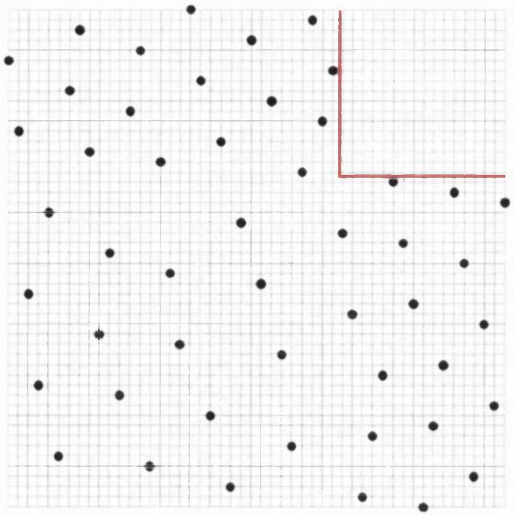


Figure 3.10: Example 3. Constrained OLH for 50 points and $nDiv = [50,50]$. The potential energy is 4.68.

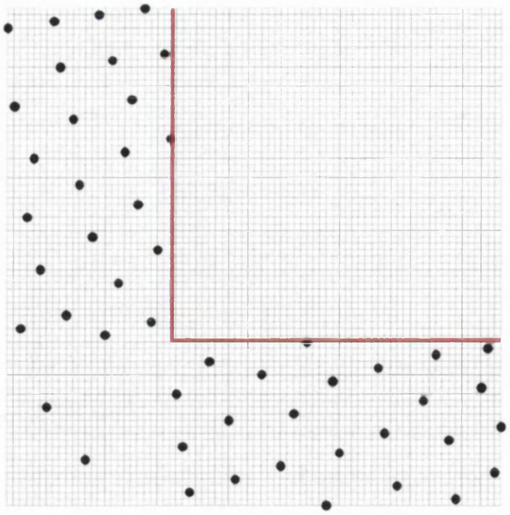


Figure 3.11: Example 4. Constrained OLH for 50 points and $nDiv = [77,77]$. The potential energy is 6.71.

along the axis, but discontinuously. Hence, the point-to-point distances drop at the location on the axis as the constraint takes effect. In Example 4 this becomes very apparent and the bottom left corner is only scarcely sampled. Again, this shortcoming of the procedure could be diminished by employing non-equidistant space divisions. The development of the potential energy throughout the optimization process for the latter four examples is illustrated in Figure 3.12.

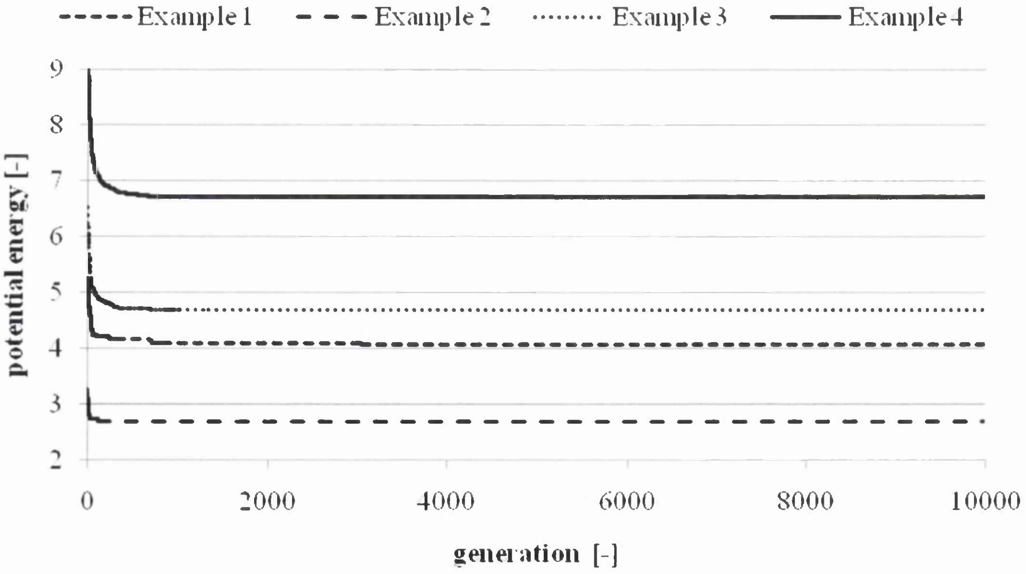


Figure 3.12: Development of the potential energy throughout the optimization process for the examples 1, 2, 3 and 4.

2D Examples with Different Space Divisions

In the following three 2D examples will be presented where infeasible regions are defined in the design space in a way that both dimensions are not affected to the same extent, i.e. the required $\underline{n}_{Div}$ will have entries of different magnitude.

Example 5: This 2D example requires the placement of 80 points in a design space where all experiments are considered to be infeasible if lying inside a semi-circle of radius equal to the length of half of the *1*-direction. The constraint used in the example reads:

$$g_1(\underline{X}^n) = R - \sqrt{\sum_{i=1}^{2} (M_i - X_i^n)^2} \leq 0 \qquad 3.25$$

$$M_1 = 1 + \frac{n_{Div_1} - 1}{2}, \qquad M_2 = n_{Div_2}, \qquad R = \frac{n_{Div_1} + n_{Div_2} - 2}{2} \qquad 3.26$$

The algorithm found $\underline{n}_{Div} = [80{,}103]$. The first entry of $\underline{n}_{Div}$ reflects that the design space projected onto the *1*-direction is not affected by the infeasible region. The resulting OLH is shown in Figure 3.13. The potential energy equals to 6.57 and was obtained after 1948 generations.

It can be seen that the experiment density is lower in the area of the beginning and the end of the *1*-direction and higher in between. This is due to the fact that the infeasible region expands in the middle of the *1*-direction. Even though the number of feasible grid points increases towards the beginning and the end of the *1*-direction, the NC prevents the algorithm from placing more points in these regions. The projection of the points onto the *2*-direction shows a good space-filling quality with fewer points in the higher constrained region but the one onto the *1*-direction shows the opposite. That means the uniformity for the unaffected direction is worse than for the affected one. Similarly to the previous examples, this shortcoming could be diminished by employing non-equidistant space divisions.

Example 6: This 2D example requires the placement of 50 points in the design space. The infeasible region is a combination of that in Examples 3 and 4. Experiments are considered to be infeasible if lying inside a rectangle having a width of 1/3 of the *1*-direction and a height of 2/3 of the *2*-direction.

The constraint reads:

$$g_1(\underline{X}^n) = min\left\{X_1^n - 1 - \left(n_{Div_1} - 1\right) \cdot \frac{2}{3},\ X_2^n - 1 - \left(n_{Div_2} - 1\right) \cdot \frac{1}{3}\right\} \leq 0 \qquad 3.27$$

The algorithm found $\underline{n}_{Div} = [59{,}50]$. The resulting OLH is shown in Figure 3.14. The potential energy equals to 5.38 and was obtained after 8858 generations.

The uniformity of the point distribution shows the same problem as in example 4 with a lower density of the placed experiments in those regions where the divisions projected onto each dimension are all feasible. The severity of this problem is lower though since in this case the two directions are both affected to some extent which is reflected in an $\underline{n}_{Div}$ with entries that are less far apart.

Example 7: This 2D example requires the placement of 50 points in a design space where all experiments are considered to be infeasible if lying inside a triangle with side lengths of 70% of each coordinate direction. Due to the orientation of the triangle, the *2*-direction is more affected than the *1*-direction. The used constraint reads:

$$g_1(\underline{X}^n) = min\left\{X_1^n - 1 - \left(n_{Div_m} - 1\right) \cdot \frac{2}{3}, \frac{X_2^n - 1}{X_1^n - 1} - \frac{n_{Div_2} - 1}{n_{Div_1} - 1}\right\} \leq 0 \qquad 3.28$$

The algorithm found $\underline{n}_{Div} = [52{,}53]$. The resulting OLH is shown in Figure 3.15. The potential energy equals to 5.12 and was obtained after 1106 generations. The entries of $\underline{n}_{Div}$ show that the degree by which the two directions are affected by the constraints are fairly similar.

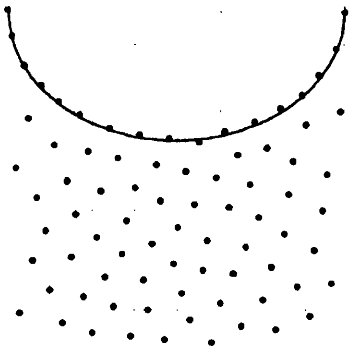


Figure 3.13: Example 5. Constrained OLH for 80 points and $\underline{n}_{Div} = [80{,}103]$. The potential energy is 6.57.

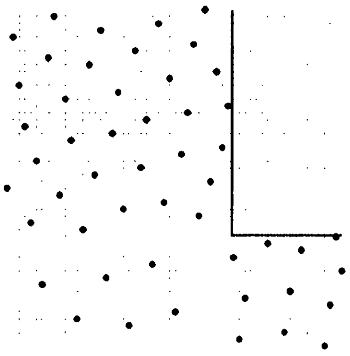


Figure 3.14: Example 6. Constrained OLH for 50 points and $\underline{n}_{Div} = [59{,}50]$. The potential energy is 5.38.

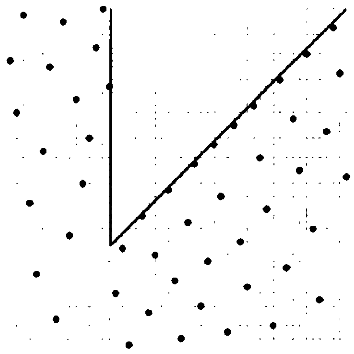


Figure 3.15: Example 7. Constrained OLH for 50 points and $\underline{n}_{Div} = [52{,}53]$. The potential energy is 5.12.

In the sense of the findings of previous two examples, the uniformity of the projected experiments can be considered better than in those. This example shows nicely how the point density decreases as the number of feasible grid-points increases, i.e. with a drop at $\frac{1}{3}$ of the I-dimension and gradually towards the end of the same axis.

The development of the potential energy throughout the optimization process for the latter three examples is illustrated in Figure 3.16.

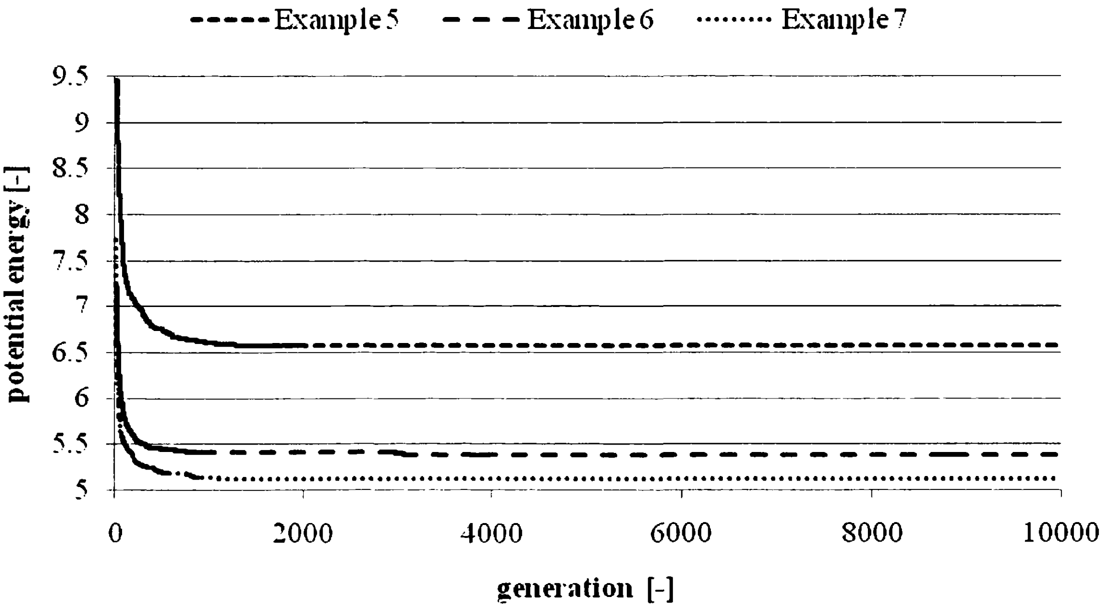


Figure 3.16: Development of the potential energy throughout the optimization process for the examples 5, 6 and 7.

2D Examples with Several Feasible or Infeasible Regions

In the following, two 2D examples will be presented where several unconnected infeasible or feasible regions are defined in each problem. Both examples are solved for one feasible region with several infeasible regions and vice versa.

Example 8: This 2D example contains three circular infeasible regions. Two quarter circles with a radius equal to a third of the average length of the two coordinate directions are placed in the upper left and lower right of the design space. A full circle of radius equal to a tenth of the average length of the two coordinate directions is placed in the middle of the design space. In addition to this case, a second case will be considered where the feasible regions are redefined as infeasible and vice versa. The three constraints used in the former case read:

$$g_k(\underline{X}^n) = R^k - \sqrt{\sum_{i=1}^{2}\left(M_i^k - X_i^n\right)^2} \leq 0, \qquad for\ k = 1,2,3 \tag{3.29}$$

$$R^1 = R^3 = \frac{n_{Div_1} + n_{Div_2} - 2}{6}, \qquad R^2 = \frac{n_{Div_1} + n_{Div_2} - 2}{20} \tag{3.30}$$

$$M_1^1 = 1, \qquad M_1^2 = 1 + \frac{n_{Div_1} - 1}{2}, \qquad M_1^3 = n_{Div_1} \tag{3.31}$$

$$M_2^1 = n_{Div_2}, \qquad M_2^2 = 1 + \frac{n_{Div_2} - 1}{2}, \qquad M_2^3 = 1 \tag{3.32}$$

In the inverted case where the previously infeasible regions are defined feasible, only one constraint is necessary. The three midpoints $\underline{M}^k$ and radii R^k remain identical to Equations 3.30, 3.31 and 3.32. The constraint reads:

$$g_1(\underline{X}^n) = min\left\{\sqrt{\sum_{i=1}^{2}\left(M_i^k - X_i^n\right)^2} - R^k, \qquad for\ k = 1,2,3\right\} \leq 0 \tag{3.33}$$

The number of points to be placed is chosen as 15. In the case of three infeasible regions the algorithm found $\underline{n}_{Div} = [15,15]$. The resulting OLH is shown in Figure 3.17. The potential energy equals to 2.75 and was obtained after 1772 generations. For the case of the inverted design space $\underline{n}_{Div} = [16,17]$. The corresponding OLH is illustrated in Figure 3.18. The potential energy equals to 6.25 and was obtained after 39 generations.

In both cases the uniformity of the experiments is good. The diagonal arrangement of the points in the first case is similar to that in the validation problem. In the inverted case it can be seen that the symmetrical feasible regions are sampled in a symmetrical way. Only in the middle region the lack of feasible grid points forces the algorithm to choose a slightly non-symmetrical pattern.

Example 9: This 2D example consists of four infeasible regions which are of circular shape. These regions have only a minor impact on the design space so that $\underline{n}_{Div} = [n_P, n_P]$. The OLH for 30 experiments is shown in Figure 3.19. The potential energy equals to 3.82 and was obtained after 416 generations. Again, the same example has been solved with an inverted design space, i.e. the former feasible space becomes infeasible and vice versa, which results in $\underline{n}_{Div} = [38,43]$. The corresponding OLH is

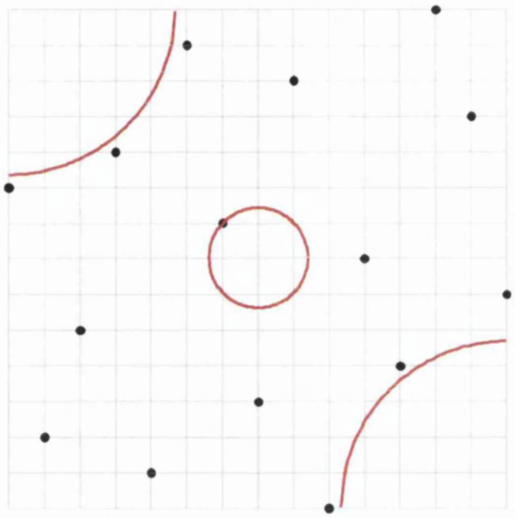


Figure 3.17: Example 8. Constrained OLH for 15 points and $n_{Div} = [15,15]$. The potential energy is 2.75.

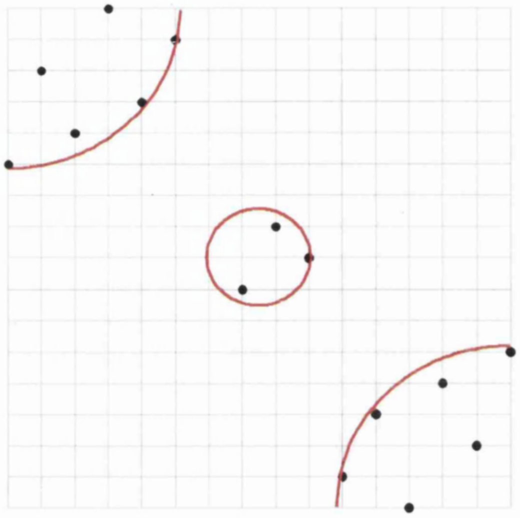


Figure 3.18: Example 8 (inverted). Constrained OLH for 15 points and $n_{Div} = [16,17]$. The potential energy is 6.25.

illustrated in Figure 3.20. The potential energy equals to 8.75 and was obtained after 7671 generations.

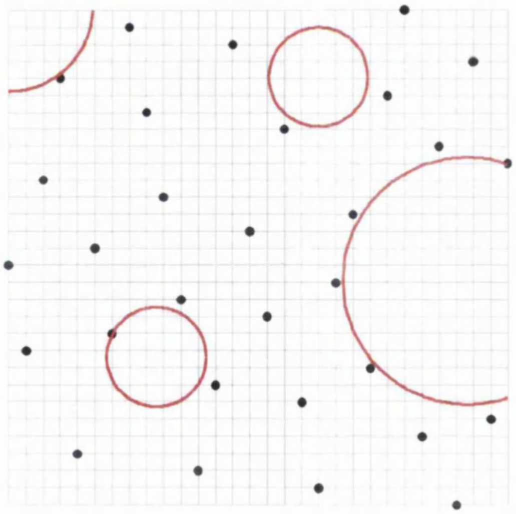


Figure 3.19: Example 9. Constrained OLH for 30 points and $n_{Div} = [30,30]$. The potential energy is 3.82.

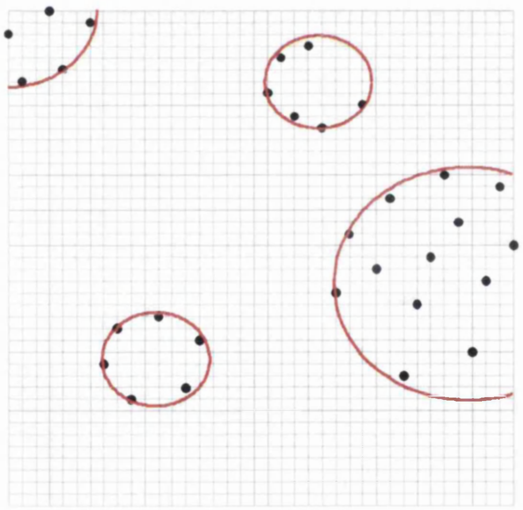


Figure 3.20: Example 9 (inverted). Constrained OLH for 30 points and $n_{Div} = [38,43]$. The potential energy is 8.75.

The development of the potential energy throughout the optimization process for the latter two examples is illustrated in Figure 3.21.

Engineering Design Example

Example 10: This example demonstrates the creation of an OLH for the optimization of a tip loaded cantilever beam made from mild steel, shown in Figure 3.22. The two

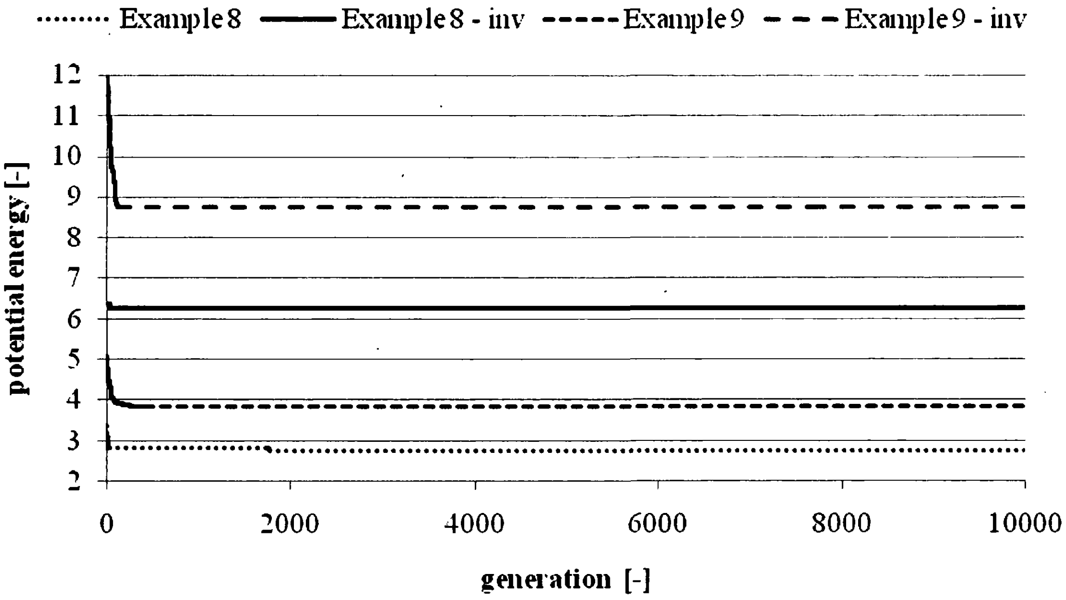


Figure 3.21: Development of the potential energy throughout the optimization process for the examples 8 and 9.

design variables are the width w and the height h of the beam's cross-section, subject to the following constraints:

maximum tip deflection

$$\delta_{max} = \frac{F \cdot l^3}{3 \cdot E \cdot I_y} \leq 5mm \tag{3.34}$$

maximum allowable bending stress

$$\sigma_b = \frac{6 \cdot F \cdot l}{w \cdot h^2} \leq \sigma_y \tag{3.35}$$

maximum allowable shear stress

$$\tau = \frac{3 \cdot F}{2 \cdot w \cdot h} \leq 0.5 \cdot \sigma_y \tag{3.36}$$

maximum height to width ratio

$$\frac{h}{w} \leq 10 \tag{3.37}$$

failure force for twist buckling

$$F_{crit} = \frac{4}{l^2} \cdot \sqrt{\frac{G \cdot I_T \cdot E \cdot I_z}{1 - \upsilon^2}} \leq 2 \cdot F \tag{3.38}$$

boundaries for width

$$5mm < w < 50\,mm \tag{3.39}$$

boundaries for height

$$20mm < h < 250\,mm \tag{3.40}$$

with $$\sigma_y = 240\ MPa \tag{3.41}$$ $$E = 216.62\ GPa \tag{3.42}$$

$$G = 86.65\ GPa \tag{3.43}$$ $$\upsilon = 0.27 \tag{3.44}$$

$$I_y = \frac{w \cdot h^3}{12} \tag{3.45}$$ $$I_z = \frac{h \cdot w^3}{12} \tag{3.46}$$

$$I_T = \frac{w \cdot h^3 + h \cdot w^3}{12} \tag{3.47}$$

20 experiments are to be placed in the constrained design space. The algorithm found $\underline{n}_{Div} = [28,32]$. The resulting OLH and the feasible design space are shown in Figure 3.23. It was obtained after 13140 generations and the corresponding potential energy amounts to 7.08.

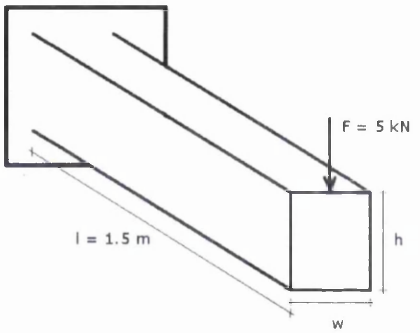


Figure 3.22: Example 10. Tip loaded cantilever beam. The design variables are the width and the height of the cross-section.

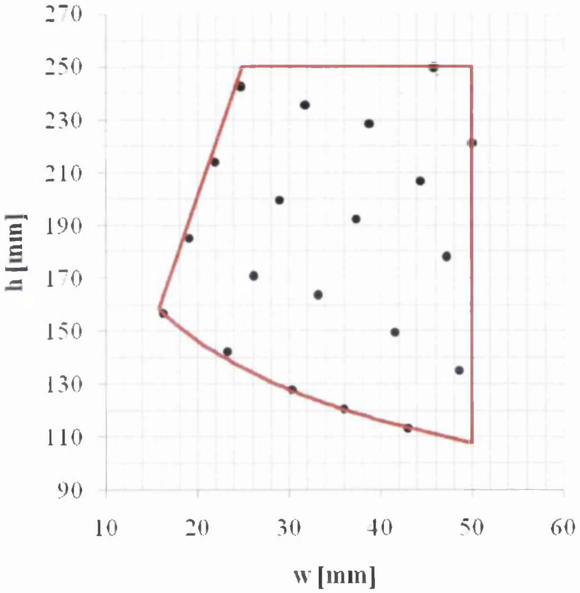


Figure 3.23: Example 10. Constrained OLH for 20 points and $\underline{n}_{Div} = [28,32]$. The potential energy is 7.08.

From an engineering point of view it might be desirable to place experiments in the corners of the feasible region of the design space. For a 2D design space as in the current example this is a seemingly simple task since it is easy to illustrate. For more dimensions though, a search algorithm which finds the corner points automatically would be necessary. This exceeds the scope of the present work.

The development of the potential energy throughout the optimization process for this example is illustrated in Figure 3.24.

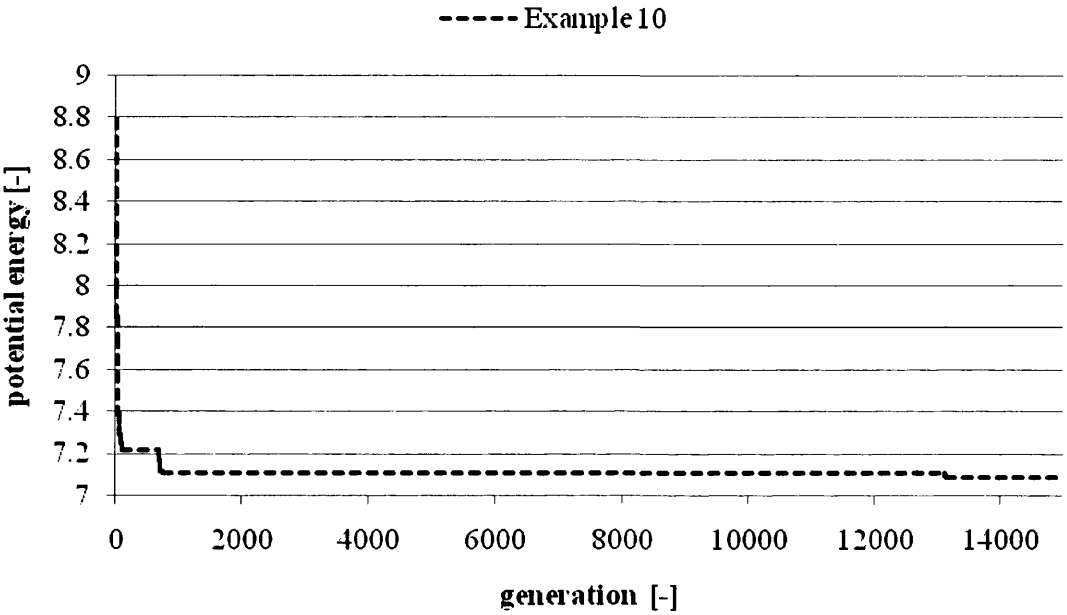


Figure 3.24: Development of the potential energy throughout the optimization process for the example 10.

Handling of Equality Constraints

Example 11: This 2D example illustrates how to deal with design spaces made up of equality constraints, i.e. design spaces with a very low feasibility ratio. The region boundaries are identical to those in example 8 but with experiments only allowed to be placed on the constraints. In this example $n_P = 30$. Contrary to inequality constraints, in this case a certain tolerance has to be defined for which an experiment is feasible, since it is very unlikely that the space division intersections perfectly intersect with the constraint boundaries. A tolerance of $10^{-3} \cdot n_{Div}$ was found to be a good working value.

Because of the nature of the problem, the entries in $\underline{n}_{Div}$ become very high. The algorithm found $\underline{n}_{Div}$ = [106,106]. Despite the high values in this case, the feasible points are still sparse and hence their final distribution far from uniform. Hence, the influence of a manual increase of the space divisions will be demonstrated by additionally using $\underline{n}_{Div}$ = [150,150] and $\underline{n}_{Div}$ = [200,200]. Figure 3.25, Figure 3.26 and Figure 3.27 show all feasible experiments for the three cases. The corresponding OLHs are shown in Figure 3.28, Figure 3.29 and Figure 3.30. Their final potential energies are

21.10, 10.08 and 9.3. They were obtained after 50, 864 and 4773 generations. Obviously, the OLH for 200 space divisions appears to have the highest space-filling quality, because the point to point distances are very uniform. Nevertheless, as mentioned before, the higher the difference between n_P and n_{Div_i} for each dimension i is, the more the NC is practically violated.

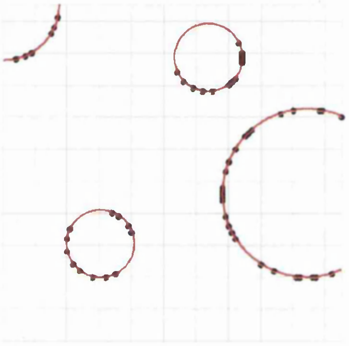


Figure 3.25: Example 11. Feasible points for $\underline{n}_{Div}$ = [106,106]. There are 58 feasible points.

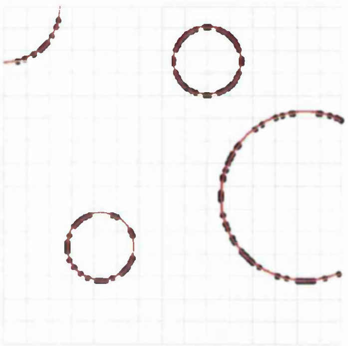


Figure 3.26: Example 11. Feasible points for $\underline{n}_{Div}$ = [150,150]. There are 133 feasible points.

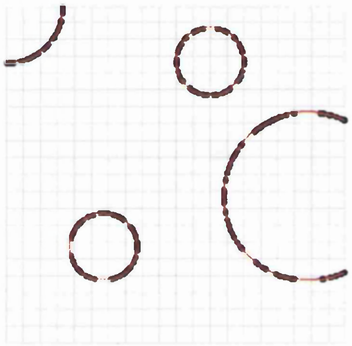


Figure 3.27: Example 11. Feasible points for $\underline{n}_{Div}$ = [200,200]. There are 200 feasible points.

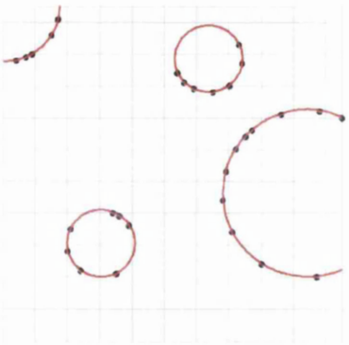


Figure 3.28: Example 11. Constrained OLH for 30 points and $\underline{n}_{Div}$ = [106,106]. The potential energy is 21.10.

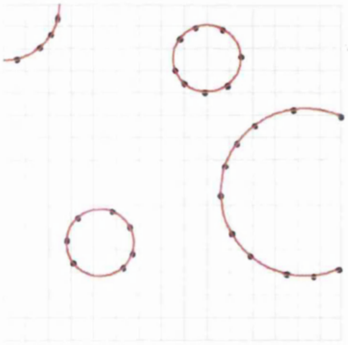


Figure 3.29: Example 11. Constrained OLH for 30 points and $\underline{n}_{Div}$ = [150,150]. The potential energy is 10.08.

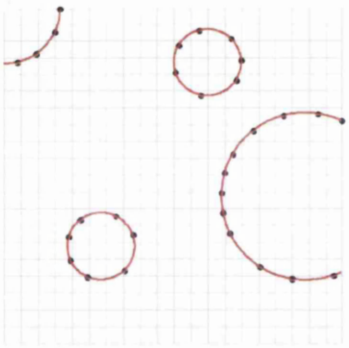


Figure 3.30: Example 11. Constrained OLH for 30 points and $\underline{n}_{Div}$ = [200,200]. The potential energy is 9.30.

The development of the potential energy throughout the optimization process for the three settings in this example is illustrated in Figure 3.31.

3D Results

Example 12: In order to show the applicability of the developed formulation to constrained design spaces of more than two dimensions a simple 3D example will be

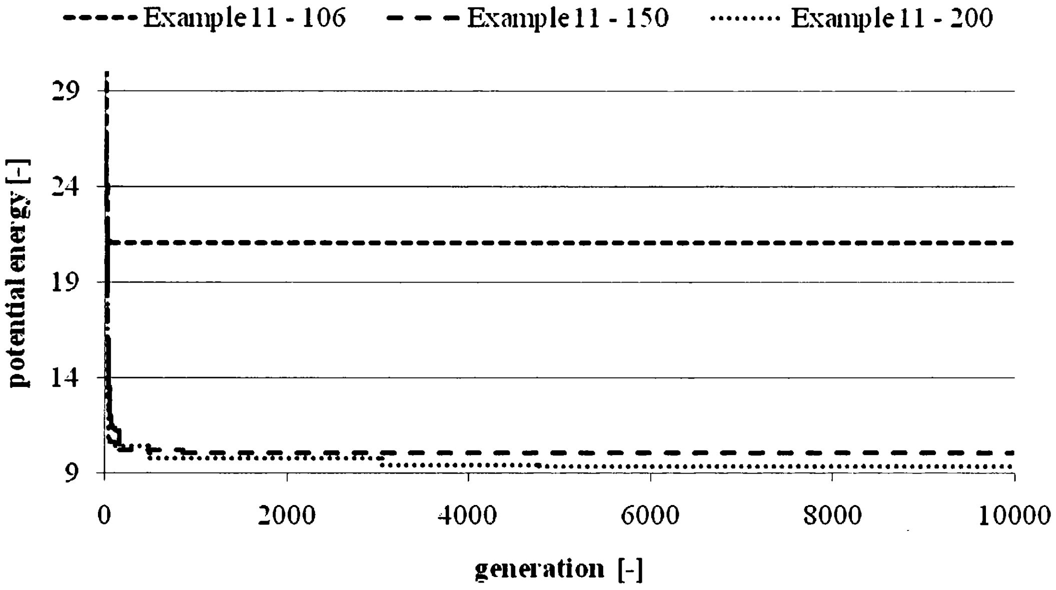


Figure 3.31: Development of the potential energy throughout the optimization process for the example 11.

solved. It requires the placement of 50 points. All experiments lying in the middle third of the first parameter are considered to be infeasible. The algorithm found $\underline{n}_{Div}$ = [75,50,50]. The resulting OLH is shown in Figure 3.32. The potential energy equals to 1.43 and was obtained after 4775 generations.

The development of the potential energy throughout the optimization process for this example is illustrated in Figure 3.33.

3.3. Design Space Approximation

As discussed before, in Surrogate Based optimization the original computationally expensive simulation code is replaced by a low cost surrogate model. To this end a small number of function evaluations are performed and the obtained results are used to train the surrogate model. Then the surrogate model can be evaluated at arbitrary locations, efficiently approximating the simulation code. A vast variety of approaches to design space approximation exist. A possible classification of popular methods can be found in Figure 3.34. (Keane & Nair, 2005) distinguish two main types: General and Physics-based approximation concepts. While methods in the former group are somewhat generally applicable, those in the latter group exploit available knowledge about the physical equations and the employed numerical methods in question. This makes them more customized and less general. For more information about Physics-

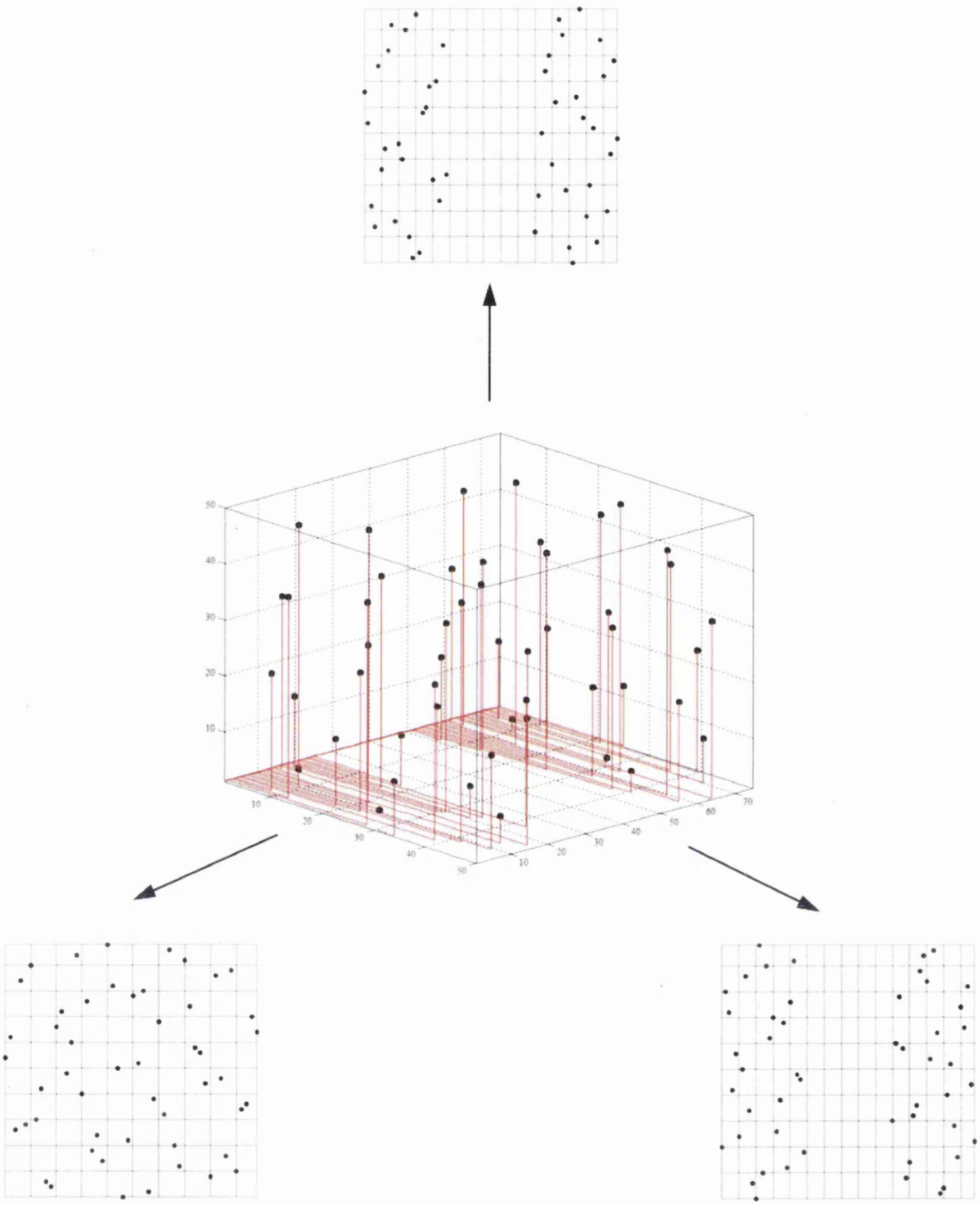


Figure 3.32: Example 12. Constrained OLH for 50 points and $nDiv = [75,50,50]$. The potential energy is 1.62.

based approximation techniques see (Keane & Nair, 2005). According to (Barthelemy & Haftka, 1993) the general approximation concepts can be split further into local, global and mid-range approximations. This distinction is based on their validity range within the design space. This chapter gives a brief overview of common approximation concepts in the three groups.

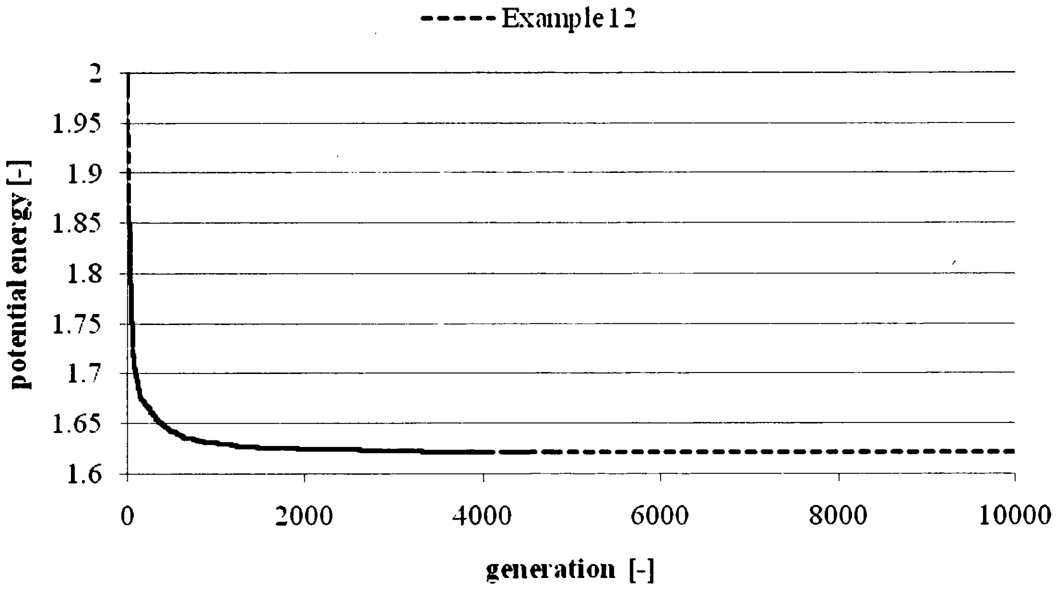


Figure 3.33: Development of the potential energy throughout the optimization process for the Example 12.

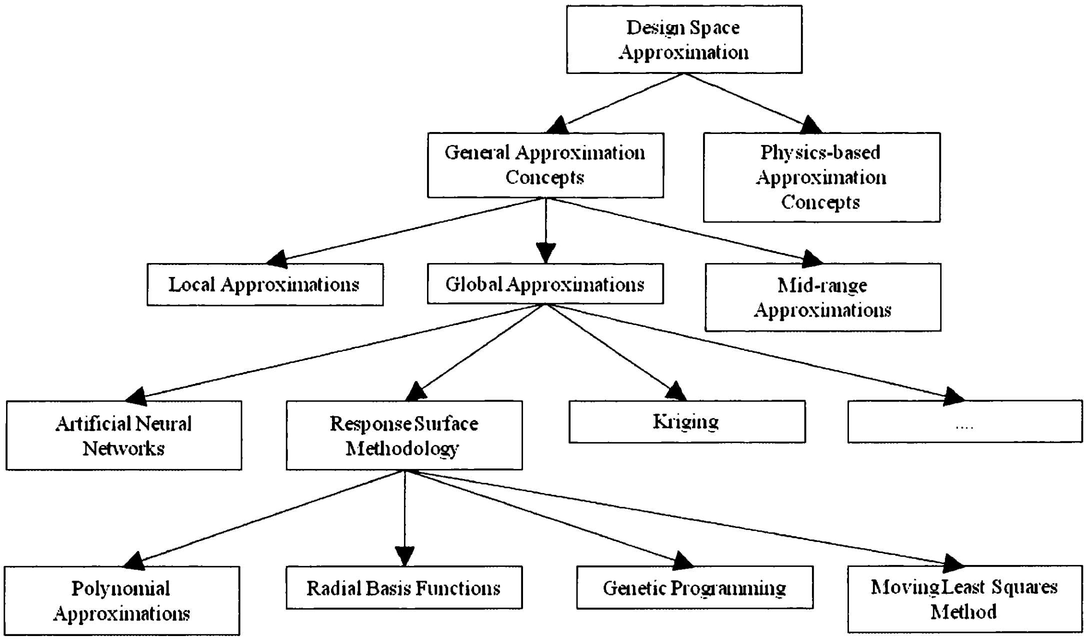


Figure 3.34: Categorization of design space approximation methods.

3.3.1. Local Approximations

Local approximations yield valid results only in the vicinity of the point which was used to generate it. These methods use variations on the Taylor series expansions for approximation. Thus, they usually require sensitivity information. Several methods such as first order, reciprocal, conservative or higher order approximations exist. For more information the reader is referred to (Barthelemy & Haftka, 1993). (Keane & Nair, 2005) also discuss how Taylor series approximations can be improved by the use of intervening variables. Another popular method is the convex linearization method (CONLIN).

3.3.2. Global Approximations

Global Approximation methods create a single approximation model for the entire design space. The advantage of this is that it is more likely that a global optimum is found. For high numbers of design variables though this approach can become very costly due to a high number of required plan points. In those cases a mid-range technique can be the preferable option. Popular global approximation techniques are Artificial Neural Networks (ANN), Kriging methods, Polynomial approximations (PA), radial basis functions (RBF), genetic programming (GP) or moving least squares (MLS) method. The latter four belong to the Response Surface methodology (RSM) group.

3.3.2.1. Artificial Neural Networks

This brief introduction to ANN is done in accordance with (Keane & Nair, 2005). ANN are based on an idealized brain structure where the neurons are represented by multiple connected elements. The most widely used version is the feed-forward system having an input, a hidden and an output layer. This system is illustrated in Figure 3.35.

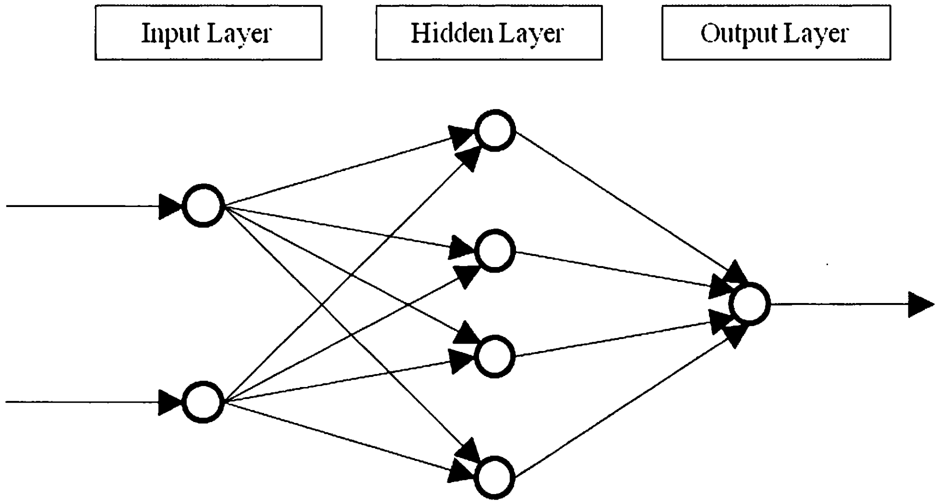


Figure 3.35: A feed-forward system with an input, a hidden and an output layer.

This model corresponds to the mathematical form in Equations 3.48 and 3.49 .

$$\overline{F}(\underline{x}) = \sum_{i=1}^{m} \alpha_i \cdot \varphi(a_i) \qquad 3.48$$

$$a_i = \sum_{j=1}^{p} w_{ij} \cdot x_j + \beta_j \qquad 3.49$$

Here, α_i, w_{ij} and β_i are the unknown parameters and $\varphi(\underline{x})$ a transfer function for instance the $tanh$ function. Furthermore, p and m are the numbers of neurons in the input and hidden layer, respectively. More information can be found in (Bishop, 1995).

3.3.2.2. Response Surface Methodology

Originally, RSM has been developed for the modeling of experimental results (Box & Draper, 1987) and was later utilized for numerical experiments. Experimental results usually exhibit a random error which occurs for instance due to the measurements or imprecise experiment set-up. As mentioned before, in computer simulations noise can occur due to incomplete convergence, round-off errors or the discretization of physical phenomena (Alvarez, 2000). In this case the noise is denoted as numerical noise. An RSM model interpolates between the results to reduce the impact of the noise and improve its smoothness, by minimizing the sum of the squared errors (Box & Draper, 1987) as shown in Equation 3.50.

$$min \qquad \sum_{i=1}^{n_P} w_i \left(F_i - \overline{F}_i(\underline{x})\right)^2 \qquad 3.50$$

where F_i is the observed response and $\overline{F}_i(\underline{x})$ its approximation. w_i are weights that can be used to control the influence of the experiments on the approximation model, i.e. the higher the weight, the closer the model will be to the corresponding response. In contrast to this, models based on Kriging methods assume that there is no numerical noise. Thus, a Kriging model that is evaluated at the location of an observed experiment will yield exactly the observed response. A comparison of the two approaches can be found in (Simpson et al., 2004).

Popular RSM methods are PA, RBF and GP. They will be described in the following.

Polynomial Approximations

The most common form of RSM approximation methods is the PA. Usually linear or quadratic polynomials are fitted to the observed data via least square error fitting which determines the unknown parameters a_i. A linear approximation function is shown in Equation 3.51 and a quadratic one in Equation 3.52.

$$\overline{F}_{lin}(\underline{x}) = a_0 + \sum_{i=1}^{n_{DV}} a_i \cdot x_i \qquad 3.51$$

$$\overline{F}_{quad}(\underline{x}) = a_0 + \sum_{i=1}^{n_{DV}} a_i \cdot x_i + \sum_{i=1}^{n_{DV}} \sum_{j=i}^{n_{DV}} a_{ij} \cdot x_i \cdot x_j \qquad 3.52$$

(Jin & Chen, 2001) highlight the straightforward application of PA and its efficient evaluation. At the same time the accuracy can be poor for large data sets.

Radial Basis Functions

According to (Queipo et al., 2005) RBF use linear combinations of radially symmetric functions h_i to approximate the target function. The model can be expressed as

$$\overline{F}_{rad}(\underline{x}) = \sum_{i=1}^{n_{RBF}} a_i \cdot h_i(\underline{x}) + \varepsilon_i, \qquad 3.53$$

with the coefficients a_i and the independent errors ε_i. The contribution of a particular basis function to the approximated function decreases with increasing distance from the corresponding centre. A typical function is the Gaussian function according to Equation 3.54

$$h_i(\underline{x}) = exp\left(-\frac{\left\|\underline{x} - \underline{x}_C\right\|^2}{\delta^2}\right), \qquad 3.54$$

where $\underline{x}_C$ is the centre and δ the radius. The tuning parameters a_i are determined in a least square sense according to Equation 3.55

$$\underline{a} = \left(\underline{H}^T \underline{H}\right)^{-1} \underline{H}^T \underline{y}, \qquad 3.55$$

where $\underline{y}$ is a vector containing the n_P observed responses and $\underline{H}$ a matrix made up of the functions h_i evaluated at the locations of the conducted experiments. $\underline{H}$ is calculated according to Equation 3.56.

$$\underline{H} = \begin{bmatrix} h_1\left(\underline{x}^{(1)}\right) & \cdots & h_{N_{RBF}}\left(\underline{x}^{(1)}\right) \\ \vdots & \ddots & \vdots \\ h_1\left(\underline{x}^{(n_P)}\right) & \cdots & h_{N_{RBF}}\left(\underline{x}^{(n_P)}\right) \end{bmatrix} \qquad 3.56$$

A response at an arbitrary location can be generated according to Equation 3.57.

$$\overline{F} = \underline{h}^T \underline{a} \qquad 3.57$$

For other basis functions and further background information the reader is referred to (Keane & Nair, 2005) and (Queipo et al., 2005). The tuning of the shape parameter δ is discussed in detail in (Keane & Nair, 2005).

(Jin & Chen, 2001) state that the implementation of RBF is relatively straightforward and that especially for small and scarce sample sets RBF outperforms the methods it was compared to (PA, Kriging, MARS (see Section 3.3.3.1)).

Genetic Programming

(Haupt & Haupt, 2004) describe Genetic Programming as software that develops new software. It was introduced by (Koza, 1992) and uses GA methodology to create expressions made from various defined programming constructs. For instance (Toropov & Alvarez, 1998) and (Alvarez, 2000) used this procedure for response surface approximations. In GP each individual consists of connected nodes in the form of a tree structure. The nodes are either terminal nodes, representing the design variables or function nodes, representing mathematical operators. The mating of two typical individuals is illustrated in Figure 3.36.

Moving Least Squares Method

A very popular approximation method is the MLS method (Lancaster & Salkauskas, 1981). It allows to build a standard regression, to interpolate or something in between the two (Forrester & Keane, 2009). A weighted least square approach is used where the weights are adjusted based on the distance from the point to be predicted, i.e. the weights for the calculation of the polynomial parameters decrease with increasing distance from the point in question. A popular choice for the decay function is the Gaussian function presented for RBF in Equation 3.56. The radius δ determines the ratio between interpolation and smoothing. For a high δ, smoothing is high and vice versa. For the determination of a suitable δ the cross-validation error can be minimized (Forrester & Keane, 2009). This naturally leads to a higher computational costs. The

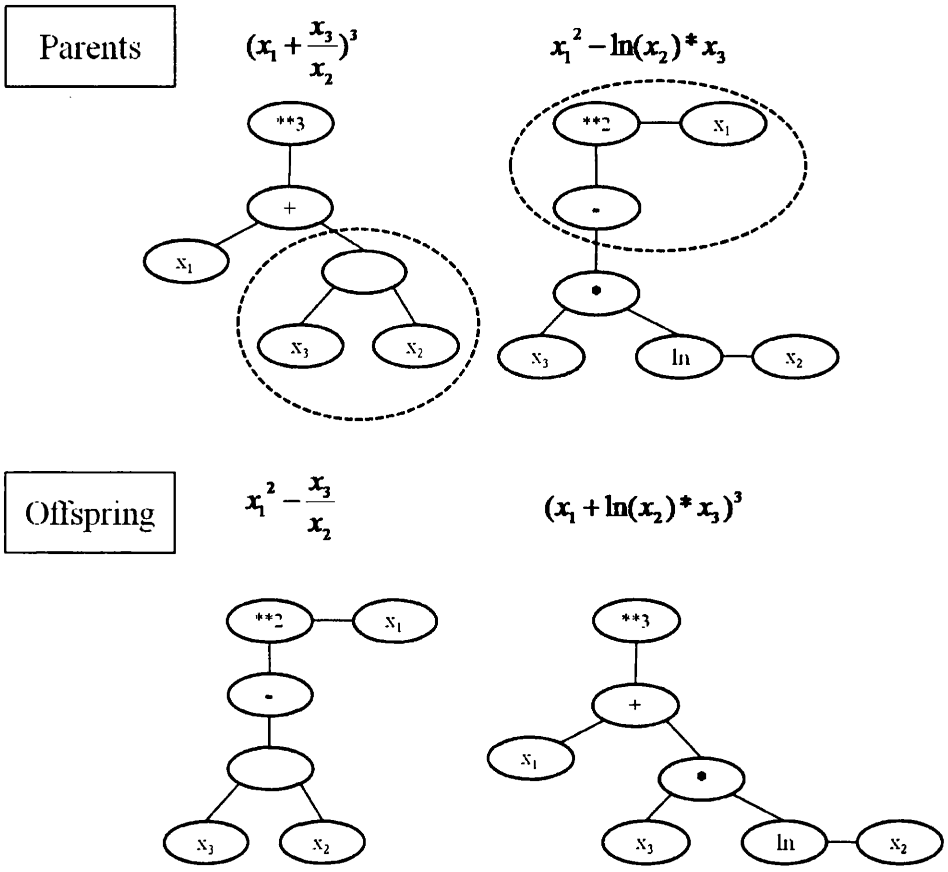


Figure 3.36: Illustration of mating two typical individuals in GP. Each individual is made up of terminal nodes, representing the design variables or function nodes, representing mathematical operators

costs are further increased since the polynomial parameters have to be determined for each point individually. (Forrester & Keane, 2009) state that MLS cannot be considered a global approximation method without a system of updating the model. Because of its relation to the response surface methodology, it is presented in this group.

3.3.2.3. Kriging

Typical linear regression approximations can be written in the form

$$\overline{F}(\underline{x}^{(i)})=\sum_h a_h\cdot f_h(\underline{x}^{(i)})+\epsilon^{(i)},\qquad (i=1,\dots,n), \qquad 3.58$$

with $f_h(\underline{x}^{(i)})$ being a linear or non-linear function and a_h unknown coefficients which are used to fit the approximation to the data. $\epsilon^{(i)}$ are independent error terms with zero mean and variance σ^2.

According to (Jones et al., 1998) the problem of applying regression for the approximation of deterministic computer code is that the assumption of independent errors is wrong. Because the experiments are deterministic, any lack of fit is a modeling error and not smoothing of noisy data. Following this, the errors should not be independent but correlated, with a high correlation if two points lie close to each other and low if far apart. (Jones et al., 1998) calculate the correlation according to Equation 3.59. The lack of random noise has been subject to discussion within the research community. For instance (Toropov, 1998) describe how a coarse mesh can lead to numerical noise.

$$Corr\left[\epsilon\left(\underline{x}^{(i)}\right),\epsilon\left(\underline{x}^{(j)}\right)\right] = exp\left[-\sum_{h=1}^{k}\theta_h\cdot\left|x_h^{(i)}-x_h^{(j)}\right|^{p_h}\right],(\theta_h \geq 0, p_h\epsilon[1,2]) \tag{3.59}$$

The parameters θ_h and p_h are estimated by maximizing the likelihood function according to Equation 3.60. In total these are $2\cdot n_{Dim}$ parameters, where n_{Dim} is the dimension of the design space.

$$\frac{1}{(2\cdot\pi)^{n/2}\cdot(\sigma^2)^{n/2}\cdot\left|\underline{R}\right|^{1/2}}\cdot exp\left[-\frac{\left(\underline{y}-\underline{1}\cdot\mu\right)^T\cdot\underline{R}^{-1}\cdot\left(\underline{y}-\underline{1}\cdot\mu\right)}{2\cdot\sigma^2}\right] \tag{3.60}$$

Here, n is the number of observed responses. The correlation matrix $\underline{R}$ and the parameters $\hat{\mu}$ and σ^2 are defined by Equations 3.61, 3.62 and 3.63, respectively.

$$R_{ij} = Corr\left[\epsilon\left(\underline{x}^{(i)}\right),\epsilon\left(\underline{x}^{(j)}\right)\right] \tag{3.61}$$

$$\hat{\mu} = \frac{\underline{1}^T\cdot\underline{R}^{-1}\cdot\underline{y}}{\underline{1}^T\cdot\underline{R}^{-1}\cdot\underline{1}} \tag{3.62}$$

$$\sigma^2 = \frac{(\underline{y}-\underline{1}\cdot\mu)^T\cdot\underline{R}^{-1}\cdot(\underline{y}-\underline{1}\cdot\mu)}{n} \tag{3.63}$$

Once the parameters are determined the response surface can be evaluated at arbitrary locations $\underline{x}^*$ using the predictor formula

$$\hat{y}\left(\underline{x}^*\right) = \hat{\mu}+\underline{r}^T\cdot\underline{R}^{-1}\cdot\left(\underline{y}-\underline{1}\cdot\hat{\mu}\right), \tag{3.64}$$

with the vector of correlations

$$r_i(\underline{x}^*) = Corr[\epsilon(\underline{x}^{(*)}), \epsilon(\underline{x}^{(i)})]. \qquad 3.65$$

To circumvent various numerical issues, (Keane & Nair, 2005) suggests to use a Log-likelihood function according to Equation 3.66 and to add a small term such as 10^{-6} to the diagonal terms of the correlation matrix.

$$\frac{1}{2} \cdot \left[n \cdot ln(2\pi) + n \cdot ln(\sigma^2) + ln|\underline{R}| + \frac{(\underline{y} - \underline{1} \cdot \mu)^T \cdot \underline{R}^{-1} \cdot (\underline{y} - \underline{1} \cdot \mu)}{2 \cdot \sigma^2} \right] \qquad 3.66$$

For other correlation functions see (Keane & Nair, 2005) and (Santner et al., 2003).

This approach is commonly called the 'DACE stochastic process model'. 'DACE' is an abbreviation for 'Design and analysis of computer experiments' which is the title of the paper which made this approach popular (Sacks et al., 1989). It is also referred to as Kriging (Keane & Nair, 2005).

3.3.3. Mid-range Approximations

As mentioned before, when dimensions of design spaces become large, global approximations may require too many function evaluations to be a feasible choice. A possible alternative are mid-range approximations. They widen the range of applicability of local approximations by either using several points (referred to as multipoint methods) or combine local and global methods (referred to as local-global methods) (Barthelemy & Haftka, 1993). Generally, multipoint approximations fit results around the current design using a simple second order regression equation which is used to improve the design within a trust region. Then a new approximation is built around that improved design until convergence occurs (Keane & Nair, 2005), (Schoofs et al., 1997) and (Toropov et al., 1993). As an example for the above category the Multipoint Approximation method based on Response Surface fitting (MARS) by (Toropov, 1998) will be discussed in the following.

3.3.3.1. Multipoint Approximation method based on Response Surface fitting (MARS)

The basic idea behind MARS is to define a current subregion and to evaluate the expensive simulation code at a number of locations defined by a DoE. Then, the newly obtained results and those that are available from previous iterations are used to fit a response surface in a least square sense. After the approximation model has been used to find the optimum within the current subregion, the subregion is moved in the design space towards the new optimum and the procedure starts again until convergence occurs. The general flow is illustrated in Figure 3.37.

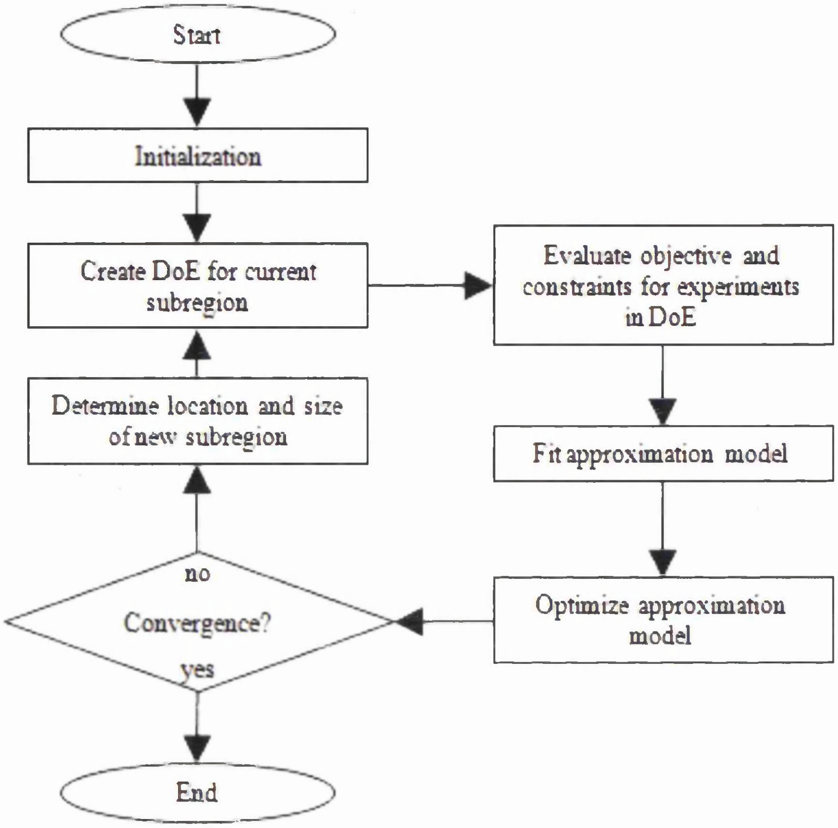


Figure 3.37: Flow of the Multipoint Approximation method based on Response Surface fitting.

The parameters for the chosen approximation method are determined by means of weighted least squares (see Equation 3.50). The weight coefficients govern the relative contribution of each response to the model, i.e. for a high weight the approximation model will be more likely to have a function value close to the actual response for the corresponding parameter values. High weights are assigned to experiments that lie on the boundary of the feasible space, that correspond to a relatively good objective function and that have a small numerical error (if information is available, for instance if an adaptive strategy is used).

After the approximation model has been used to find the optimum in the current subregion, a new location and size has to be established. To this end a move limit strategy is employed that requires the evaluation of several indicators that determine the new subregion. These indicators are:

- quality index (based on the largest relative approximation error)
- location index (dependent on the location of the optimum in the current subregion)
- two move direction indicators (governed by the move direction of the last two steps)
- size indicator (based on the size of the current subregion)
- feasibility indicator (based on the distance from the boundary of an infeasible region)

Depending on the distance, experiments outside the current search domain can improve the approximation model if they are included in the model generation. (Toropov, 1998) found a domain size 1.5 to 2.0 times larger than the search domain to work well as a neighborhood where experiments are still considered in the model generation.

Since old experiments are considered in the following iterations, a DoE generation scheme has to be employed that allows for new points to be placed among existing ones. (Toropov, 1998) proposes a strategy where one new design is created at a time. A cost function is used that enforces a homogenous distribution with a maximum distance between the points without allowing them to be placed on the boundary of the subregion.

The main motivation for the development of this method is the case of a high dimensional design space where a global approximation method would require too many experiments to reach a satisfactory degree of accuracy. Since MARS samples only a smaller subspace the number total number of experiments is typically lower in such cases.

3.3.4. Model Validation

Several methods exist that allow for the assessment of the quality of surrogate models or that help selecting an approximation method among a number of candidates. The most prominent methods in the context of Surrogate based optimization are Split Sample, Bootstrapping and Cross-validation (Queipo et al., 2005). In the Split Sample scheme the sample data is split into a training and a test set. The former is used to build the model. This is evaluated at the locations of the test set and the comparison between the evaluated and observed responses gives an indication on the accuracy of the model. Bootstrapping is an effective assessment method in the context of Neural Networks but it has been applied with regression methods as well (Queipo et al., 2005). The idea of cross-validation is to predict one response with the model being based on the remaining $n-1$ observations. This response is denoted as $y_{-1}(\underline{x}^{(i)})$. Following (Jones et al., 1998), in the context of Kriging the so called standardized cross-validation residual, abbreviated as SCVR, is calculated via

$$SCVR = \frac{y(\underline{x}^{(i)}) - y_{-1}(\underline{x}^{(i)})}{S_{-1}(\underline{x}^{(i)})}, \qquad 3.67$$

with the mean squared error of the predictor

$$S^2(\underline{x}^*) = \sigma^2 \cdot \left[1 - \underline{r}^T \cdot \underline{R}^{-1} \cdot \underline{r} + \frac{\left(1 - \underline{1}^T \cdot \underline{R} \cdot \underline{r}\right)^2}{\underline{1}^T \cdot \underline{R} \cdot \underline{1}}\right] \qquad 3.68$$

The model is considered to be valid if the absolute value of the SCVR does not exceed 3 for any observation in the model. To be precise, one would have to re-estimate the Kriging parameters for each prediction in the cross-validation, but unless the number of observations the model is based on is very small, omitting one point has a negligible effect. Thus the initial parameters are typically reused throughout the check. Thus, only $\underline{R}$, μ, σ^2 and $\underline{r}$ have to be recalculated for the remaining $n-1$ observations, whereas θ_i and p_i remain unchanged.

3.4. Application

Before surrogate based optimization will be applied to the Engineering design application in Chapter 6, the individual tools will be tested and validated. To this end, they will be applied to several benchmark problems. These are two unconstrained and one constrained problem. The latter will be solved with and without the use of OLHs for constrained design spaces. The tools that will be used are:

- For the creation of DoEs: Formulation for the creation of OLHs (see Section 3.2.3)
- As the approximation method: Kriging model (see Section 3.3.2.3)
- For the optimization using the surrogate model: Continuous GA (see Section 2.3.5.3)

3.4.1. Application to Unconstrained Problems

The following two unconstrained benchmark problems were found in (Trelea, 2003). The first one that will be solved is the so called Rastrigin function. The problem will be solved for two design variables on the interval [-1.5, 1.5]. The problem and its solution read:

objective function $$y = \sum_{i=1}^{2} [x_i{}^2 - 10 \cdot cos(2 \cdot \pi \cdot x_i) + 10] \qquad 3.69$$

interval $-1.5 \leq x_i \leq 1.5, \forall\, i$

location of global minimum $x_i = 0, \forall\, i$

function value at global minimum $y = 0$

The function on the defined parameter interval is illustrated in Figure 3.38. The function values vary between 0 and 45.

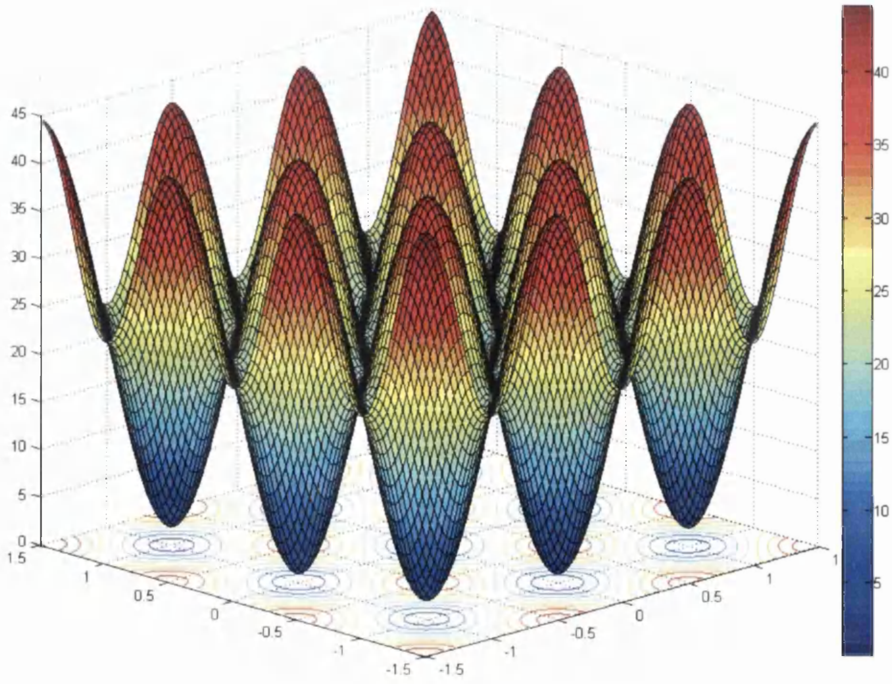


Figure 3.38: The Rastrigin function plotted on the defined design space.

Five Kriging models are built to test the influence of the number of points on the accuracy on the solution. The number of experiments n_P are: 40, 80, 120, 180 and 400. The solutions of the optimization runs are listed in Table 3.1.

Table 3.1: Creation and optimization of the five surrogate models approximating the Rastrigin function. The analytical global optimum is: $\underline{x} = [0 \quad 0]^T, y = 0$.

OLH		***Kriging***	***Optimization solution***		
nP	*pot. energy*	*max. SCVR*	*x1*	*x2*	*y*
40	*3.929*	*1.69*	*0.021*	*-0.037*	*-0.777*
80	*4.920*	*1.44*	*0.002*	*-0.004*	*1.047*
120	*5.513*	*1.82*	*-0.001*	*0.002*	*0.086*
180	*6.113*	*1.57*	*0.000*	*0.002*	*0.169*
400	*7.333*	*1.02*	*0.002*	*0.011*	*0.125*

It can be seen that the solution that is closest to the analytical solution has been found by the model which is based on 120 points, i.e. the models with 180 and 400 points yield an optimum that is less accurate. Naturally, the accuracy of the solution depends on how close the experiments are to the location of the global optimum and it is possible that a point in the OLH with 120 experiments is closer to the optimum than for higher numbers. Nevertheless, the probability of this to happen decreases with increasing numbers of experiments. Considering function values between 0 and 45, within the searched design space, the accuracy of the models with 120, 180 and 400 can be considered sufficiently accurate. With regards to the location of the global optimum, all models yield accurate results. Furthermore, despite the existence of several local minima, all models converged towards the global minimum.

One can see that the SCVR is not an indicator on the accuracy of the optimum solution. In fact, the model (120 points) which yields the most accurate solution exhibits the highest SCVR of 1.82. As discussed in Section 3.3.4, the SCVR measures how accurately function values are predicted when the corresponding experiment is not included in the model. Hence, it does indicate the overall reliability of the model but not necessarily the accuracy of the function value in an arbitrary location in the design space. Function plots based on the Kriging interpolation and the cross-validation plots can be found in Appendix A. The development of the function value during the optimization of the Kriging model is illustrated in Figure 3.39.

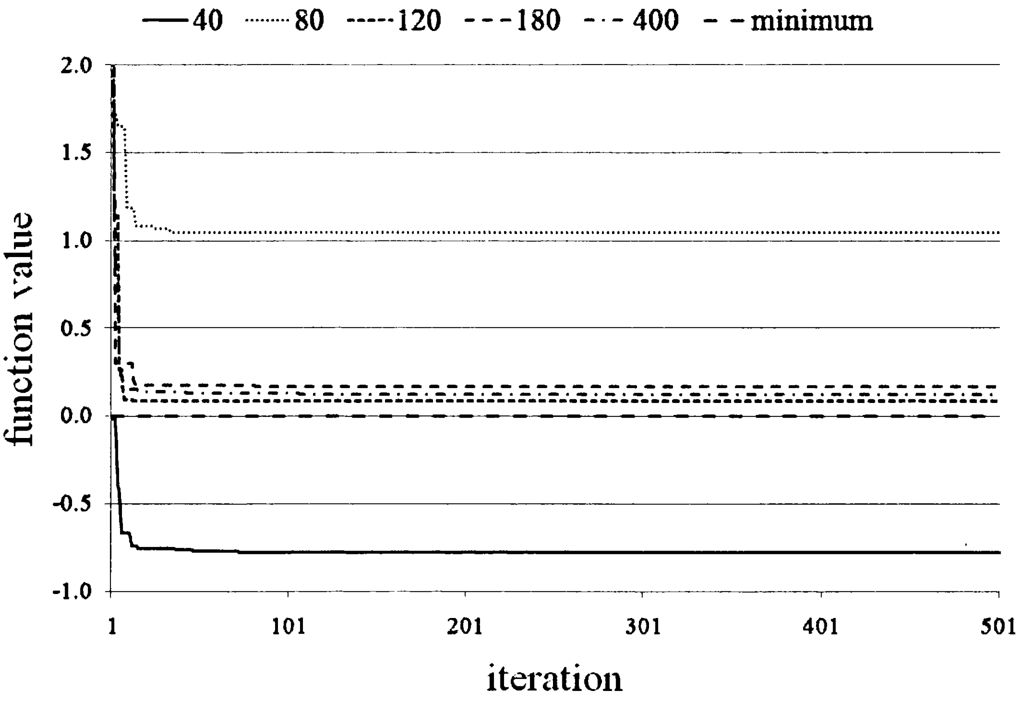


Figure 3.39: Development of the function value during the optimization of the Kriging model approximating the Rastrigin function.

The second unconstrained benchmark problem that will be solved here is a sphere function. The problem will be solved for two design variables on an interval [-100, 100]. In this case the parameter interval and the function values are significantly higher than in the first example. The problem and its solution read:

objective function $$y = \sum_{i=1}^{2} x_i^2 \qquad 3.70$$

interval $$-100 \leq x_i \leq 100, \forall\, i$$

location of global minimum $$x_i = 0, \forall\, i$$

function value at global minimum $$y = 0$$

The function on the defined parameter interval is illustrated in Figure 3.40. The function values vary between 0 and 20,000.

Five Kriging models are built which are based on the same OLH as in the first example. The solutions for the optimization runs are listed in Table 3.2.

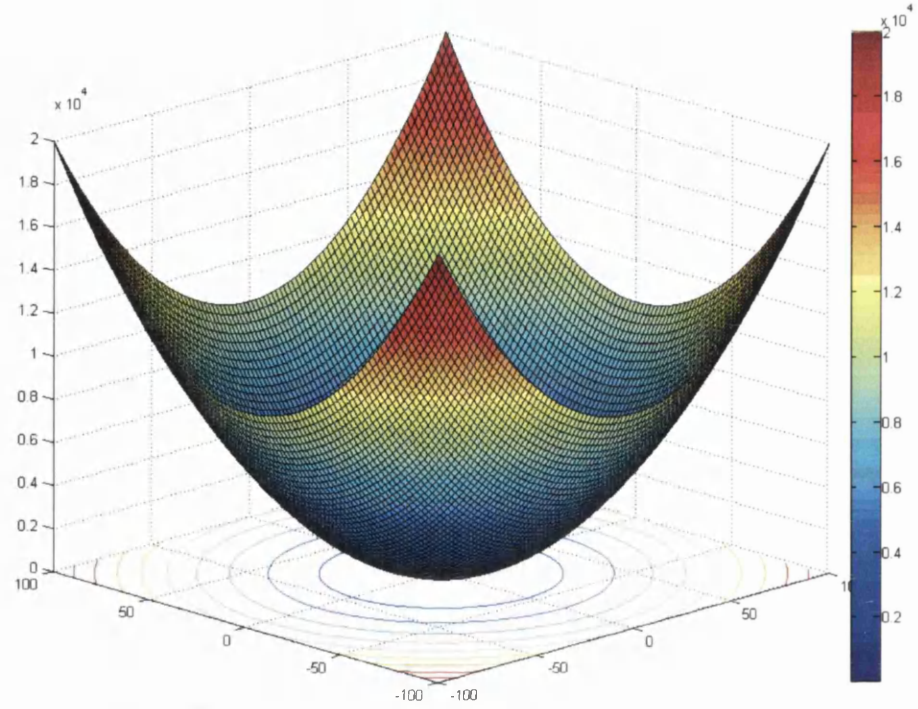


Figure 3.40: The sphere function plotted on the defined design space.

Table 3.2: Creation and optimization of the five surrogate models approximating the sphere function. The analytical global optimum is: $\underline{x} = [0 \quad 0]^T, y = 0$.

OLH		***Kriging***		***Optimization solution***	
nP	*pot. energy*	*max. SCVR*	*x1*	*x2*	*y*
40	*3.929*	*1.08*	*0.372*	*1.196*	*10.569*
80	*4.920*	*1.42*	*1.014*	*-0.662*	*5.436*
120	*5.513*	*1.52*	*0.390*	*-0.251*	*-9.043*
180	*6.113*	*1.91*	*-0.230*	*-0.035*	*0.949*
400	*7.333*	*1.66*	*0.239*	*-0.555*	*2.757*

Considering the high variable and function value range, all solutions can be considered sufficiently accurate. Similarly to the previous example, here the least accurate model (40 points) exhibits the lowest SCVR while the most accurate one (140 points) exhibits the highest one. Function plots based on the Kriging interpolation and the cross-validation plots can be found in Appendix A. The development of the function value during the optimization of the Kriging model is illustrated in Figure 3.41.

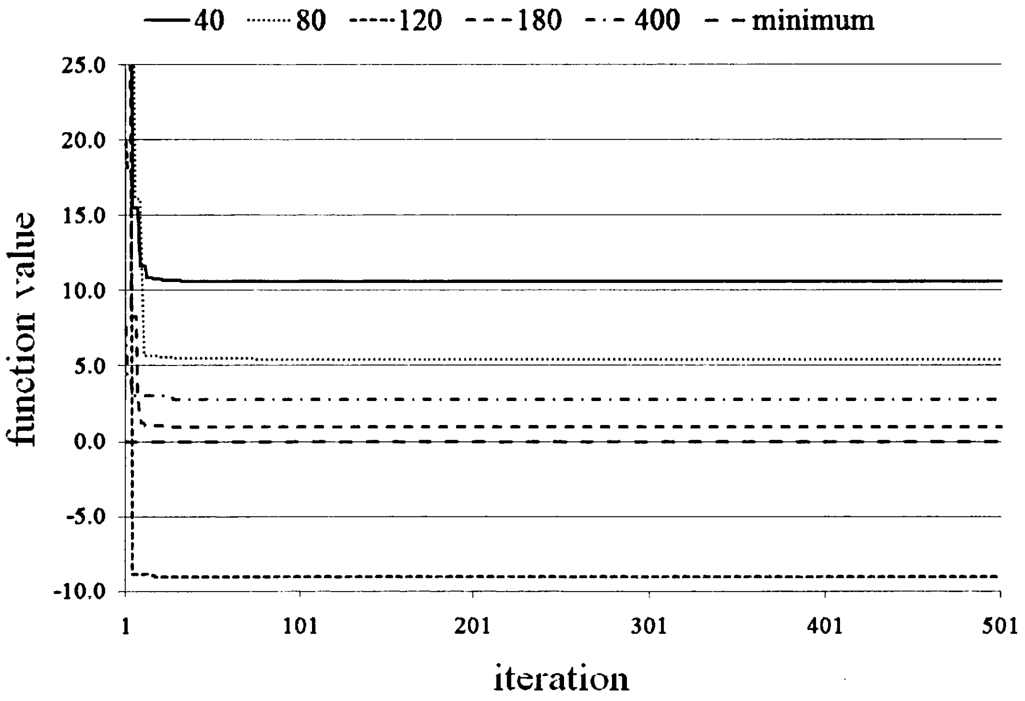


Figure 3.41: Development of the function value during the optimization of the Kriging model approximating the Sphere function.

In Figure 3.42 the numbers of points in the Kriging model are plotted against the function values for the last two examples.

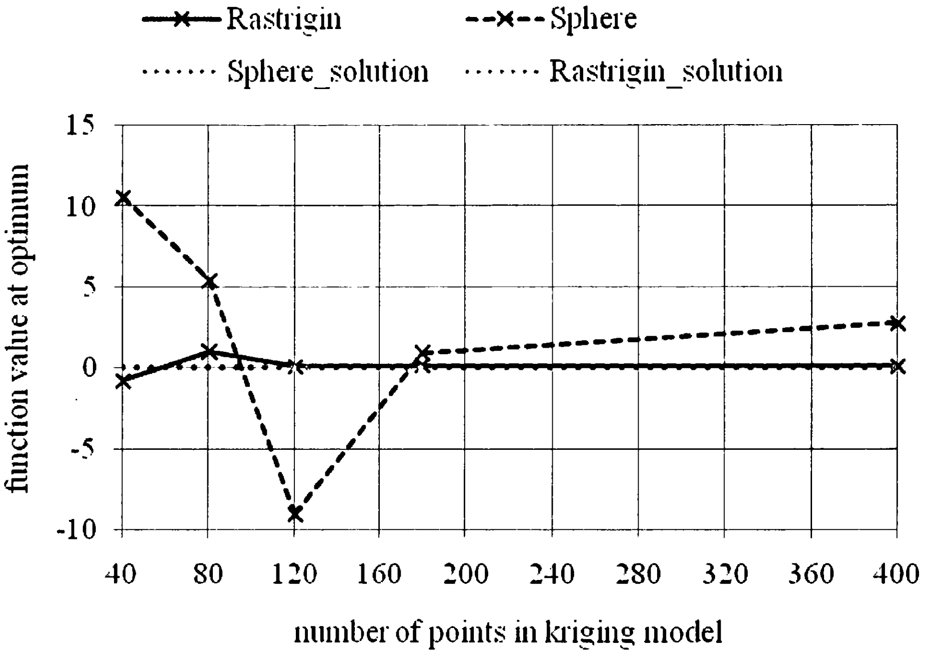


Figure 3.42: The numbers of points in the Kriging model plotted against the function values for the Rastrigin and sphere function.

3.4.2. Application to a Constrained Problem

After the application of the developed surrogate based optimization framework to two unconstrained problems, now a constrained function will be considered. This 2D problem, in (Hamida & Schoenauer, 2002) denoted as g08, will be solved on an interval of [0.5, 3.5] for x_1 and [2.0, 6.0] for x_2. The problem reads:

objective function

$$y = -\frac{\sin^3(2 \cdot \pi \cdot x_1) \cdot \sin(2 \cdot \pi \cdot x_2)}{{x_1}^3 \cdot (x_1 + x_2)} \qquad 3.71$$

subject to

$$x_1{}^2 - x_2 + 1 \leq 0$$

$$1 - x_1 + (x_2 - 4)^2 \leq 0$$

interval

$$0.5 \leq x_1 \leq 3.5, \qquad 2.0 \leq x_2 \leq 6.0$$

location of global minimum

$$\underline{x} = \begin{bmatrix} 1.22797135260752599 \\ 4.24537336612274885 \end{bmatrix}$$

function value at global minimum

$$y = -0.0958250414180359$$

The problem will be solved using Kriging models that are based on OLHs that have been created either in the feasible region of the design space only or in the complete design space. Three Kriging models are built using the former type of OLHs and four using the latter. The constraints are illustrated in Figure 3.43 and Figure 3.44. Figure 3.43 shows an OLH with 60 experiments which have been created only in the feasible region. Figure 3.44 shows an OLH with 400 experiments which are spread over the complete design space. In the latter only approximately 30 feasible experiments have been created. The function on the defined parameter interval is illustrated in Figure 3.45. The function values vary between -0.731 and 0.859.

The solutions for the seven optimization runs are listed in Table 3.3. Obviously, the Kriging model based on the OLH with 60 experiments created in the feasible region yields the most accurate results. In this case also the SCVR is the lowest among the created models. Even the model based on 40 feasible experiments predicts the global minimum more accurately than the model based on 400 experiments spread over the

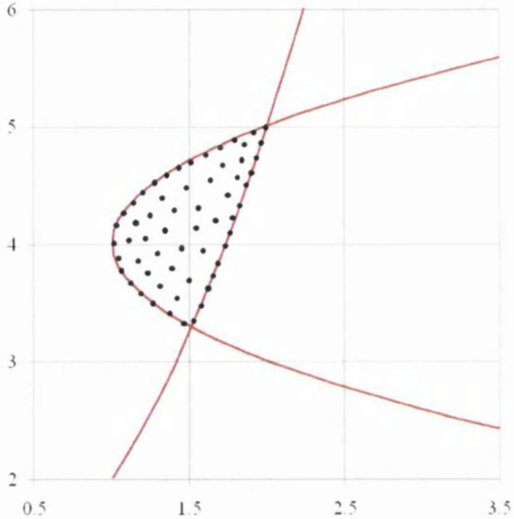


Figure 3.43: OLH with 60 experiments created only in the feasible region of the design space.

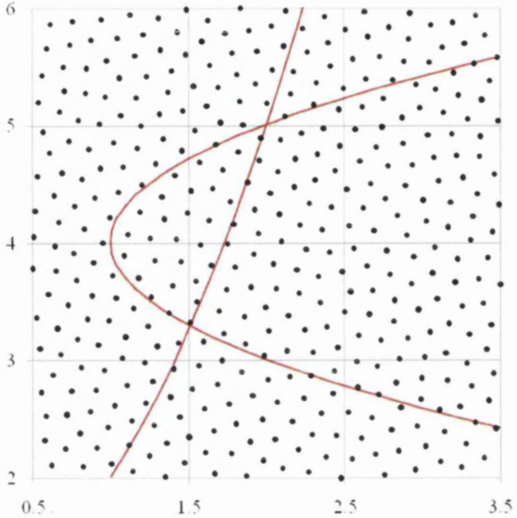


Figure 3.44: OLH with 400 experiments created in the complete design space.

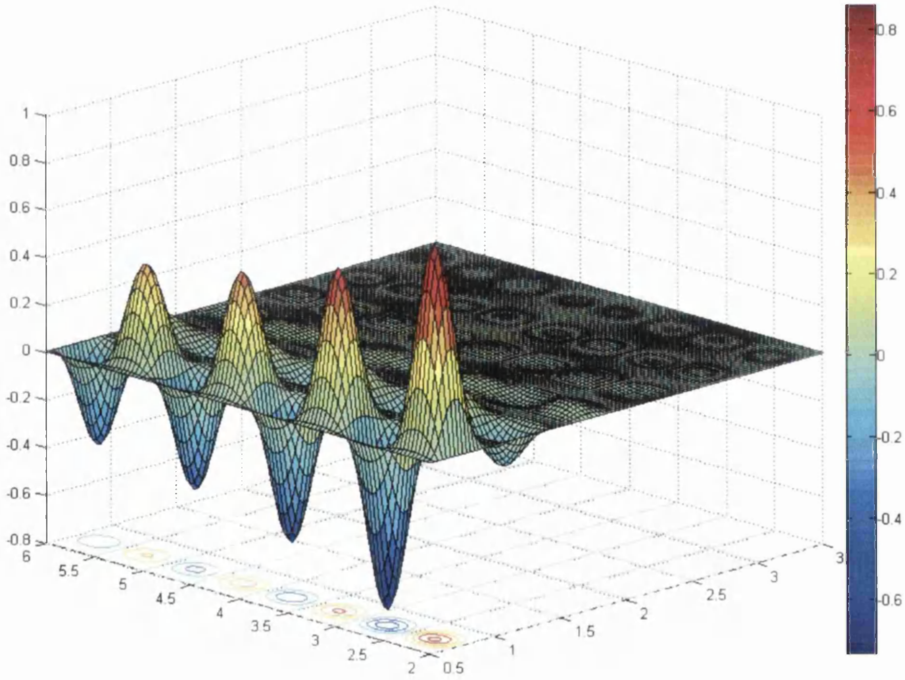


Figure 3.45: The g08 function plotted on the defined design space.

entire design space. This is not surprising since, as mentioned before, this OLH contains only around 30 feasible experiments. Function plots based on the Kriging interpolation and the cross-validation plots can be found in Appendix A.

Table 3.3: Creation and optimization of the seven surrogate models approximating the g08 function. The analytical global optimum is: $\underline{x} = [1.22797 \quad 4.24537]^T, y = -0.09582$.

OLH			***Kriging***	***Optimization solution***		
nP	*constraints*	*pot. energy*	*max. SCVR*	x_1	x_2	y
20	*yes*	*32.199*	*3.95*	*1.172*	*4.357*	*-0.061*
40	*yes*	*42.557*	*3.02*	*1.230*	*4.229*	*-0.090*
60	*yes*	*50.351*	*2.04*	*1.232*	*4.245*	*-0.095*
40	*no*	*3.929*	*4.42*	*1.039*	*3.802*	*-0.021*
80	*no*	*4.920*	*7.27*	*1.108*	*3.671*	*-0.335*
180	*no*	*6.113*	*8.86*	*1.338*	*4.190*	*-0.042*
400	*no*	*7.333*	*7.89*	*1.184*	*4.266*	*-0.084*

The development of the function value during the optimization of the Kriging model is illustrated in Figure 3.46.

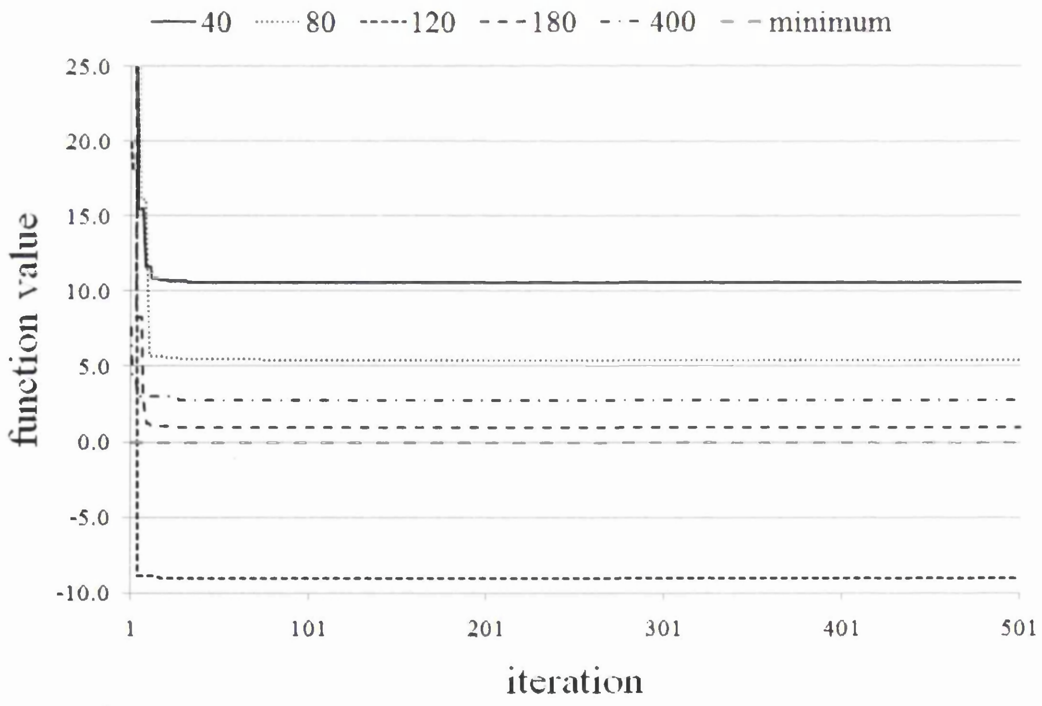


Figure 3.46: Development of the function value during the optimization of the Kriging model approximating the Sphere function.

In Figure 3.47 the numbers of points in the Kriging model are plotted against the function values for the last example.

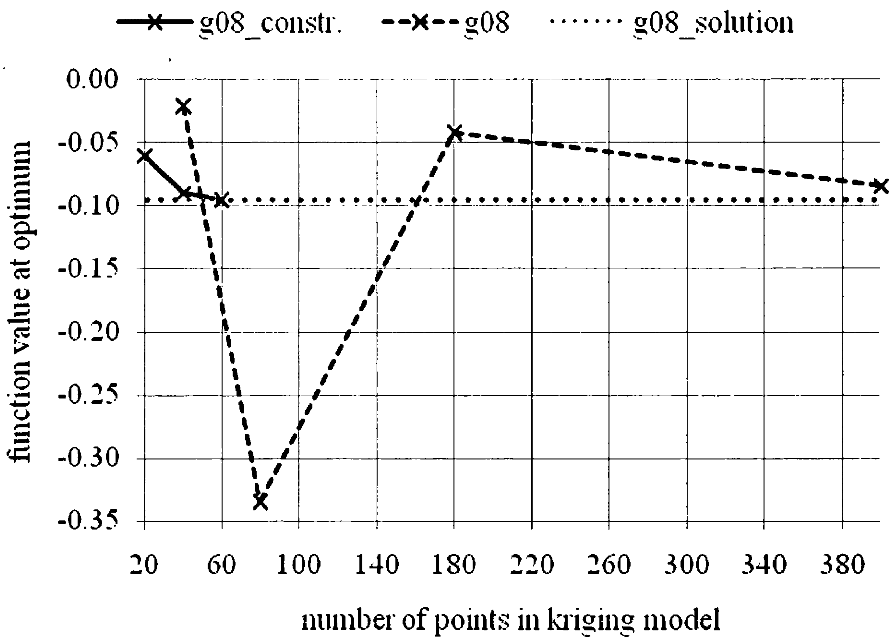


Figure 3.47: The numbers of points in the Kriging models plotted against the function values for the g08 function.

3.5. Concluding Remarks

This chapter has given an overview of the elements in surrogate based optimization. The main ingredients being DoEs (Section 3.2) and design space approximation methods (Section 3.3). A formulation for the creation of OLH for constrained design spaces has been developed and discussed in detail in Section 3.2.3. Several examples showed the validity of the chosen approach and its realization. A shortcoming of the formulation has been identified. That is, in some cases, areas whose parameter projections on several dimensions are unconstrained, can exhibit a sampling density that is lower than in areas where this is not the case. This shortcoming of the procedure could be diminished by employing non-equidistant space divisions with larger spacing towards the constraint boundaries and vice versa. Nevertheless, in most cases the uniformity of the placed points can be considered good and naturally better than that of the feasible experiments in an OLH that has been created for the entire design space.

The Kriging approximation method has been implemented and, together with the developed method for the creation of OLHs as well as an implementation of a continuous GA, formed into a framework for surrogate based optimization. This framework has been tested and validated on three benchmark problems. Here, two unconstrained and one constrained problem were considered. The latter illustrated the advantage of creating OLHs in the feasible region of the design space rather than in the entire design space. The three benchmark problems showed that the SCVR is not an indicator on the accuracy of the optimum solution when a Kriging approximation is used. The SCVR measures how accurately function values are predicted when the corresponding experiment is not included in the model. Hence, it does indicate the overall reliability of the model but not necessarily the accuracy of the function value in an arbitrary location in the design space.

The developed framework for surrogate based optimization has been applied in one industrial Engineering design example and will be presented in Chapter 6.

4. Optimization Model

Summary: This chapter gives an overview of the aspects of the optimization model which are of importance in this thesis. The main methods in structural optimization, i.e. topology, shape and size optimization are introduced, aspects of parameterization in shape optimization are discussed and approaches to the automation of the simulation model creation process presented. In particular the approach used within this thesis is explained in detail. Furthermore, the decomposition of large optimization problems is discussed. A new method for the decomposition of large coupled size and shape optimization problems is proposed.

4.1. Introduction

As described in Chapter 1, the optimization model represents the link between the simulation model and the optimization algorithm. It receives values for the design variables from the former and transforms these into a model which is solved by the latter. The form of the optimization model determines the type of design variables, i.e. the optimization method in question and effectively the parameterization of the structure to be optimized. In the other direction it receives the structural responses from the simulation model and uses these to evaluate the objective and constraint functions. Naturally, this process has to work in an automated manner which requires particular attention on the realization of the transformation of the design variable values into the simulation model.

In this chapter, the most important aspects of the optimization model with respect to the work in this thesis will be discussed. Firstly, the three main optimization methods, i.e. the different types of design variables, used in structural optimization will be presented. These are topology, shape and size optimization. A close look at different approaches to the parameterization for shape optimization follows. Subsequently, the various possibilities for the decomposition of large optimization problems into smaller sub problems are presented. A novel approach for the decomposition of coupled size and shape optimization problems is proposed. Finally, different approaches to automation of the simulation model creation process will be discussed. Here, particular attention is paid to the approach used within two industry examples in this thesis.

4.2. Structural Optimization Methods

The chosen type of design variables governs the characteristic of the structural optimization problem and thus its complexity and the complexity of the solution that can be found. The notion 'structural optimization method' is typically used to describe this character of the optimization problem in question. In general three different structural optimization methods can be identified (Christensen & Klarbring, 2008):

- **Topology optimization**: Determines the optimal topology of the domain – holes of arbitrary shape and size can be introduced in the domain
- **Shape optimization**: Determines the optimal shape of the domain
- **Size optimization**: Determines for instance the optimal size of truss members or an optimal thickness distribution of a plate

To illustrate these optimization methods, a simple example will be solved in the following. This problem involves a cantilever beam of 100mm length and 10mm height. Its left side is fully clamped and a force of 100N is applied at the bottom end of the right side. The objective function that is to be minimized in these problems is the overall volume while a maximum displacement of 3mm of the node, where the force is applied at, shall not be exceeded. The initial beam thickness is 1mm and the beam material is steel. The problems are solved with Altair OptiStruct (Altair HyperWorks (b), 2009). Figure 4.1 illustrates the resulting designs for the exemplary topology, shape and size optimization

Figure 4.1 a) shows the beam in its initial form. The corresponding volume is 1000mm^3 and the tip deflection is 1.92mm.

Figure 4.1 b) shows the optimum solution for the topology optimization where the entire cantilever has been declared as design space. During optimization, openings have been created in the middle of the beam with increasing size towards the unsupported end. The optimum topology resembles the form of an I-beam where the top and bottom fibres of the cross-section carry the majority of the load and shear forces are transferred via inclined members in between. Also the material in the cross-section increases towards the support, where the bending moment is maximal. The volume of the final design is 382mm^3. This corresponds to a material saving of 61% with respect to the initial design.

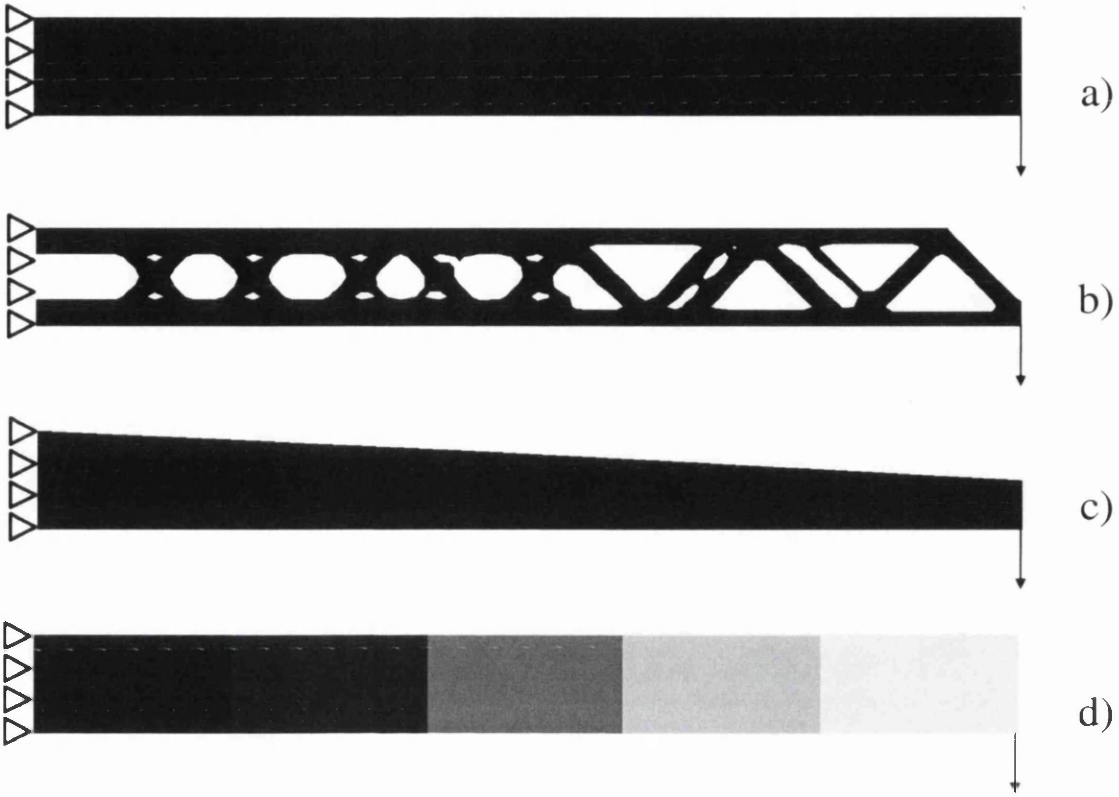


Figure 4.1:.Exemplary topology, shape and size optimization of a cantilever beam subjected to a vertical force at the unsupported end. The objective function to be minimized in these problems is the overall mass while a displacement constraint of the beam tip has to be satisfied. a) shows the initial design, b) the optimum design for topology optimization, c) for shape optimization and d) for size optimization.

Figure 4.1 c) shows the optimum solution for the shape optimization where the shape of the upper edge of the beam is the design variable. In the optimum design the height of the beam is decreased linearly from the supported end to the unsupported end. Due to the definition of the design variables only the variation of the bending moment can be taken into account resulting in the varying cross-sectional height. The varying material distribution across the cross-section, as it was found in the topology optimization, does not occur. Naturally, the solution of the topology optimization is superior, i.e. lighter, to the one obtained in the shape optimization. Here, the volume of the final design is 761mm^3 which corresponds to a material saving of 24%.

Figure 4.1 d) shows the optimum solution for the size optimization where the beam has been divided into five sections of equal length. The design variables are the sections' thicknesses. In the optimum design, the thickness decreases from the supported end to the unsupported end. The resulting design is similar to that in the shape optimization. Only here, the varying bending moment is accounted for by reducing the beam's thickness along its length rather than its cross-sectional height. The volume of the final design is 497mm^3 which corresponds to a material saving of 50%, which is by about 25% greater than that of the shape optimization and by about 10% smaller than in the topology optimization. To make the best use of size and shape optimization, they are usually performed at the same time.

(Hinton et al., 2003) describe a typical design process as a sequence of three phases. In the first phase the optimum topology is found for the given loading and support conditions using topology optimization. In the second phase the obtained topology is redefined to form a smooth geometric description which is parameterized subsequently. In the third and final phase the geometry is fine-tuned using shape and size optimization.

The present work focuses on the last two phases in the design process, assuming that the structural topology is given through either a preliminary topology optimization, experience or manufacturing constraints.

4.3. Parameterization for Shape Optimization

While size optimization usually does not require the parameterization of the geometry as the design variables are mere properties such as thickness or cross-sections of a shell or beam element, shape optimization involves changes in the geometry boundary. Thus, the geometry has to be parameterized to some extent to allow for its automatic update. In the following some of the most common approaches to shape parameterization will be presented.

4.3.1. Discrete Parameterization

The most basic form of parameterization is the discrete parameterization. It can give complete flexibility over the geometry by controlling the lowest level of the geometry definition (Keane & Nair, 2005). Possible design variables in this sense are the coordinates of points that define lines and elements as illustrated in Figure 4.2. Then changes are applied to the geometry and not the mesh. Hence, the geometry has to be remeshed after manipulation. This approach has two obvious drawbacks. Firstly, the geometry often features a lack of smoothness which can lead to stress concentrations and unacceptable designs. Secondly, the number of design variables required to sufficiently parameterize a geometry can become very high (Keane & Nair, 2005).

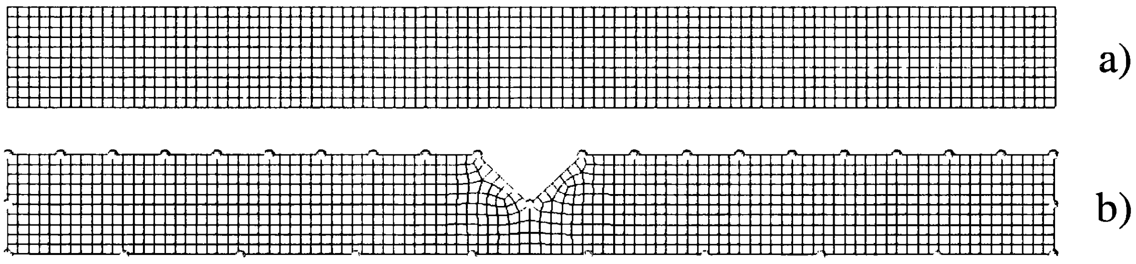


Figure 4.2:Discrete parameterization of a beam. Here, the design variables are the nodal coordinate defining the boundary lines of the structure. a) shows the original geometry and b) the geometry after moving one of the nodes. Here, changes are applied to the geometry and not the mesh. Hence, the geometry has to be remeshed after manipulation.

Instead of altering discrete entities governing the geometry the nodal coordinates of the FE mesh can be altered themselves (Arora, 2007) but in this approach the previous drawbacks are exacerbated and since no remeshing is performed the resulting mesh distortion limits its applicability further.

In some cases the geometry is simple enough that a discrete parameterization leads to both, a small number of design variables and a satisfactory flexibility. It has been applied in two industry examples. In Section 5.5.3 the shape of the pipe cross-section has been parameterized in a discrete way. In Chapter 7, the location of the shear web in a wind turbine blade has also been realized in this way.

4.3.2. Morphing

Within structural optimization, morphing describes the distortion of basis geometry while maintaining its original topology (Keane & Nair, 2005). To this end features of the basis geometry such as boundary lines have to be defined. An example of a morphed geometry is shown in Figure 4.3. Here, the middle of the upper line of the original beam in Figure 4.3 a) is moved downwards. This is done to the meshed geometry, i.e. no remeshing is necessary. This method is limited to small geometrical changes, because otherwise the element distortion would become too large (Duddeck, 2008).

In Altair OptiStruct (Altair HyperWorks (b), 2009) the so called perturbation vector approach to shape optimization is used. Here, the structural shape is defined as a linear combination of perturbation vectors. Each perturbation vector is constructed using the morphing capability of Altair HyperMesh (Altair HyperWorks (a), 2009). Hence, the designer needs to manually create the morphs that the final solution will be made up of.

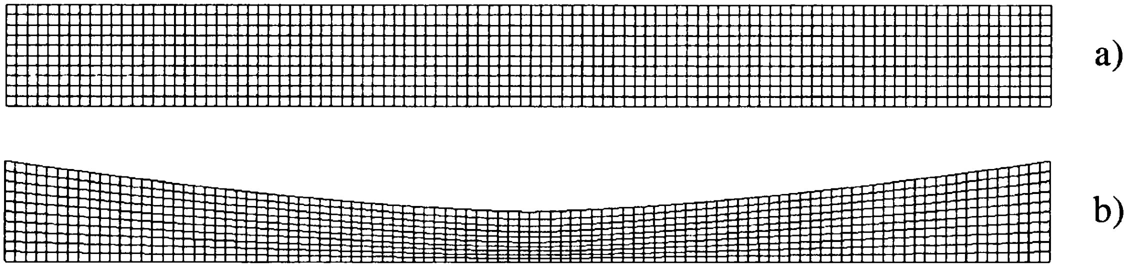


Figure 4.3: Morphing of a beam. a) shows the original geometry and b) the geometry after moving the middle of the upper line downwards. Here, changes are applied to the FE mesh. Hence, no remeshing is necessary.

4.3.3. Spline-based Parameterization

The two drawbacks of the discrete parameterization described before can be mitigated by employing a spline-based parameterization. Spline curves are low-order piecewise polynomial functions (Christensen & Klarbring, 2008). High order polynomials can lead to oscillation of the boundary (Christensen & Klarbring, 2008). This is a problem that does not occur when using spline curves. In addition to that the control points, that govern the splines, have a more intuitive relationship to the geometry they describe (Keane & Nair, 2005). Here, features such as lines, describing the boundary of the geometry are described using splines. The splines in return are governed by a certain number of control points. A manipulation of the coordinates of the control points leads to a change of the line which remains smooth. In an optimization process the control points are the design variables.

Typical splines are Bezier splines, B-splines and non-uniform rational B-splines (NURBS). The difference between Bezier and B-splines is that the number of control points is not directly governed by the degree of the spline (Christensen & Klarbring, 2008). In contrast to B-splines, NURBS are capable of forming conic sections (Keane & Nair, 2005).

Within this work B-splines are used exclusively. In more detail an arbitrary B-spline representation is discussed in Section 4.3.3.1 and a cubic B-spline representation in Section 4.3.3.2.

Figure 4.4 illustrates the use of a cubic B-spline in geometry parameterization. The boundary lines of a beam Figure 4.4 a) are described using B-splines. The y-coordinate of a control point in the middle of the upper line is moved downwards and the resulting geometry remeshed Figure 4.4 b).

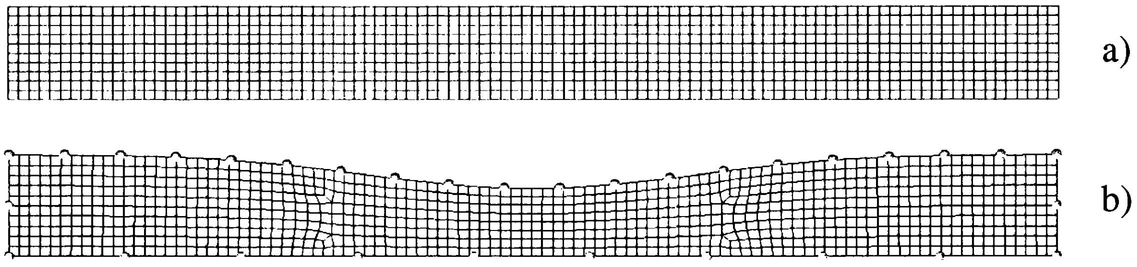


Figure 4.4: B-spline parameterization of a beam. Here, the design variables are the coordinates of the control points defining the boundary lines of the structure. a) shows the original geometry and b) the geometry after moving a control point in the middle of the upper line downwards. Here, changes are applied to the geometry and not the mesh. Hence, the geometry has to be remeshed after manipulation.

4.3.3.1. Arbitrary B-spline Representation

In order to describe a line in two dimensions by means of B-splines, effectively two splines are necessary: One for the *x* and one for the *y*-coordinate. Often the two splines are combined in a vector form, where the corresponding control values for one point are combined in one vector. In the literature this is often referred to as a control point. Since B-splines can be used to interpolate one dimensional values too, the following discussion will remain one dimensional.

(Christensen & Klarbring, 2008) define an arbitrary B-spline representation of degree p with $k+1$ control values b_i. It is evaluated according to Equation 4.1 by means of an interpolation parameter $h \epsilon [0,1]$.

$$r(h) = \sum_{i=0}^{k} M_{i,p}(h) \cdot b_i \qquad 4.1$$

The basis functions $M_{i,p}(h)$ are defined recursively according to Equations 4.2, 4.3 and 4.4.

$$M_{i,0}(h) = \begin{cases} 1 \ \ if \ h_i \leq h < h_{i+1} \\ 0 \qquad otherwise \end{cases} \qquad 4.2$$

$$M_{i,p}(h) = \frac{h - h_i}{h_{i+p} - h_i} \cdot M_{i,p-1}(h) + \frac{h_{i+p+1} - h}{h_{i+p+1} - h_{i+1}} \cdot M_{i+1,p-1}(h), \qquad p \geq 1 \qquad 4.3$$

$$M_{m-p-1,p}(1) = 1 \qquad 4.4$$

In Equation 4.3 the two fractions are interpreted as zero whenever the denominator becomes zero.

The $m + 1$ scalars h_i are called knots, where $m + 1 = p + k + 2$. They are specified prior to setting up the B-spline. The first $p + 1$ knots are zero and the last $p + 1$ one. The remaining knots are usually equally spaced. A general representation of the knot vector $\underline{\mathcal{H}}$ is shown in Equation 4.5

$$\underline{\mathcal{H}} = [0, \dots, 0, h_{p+1}, \dots, h_{m-p-1}, 1, \dots, 1]^T \qquad 4.5$$

4.3.3.2. Cubic B-spline Representation

(Hinton et al., 2003) give a closed form description of a cubic B-spline representation with $k + 2$ control values b_i. It has been derived as a connection of several splines where four knots build one segment, i.e. the number of segments depends on the number of control values. Here, the knots are equally spaced. The control values interpolating k discrete values p_i can be obtained by solving the following set of linear equations.

$$\begin{bmatrix} a & b & & & & & \\ 1 & 4 & 1 & & & & \\ & & \ddots & & & & \\ & & 1 & 4 & 1 & & \\ & & & & \ddots & & \\ & & & & 1 & 4 & 1 \\ & & & & & c & d \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_i \\ \vdots \\ b_{k-1} \\ b_k \end{bmatrix} = \begin{bmatrix} e \\ 6 \cdot p_2 \\ \vdots \\ 6 \cdot p_i \\ \vdots \\ 6 \cdot p_{k-1} \\ f \end{bmatrix} \qquad 4.6$$

The end-conditions of the spline determine the parameters a to f as well as the first and the last control values b_0 and b_{k+1}. Either the tangents at the ends of the spline are specified or the natural spline option is chosen. The latter induces zero curvature at the ends. The parameters a to f as well as b_0 and b_{k+1} can be determined from Table 4.1.

Once the control values have been computed the spline can be evaluated at arbitrary points. Each segment is evaluated by means of the interpolation parameter h according to Equation 4.7.

$$r(h) = \underline{h}\, R\, M \qquad 4.7$$

with $$\underline{h} = [h^3 \quad h^2 \quad h \quad 1], \qquad 0 \leq h \leq 1 \qquad 4.8$$

$$\underline{R} = \frac{1}{6} \cdot \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \tag{4.9}$$

$$\underline{M} = [b_0 \quad b_1 \quad b_2 \quad b_3]^T \tag{4.10}$$

Table 4.1: The end conditions determine the parameters a to f as well as b_0 and b_{k+1}. Either the tangents at the ends of the spline are specified or the natural spline option is chosen.

	Specification of end tangents	**Natural spline**
e	$3 \cdot p_1 + p_1{}'$	p_1
f	$3 \cdot p_k + p_k{}'$	p_k
a,d	2	1
b,c	1	0
$\boldsymbol{b_0}$	$b_2 - 2 \cdot p_0{}'$	$2 \cdot b_1 - b_2$
$\boldsymbol{b_{k+1}}$	$b_{k-1} - 2 \cdot p_k{}'$	$2 \cdot b_k - b_{k-1}$

According to Equation 4.7 the first segment evaluated at location $h = 0$ yields

$$r(0) = \frac{1}{6} \cdot b_0 + \frac{2}{3} \cdot b_1 + \frac{1}{6} \cdot b_2 \tag{4.11}$$

This means that the evaluated value at the beginning of a B-spline changes if one of the first three control values is altered. In an optimization task where this value is not supposed to change but remain a constant value $\hat{r}_{start}$, the corresponding three control values could not be defined as design variables. Since the fourth control value b_3 in the segment has only a very small influence on the evaluated value near $h = 0$, it can be beneficial to define b_2 as design variable too and update the remaining control values in a way that $r(0) = \hat{r}_{start}$. In the case of a natural spline, Table 4.1 yields Equation 4.12.

$$b_0 = 2 \cdot b_1 - b_2 \tag{4.12}$$

Equation 4.12 inserted into Equation 4.11 yields Equation 4.13 which shows that control value b_1 has to remain equal to the desired start point $\hat{r}_{start}$ throughout the optimization while b_0 has to be updated according to Equation 4.14.

$$b_1 = \hat{r}_{start} \tag{4.13}$$

$$b_0 = 2 \cdot \hat{r}_{start} - b_2 \qquad 4.14$$

Similarly, the last point remains unchanged only if the last three control values in the spline do not change.

$$r(1) = \frac{1}{6} \cdot b_{k-1} + \frac{2}{3} \cdot b_k + \frac{1}{6} \cdot b_{k+1} \qquad 4.15$$

Thus, control value b_k value remains $\hat{r}_{end}$ and b_{k+1} is updated according to Equation 4.17.

$$b_k = \hat{r}_{end} \qquad 4.16$$

$$b_{k+1} = 2 \cdot \hat{r}_{end} - b_{k-1} \qquad 4.17$$

The preceding representation can be used to interpolate one dimensional values such as a thickness along a line. More importantly for shape optimization it can be used to describe a line in two dimensions by using two B-splines: One for the *x* and one for the *y*-coordinate. As discussed before the control values for the two dimensions can be combined in vector form. These vectors are called control points. An example of a B-spline with seven control points describing a line can be seen in Figure 4.5. Since four control points form a segment, the spline consists of four segments.

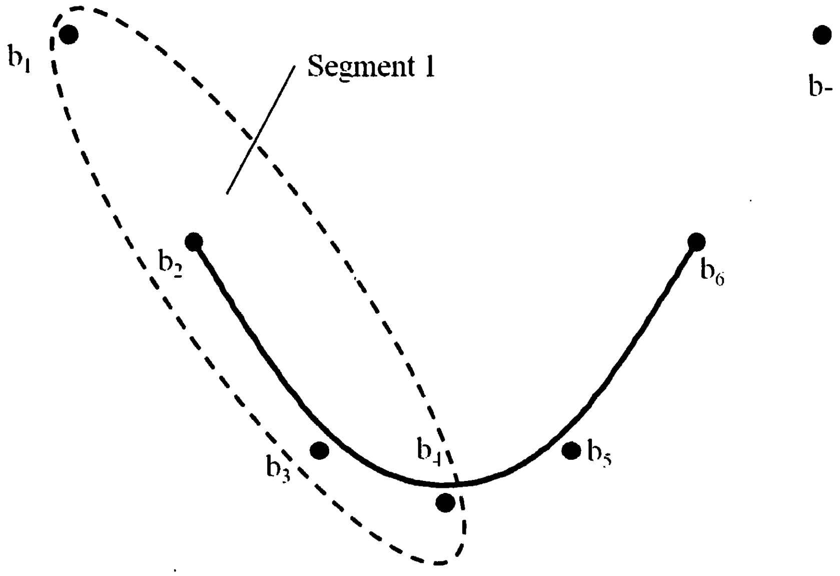


Figure 4.5: Two B-splines with seven control points describing a line. Four control points form a segment, i.e. the spline contains four segments.

The influence of a change of the *y*-coordinate of control point $\underline{b}_5$ is illustrated in Figure 4.6. To prevent the end point of the spline to move, control value $\underline{b}_7$ is updated according to Equation 4.17.

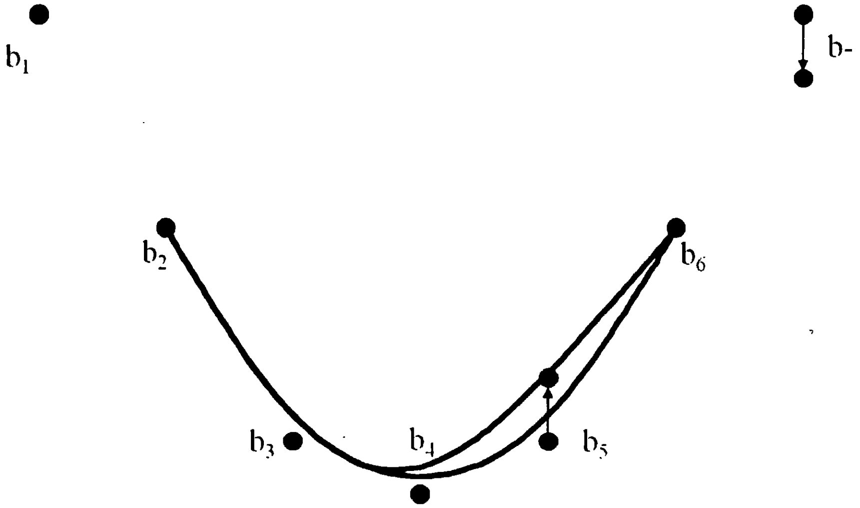


Figure 4.6: Influence of a change of control point $\underline{b}_5$ of the *y*-coordinate B-spline. To prevent the end point of the spline to move, control point $\underline{b}_7$ is updated according to Equation 4.17.

4.4. Optimization Decomposition

The relationship between changes in dimensionality of optimization problems and the resulting computational effort is not a linear one. That is to say, reducing the number of design variables by a certain factor reduces the time required to solve the problem by a higher factor (Haftka & Gürdal, 1992). Effectively this means, that solving two optimization problems with n design variables each is more efficient than solving one with $2 \cdot n$ design variables. Thus, breaking a large optimization problem into several smaller sub problems is an extensively used approach. One of the most commonly used methods is decomposition (Haftka & Gürdal, 1992).

Optimization decomposition requires the split of the design variables into several groups, where the groups are only weakly or not at all interconnected via the objective and constraint functions. The optimization process then contains two levels: A top level, coordinating the process and a subordinate level, with the small sub problems. This structure is denoted as wide tree structure (Haftka & Gürdal, 1992). It is illustrated in

Figure 4.7. Naturally, the sub problems can be split further, such that the process contains more than two levels.

Some problems feature a narrow tree structure, with only one sub problem per lower level. This structure is illustrated in Figure 4.8. A possible example for this is the case where each structural simulation is a minimization of the total potential energy of the structure (Haftka & Gürdal, 1992).

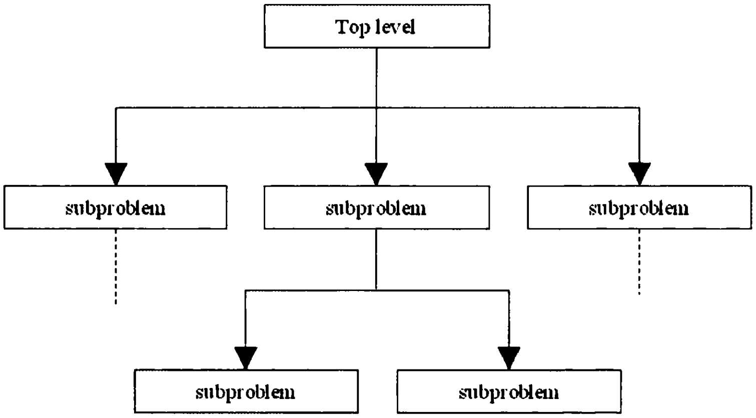


Figure 4.7: Wide tree structure in optimization decomposition.

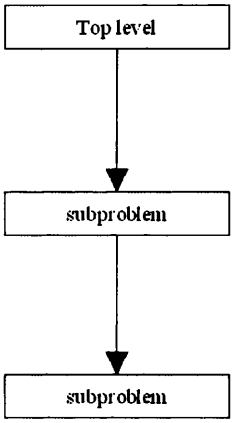


Figure 4.8: Narrow tree structure in optimization decomposition.

When decomposition is applied to optimization problems the design variable vector $\underline{x}$ can be written in terms of s groups

$$\underline{x}^T = [\underline{x}_1 \quad \cdots \quad \underline{x}_s] \tag{4.18}$$

In the simplest case the objective function is a sum of terms where each one depends on one group of design variables only.

$$f(\underline{x}) = \sum_{i=1}^{s} f_i(\underline{x}_i) \tag{4.19}$$

The same has to apply to the constraints, too.

$$g_i(\underline{x}_i) \geq 0, \qquad i = 1, \ldots, s \tag{4.20}$$

In that case, each sub problem can be solved individually since there is no interdependency between the design variable groups. It is very uncommon though that no connection exists at all, but that there is at least a weak link between the groups. In this case the sub problems could be treated as being totally independent. But then, reiteration of the optimization becomes necessary for it to converge to the optimum solution (Haftka & Gürdal, 1992).

A more common situation is the one where a certain number of global design variables $\underline{x}_g$ exist (see Equations 4.21, 4.22 and 4.23)

$$f(\underline{x}) = f_0(\underline{x}_g) + \sum_{i=1}^{s} f_i(\underline{x}_i) \tag{4.21}$$

$$g_0(\underline{x}_g) \geq 0 \tag{4.22}$$

$$g_i(\underline{x}_i) \geq 0, \qquad i = 1, \dots, s \tag{4.23}$$

A simple approach to such classes of problems is a nested procedure where the optimization of the sub problem variables is performed as a nested optimization inside an outer optimization which determines the global variables. A more sophisticated approach coordinates the process at both levels, i.e. the derivatives of the optimum solutions of the sub problems with respect to the global variables are taken into account when updating the global variables (Haftka & Gürdal, 1992).

Existing approaches to optimization decomposition will be outlined in the next section, followed by a new approach for the split of an optimization problem into a shape optimization problem with a size optimization sub problem.

4.4.1. Optimization Decomposition Methods

The following review of decomposition methods is based on (Shan & Wang, 2009). Decomposition methods can be grouped into three main categories: Product decomposition, where the product is partitioned into physical components (Kusiak & Larson, 1995), process decomposition, where the problem is divided according to the flow of elements or information such as the design process (Kusiak & Wang, 1993) or the problem decomposition, where a problem is divided into different sub-problems, often used in multidisciplinary design optimization (Kodiyalam & Sobieszczanski-Sobieski, 2000) (Sobieszczanski-Sobieski & Haftka, 2000).

A relationship matrix is often used which reflects the relationships between the design variables. Two types of relationship matrices can be identified: A design structure matrix (DSM) and a function dependent matrix (FDM). A DSM is a square matrix that has identical row and column listings which represents a single set of objects

(Browning, 2001). A FDM has different row and column listings to represent two sets of objects (Krishnamachari & Papalambros, 1997). The partitioning of the matrix is often formed with tools that fall into three categories. The first group models the decomposition as a hyper-graph (Michelena & Papalambros, 1997), network reliability (Michelena & Papalambros, 1995) or an integer programming problem (Krishnamachari & Papalambros, 1997). The second group are the heuristic methods (see for instance (Wagner & Papalambros, 1993)) and the third group contains the clustering approaches (Chen et al., 2005).

The decomposition strategies are typically characterized as hierarchical or non-hierarchical. Hierarchical decompositions are have a tree structure where no interaction between the blocks exists and non-hierarchical decomposition by a network structure where interaction between the blocks is considered (Renaud & Gabriele, 1991).

An example for the application of decomposition methodologies can be found in (Di Lorenzo et al., 2008), where two splitting approaches are tested. The first one is a technological criterion where the design space is divided into design variables governing the pressure loading and those governing the counterpunch force. The second criterion is denoted as 'gradient criterion' where the three highest partial derivatives of the objective and the three lowest are grouped into two subspaces. The sub problems are solved in a static procedure where the design variables are updated after a sub problem fully converged and in a dynamic procedure where each sub problem performs one steepest descent step and passes the updated design variables on to the other sub problem.

Another example is discussed in (Lee & Lee, 2003). Here, an approach is used where the design variables are divided into global and local variables. Here, the global variables are those that govern global performances such as global stiffness and global strength and the local variables those that govern local performances such as moment of inertia or joint stiffnesses.

4.4.2. A Novel Approach to Decomposition of Coupled Size and Shape Optimization Problems

As discussed in Section 4.2, the design variables in size optimization are usually thicknesses of shell or cross-sectional properties of beam elements. Hence, the parameterization of models with regards to such properties is comparably straightforward since a change of them does not affect the geometry at all. The pre-processor Altair HyperMesh (Altair HyperWorks (a), 2009) and the FE solver Altair OptiStruct (Altair HyperWorks (b), 2009) offer a wide variety of optimization tools one of which supports size optimization. The sensitivities within the software are calculated analytically which makes the process very efficient even for high dimensional problems. In addition to that, for most applications, no parameterization flexibility is sacrificed in comparison to a bespoke in-house optimization tool. After the creation of a regular FE model, the steps to set up a size optimization with HyperMesh (Altair HyperWorks (a), 2009) bear little extra effort. For instance, in case of a thickness sizing of a model meshed with shell elements, these steps are merely defining a set of design variables and assigning them to the corresponding shell properties. This makes the use of commercial optimization tools very beneficial especially when they are directly incorporated in the pre-processor and the solver.

Contrary to that, the parameterization for shape optimization is not as straightforward since the parameters govern the geometry outline. As mentioned before Altair HyperMesh (Altair HyperWorks (a), 2009) offers the possibility to morph meshes and set up a shape optimization problem but the generalization of shape optimization to arbitrary structures naturally leads to a sacrifice in flexibility and a more complicated set-up. For the sake of flexibility, especially for complex geometries, it can be beneficial to develop a bespoke optimization tool that determines the optimum shape of the structure. Since size and shape optimization are often performed simultaneously this leads to a decomposition of the problem because two different optimization tools are to be employed.

As discussed before, decomposition usually requires the investigation of the objective and the constraint functions' dependencies upon the design variables in question. The following approach makes this investigation redundant. In this approach the size optimization is a sub problem and the shape optimization the top level problem. The size optimization is solved by means of an efficient optimization tool such as Altair OptiStruct (Altair HyperWorks (b), 2009) and the shape optimization by means of surrogate based optimization. That is to say, each simulation that the surrogate model is based on is a full size optimization. Each size optimization is subject to all constraints that depend on its design variables. That means all those constraints become irrelevant in the top level optimization since they are a priori satisfied as long as a feasible solution exists. Hence, no additional approximation model for those eliminated constraints needs to be set up. This improves the efficiency further. The proposed decomposition strategy is very beneficial for a high number of size parameters for instance for the optimization of a composite structure.

The flow of this strategy can be described as follows: In a first step a DoE is created for the n_{DV}^{shape} shape design variables in the problem. Then, for each of the n_P experiments in the DoE an optimization of the n_{DV}^{size} size design variables is performed. The responses from these are used to create an approximation model for the objective function and those constraints that are not a priori satisfied. This approximation model can be evaluated for an arbitrary set of shape parameters and the obtained response corresponds to an optimum set of size parameters. An optimization of the approximation model yields an optimum set of shape variables. To obtain the corresponding set of optimum size variables, a final size optimization is performed. If the model would be perfectly accurate the objective function at the end of this size optimization would be identical to that obtained from the evaluated approximation model. If the values differ to an unsatisfactory degree, another run with an increased number of experiments should be performed. The flow of the strategy is outlined in Figure 4.9.

The proposed strategy is applied within the carbon-fibre bicycle frame optimization in Chapter 6. Here, the size optimization design variables are the composite ply thicknesses and the shape design variables determine the shape of the tubes in the frame.

4.5. Process Automation

The main goals of automating processes in the context of engineering design are the reduction of costs and design cycle times (Keane & Nair, 2005). To this end structural models are parameterized to allow for easy and quick design changes. This helps in eliminating repetitive and time consuming pre-processing tasks.

A parameterized and automated model creation scheme is also a prerequisite for optimization processes because there, simulations for different designs have to be performed, where the design parameters are determined in an automatic manner. One of the greatest benefits of the set-up of an optimization process is often the automated

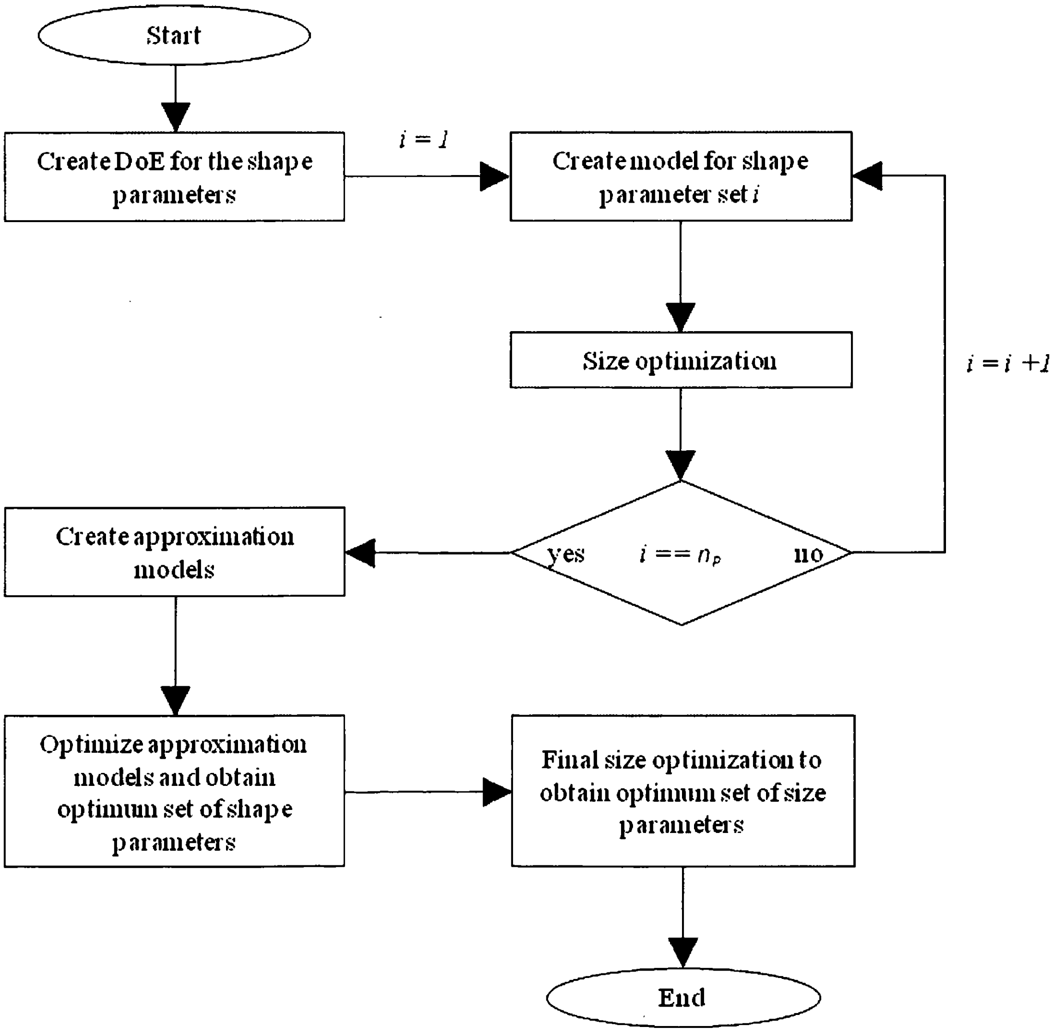


Figure 4.9: Flow of the proposed decomposition approach. The size optimization is a sub problem which is solved by means of an efficient optimization tool. The shape parameters are determined by means of a surrogate model where each simulation is a full size optimization.

simulation itself (Keane & Nair, 2005). Even if optimization is not performed regularly the simulation process is simplified significantly.

In the literature three main approaches towards process automation in engineering design can be identified. They generally differ in the level on which the model parameterization takes place. These methods will be discussed in Section 4.5.3. In any case, the tools need to be controlled in a batch mode manner to fully automate the procedure and to allow for the integration in an optimization framework. That means the typical GUI needs to be avoided. A detailed discussion on software control will be given in Section 4.5.2. Finally, the two methods used within this thesis will be presented in Sections 4.5.4 and 4.5.5. Firstly though, some necessary background information on the basic differences between interpreted and compiled languages are outlined.

4.5.1. Interpreted versus Compiled Languages

A good introduction to scripting languages (also denoted as interpreted languages) and their differences to system languages (also denoted as compiled languages) can be found in (Ousterhout, 1998). System languages such as Fortran, Algol, C, C++ and Java appeared as an alternative to assembly languages where each statement represents a certain action of the machine that it is running on. In system languages a compiler takes over some of the work by translating instructions into binary code. The fact that part of the work is done automatically led to the notion of 'higher level languages'. On the other hand, this automation makes them less efficient than code written in assembly languages. In another step towards programming simplification, scripting languages evolved. They are generally not used for large and complex programs but for the extension of existing code. Most scripting languages do not require the declaration of variable types, such as integer or floating point variables, which makes the coding faster and the code more flexible. The main difference to system languages is that they are not compiled but interpreted, which means they are not translated into binary code prior to execution but remain in text form. Upon execution, an interpreter reads every statement and performs the corresponding action. All these simplifications compared to system languages, lift scripting languages another level higher as the number of machine instructions per statement increased significantly leading to shorter code. But just as the higher level of system languages compared to assembly languages came at the price of reduced efficiency, it does in this case, too. System languages can often be 10 to 20 times faster than scripts while scripts can be developed 10 times faster than compiled software. This is the reason why (Ousterhout, 1998) concludes that scripting languages are not a substitute for system languages. The flexibility and simplicity make it perfect for software control and extension while the superior efficiency makes system languages ideal for complex algorithms. Another recommendation in (Ousterhout, 1998) is to use both language types in a symbiotic way, meaning to use them in environments where they tend to excel. A more recent publication (Loui, 2008) takes a similar stand and lists the similar advantages of interpreted languages while admitting the necessity for compiled ones for CPU-intensive tasks. Many of the previous characteristics and differences are confirmed by (Prechelt, 2000). Here, the same software has been written in different languages by many different programmers and is compared in terms of efficiency, memory consumption, development time and code length. It was found that C and C++ can be up to 10 times faster than the tested scripting languages and require twice as much memory but also that the development is twice as time consuming and the code is about two to three times as long.

4.5.2. Software Control

Virtually all sophisticated CAE tools offer the possibility to operate them via commands rather than clicking buttons or inserting numbers into text boxes. These commands are written to files and are processed successively when loaded. They can either be loaded from within the GUI or in batch mode from the command prompt. Any possible operation can be performed that way. This also means that no user interaction is possible once this procedure is started so that all operations have to be planned in advance. There are several advantages from this way of software control. Most importantly, it allows for process automation, i.e. tedious tasks can be performed automatically. Furthermore, many CAE tools do not have an 'undo' function. For instance when a step in the model creation does not lead to a satisfactory result it can be advisable to repeat the entire procedure. Often this is only feasible if written commands are used. Another important aspect is that it allows to remotely use a tool from within a different master tool. This is a prerequisite for optimization processes where the CAE tools are mere function evaluators which have to be called repeatedly to create and solve models that vary only by details or for software coupling where several programs have to interact which might not provide an appropriate interface.

Most CAE tools incorporate an interpreter for at least one scripting language. Widely supported languages are Python (e.g. Abaqus and ANSYS) and TCL/TK (e.g. Altair HyperWorks (Altair HyperWorks (a), 2009)). The CAE tools extend the languages by specific commands and routines which allow to ‘modify’ and ‘query’. The former can be used to call a function in the software similar to using the GUI and the latter to access the database to query information. Examples in an FE context for the two categories could be the commands to mesh a geometry and to query the number of nodes.

Often the programs offer the possibility to record performed steps in a script, i.e. once the 'record-button' is pressed, all actions are saved in the form of commands in a text file. Loading this text file at a later stage will execute the exact same functions. With the help of the software manual the parameters of each command can be decoded and altered for another run. At the same time, by default, most CAE tools record everything the user does in a command file. These commands can be extracted and rerun to end up at the same state as before. For instance, when the system crashes the user does not have to start from the last saved status. Furthermore, this feature can be deliberately used to create macros or entire programs. Starting with a clean command file the user performs a step and saves the corresponding command in another location. Again, the program's user manual helps in understanding the commands' structures and for reproducing them.

There are a some basic differences between scripting and command files. Firstly, the syntax varies. For instance in Altair HyperWorks the commands' names as well as the parameters are identical. They only differ in the fact that the scripts do not contain any brackets or commas and end with a semicolon. The parameters of commands in the command files on the other side are written in between brackets and are separated by commas. A new command is indicated by an asterisk. More importantly, contrary to a command file, a script can contain expressions of the underlying scripting language like loops and logical comparisons. Furthermore, the database can be accessed from within a script, while the command files can only execute modifying functions.

4.5.3. Parameterization Level

As mentioned before, the three main approaches to process automation differ in the level on which the parameterization takes place. It can take place on the Computer Aided Design (CAD) level, the Computer Aided Engineering (CAE) level or on the level of a specific parameterization tool.

4.5.3.1. Parameterization on CAD Level

Within the first approach the CAD model has to be converted into a CAE model whenever a simulation needs to be performed. For instance (Mencaglia et al., 2009) link Catia with Abaqus using two commercial software tools: CADNexus Capri for the handling of the CAD file and Simmetrix as the meshing tool. Their approach is applied for the multi-disciplinary optimization of composite structures. (Merkel & Schumacher, 2003) use a parameterized Unigraphics CAD model and MSC/Patran as the pre-processor for the automated simulation of an automotive part. The link between the two is realized using the PARASOLID approach.

4.5.3.2. Parameterization on CAE Level

In the second approach, the parameterization takes place directly on CAE level, i.e. no interaction between two tools is required. To this end the model generation capabilities of the CAE tool have to be used which need to be sufficiently powerful for the problem in question. An example for this approach can be found in (Gassara et al., 2008) where a sheet metal L-bending process is optimized. The parameterized model creation and pre-processing are both performed within Abaqus. Thus, no transfer between CAD and CAE tool is necessary. A Python script controls Abaqus and performs the optimization at the same time.

4.5.3.3. Parameterization on Parameterization Tool Level

In the third approach another tool is introduced into the simulation process chain. This is a bespoke parameterization or process integration tool. It either serves as a pre-processor where a parameterized CAE model is generated and links it directly into the commercial simulation tool or it uses an existing CAE model and supports the user in the extraction of the desired parameters. In some cases additional functionality such as optimization features are included. Effectively, here too, the parameterization takes place on CAE level. Hence, this approach can be seen as a special case of the second approach, only here, the pre-processing and parameterization capabilities of the CAE tool are extended.

For instance in (Hilmann et al., 2005) the software SFE Concept is used to create the FE model which is solved by RADIOSS. SFE Concept is a specialized tool for the efficient creation of parameterized CAE models. Another example can be found in (d'Ippolito et al., 2009) where a software called OPTIMUS is used for the optimization of multi-body models. This tool is used for the parameterization, geometry and mesh update as well as optimization. The CAE model is created and solved by means of Virtual.Lab Motion.

The first two methods both have advantages and disadvantages. (Mencaglia et al., 2009) stresses that the parameterization capabilities of CAE tools are commonly more limited than those of CAD tools and that parameterization work done on CAE level is non-reusable and has to be redone for each new model after changes occurred in the design. Furthermore, the authors argue that the design has to be recreated in a CAD environment after the design process to make it available to everyone in the product development. Especially the last two points are valid in an CAD-centric environment where several departments work on the same design and interaction between them requires a standard format. This approach is not always practical though. (Schramm,

2009) calls the iterative loop, utilizing CAD and CAE methods until convergence, 'the classical design process'. The author sees a new paradigm in performing all design iterations on CAE level where CAD is only used at the end for performance and manufacturing validation. According to (Bartholomew & Morris, 2008) the potential for design optimization to contribute to the performance and profitability of the resulting product increases with an earlier introduction of CAE into the design process. It is admitted though that for most designers, the design remains a CAD-led activity.

Naturally, creating the parameterized geometry within a CAE tool that directly converts it into the numerical model, has the benefit that no interaction between two tools is necessary. For instance (Merkel & Schumacher, 2003) highlight the difficulties in setting up the link between CAD and CAE tools. Within this thesis, the parameterization will take place on CAE level.

4.5.4. A Mixed Approach to Process Automation

As mentioned before, Process Automation describes the procedure of automating several steps in processes or the entire process by developing software that takes over the tasks that had to be done manually. This usually involves software that was controlled via a GUI and is then remotely accessed in batch mode.

When the parameterization takes place on the CAE level, the advantages of scripting such as fast and easy code generation and amendment as well as the possibility of directly accessing other software's features and databases, makes it tempting to create the desired procedure entirely in the supported scripting language. The one significant disadvantage of scripts can be the inferior run time. This inferiority becomes more and more significant the smaller the fraction between the uses of CPU intensive external functions in binary code to those inside the script gets. In many applications involving automated parameterized simulations such CPU intensive custom made functions are necessary.

As mentioned before, the external functions are divided into two categories: 'Modify' and 'Query' commands. The former are used to execute the software's features and the latter to access the database that is built by those actions. They are a typical example of a field of application where scripting languages should be used. The functions called by these commands execute external sophisticated routines that are written in binary code and thus are very efficient. The latter though, can yield large amount of data that the calling procedure needs to process. This can lead to long nested loops that easily become very time consuming in a scripting language. Following the recommendation of (Ousterhout, 1998) to use scripting and system languages in areas where they excel, it is proposed to use a scripting language for the execution of the 'Modify' commands and a system language for the processing of the databases' content. Naturally, this makes it necessary to export the required data from the external software into the processing one.

Another advantage apart from the speed is the modularization of the procedure. If the software to be controlled shall be substituted by a different one, all the 'Query' commands would change and perhaps even the supported scripting language. In the proposed framework, all that needs to be recoded are the script-writing and import-routines, while the data-processing remains the same. This approach respects the areas of expertise of scripting and system languages while the modular structure offers great flexibility in the choice of the external software.

In general the flow of the proposed approach to process automation can be described as follows: The master program that manages the process and controls the slave program (in this case the FE pre-processor) is written in a compiled programming language. It writes a file that contains all the commands for the model generation until it requires information about the created geometry, for instance indices of nodes in a certain location. Then, it calls the slave program which executes the commands and exports the data at the end. The required information from this data is read by the master tool and the next command file is written. This process is repeated until the model creation is finished. The flow is illustrated in Figure 4.10.

The proposed approach to Process Automation has been applied in two industrial examples. These are the carbon-fibre bicycle frame optimization in Chapter 6 and the optimum wind turbine blade design in Chapter 7.

4.5.5. Direct Process Automation

In cases where the geometry and its parameterization is straightforward to create or where financial constraints rule commercial and sophisticated pre-processing software out, it might be necessary to develop bespoke software tools that directly write the FE input file. Public domain meshing tools might be incorporated in this process to convert the geometry into an FE mesh. Naturally, due to long development times this should be avoided wherever possible. Within this thesis this approach has been adopted for the

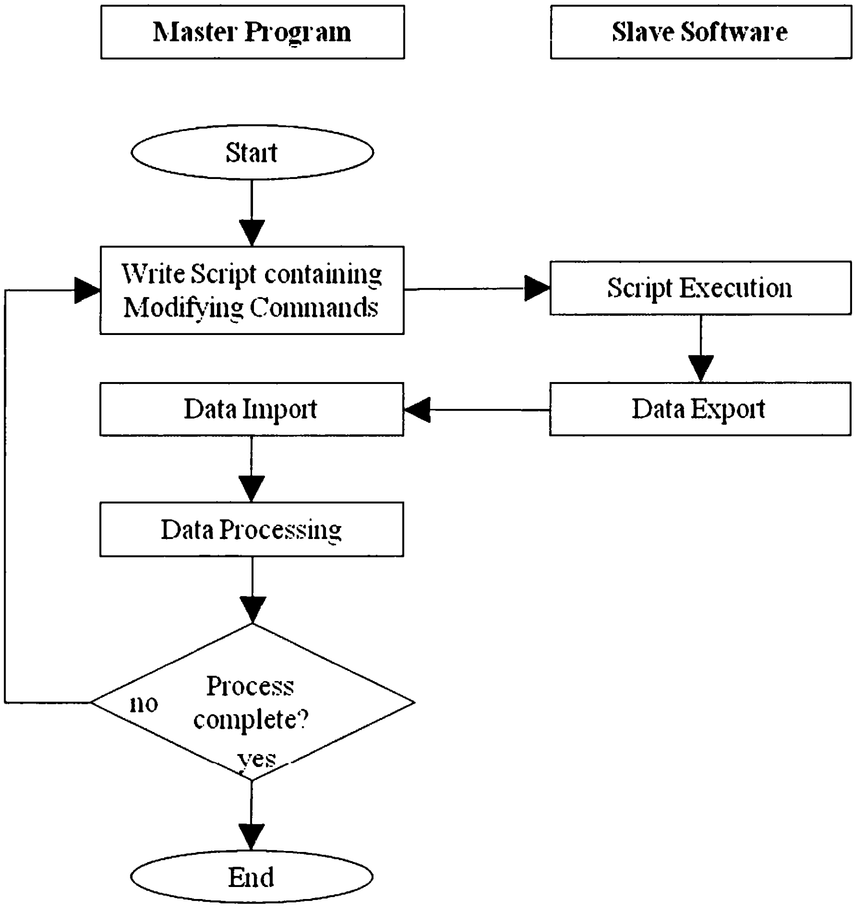


Figure 4.10: Flow of the proposed programming approach to software control in process automation. Here the master program is developed in a system language issuing scripts which are read and executed by the slave software.

industrial example in Chapter 5. The FE model for the ringstiffness test and for the buried simulation are both created without the use of commercial pre-processing tools. In case of the buried pipe simulation a public domain meshing tool is used.

4.6. Concluding Remarks

This chapter has given an overview of various aspects of the optimization model that are of importance to the work carried out in this thesis. The three main optimization methods (size, shape and topology optimization), different approaches to the parameterization in shape optimization and aspects of optimization decomposition as well as automation of the simulation model creation process have been discussed.

A novel approach for the decomposition of large coupled size and shape optimization problems has been developed and discussed in detail in Section 4.4.2. Here, a sophisticated optimization tool is used to efficiently solve the size optimization as a sub problem, while the top level problem, which determines the shape parameters, is solved by means of a surrogate model. The advantage of this approach is shown in Chapter 6 where it is applied in the optimization of carbon-fibre bicycle frames.

An approach to process automation has been proposed in Section 4.5.4. It is applied in Chapters 6 and 7. Here, a compiled programming language is used to write command files that are interpreted in batch mode by external pre-processing tools to automatically create the simulation model.

5. A Software Tool for Automated Simulation and Optimum Design of Structured-Wall PE Pipes

Summary: This chapter presents a software tool that is developed as a hands-on decision guidance for many occurring design situations for structured wall HDPE pipes. In order to avoid licensing fees only public domain software or in-house code is used in developing the tool. It offers the efficient and automated simulation of the ringstiffness test as well as pipe installation scenarios. In addition, an optimization feature is implemented for the design of optimum pipe profiles with regards to the ringstiffness test. Here a simplified model is used which contains only one box-section with a horizontal spring support. To improve the accuracy of the simulations, material tests are performed and a guideline for the extraction of a secant modulus from the obtained data is developed. For the accurate description of the profile geometry a feature was set up that allows for the generation of shell element representations based on laser-scan geometry data. The application of the developed tool to various design problems will show its validity and usability.

5.1. Introduction

Companies whose products are expensive and difficult to manufacture often take advantage of extensive simulation and sometimes even optimization tools. For instance the choice between building a prototype of a car only to use it in a crash test and simulating this process on the computer is an obvious one. Only in the final stages of the product development the real life tests are the preferred choice. When products are relatively cheap and easy to be produced though, the use of simulation tools is rarer and an experience based or trial and error product development is a more common approach. This is because knowledge about the necessary simulation tools is missing and the financial and time benefits are not as drastic and obvious. Also, the cost of sophisticated and user-friendly tools is another drawback. On the other hand, the use of public domain programs is often cumbersome and not particularly intuitive. One way of introducing simulation tools to such companies could be to offer license free public domain software, but to simplify their use by automating processes with few parameters to determine and to make them as efficient as possible.

The production complexity and costs of the structured wall pipes (see Figure 5.1) studied within this chapter falls into the latter of the two company types. At the same time the geometry of the pipes is simple and the loading scenarios have a high rate of recurrence. Thus, in this case process automation is feasible and desirable.

Figure 5.1: A Weholite pipe being transported by a forklift truck.

Within this piece of work, easy to use and efficient simulation and optimization tools are sought to be developed for the guidance in the majority of steps during the pipe design. To avoid licensing fees only public domain software or in-house code will be used. For improved user-friendliness a Java based graphical user interface (GUI) will be developed from which the underlying programs are controlled.

A key quality control measure is the ringstiffness to BS EN 1446: 1996 (British Standards Institution, 1996). The ability to predict this accurately as a function of the pipe wall geometry is a pre-requisite for optimization of the pipe design and during the manufacturing process. In addition to that, a thorough understanding of the pipe's behaviour during and after underground installation is of great importance. Thus, the accurate simulation of the ringstiffness test and the installation process for the buried pipes are vital in their design. These are the main aspects that the developed tool will provide. It has been developed in close collaboration with Asset International Ltd. based in Newport, Wales and KWH Pipe Ltd. based in Vaasa, Finland.

This chapter is structured as follows: In Section 5.2 a general outline of the software structure is given. In Section 5.3 some background information on the studied structured wall pipes is given, i.e. the production, structural behaviour, typical fields of application and typical loading scenarios, followed by data from conducted material tests as well as a guideline on how to extract a secant modulus which is dependent upon strain, strain rate and temperature in Section 5.4. The material shown there is the one used at KWH. For data from the material used at Asset the reader is referred to Appendix B. Subsequently, in Section 5.5, the approach for the automated simulation of the ringstiffness test is presented. Here also the profile optimization method is discussed. In Section 5.5.2 the approach to the automated simulation of the pipe installation is presented. Section 5.7 concludes this chapter.

More information on the use of the developed tool can be found in the user manual provided on the attached DVD.

5.2. Software Structure

A Java based Graphical User Interface (GUI) has been developed that allows for the convenient execution of all available functions in the tool. It is divided into three main category tabs. These are 'profile creation', 'ringstiffness test' and 'buried pipe'.

In the first tab an FE representation of a profile cross-section can be generated. To this end a midsurface and thickness distribution have to be created. Here, three methods are supported. They are either extracted from an ASCII file containing a point cloud from a laser scan (see Section 5.5.1.1) or a DXF file containing a drawing of a profile (see Section 5.5.1.2). Also, a rectangular box-section can be created by specifying width, height and thicknesses (see Section 5.5.1.3). The second tab contains the ringstiffness functionality, i.e. the automated FE simulation using one of the previously created profiles. A detailed discussion of this functionality is given in Section 5.5.1.4. This tab also contains the feature for the optimization of the profile, which minimizes the expended material while meeting a stiffness and stress constraint. This feature is explained in Section 5.5.2.2. The third tab allows for the access to the pipe installation simulation which can be used to estimate deflections, stresses and strains in pipes and soils for various installation and loading scenarios. Furthermore, different model complexities can be chosen to improve the efficiency or the accuracy. The buried pipe simulation is presented in Section 5.6. The structure of the developed software is illustrated in Figure 5.2 and a screenshot can be seen in Figure 5.3.

In order to reduce development time and improve reliability, wherever possible, open source and public domain programs are utilized. The simulation and optimization of the ringstiffness test uses FEAPpv (Berkeley University, 2005) as the FE solver. The obtained results are displayed with Paraview (Kitware Inc., 2008). For the simulation

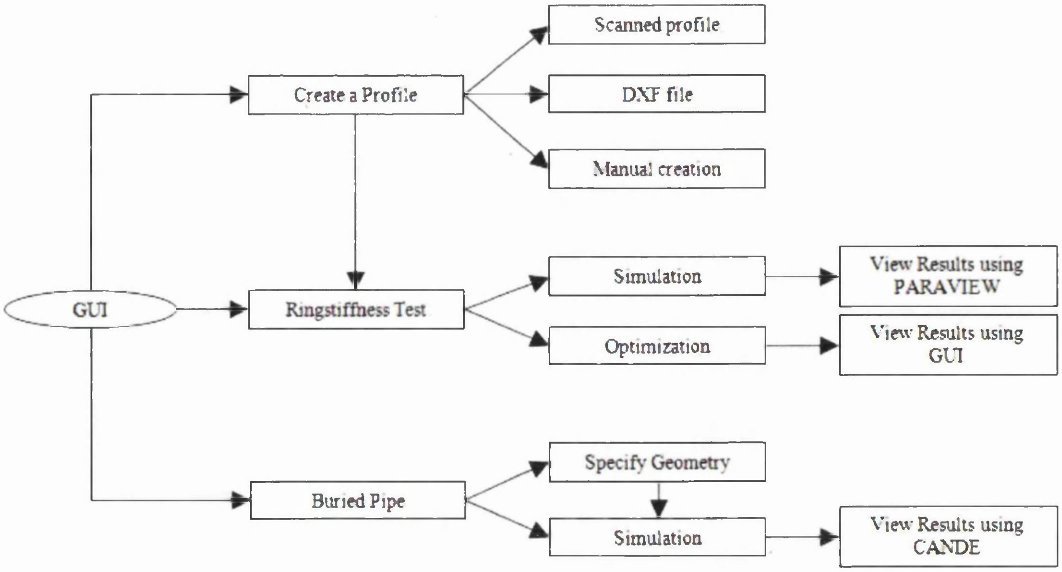
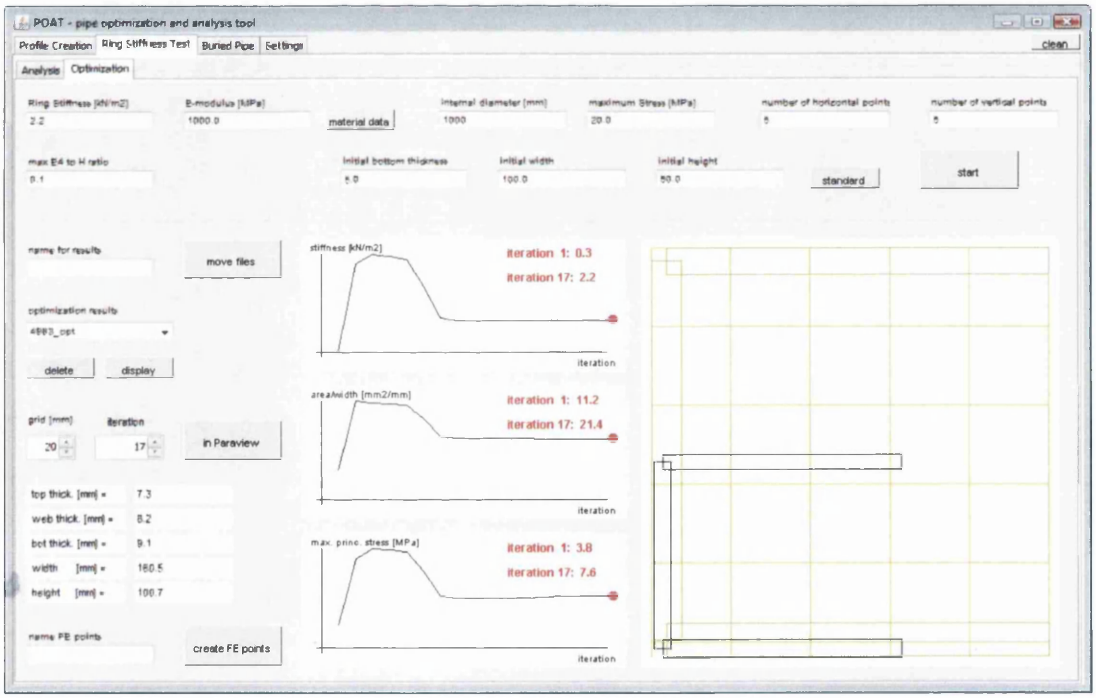


Figure 5.2: Structure of the developed software.



Ring Stiffness [kN/m2]: 2.2
E-modulus [MPa]: 1000.0
internal diameter [mm]: 1000
maximum Stress [MPa]: 20.0
number of horizontal points: 5
number of vertical points: 5
max E4 to H ratio: 0.1
initial bottom thickness: 5.0
initial width: 100.0
initial height: 50.0
optimization results: 4993_opt
grid [mm]: 20
iteration: 17
top thick. [mm] = 7.3
web thick. [mm] = 8.2
bot thick. [mm] = 9.1
width [mm] = 180.5
height [mm] = 100.7

stiffness [kN/m2]
iteration 1: 0.3
iteration 17: 2.2

area/width [mm2/mm]
iteration 1: 11.2
iteration 17: 21.4

max. princ. stress [MPa]
iteration 1: 3.8
iteration 17: 7.6

Figure 5.3: Screenshot of the developed GUI. The selected tab shows the ringstiffness optimization feature.

and the post-processing of pipe burying processes CANDE 2007 (Transportation Research Board, 2007) is used. The open-source in-house meshing tool SWAN (Sienz & Hinton, 1993) is used for the generation of the FE mesh for the buried pipe simulation. More information on the use of the developed tool can be found in the user manual provided on the attached DVD.

5.3. Weholite High Density Polyethylene Structured Wall Pipes

5.3.1. Pipe Production

The production of the structured wall pipes studied within the present work starts by melting high density polyethylene (HDPE) granules and forming the liquid mass into the desired box profile inside a vacuum chamber (see Figure 5.4). This profile is conveyed along the production line where it is gradually cooled down (see Figure 5.5).

Figure 5.4: The HDPE granules are melted and using vacuum pressure formed into the desired profile.

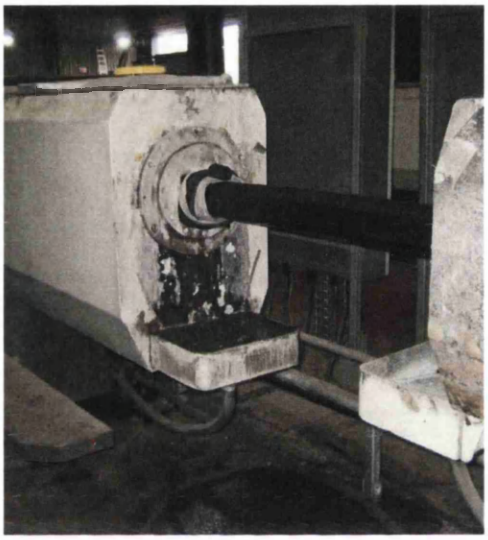


Figure 5.5: The extruded profile is cooled down until it reaches the desired temperature for winding it onto the mandrel.

When the material reaches the correct temperature it is wound around a rotating mandrel where it is continuously welded to the adjacent profile (see Figure 5.6). The process forces the profiles to move horizontally along the mandrel, i.e. the pipe grows on the mandrel (see Figure 5.7).

A box profile after extrusion is depicted in Figure 5.8. The theoretical dimensions of the box-profiles are the width w, the height h and the wall thicknesses t_1, t_2 and t_3. They are illustrated in Figure 5.9.

The profiles are welded on the inside and outside surface. Depending on the profile's height, a gap can occur in between the individual boxes. A welded profile which was cut from a pipe, with its inside and outside welds as well as the gap between them highlighted, can be found in Figure 5.10. Comparison of Figure 5.9 and Figure 5.10 shows that, due to a curvature of the top and bottom walls, the shape of the theoretical rectangular box-sections often significantly differs from its real produced counterpart.

Figure 5.6: The extruded HDPE profile is wound around the mandrel and continuously welded to the adjacent one.

Figure 5.7: The pipe grows on the rotating mandrel and is gradually moves along its axis.

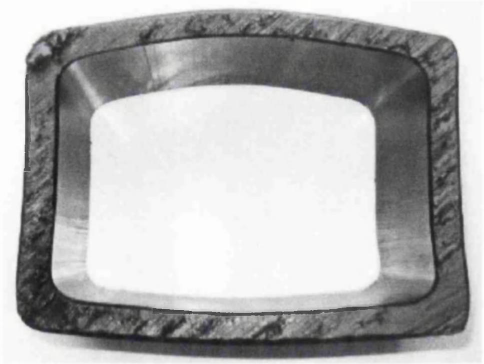


Figure 5.8: An extruded profile before welding.

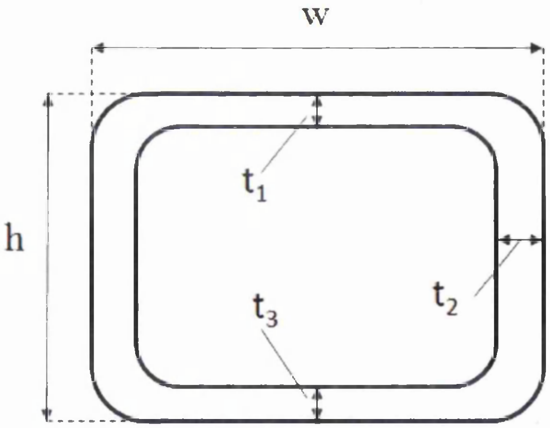


Figure 5.9: Theoretical dimensions of box-profile: Width w, height h and wall thicknesses t_1, t_2 and t_3.

Experience showed that in some cases differences of up to 20% can occur. Difficulties in controlling the actual dimensions of the extruded box-section as well as changes to the shape of the profiles during winding and welding lead to these changes. Furthermore, the web thickness t_2 does not include the additional material due to welding.

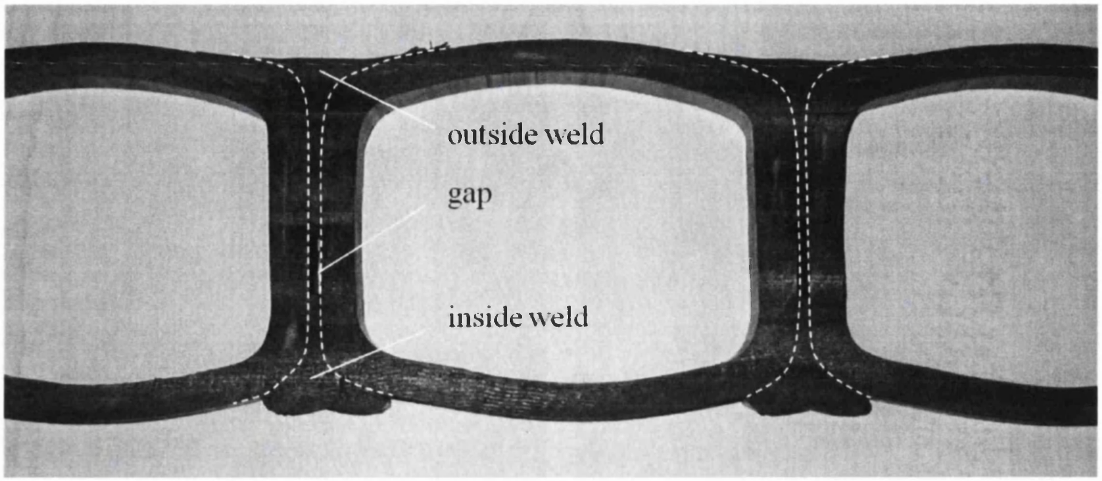


Figure 5.10: A pipe cross-section. The extruded profiles are welded on the inside and outside surface . The inside and outside welds as well as the gap between them.

Due to the continuous nature of the manufacturing process, the profiles are arranged spirally, i.e. the location of a profile along the pipe's axis is offset by the width of a profile. This arrangement can be seen in Figure 5.6. The available pipes range from 360 to 3500 mm internal diameter.

5.3.2. Flexible versus Rigid Pipes

Generally pipe materials are classified into two major groups: Flexible and rigid materials. Pipes classified as flexible can deflect at least 2% of its diameter without exhibiting any plastic deformation. Example materials are concrete or clay for rigid and plastic or steel for flexile pipes (Moser, 2001). The two categories differ in the way they withstand the imposed loads in buried conditions. While rigid pipes tend to carry the majority of the dead and live loads themselves, flexible pipes transfer a significant part of the loads to the surrounding soil. This is due to the fact that, as the soil tries to follow the deflected pipe, it is restricted by the shear resistance of the surrounding soil (Samaras et al., 2008). This phenomenon is usually referred to as arching (Moser, 2001). The vertical deflection of the pipe is further reduced to some extend by the horizontal resistance of the backfilling material as a vertical deflection of the pipe's crown would lead to a horizontal expansion of the pipe (Samaras et al., 2008).

Following the results of the study 'Design of Buried Thermoplastic Pipes', performed by the European Plastics Pipe and Fitting Association (TEPPFA) and the Association of Plastics Manufacturers in Europe (APME), the flexibility of pipes is advantageous as it reduces the loading on the pipe. Thus, external loading has little effect on the pipe deflection (The European Plastics Pipe and Fitting Association, 1999). That is to say, failure due to high loading is more likely to occur with rigid pipes. The study suggests that the most important factor governing the deflection of flexible pipes is the quality of the installation. The relative contributions to the pipe deflection are 80% for the installation quality and 15% for the height of pipe cover. The pipe's stiffness and the pipe material only have a negligible contribution (The European Plastics Pipe and Fitting Association, 1999).

5.3.3. Fields of Application

The HDPE pipes studied in this work are used to convey or store liquids or gases under or above ground or water in low pressure applications up to 1.5 bar of internal pressure (Asset International Ltd. (b), n.d.). Typical fields of application are storm water attenuation, culverts, surface drainage, gravity sewers, pipelines (see Figure 5.11) or storage solutions (see Figure 5.12).

Figure 5.11: 3.0m diameter pipeline in British Columbia, Canada.

Figure 5.12: 1.8m diameter storage tank in Kent, UK.

5.3.4. Ringstiffness Test

A key quality control measurement is the ringstiffness test to BS EN 1446: 1996 (British Standards Institution, 1996). In the following section, its procedure will be presented, followed by sources for variability of the test results.

5.3.4.1. Test Procedure

In the ringstiffness test a pipe sample is placed in between two parallel horizontal plates and compressed up to a vertical deflection equal to 3% of the pipe's inner diameter. The machine monitors the force (*F*) that is necessary to move the top plate with constant velocity. The plate velocities, dependent on the nominal diameter d_n are listed in Table 5.1. The nominal diameter is the average between the outer and the inner diameter.

Table 5.1: Deflection speeds depending on the nominal diameter d_n of the pipe.

d_n [mm]	Deflection speed [mm/min]
$d_n \leq 100$	2 ± 0.4
$100 \leq d_n \leq 200$	5 ± 1
$200 \leq d_n \leq 400$	10 ± 2
$400 \leq d_n \leq 1000$	20 ± 2
$d_n > 1000$	50 ± 5

According to the standard, the ringstiffness (*SN*) is calculated via Equation 5.1.

$$SN = \frac{\left(0.0186 + 0.025 \cdot \frac{y}{d_i}\right)}{y \cdot L_{sample}} \cdot F \qquad 5.1$$

where y is the vertical deflection, L_{sample} is the length of the pipe sample, and d_i is the inner diameter (all values in mm). The length of the pipe sample shall be chosen according to Table 5.2. For spirally wound pipes such as in the present case, the sample should also at least be five times as wide as the width of a single profile. A pipe in the ringstiffness machine is depicted in Figure 5.13.

Table 5.2: Required pipe sample length L_{sample}.

d_n [mm]	Pipe sample length L_{sample} [mm]
$d_n \leq 1500$	$L_{sample} = 300 \pm 10$
$d_n > 1500$	$L_{sample} \geq 0.2 \cdot d_n$

At the time of testing the pipes should be at least 24h old and kept at a constant temperature of 23°C (allowable tolerance: ±2°C).

5.3.4.2. Sources for Variability of Test Results

Processing Variability

The nature of the pipe production leads to pipes that are prone to a certain variability of the profile shape. For instance experience shows that the condition of the mandrel's surface has a high impact on the amount of material that is required to produce a certain

Figure 5.13: Pipe placed in the ringstiffness machine. The pipe is compressed in between two horizontal plates up to a vertical deflection of 3% of the internal diameter.

length of pipe. This is due to the amount of friction between pipe and mandrel, i.e. as the friction increases more material is ‘squeezed’ into the pipe. It was found that polishing the mandrel prior to production, i.e. removing rust and dirt has a significant impact on the material expenditure. Also the profile geometry is influenced by the surface condition of the mandrel. Needless to say, two pipes with different amounts of material per unit length or different profile geometries have a different structural behaviour. Since the mandrel is not always in the same condition during production, test results for supposedly identical pipes can differ substantially. In a similar manner the speed of the mandrel’s rotation and that of the extruded profile cannot be controlled to 100% accuracy. Again, different speeds lead to differently behaving pipes.

As stressed in Section 5.4.1 the behaviour of HDPE, as that of all Thermoplastics, is highly dependent on its temperature. Naturally, the ambient temperature in a production facility is difficult to control, i.e. the pipes are typically produced at varying factory hall temperatures. This change in temperature changes the profile geometry to some extent as the material's properties can change substantially when wound around the mandrel.

Testing Variability

Ideally the pipe samples should be stored at the required constant temperature for a long period, such that the pipe's temperature is uniform across its volume. According to standard this temperature is 23°C (British Standards Institution, 1996). In practice though this is not always possible, so that the pipes temperature might not be uniform. Similarly to the influence of the temperature on the pipe production, here too, the variability in temperature leads to variability of the test results.

In order to conduct a ringstiffness test, a sample needs to be cut from the produced pipes. Firstly, especially for larger pipes and stronger profiles this becomes more and more difficult to do and the sides will not be perfectly flat. Secondly, due to the spiral formation of the profiles the sides of a sample are not uniform. More so, parts of the profiles are cut open. The variability of the pipe's end conditions leads to further inaccuracies of the test results.

Raw Material Variability

In addition to the variability of the test results caused by the pipe manufacturer another source can be identified which is not in their control. This is the variability of the properties of the raw material.

The previous discussion showed that the results of the ringstiffness test are expected to inherit a relatively large variability and that even two seemingly identical tests will not yield identical results (see Section 5.5.4.2). Thus, a tolerance for the reproduction of those results by means of FE simulations of around 10% will have to be accepted.

5.3.5. Pipe Installation

The previously mentioned study 'Design of Buried Thermoplastic Pipes' showed that 80% of the final maximum deflection of a buried plastic pipe is governed by the installation quality. 15% and 3% are influenced by the pipe cover and pipe stiffness, respectively (The European Plastics Pipe and Fitting Association, 1999). That is to say, that a careful installation is of utmost importance in restricting pipe deflection and avoiding pipe failure.

The installation of a pipe can be divided into three steps (Asset International Ltd. (a), n.d.). The first step is the preparation of the pipe bedding. Typically, this is a 150mm thick layer of well compacted granular material. For the installation in wet and soft soil, a geotextile is placed below the bedding layer. After placing the pipe on the bedding the primary backfill follows in steps of 150-300mm layers of granular material of which each is compacted thoroughly. Compaction directly above the pipe crown is performed once the backfill has reached 300mm above the crown. For the final backfill native soil is used and compacted in several layers. Trafficking of the pipes should be avoided until a backfill height of 0.8m above the pipe has been reached. An illustration of a pipe installation in trench conditions can be found in Figure 5.14.

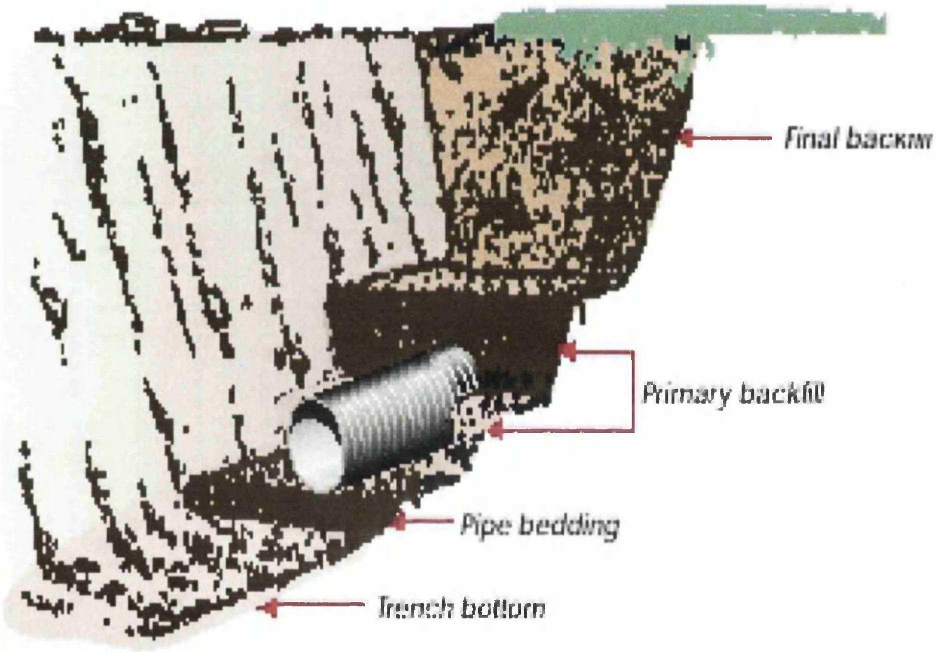


Figure 5.14: Illustration of a pipe installation in trench conditions. The installation is divided into three steps: The bedding, the primary and the final backfill. A thorough compaction is vital in restricting maximum pipe deflections (The European Plastics Pipe and Fitting Association, 1999).

The primary and final backfill materials are shown in Figure 5.15 and the installation using heavy machinery in Figure 5.16.

To ensure sufficient working space the trench width should be chosen in accordance with EN 1610: 1997 (British Standards Institution, 1997) as summarized in Table 5.3. For multi-leg installations a minimum distance between the pipes of 300mm should be used.

Figure 5.15: Illustration of installation soils. The primary backfill is a granular material and ends 0.3m above the pipe. The final backfill ends at the cover level and is a cohesive soil.

Figure 5.16: Additional installation loads induced by heavy machinery. This loading occurs after the installation is finalized for the particular area.

Table 5.3: Recommended trench width for sufficient workspace, depending on the outer diameter (d_o).

Internal Diameter [mm]	Minimum Supported Trench Width [mm]
400 – 700	$d_o + 700$
700 – 1200	$d_o + 850$
> 1200	$d_o + 1000$

5.4. A Temperature, Strain and Strain Rate Dependent Linear Elastic Model for High Density PE Implemented via a Secant Modulus

The response of a strip of a linear elastic material subject to a constant uni-axial stress σ_0 can be obtained with Hooke's law $\sigma_0 = E\ \varepsilon_0$. That means the resulting strain ε_0 is independent with respect to time. For plastics, this relationship is not valid. According to (Janson, 2003) the material behaviour of a plastic material can be simulated by the combination of a spring, a dashpot and a Voigt element (see Figure 5.17).

Thus, the strain occurring due to a constant stress is made up of the sum of three types of strain. The strain that is independent upon time and thus follows Hooke's law is called elastic strain ε_I. The strain that recovers sometime after unloading is called retarded elastic strain ε_{II}. The remaining part is the viscous strain ε_{III}, which will recover only in infinitely long time.

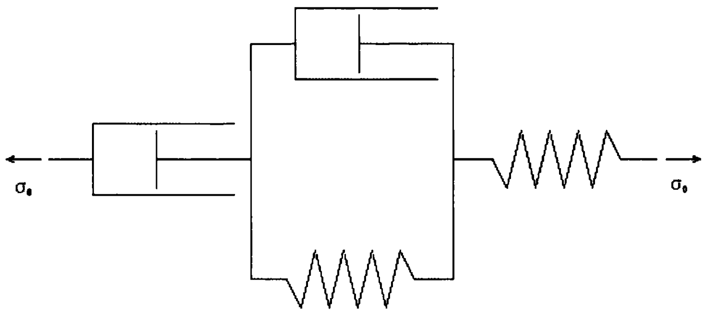
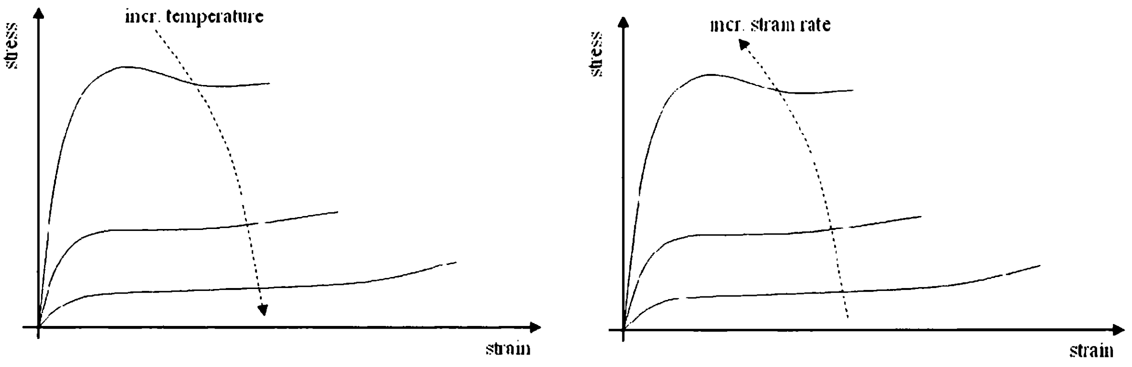


Figure 5.17: Model representing the behaviour of a plastic subjected to uni-axial tension – a series of a spring, a dashpot and a Voigt element.

The behaviour of this model certainly changes if the total stress is not immediately but gradually applied. In that case the rate at which it is increased changes the response of the material. That is for a higher strain rate the material is stiffer.

Furthermore, stress-strain relationships in uni-axial tension at constant strain rates are non-linear, because stress relaxation occurs throughout the test (Bilgin et al., 2007). The behaviour of thermoplastic materials, such as polyethylene (PE), is also dependent on the temperature: They soften on heating and harden on cooling. Typical stress-strain curves for a thermoplastic are illustrated in Figure 5.18.



a) b)

Figure 5.18: Typical behaviour of thermoplastic materials. Material stiffness increases with a) decreasing temperature and b) increasing strain rate.

Following the above discussion, an accurate simulation of polyethylene pipes subject to the ringstiffness requires a time dependent material model. (Moore & Hu, 1996) accurately predicted pipe displacements up to 3% using a linear visco-elastic model. Usage of such a time dependent model naturally leads to higher computational effort. In this work though, the focus lies on simple and automated simulations that require minimum user interaction and yield results in a timely fashion. Furthermore, the final state of the pipe is of interest rather than the entire history. This piece of research aims at maximizing the degree of accuracy of the simulation of the ringstiffness test while employing a linear elastic material model.

To this end, the following approach has been chosen: Specific material data will be obtained and a guideline on how to extract an equivalent secant modulus, depending on the input parameters temperature, strain rate and strain level will be developed. With regards to the temperature, one can assume that a pipe that is stored in a location of constant ambient temperature has no variation across its volume. Therefore, no error is made if material properties are used for one specific temperature. Uniformity in that sense naturally cannot be assumed for the strain field. That means both, the strain level and the strain rate, cannot be respected in a satisfying way with a single set of material properties for the entire pipe. One way to reduce the error that is made by utilizing a linear material model is to calculate a representative strain level and, with the pipe's deformation speed at hand, a mean strain rate. With those values, corresponding material properties can be obtained. The basic idea of the approach is illustrated in Figure 5.19. That is, given the specific stress-strain relationship for the pipe temperature T_{pipe} and a representative strain rate $\dot{\varepsilon}_{pipe}$ the secant modulus E_{FE} for a representative strain level ε_{pipe} is calculated as $E_{FE} = \sigma(\varepsilon_{pipe})/\varepsilon_{pipe}$.

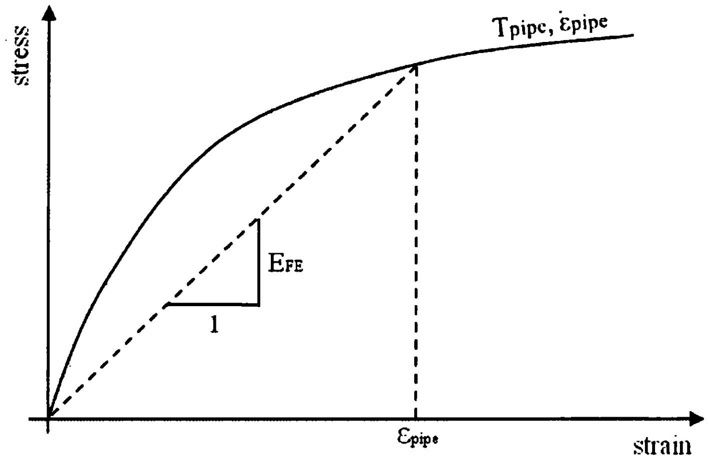


Figure 5.19: The approach for the calculation of the secant modulus E_{FE} based on the pipe temperature T_{pipe} , a representative strain level ε_{pipe} and a representative strain rate $\dot{\varepsilon}_{pipe}$.

For a linear elastic isotropic material model two material properties are required. These are the E-modulus and the Poisson's ratio. The dependence upon the temperature, strain level and the strain rate of the E-modulus is going to be investigated. The Poisson's ratio will be taken as described in (Bilgin et al., 2007) being dependent only upon the temperature.

In order to investigate the material's stiffness several material tests of two different types will be conducted. These are DMTA and tension tests. In the following sections only the material data from the KWH material will be shown. The corresponding plots for the Asset material can be found in Appendix B.

5.4.1. Dynamic Mechanical Thermal Analysis

In a DMTA test a specimen is typically tested by applying an oscillating force or displacement, while passing through a specified temperature interval. Furthermore, within one test, the oscillation frequency can take several values. This testing procedure is a convenient way to investigate the material stiffness at various temperatures, while analyzing the visco-elastic behaviour at the same time, by utilizing different frequencies.

5.4.1.1. Theory

In the following the theoretical background on DMTA tests will be given. Background information on the DMTA test itself can be found in (Sepe, 1998). Further information on the behaviour of polymer materials can be found in (Hall, 1989) and (Ferry, 1980).

Strain Level and Strain Rate

Within this study the DMTA specimens are tested in a three point bending test, where all three supports are fully clamped. A typical specimen can be seen in Figure 5.20.

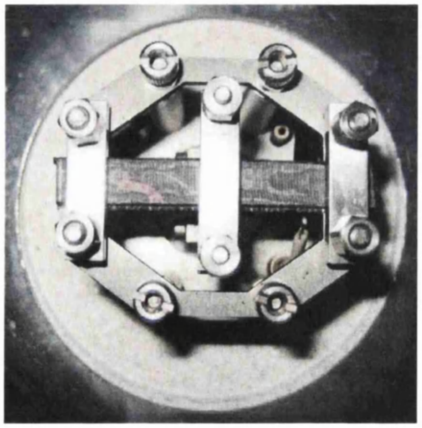


Figure 5.20: Specimen in clamped three point bending device of the DMTA machine.

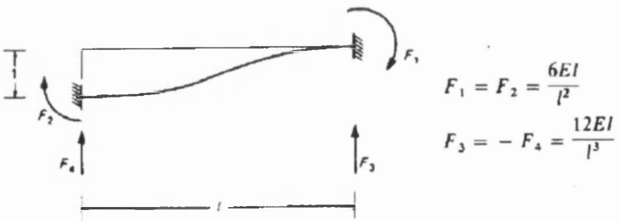


Figure 5.21: Theoretical moments and forces at the supports of the clamped bending test.

As illustrated in Figure 5.21 the theoretical moment at the supports of a clamped beam with a flexural rigidity *EI* and a length *l* equals to

$$M_{support} = \frac{6 \cdot EI \cdot d}{l^2} \tag{5.2}$$

if one of the supports is displaced by d. Due to the linearity of the bending moment its function along the beam axis x is:

$$M(x) = \frac{6 \cdot EI \cdot d}{l^2} \cdot \left(1 - 2\frac{x}{l}\right) \quad with \;\; 0 \leq x \leq l \tag{5.3}$$

The bending strain is:

$$\varepsilon(x,y) = M(x) \cdot \frac{y}{EI} \tag{5.4}$$

Here, y is the vertical distance from the center of gravity to the location of the desired strain. Inserting Equation 5.3 into Equation 5.4 yields:

$$\varepsilon(x,y) = \frac{6 \cdot d \cdot y}{l^2} \cdot \left(1 - 2\frac{x}{l}\right) \tag{5.5}$$

In order to obtain an average strain $\bar{\varepsilon}$, the strain function in Equation 5.5 is integrated along half the beam and from the centre of gravity of the cross-section to the extreme fibre at the top. This integral is divided by the product of half the length and half the height.

$$\bar{\varepsilon} = \int_0^{y_{max}} \int_0^{l/2} \varepsilon(x,y)\, dx\, dy \cdot \frac{1}{l/2 \cdot y_{max}} = \frac{6 \cdot d \cdot y_{max}}{l^2} \tag{5.6}$$

In Figure 5.22 the qualitative displacement of a specimen in a typical DMTA test is plotted against time.

The average strain rate $\dot{\bar{\varepsilon}}$ is idealized as the average strain $\bar{\varepsilon}$ divided by a quarter of the period T, which is the strain amplitude divided by the time required to reach it starting from zero strain.

$$\dot{\bar{\varepsilon}} = \frac{\bar{\varepsilon}}{T/4} = 4 \cdot \bar{\varepsilon} \cdot f \tag{5.7}$$

Here, f is the frequency of the oscillating displacement.

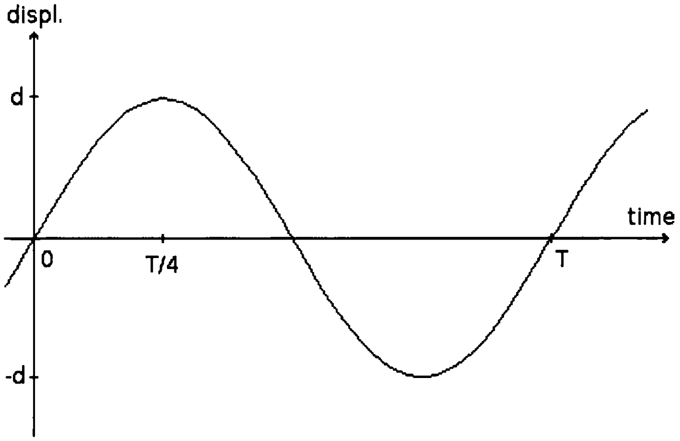


Figure 5.22: Displacement of the specimen in DMTA test plotted against time.

Output

The induced strain in a DMTA test is sinusoidal and reads for amplitude ε_0 and frequency f:

$$\varepsilon = \varepsilon_0 \cdot \sin(f \cdot t) \qquad 5.8$$

Due to the visco-elasticity of the material the induced strain and the measured stress are out of phase by an angle δ because the material does not react instantly but with a certain time lag. The stress reads:

$$\sigma = \sigma_0 \cdot \sin(f \cdot t + \delta) \qquad 5.9$$

The stress can be decomposed into

$$\sigma = \sigma_0 \cdot \sin(f \cdot t) + \sigma_0 \cdot \cos(f \cdot t) \cdot \sin \delta \qquad 5.10$$

Stress, strain and out-of-phase angle are illustrated in Figure 5.23 for two different frequencies. The two out of phase angles are identical but due to the different frequencies, the offsets are between the curves are different. Now, the stress is divided into an in-phase part and one that is shifted by 90°, denoting the former elastic and the latter viscous stress. With that distinction the storage and loss moduli are defined as

$$E' = \frac{\sigma_0}{\varepsilon_0} \cdot \cos \delta \qquad 5.11$$

$$E'' = \frac{\sigma_0}{\varepsilon_0} \cdot \sin \delta \qquad 5.12$$

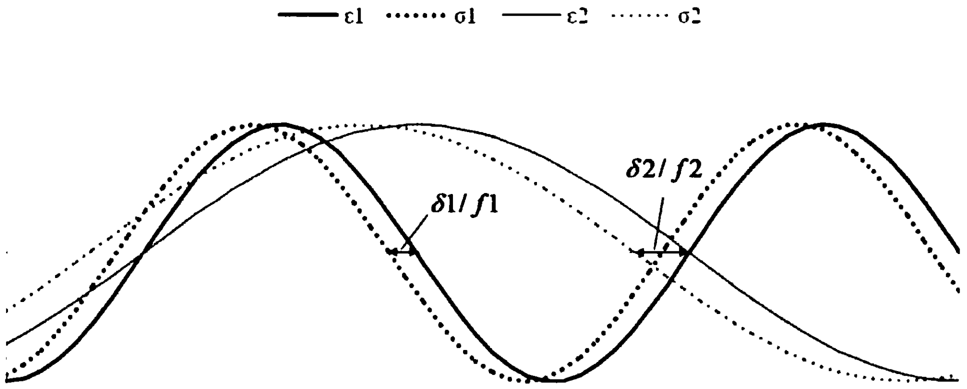


Figure 5.23: Illustration of dynamic stress, strain and out-of-phase angle δ for two different frequencies with f1 > f2.

The relation between the two moduli defines the loss tangent $\tan\delta$.

$$\tan\delta = \frac{E''}{E'} \tag{5.13}$$

The DMTA tests output two material properties. The storage modulus E' and the loss tangent $\tan\delta$.

5.4.1.2. Conducted Tests and Results

In the following the material data for the KWH material will be shown. The data for the Asset material can be found in Appendix B. In order to cover possible future applications of the data that go beyond that of the simulation of the ringstiffness test, the temperature and frequency settings were widely chosen. Hence, four tests were conducted at frequencies 3, 1, 0.3, 0.1, 0.03 and 0.02 Hz and a temperature range between 5 °C and 75 °C, which changed at a rate of 1 °C/min. The relation between the DMTA test frequency and the strain rates will be derived in Section 5.4.1.2. The samples are 7 mm wide and 2.4 mm thick. Each of the two spans of the three point bending is 14 mm long. The middle support's maximum deformation was set to 20μm in either direction.

Inserting the deformation of the support $d=0.02mm$, the length of the beam $l=14mm$ and the location of the extreme fibre $y_{max}=1.2mm$ into Equation 5.6 yields the average strain level

$$\bar{\varepsilon} = 1.83674 \cdot 10^{-4} \qquad 5.14$$

Insertion of Equation 5.14 into Equation 5.7 produces the average strain rate, naturally dependent upon the frequency

$$\dot{\bar{\varepsilon}} = 7.34694 \cdot 10^{-4} \cdot f \qquad 5.15$$

All obtained curves have been fitted by means of polynomials and those for the same frequency were averaged. The fitted and averaged storage moduli for the changing temperatures and frequencies are shown in Figure 5.24.

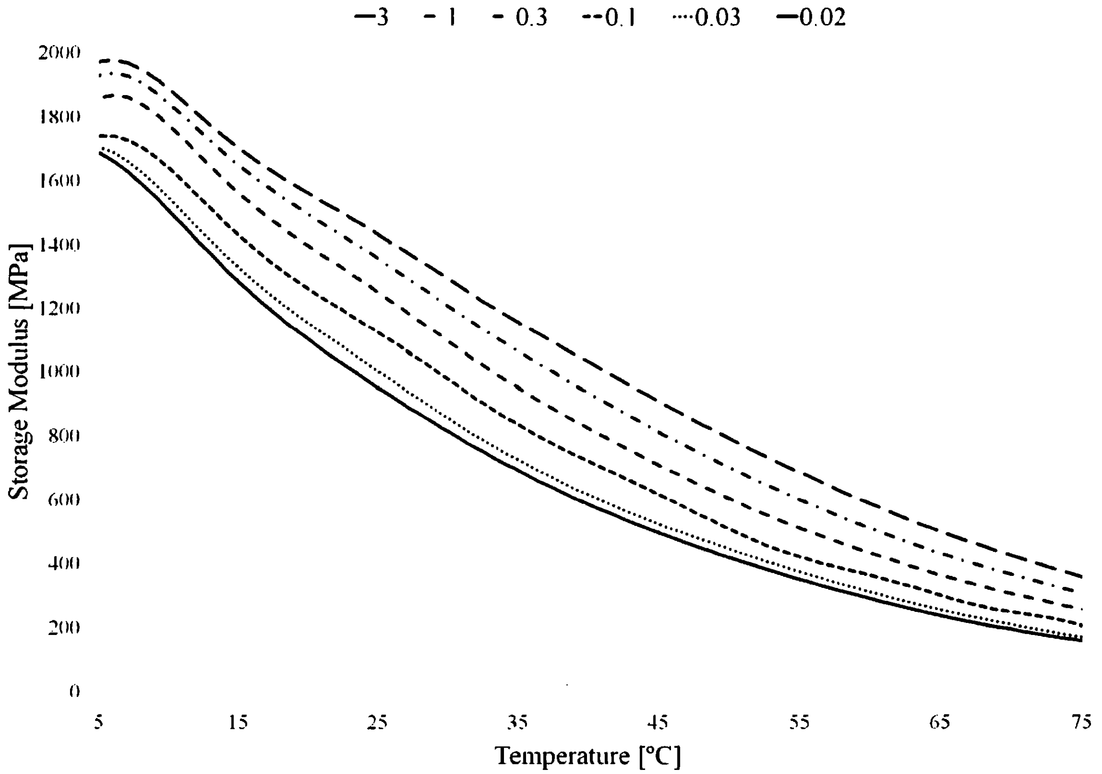


Figure 5.24: KWH material. Storage modulus E' plotted against temperature for various frequencies obtained from DMTA tests.

One can see the decrease in material stiffness with increasing temperature and decreasing frequency. For instance for a frequency of 0.3Hz the storage modulus drops from 1971MPa at 5°C to 1422MPa at 25°C.

The loss tangents for the temperature range and various frequencies are shown in Figure 5.25. The behaviour of the loss tangent is obviously opposite to that of the storage modulus. It increases with increasing temperature and decreasing frequency. For a

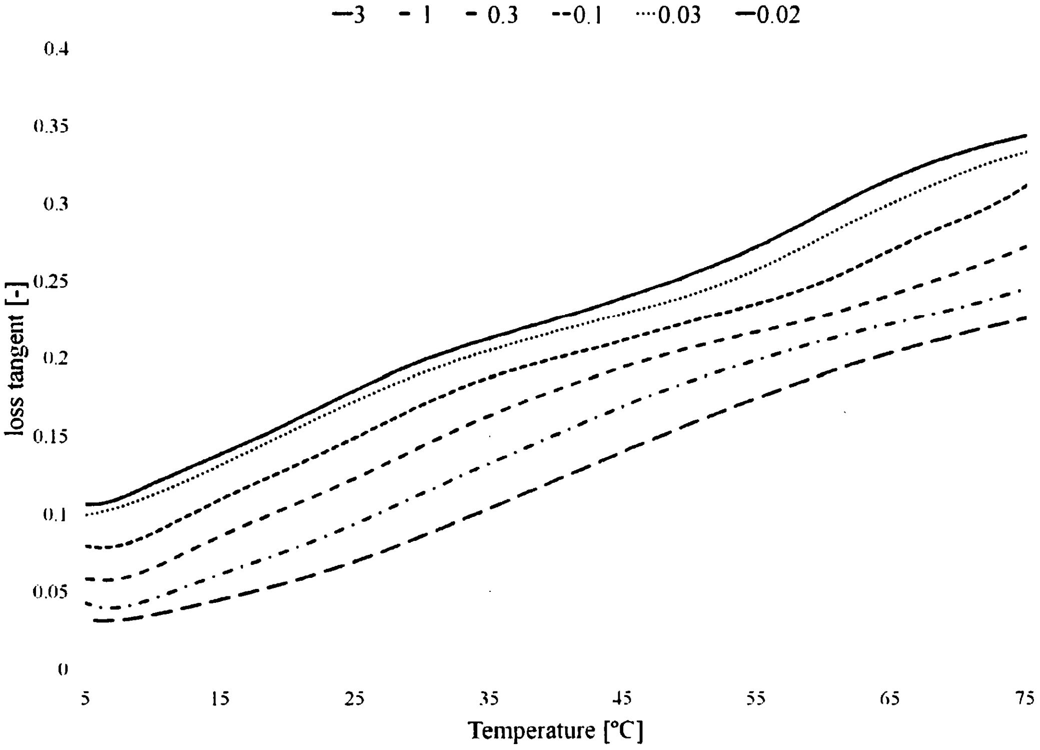


Figure 5.25: KWH material. Loss tangent tan δ plotted against temperature for various frequencies obtained from DMTA tests.

frequency of 0.3Hz the loss tangent increases from 0.032 at 5°C to 0.071 at 25°C. Since $tan\,\delta$ is an indicator for the degree of the visco-elastic effect, with $tan\,\delta = 0$ being purely elastic, it is not surprising that it increases with increasing temperature. It is surprising though that it decreases with increasing frequency because one would expect that, similar to an ideal damper, the damping effect becomes more significant with higher deformation speeds. The explanation for that contradiction can be found by studying Figure 5.23. The two out-of-phase angles are of similar size. Thus, the DMTA machine would output identical loss tangents for both tests even though the offset of the second curve is higher, due to the lower frequency. That means the loss tangent has to be set in relation to its frequency for it to represent the degree of the visco-elastic effect. In reality the effect naturally increases with increasing frequency. This will be shown when the results of the tension tests are investigated in Section 5.4.2.2.

5.4.2. Tension tests

Contrary to the DMTA tests, tension tests have a constant temperature and a constant strain rate throughout one test, but they allow for the investigation of the influence of strain levels on the material stiffness. Several tests with various strain rates and temperatures have been conducted to measure to what extent the influence of the strain level changes the results.

In Figure 5.26 the tension test set-up for experiments at room temperature is shown. The material sample is placed in between two clamps of which the upper one is used to deform the specimen. An extensometer is attached to the sample that measures the strain. Figure 5.27 shows the oven that is used to conduct tests at elevated temperatures. The remaining set-up inside the oven is identical to that in Figure 5.26.

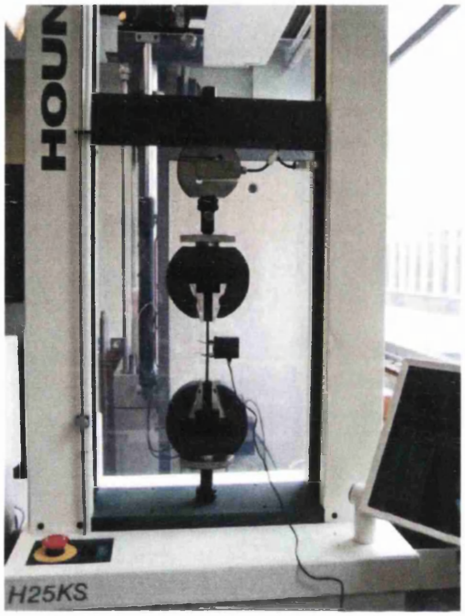


Figure 5.26: Tension test at room temperature: Clamped specimen with attached extensometer.

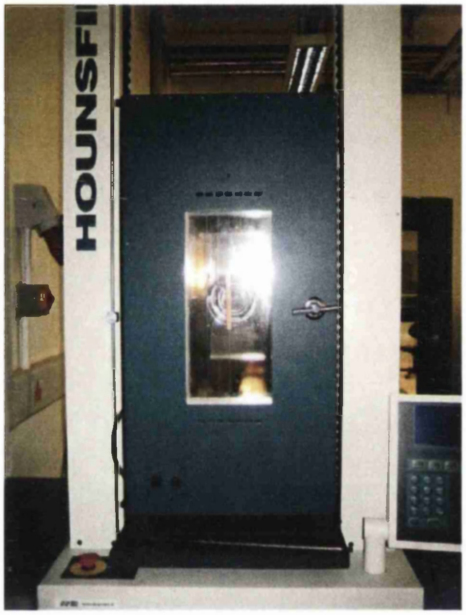


Figure 5.27: Tension test at elevated temperature: Chamber where specimen is placed in.

5.4.2.1. Theory

The tension tests are displacement controlled and have a constant velocity throughout the test. The strain is the extension of the specimen over its original length

$$\varepsilon = \frac{\Delta l}{l} \qquad 5.16$$

Due to the clamping of the specimen a small force is induced in the beginning of the test. Therefore the E-modulus takes into account the initial stress σ_0

$$E = \frac{\sigma - \sigma_0}{\varepsilon} \qquad 5.17$$

The theoretical strain rate of the test is the displacement velocity v over the length of the specimen:

$$\dot{\varepsilon} = \frac{v}{l} \qquad 5.18$$

In practice though, mainly due to compliance of the machine and the visco-elastic effect, the strain rate is not constant. Therefore the strain rate is calculated with the strain and the elapsed time t according to:

$$\dot{\varepsilon} = \frac{\varepsilon}{t} \qquad 5.19$$

In order to obtain the corresponding $tan\ \delta$ for a tension test the strain rate has to be converted into the corresponding frequency in the DMTA test. The strain rate is converted to the corresponding DMTA frequency via Equation 5.20, which is the inverted Equation 5.15.

$$f_{DMTA} = 1361.11088 \cdot \dot{\varepsilon}_{Tension} \qquad 5.20$$

5.4.2.2. Conducted Tests and Results

In the following the material data for the KWH material will be shown. The data for the Asset material can be found in Appendix B. Four different settings have been investigated by means of tension tests. They are listed in Table 5.4.

Table 5.4: Four different tension test settings. The number of tests, temperature, strain rate, frequency and loss tangent are listed.

	Unit	1	2	3	4
number of tests	*[-]*	*4*	*4*	*6*	*6*
T	*[°C]*	*23*	*23*	*23*	*56*
$\dot{\varepsilon}_{Tension}$	$[s^{-1}]$	$2.0 \cdot 10^{-4}$	$4.25 \cdot 10^{-5}$	$2.8 \cdot 10^{-5}$	$3.8 \cdot 10^{-5}$
f_{DMTA}	$[s^{-1}]$	*0.272*	*0.058*	*0.038*	*0.052*
tan δ	*[-]*	*0.12*	*0.15*	*0.16*	*0.25*

It shows how many samples have been analyzed, the temperature T and the strain rate $\dot{\varepsilon}$. The latter is converted into the corresponding DMTA frequency according to Equation 5.20. The loss tangent $tan\ \delta$ is then obtained from the DMTA data by interpolation of temperature and frequency. One setting is characterized by elevated and the rest by room temperature. Each obtained stress-strain curve was fitted by means of polynomials and those corresponding to one setting were averaged.

Investigation of the Influence of Strain Rates

Firstly, the influence of a change of the strain rate on the stress-strain curves is investigated. The resulting curves for test settings 1, 2 and 3 are shown in Figure 5.28. It is clearly observable that tests with a higher strain rate have a higher resistance leading to higher stresses. Their degree of nonlinearity is also higher, which means the degree of the visco-elastic effect increases with increasing strain rate. As explained in Section 5.4.1.2 this is not directly represented in the loss tangent. In case of strain rate the loss tangent is not an indicator on the degree of the visco-elastic effect.

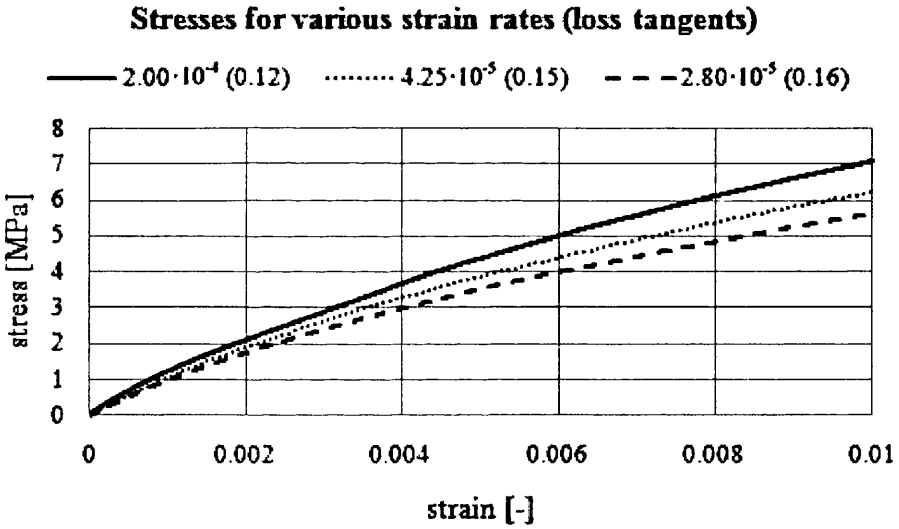


Figure 5.28: KWH material. Stress plotted against strain for various strain rates (loss tangents) obtained from tension tests.

This leads to the development of the E-moduli as shown in Figure 5.29 with high values at low strain levels for large strain rates decreasing with decreasing strain rate and increasing strain level.

Investigation of the Influence of Temperature

After the strain rate, now the influence of a change of the temperature on the stress-strain relationship is investigated. The curves for test settings 1 and 4 are shown in Figure 5.30. The two settings do not have identical strain rates; the small difference is negligible though.

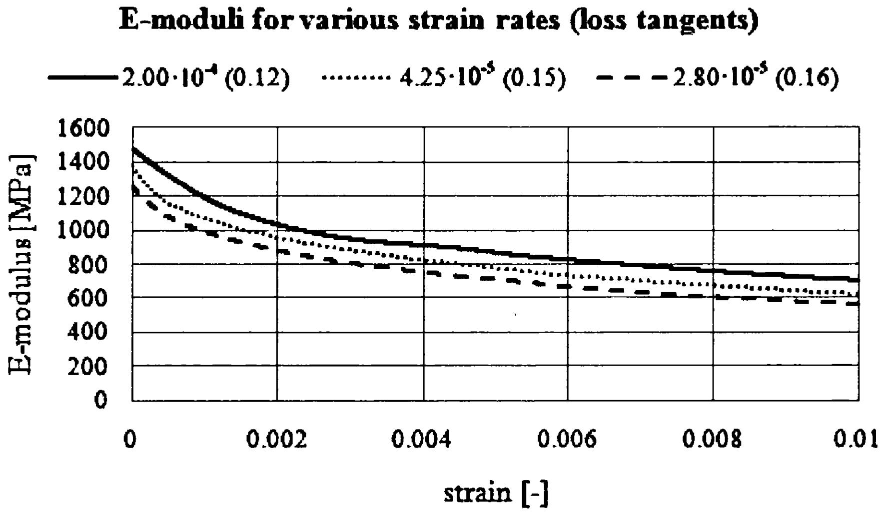


Figure 5.29: KWH material. E-modulus plotted against strain for various strain rates (loss tangents) obtained from tension tests.

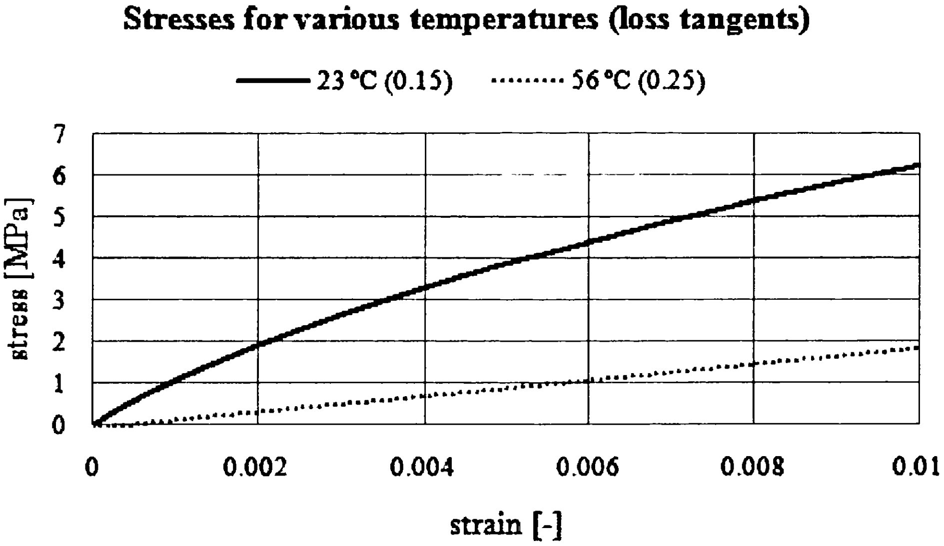


Figure 5.30: Stress plotted against strain for various temperatures (loss tangents) obtained from tension tests.

Note that the curve at elevated temperature starts at the DMTA strain level, because it has been manually shifted to that strain level. This has been done because the heated oven had to be opened in order to place the sample in the device. At that point the temperature drops and hence the specimen shortens. This corresponds to applying a tensile force and thus, because of the elapsed time until the experiment is started, the strain does not start at zero. Therefore, the starting strain level was assumed to be the DMTA strain level as the E-modulus at that level approximates that anticipated from the DMTA results (see Section 5.4.3).

It is clearly observable that the test at room temperature has a higher resistance and leads to higher stresses. The nonlinearity of the curve at elevated temperature is hardly visible, which might lead to the conclusion that the visco-elastic effect would decrease with increasing temperature and loss tangent. Consideration of the development of the E-moduli as shown in Figure 5.31 though shows that the curve is in fact highly nonlinear for a small strain interval and only becomes linear after a certain strain level.

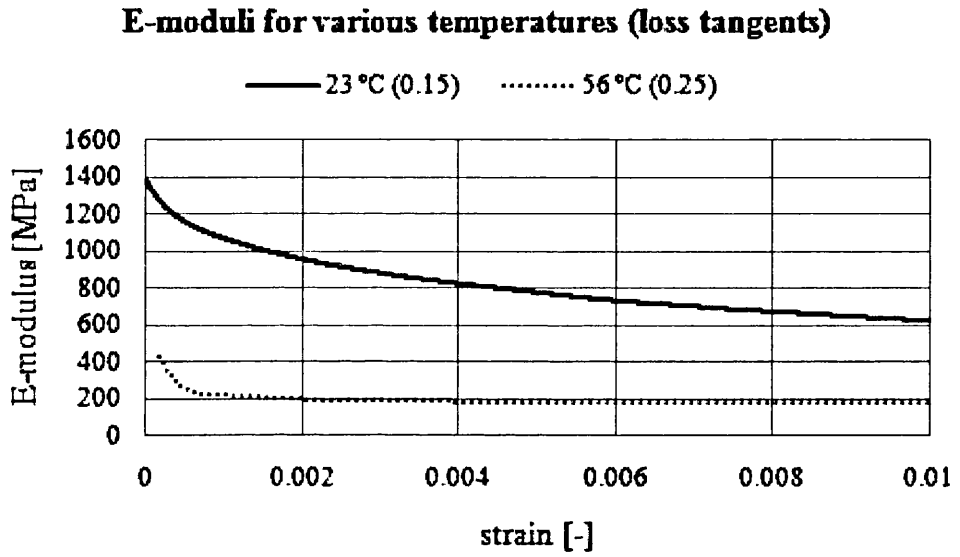


Figure 5.31: KWH material. E-modulus plotted against strain for various temperatures (loss tangents) obtained from tension tests.

Furthermore, the reduction of the E-modulus is even higher than for lower loss tangents, since the ratio between the E-modulus at the DMTA strain level and that at 1% strain equals to *2*. For the test at elevated temperature this value equals to *2.5*. Hence it can be concluded that with increasing loss tangent the decrease of material stiffness occurs in a smaller strain interval and is greater than for lower loss tangents.

5.4.3. Application in Numerical Modeling

The combination of the results obtained from the DMTA tests and the tension tests yields a secant modulus that takes temperature, strain rate and strain level as well as their correlation into account. This modulus will be called E_{FE} as it will be used within the FE simulations. The starting point to define that value is the storage modulus *E'* obtained from the DMTA test. For the FE model naturally the total E-modulus is required and not the elastic part of it. Therefore according to Equation 5.13 this elastic part has to be multiplied by $1 + \tan\delta$, called fac_{visco} from here forth, to obtain the total E-modulus which is the sum of the elastic and the inelastic part

$$E = E' + E'' = E' \cdot (1 + \tan\delta) = E' \cdot fac_{visco} \qquad 5.21$$

In order to show the validity of the developed procedure, Equation 5.21 will be used to reproduce to the E-moduli obtained in the tension tests at the DMTA strain level (see Equation 5.14). This was done for all four tension test settings. The results are listed in Table 5.5.

Table 5.5: Comparison of the total E-modulus obtained from DMTA tests with the E-modulus at the DMTA strain level from the tension tests. The total E-modulus is obtained by multiplying E' by fac_{visco}.

	Unit	1	2	3	4
T	*[°C]*	*23*	*23*	*23*	*56*
$\dot{\varepsilon}_{Tension}$	*[s⁻¹]*	$2.0 \cdot 10^{-4}$	$4.25 \cdot 10^{-5}$	$2.8 \cdot 10^{-5}$	$3.8 \cdot 10^{-5}$
f_{DMTA}	$[s^{-1}]$	*0.272*	*0.058*	*0.038*	*0.052*
$\tan\delta$	*[-]*	*0.12*	*0.15*	*0.16*	*0.25*
E'	*[MPa]*	*1289*	*1106*	*1073*	*375*
$E' \cdot fac_{visco}$	*[MPa]*	*1442*	*1278*	*1247*	*470*
$E_{Tension}(\varepsilon_{DMTA})$	*[MPa]*	*1420*	*1274*	*1177*	*433*

The total E-modulus from the DMTA test and the one from the tension tests differ by at most 8 %.

The results from the tension tests are now exploited in order to take the strain level into account. Therefore another factor fac_{level} is introduced.

$$fac_{level} = \frac{E_{Tension}(\varepsilon_{Pipe})}{E_{Tension}(\varepsilon_{DMTA})} \qquad 5.22$$

It will be used to reduce the E-modulus obtained from the DMTA tests according to a representative strain level. Firstly, fac_{level} will be discussed for the different strain rates. The corresponding curves are illustrated in Figure 5.32. One can see that the variation between the curves is rather small. This is due to the fact that the strain rate has a small influence on the loss tangent, i.e. an increase of the strain rate by a factor of more than 10 leads to a decrease of the loss tangent by a factor of 1.3.

Now the influence of varying temperatures on fac_{level} will be discussed. The corresponding curves are illustrated in Figure 5.33.

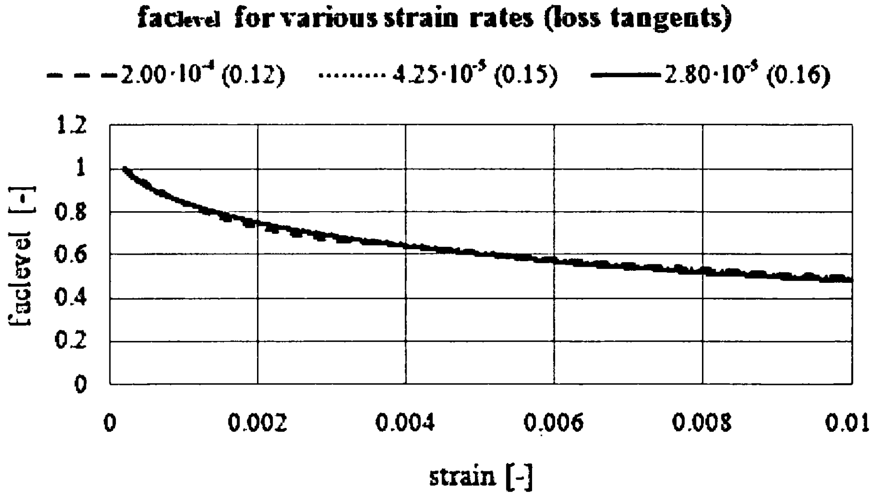


Figure 5.32: KWH material. fac_{level} plotted against strain for various strain rates (loss tangents) obtained from tension tests.

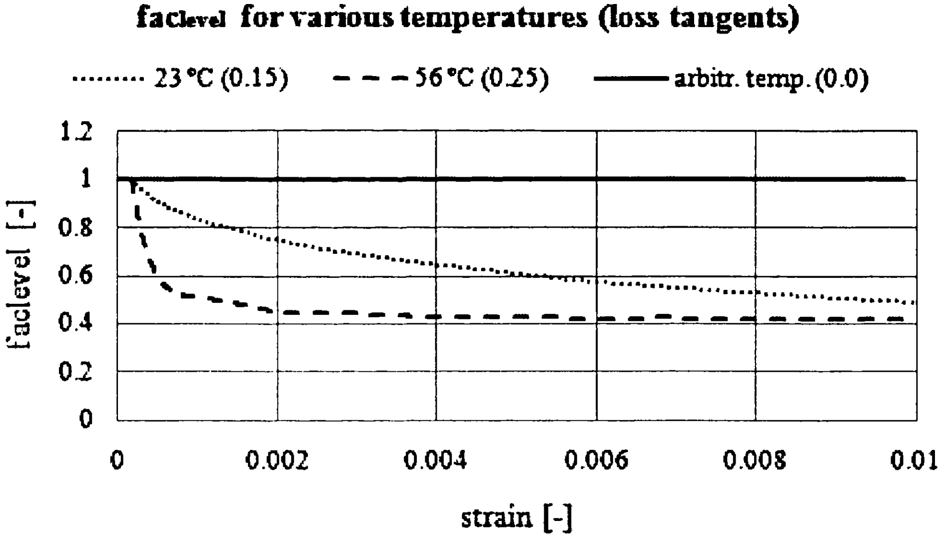


Figure 5.33: KWH material. fac_{level} plotted against strain for various temperatures (loss tangents) obtained from tension tests.

One can see that the influence of the temperature is more significant than the strain rate. Not only is the quantitative difference more significant but also the shape of the curves changes. The degree of the visco-elastic effect is measured by means of $tan\,\delta$. For $tan\,\delta = 0$ the material behaviour is purely elastic, thus the E-modulus is constant in that case. The theoretical curve for that case has been added in Figure 5.33. To obtain fac_{level} for any other temperature, the values will be interpolated by means of the actual loss tangent.

With the two correction factors the final E-modulus for the FE analyses reads:

$$E_{FE} = E' \cdot fac_{visco} \cdot fac_{level} \quad 5.23$$

In order to evaluate Equation 5.23 $\varepsilon_{Pipe}, \dot{\varepsilon}_{Pipe}$ and T_{Pipe} have to be available. As mentioned before, for a pipe stored at a constant temperature for a sufficiently long time the pipe temperature can be assumed uniform across its volume. For the strain level and strain rate though a representative value has to be found. Areas in the pipe that exhibit higher strains than others will be more effective in determining the overall structural performance. In other words, the local stiffness in a region of the pipe that is not strained at all does not affect the overall behaviour. Thus, the average strain level will be chosen to lie at the upper end of the scale. Experiments showed that a value of 90% of the maximum membrane strain is a good working value.

$$\varepsilon_{Pipe} = 0.9 \cdot \varepsilon_{mem}^{max} \quad 5.24$$

The representative strain rate is the representative strain level divided by the duration of the ringstiffness test t_{test}.

$$\dot{\varepsilon}_{Pipe} = \frac{\varepsilon_{Pipe}}{t_{test}} \quad 5.25$$

Equation 5.23 has been evaluated for a temperature range 5°C to 49°C and a strain rate range from $2 \cdot 10^{-5}$ s^{-1} to $1 \cdot 10^{-4}$ s^{-1}. The results are plotted in Figure 5.34. The strain level at which fac_{visco} has been evaluated was set to $4 \cdot 10^{-3}$.

The results from the ringstiffness test simulations that will be discussed in Section 5.5.4.1 have been used to plot the maximum and minimum membrane stresses against the obtained ringstiffness. This illustration can be found in Figure 5.35. A clear trend of increasing strain with increasing ringstiffness can be seen. In order to decrease the necessary user-interaction a simple equation will be derived from this data for the user to obtain the expected absolute maximum membrane strain for a simulation, depending on the expected ringstiffness.

$$|\varepsilon_{mem}^{max}| = 0.0034 + 0.0003 \cdot SN \quad 5.26$$

The obtained value has to be inserted into Equation 5.24.

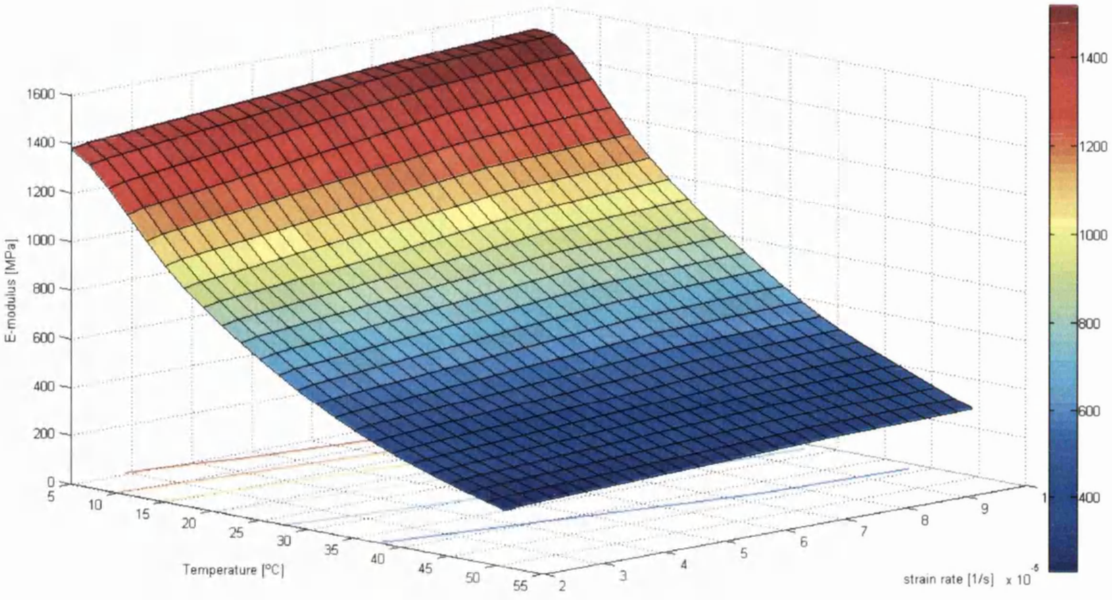


Figure 5.34: E_{FE} plotted against temperature and various strain rates, evaluated at 0.4%.

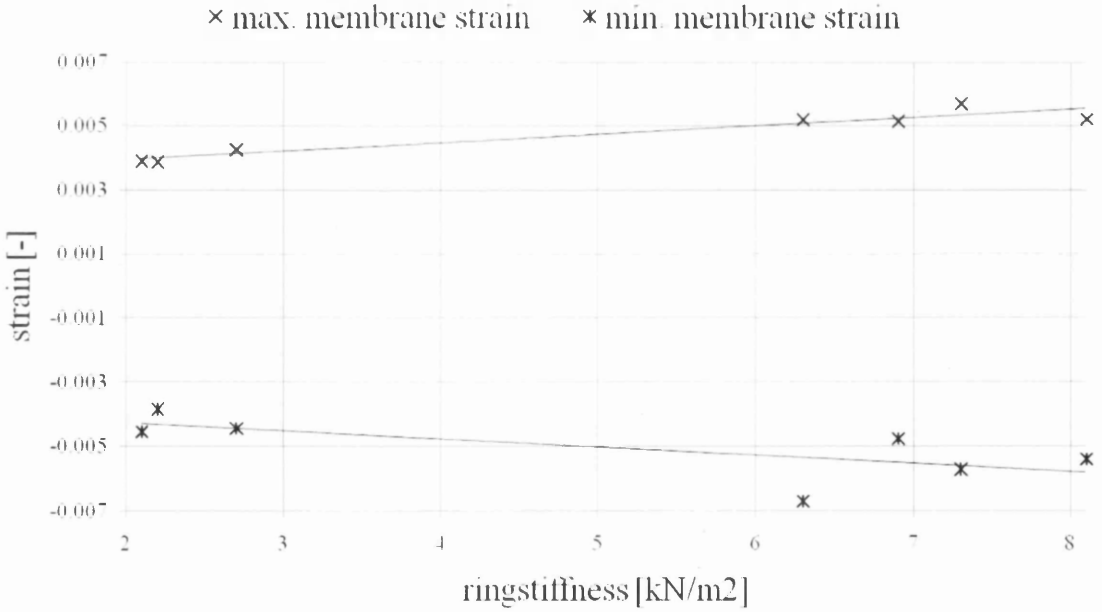


Figure 5.35: The maximum and minimum membrane stresses plotted against the obtained ringstiffness from the results from the ringstiffness test simulations that will be discussed in Section 5.5.4.1.

5.5. Ringstiffness Test Simulation and Optimization

5.5.1. Profile Creation

The cross-sections within the ringstiffness simulation and optimization are represented by shell elements. For a shell element representation of the box-sections a centre-line needs to be generated that describes the profile's shape. At a finite number of locations the corresponding thickness of the profile is measured, too. Due to the recurrence and the symmetry of the profiles this is only done for half of one box, denoted as C-profile. This C-profile can be generated in three different ways: By processing the output of a laser scan (see Section 5.5.1.1), by processing the data imported from a dxf-file (see Section 5.5.1.2) or by directly specifying the profile's dimensions (see Section 5.5.1.3).

As discussed in Section 5.3.1, the theoretical, rectangular box-section profiles of the pipes, extruded at the very beginning of the production process – completely defined by the height, width and wall thicknesses – often differ significantly from the cross-sections obtained after production, exhibiting differences of up to 20%. Therefore, the laser scan is the most precise way to capture the geometry of the profile. The other two options are more efficient but can lead to inaccuracies.

After a C-profile has been created with one of the three methods, it has to be mirrored and merged with the previous one so many times until the entire profile reaches the desired length of the sample to be modeled. This procedure is performed automatically in the automatic pre-processing prior to the simulation. The procedure itself is presented in Section 5.5.1.4.

5.5.1.1. Cross-Section Laser Scanning

In order to obtain a geometry representation of a pipe profile by means of a laser scan, a profile sample needs to be cut from the pipe and scanned. The use of the laser scanner is illustrated in Figure 5.36 and Figure 5.37. The scanner exerts a laser-beam that captures the geometry of the structure that it falls on. The geometry is sent to a computer in the form of discrete point coordinates. A point cloud describing a typical cross-section is depicted in Figure 5.38. These points have to be processed prior to the creation of the FE model. The programs carrying out that task are written in Fortran 77 and the whole procedure is controlled from within the Java tool to simplify the use of the command-line-type programs. The main steps the point cloud goes through are illustrated in Figure 5.39 through Figure 5.44 and will be discussed in more detail in the following.

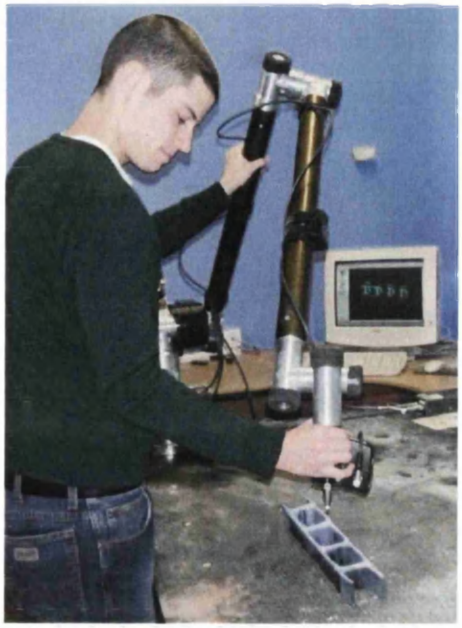


Figure 5.36: Use of the 3D scanner. The geometry is captured by means of the scanner arm.

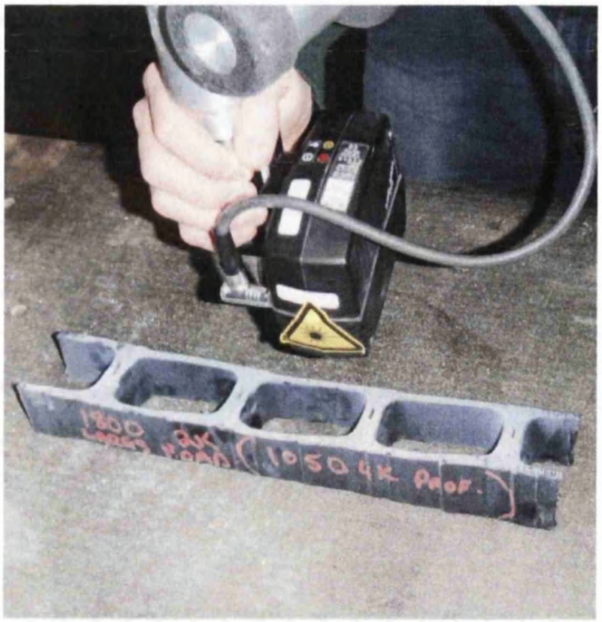


Figure 5.37: By using the laser scanner, a precise description of the pipe's cross-section can be obtained.

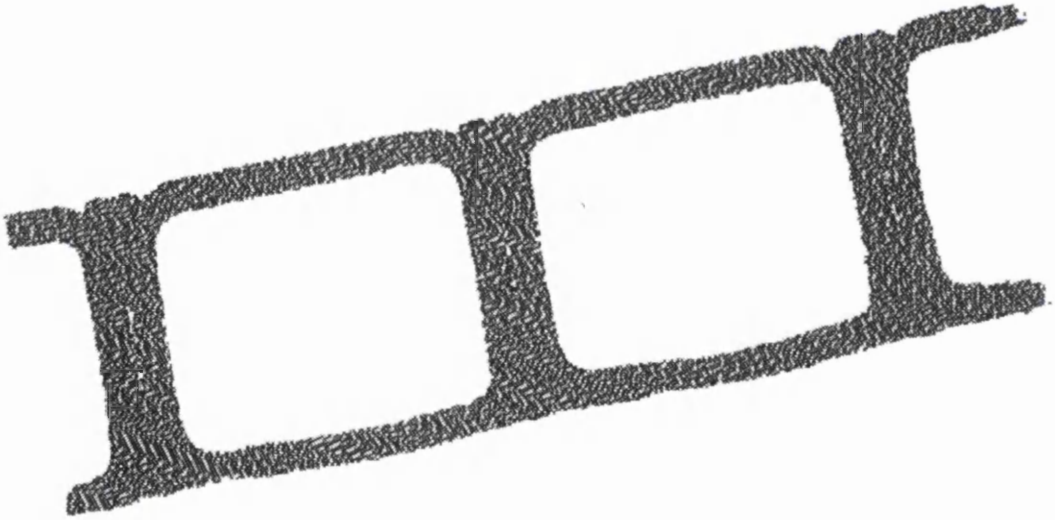


Figure 5.38: Typical point cloud from a laser scan.

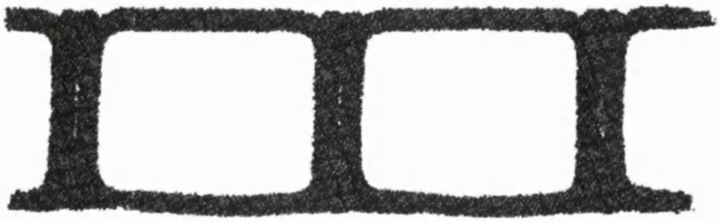


Figure 5.39: Rotated profile.

Figure 5.40: Repeating pattern of profile.

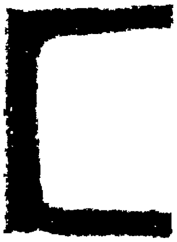


Figure 5.41: Profile vertically swapped.

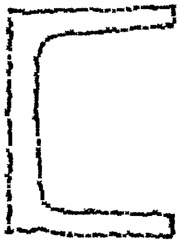


Figure 5.42: Surface points identified.

Figure 5.43: Three lines identified.

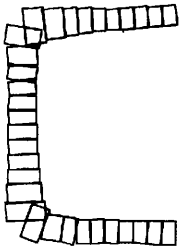


Figure 5.44: Shell representation.

The scan processing starts with the rotation of the profile in a way that all walls are either horizontal or vertical. To this end the program reads the discrete points returned by the scanner. Then, a vertical and a horizontal axis are created crossing at the centre of gravity of the data points, dividing the plane into quadrants. Assuming uniform density of points, the two axes are rotated so that the difference in the number of points in each quadrant is minimized. Finally, the axes and the data points are rotated so that the 'symmetry' axes become vertical and horizontal. A rotated profile is illustrated in Figure 5.39.

After that the repeating pattern, i.e. a C-profile, has to be identified. To this end, the centre of a hole in the one of the boxes in Figure 5.39 is found by making sure the distance from it to the boundaries are the same above and below, and the same to the right and to the left. Next, the web thickness and box width is measured. Now two vertical lines are created passing through the middle of the web (to the left of the centre point) and the centre point itself, respectively. The data points located between the two lines are considered to be the repeating pattern which is shown isolated in Figure 5.40.

If the profile has been scanned upside down, it is swapped now. The swapped profile is shown in Figure 5.41.

The next step is to determine which points lay on the boundary of the cross-section, since only those will be of use for the model generation. This is done considering one point at a time. For each point the ten nearest ones are found. Next, a straight line is created containing the point at issue and one of the ten points, and the angle between this line and the vertical axis is computed. The procedure is repeated for the remaining 9 points, and the angles are rearranged in increasing order so that the relative angles can be calculated by subtracting the lowest angle from the actual one. The greatest of the relative angles is then used as an indicator of whether a point lies on the surface or not. The closer it is to 180° the more likely it is to belong to the boundary. This concept is illustrated in Figure 5.45 and the determined surface nodes for the example scan are depicted in Figure 5.42.

The surface nodes are now divided into three lines: the 'top', the 'bottom', and the 'middle' lines. The starting point of each line is identified, and the program finds the next point by using the following strategy: The last point found to be part of the line at issue is used to create a straight line with one of the remaining surface points. A second straight line is created containing the last and the second last point. The angle between the two lines can be used as an indicator whether the point at issue is the next one on the line (its value should be close to 180°). All points that have a value no farther from 180° than a given threshold are stored. Once all points are checked, the nearest one is the next point on the line. This concept is illustrated in Figure 5.46 and the three lines identified in the example scan are shown in Figure 5.43.

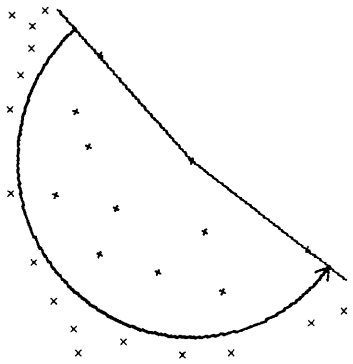


Figure 5.45: Decision whether or not a point lies on the surface.

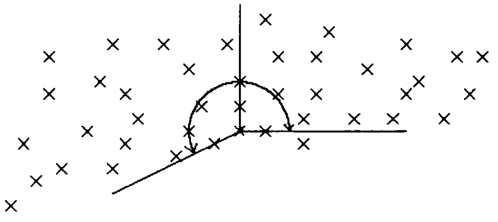


Figure 5.46: Process of finding the nodes that lie on a surface line.

Now that the three lines have been identified, a shell element representation, i.e. a centre line and the corresponding thickness distribution can be generated. When the point density is scarce it can be beneficial to replace the line points by points from evaluated polynomials. This option is available in the Java tool. The procedure for the generation of the shell representation starts by dividing the profile into an upper and a lower part and placing points on the middle of the wall at the ends of those parts and connecting the points by two lines. These lines are shown in the first image of Figure 5.47. Now, a new set of points is found by creating new lines, perpendicular to the old ones and finding the middle of the profile walls from the intersection of the profile walls and the new lines as shown in the first image of Figure 5.47. The newly created points are added to the previous set and the process continues until the specified number of steps has been reached. Steps two and three are illustrated in the second and third images of Figure 5.47.

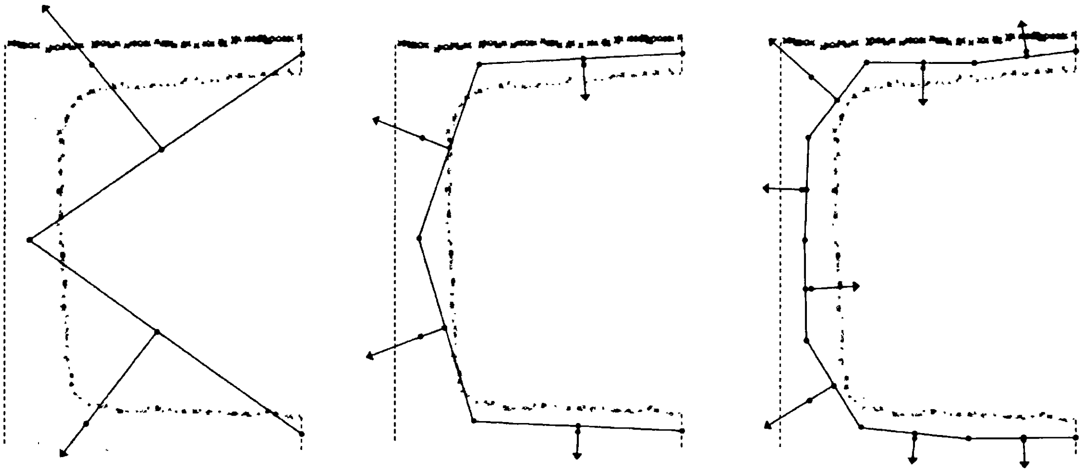


Figure 5.47: The first three steps in the shell model creation. In that process, the profile midpoints and the wall thicknesses are computed.

The scanned outline of the example profile and its corresponding shell representation are illustrated in Figure 5.48.

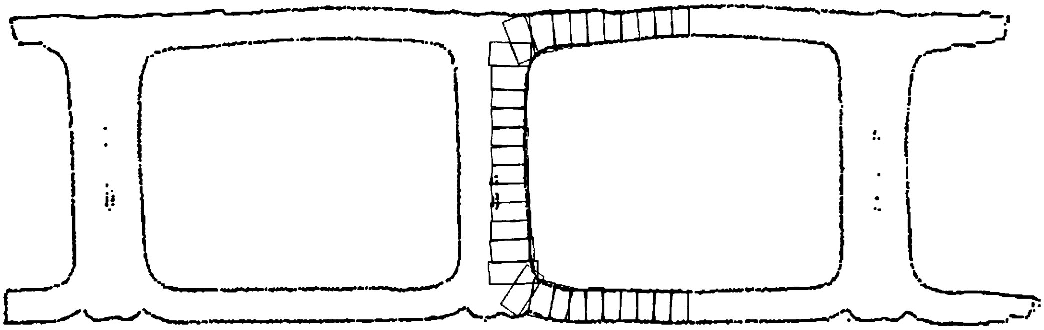


Figure 5.48: A profile imported from a laser scan and corresponding generated shell elements in thickness view.

5.5.1.2. DXF-File Import

Another possibility that has been incorporated in the Java tool for the generation of a shell representation of a pipe profile is the import of a dxf-file. Entities, such as lines, arches, splines and ellipses are read from the chosen file and discrete points are evaluated from them. These points describe the surface of the profile and the same procedure for the identification of the three lines and the shell representation as for the scanned profile are performed next. A shell representation from a dxf-file import is illustrated in Figure 5.49. The accuracy of the representation naturally depends on the accuracy of the data in the dxf-file.

5.5.1.3. Rectangular Profile Creation

The third possibility for the generation of a shell representation of a pipe profile is the creation of a rectangular box-section by specifying the width, height and the three wall thicknesses. Also the number of elements needs to be specified. A rectangular box-section can be seen in Figure 5.50. This representation is the least accurate of the three options but requires the least user-interaction. For more details on the accuracy issues due to the sharp corners in the profile see Section 5.5.4.3.

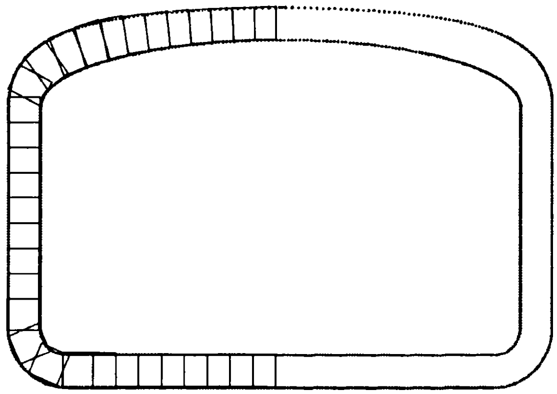


Figure 5.49: Profile imported from a DXF file and corresponding generated shell elements in thickness view

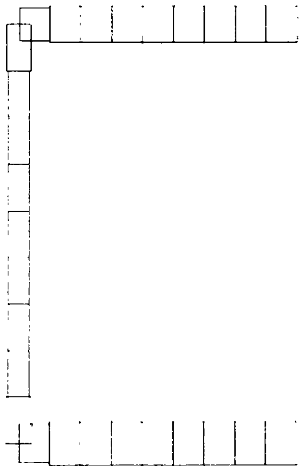


Figure 5.50: Shell elements for rectangular box-section in thickness view.

5.5.1.4. Mirroring and Merging of C-profiles

Before the shell representations of the C-profiles can be used for the FE simulation of the ringstiffness test, they have to be mirrored and merged so many times, until the whole profile reaches the desired sample length. This process is performed within the automatic pre-processing step discussed in Section 5.5.2. The process starts by mirroring the original C-section with respect to the middle of the vertical web, i.e. half of the web thickness left of the vertical centre-line (see Figure 5.51). Now, this newly created profile is mirrored again, this time with respect to the left ends of it (denoted as ‘A’ in Figure 5.51). This process is repeated until half of the desired sample length has been reached (due to symmetry only half of the whole profile is generated). The profile should be closed on the outside, i.e. it should not end at the points denoted as ‘A’ in

Figure 5.51. Because, the sample length in the simulation will not be exactly the same as in the experiment, the profiles are mirrored as many times as it is necessary so that the difference between the two is minimal.

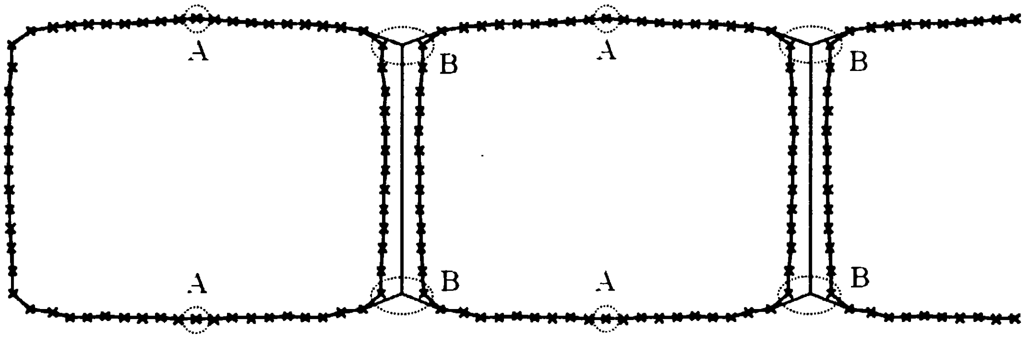


Figure 5.51: Merging the mirrored center-lines. The thicknesses at the ends of the C-profiles are averaged and doubled along the vertical web. Those in the corners are adjusted in way that the area of the entire profile is identical to that of the correct multiple of the original profile.

After the mirroring the profiles have to be combined. Because, the center-lines do not pass through the middle of the original web, the points have to be moved onto it and then merged. This action is performed for the points denoted as ‘B’ in Figure 5.51 as well as for all points in between. The points at the ends of the profiles (denoted as ‘A’ in Figure 5.51) do not require movement. They are simply merged.

After the points have been moved and merged, the thicknesses have to be updated accordingly. For points where the profile is horizontal (denoted as ‘A’ in Figure 5.51), the thicknesses require no update. Wherever the merged points lie on a part of the profile that is vertical, i.e. the web, the thicknesses have to be doubled in size. Following this logic, points where the profile is neither vertical nor horizontal (denoted as ‘B’ in Figure 5.51) require a thickness update that lies in between the previous two extremes. Based on the angle of inclination of the point in question the factor is simply interpolated.

After the thicknesses have been updated, a final check of the resulting profile's cross-sectional area is performed. If this area is not equal to the correct multiple of the original one, the thicknesses of the points in the corners of the profile (denoted as ‘B’ in Figure 5.51) are amended such that this requirement is fulfilled.

The mirrored C-profiles in thickness view as well as an outline of the scanned profile are shown in Figure 5.52. The same illustration for the merged profiles can be found in Figure 5.53.

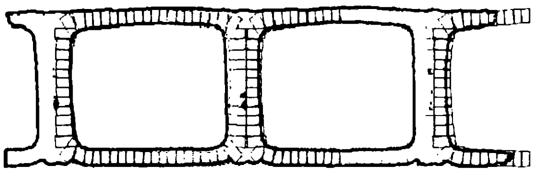


Figure 5.52: The mirrored C-profiles in thickness view, as well as the outline of the original scanned profile.

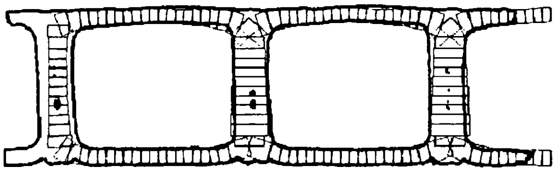


Figure 5.53: The mirrored and merged C-profiles in thickness view, as well as the outline of the original scanned profile.

5.5.2. Simulation

5.5.2.1. FE Model

Using FE simulations the ability to efficiently predict the ringstiffness is sought. The fact that the pipe geometry is governed only by a few parameters makes process automation desirable and feasible. The necessary input parameters are:

- Profile geometry (in the form of a shell representation of a C-profile)
- Internal pipe diameter
- Length of the pipe sample
- Young's Modulus (dependent upon strain rate, strain level and temperature as discussed in Section 5.4)

After provision of these, the FE model is created and solved automatically. The flow of an automated simulation is illustrated in Figure 5.54. The process can be summarized as follows: The created half profiles are mirrored and merged so many times until they reach the desired sample length as discussed in Section 5.5.1.4. This process creates the 2D profile for the pipe sample. Now, this cross-section is spun around the pipe's centre to form a quarter of a circle (due to symmetry only 1/8th of the total the pipe is modeled). A typical FE model is shown in Figure 5.55.

5.5.2.2. Simplifications in the FE Model

The proposed FE model inherits a number of simplifications. Firstly, the real spiral formation of the profiles (see Section 5.3.1) is neglected in the model, i.e. the profiles are not offset by its width when spun. As a result, the sides of the pipe model are flat. In reality, due to the spiral formation, the profiles in a pipe sample on either end are partly cut open.

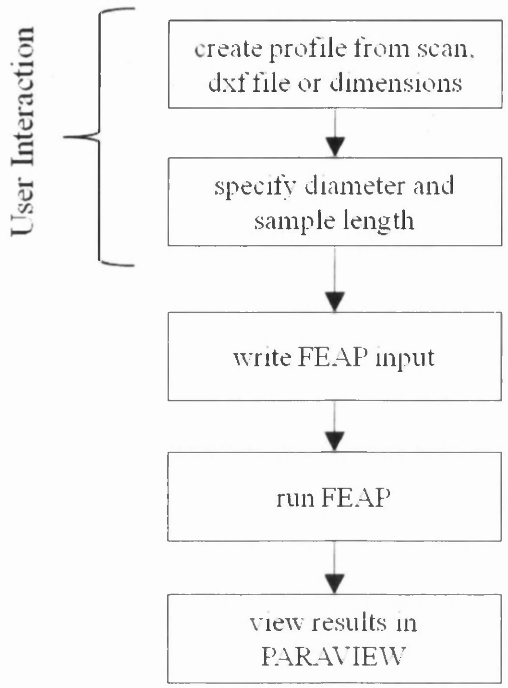


Figure 5.54: Flow of the automated ringstiffness test simulation.

Figure 5.55: FE model for ringstiffness test. Due to symmetry only 1/8th is modeled.

Another approximation in the model corresponds to the loading plates. Rather than simulating the plates themselves which would require solving a more complex contact problem, the loading is idealized by a prescribed displacement of the FE nodes at the pipe crown. That means the flattening of the pipe crown as well as the friction between the plate and the pipe are not simulated.

Further approximation stems from the fact that a linear elastic material model is used. The corresponding error was sought to be limited by the use of a secant modulus that takes a mean strain level, a mean strain rate and the pipe's temperature into account (see Section 5.4).

A final approximation corresponds to the geometrical representation of the pipe profiles. Despite the scanning of profile samples, the shell representation is not capable of simulating all details. For instance the possible gap between the inside and the outside weld (see Figure 5.10) is omitted. In a separately conducted study (Sienz et al., 2008) various aspects on the accuracy of FE simulations of the ringstiffness tests have been investigated, for instance the difference of a discretization using solid and shell elements. A typical model using solid elements is shown in Figure 5.56.

Obviously, small details, such as the additional material of the inside weld can be grasped more accurately than with shell elements. The profile geometry of three pipes

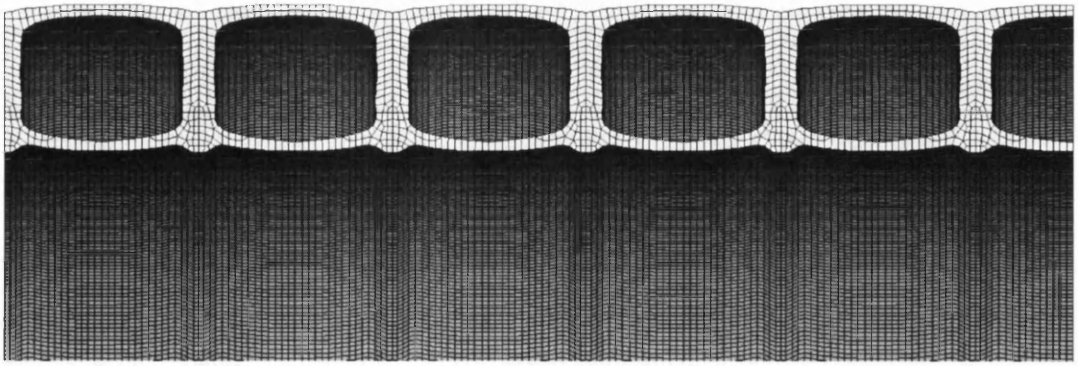


Figure 5.56: Typical model using solid elements studied in (Sienz et al., 2008).

have been obtained by means of laser scans and used to created FE models meshed with solid and shell elements. The results of the models using shell elements matched those of the models meshed with solid elements with an accuracy of 1.5, 2.2 and 2.7%, while the latter was about four times as efficient.

5.5.3. Optimization

The objective to incorporate an optimization feature in the design tool is to provide the option to generate a profile with minimum material expenditure which exhibits a desired ringstiffness for a defined diameter while meeting stress and manufacturing constraints.

5.5.3.1. FE Model for Optimization

During optimization processes simulations have to be performed repeatedly, hence the computational effort of such a simulation should be as small as possible. The geometrically repeating pattern of profiles in the structured wall pipes makes it interesting to further reduce the model, although the geometrical symmetry is not a structural one. That is to say, the boundary conditions are not the same for every isolated C-profile in the model. If one isolated C-profile is to be used to model the ringstiffness test within the optimization procedure, the question that arises is what the appropriate boundary conditions could be. The great advantage of using only one C-profile in the model would be that it is small enough to allow numerous analyses during the optimization.

As it can be seen in Figure 5.57, the horizontal displacement of the profile webs increases as one travels from the middle of the pipe sample towards one side. That is to say, the web near the symmetry plane of the pipe sample (in Figure 5.57 on the right side) remains relatively vertical and that near the end of the sample (in Figure 5.57 on the left side) is more inclined.

The C-profile on the right exhibits a behaviour that resembles the one it would have if the x-displacements on its face were restricted. On the contrary, the C-profile on the left effectively has its face unrestricted. Therefore, there is no one C-profile that can realistically model the behaviour of the pipe subjected to the ringstiffness test. The best that can be aimed for is to develop some artificial boundary conditions for a C-profile (without a real counterpart) that can 'represent' rather than 'simulate' the real problem. That is to say, this C-profile and its boundary conditions would not be simulating any of the C-profiles in Figure 5.57. It must be kept in mind that these boundary conditions are artificial.

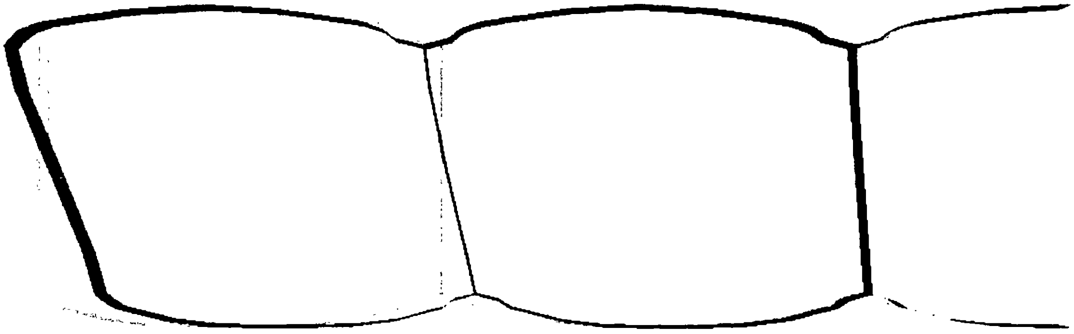


Figure 5.57: Exaggerated horizontal displacement of a profile in the ringstiffness test simulation. The displacement of the web increases from the middle towards the side.

It seems self-evident that the ringstiffness obtained with the model of the full pipe, i.e. with the real boundary conditions, would be higher than that of the C-profile with its web face unrestrained (see Figure 5.58). Similarly, the ringstiffness of the whole pipe would be lower than that of the same C-profile but now having a perfect restriction of the horizontal displacement of the nodes on the web face (see Figure 5.59).

Hence, the C-profile with the free web lower-bounds the ringstiffness while the C-profile with the restricted web upper-bounds it. However, the range of ringstiffness values bounded by these two C-models might be quite wide. Therefore, some artificial intermediate boundary condition for the web face needs to be conceived if a C-profile is to be used to model the ringstiffness test.

One way to account for the stiffening effect in the pipe axis direction of the omitted pipe sections is to use spring elements as supports on the side, where the higher or lower

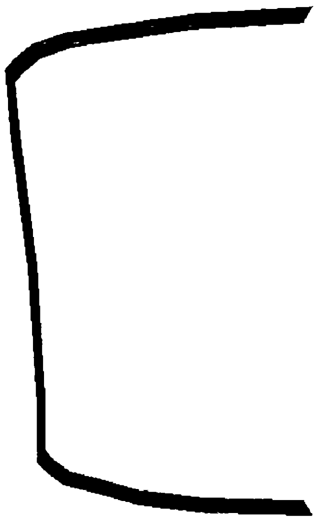


Figure 5.58: Exaggerated horizontal displacement of a C-profile in the ringstiffness test simulation with the horizontal displacements of the web unconstrained.

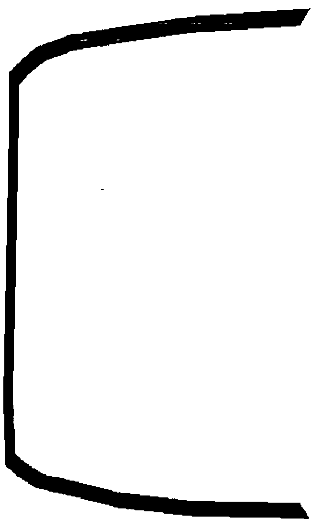


Figure 5.59: Exaggerated horizontal displacement of a C-profile in the ringstiffness test simulation with the horizontal displacements of the web fully constrained.

restrictive effect can be controlled by 'tuning' the springs' stiffnesses. Thus, the latter can be adjusted in such a way that the ringstiffness obtained by using the C-profile with springs matches that of the full pipe. It is important to keep in mind that this model cannot be used to analyze the local behaviour of specific C-profiles, but to efficiently analyze the global behaviour such as the calculation of the ringstiffness.

The spring stiffness assigned to every single spring depends upon the size of the area that the springs stands for. Thus, the concept 'lateral pipe stiffness' (lps) is introduced in Equation 5.25, where S_{spring} is the spring stiffness, n_{spring} is the number of springs, A_{web} is the area that the springs are attached to, d_i and d_o are the inner and outer diameters of the pipe, and h is its height of the cross-section.

$$lps = \frac{S_{spring} \cdot n_{spring}}{A_{web}} \tag{5.27}$$

$$A_{web} = h \cdot \frac{\pi}{2} \cdot \frac{d_i + d_o}{4} \tag{5.28}$$

This concept allows that, if changes in the geometry or discretization are made, the new spring stiffness can be determined from the constant lps without having to tune the former until the results of the C-model match those of the whole model.

Since the pipe should only be stiffened in the x-direction, the support of each spring on the extreme that is not attached to the pipe should only be restricted in that direction.

This is a kinematic configuration (i.e. not entirely constrained), which cannot be handled by linear solvers such as FEAP (Berkeley University, 2005). This is overcome by restraining the support in all directions, but placing it far away from the pipe so that the components on directions other than x are comparatively negligible. A typical FE model using the C-profile with spring support can be seen in Figure 5.60. In this illustration the springs are cut i.e. not shown with their full length to better illustrate the model.

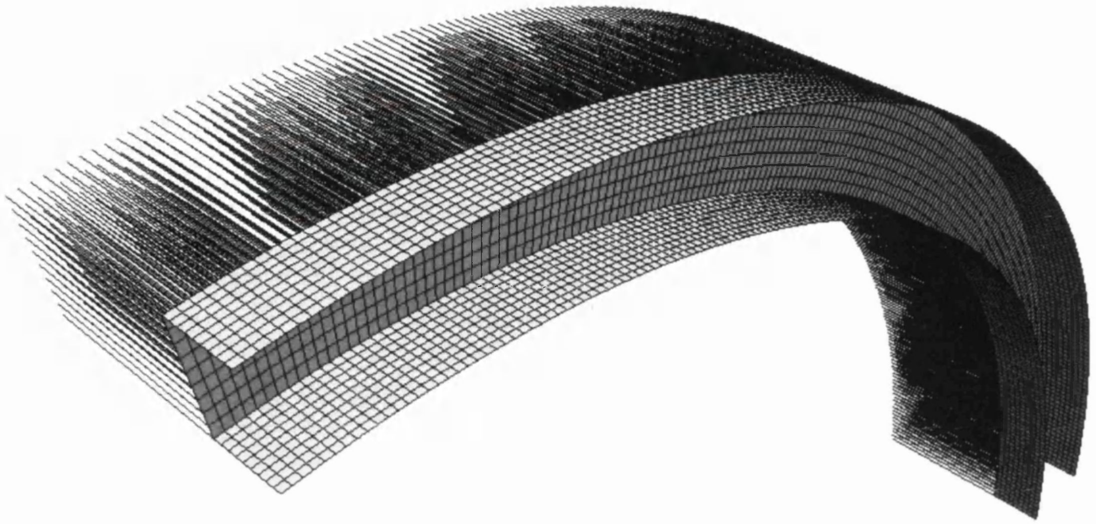


Figure 5.60: FE model used for ringstiffness optimization. For improved efficiency only one C-profile is used here. The reduced horizontal stiffness is accounted for by applying horizontal springs in the web area. For better illustration the springs are cut i.e. not shown with their full length.

Because the correct value for *lps* is unknown before an optimization process, a value has to be chosen prior to optimization. In order to give a guideline on this choice, 15 simulations of pipes with various diameters and stiffnesses have been conducted. Furthermore, for each simulation the correct *lps* has been found such that the C-model yields the result of the whole model. The results showed that irrespective of the resulting stiffness, a correlation between the internal diameter of the pipe and the required *lps* can be observed. In Figure 5.61 the required *lps* is plotted against the internal diameter. An exponential relationship between the two values can be defined as:

$$lps \approx 1.78908 \cdot 10^4 \cdot e^{-9.1475 \cdot 10^{-4} \cdot d_i} \qquad 5.29$$

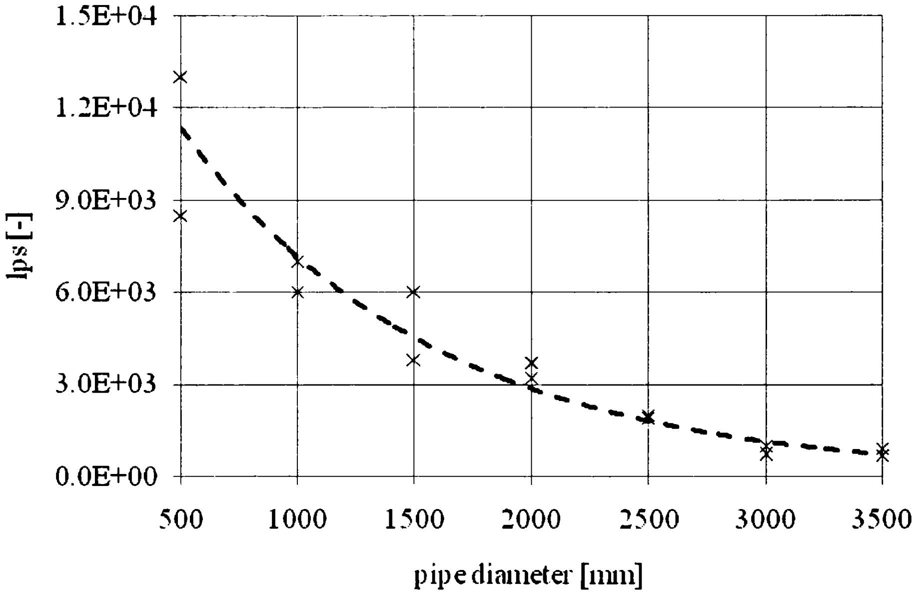


Figure 5.61: The required *lps* to obtain the same ringstiffness as with the whole model, plotted against the pipe diameter.

For smaller diameters this relationship becomes less accurate and thus, after convergence the obtained optimum profile dimensions should be used to run an analysis with the entire pipe modeled. If the ringstiffness of this analysis is lower than the desired one, the process is repeated with a decreased *lps* since the horizontal resistance of the pipe has been overestimated. In a similar manner the value for *lps* is increased if the ringstiffness is too high.

5.5.3.2. Optimization Process

The manufacturability of the profiles is controlled via several dimensional relationships. Two relationships govern the ratio between the thicknesses, which make the web thickness t_2 and the bottom thickness $t_3$dependent upon the top thickness t_1. They read:

$$t_1 = a_1 \cdot t_3 \tag{5.30}$$

$$t_2 = \frac{t_1 + t_3}{a_2} \tag{5.31}$$

Thus, the design variables governing the profile are the height h, the width w and the top thickness t_1. The optimization problem can be summarized as follows: The objective function f is the area of the profile divided by its width w, i.e. the expended material per unit pipe length is minimized. A first constraint g_1 enforces that the pipe exhibits at least the desired ringstiffness. Furthermore, the maximum stress in the pipe sample is limited to a limit (constraint g_2). Due to the assumptions in the model this value should be chosen conservatively, i.e. lower than the actual yield stress of the material. Five more constraints g_3, g_4, g_5, g_6 and g_7 govern the profile dimensional relationships also ensuring a manufacturable design (the values for the parameters a_1 through a_7 are confidential). The optimization problem is fully defined as:

$$min \rightarrow f = \frac{w \cdot (t_1 + t_3) + h \cdot t_2 \cdot 2}{w} \tag{5.32}$$

$$g_1 = SN_{des} - SN \leq 0 \tag{5.33}$$

$$g_2 = |\sigma|_{max} - \sigma_{limit} \leq 0 \tag{5.34}$$

$$g_3 = a_3 - \frac{t_3}{h} \leq 0 \tag{5.35}$$

$$g_4 = \frac{t_3}{h} - a_4 \leq 0 \tag{5.36}$$

$$g_5 = \frac{h}{d} - a_5 \leq 0 \tag{5.37}$$

$$g_6 = a_6 - \frac{w}{h} \leq 0 \tag{5.38}$$

$$g_7 = \frac{w}{h} - a_7 \leq 0 \tag{5.39}$$

An open source SQP tool is used to solve this problem (Hinton et al., 2003). For more information on this type of optimization algorithm, see Section 2.2.3.

The flow of the optimization process can be described as follows: It starts after specification of the desired ringstiffness, maximum stress, the internal diameter of the pipe and *lps*. For an initial profile the FE model is created and written to an input file which is read by FEAP. After the model has been solved the results are used to calculate the objective and the constraint function values. Now the three design variables are perturbed one at a time and the corresponding simulation is run again. These results are used to calculate the sensitivities. The gathered data is passed on to the SQP tool which improves the design by modifying the variables and a new iteration starts. This procedure continues until convergence occurs. The flow is illustrated in Figure 5.62.

5.5.4. Applications

In the following, the developed profile generation feature, the ringstiffness test simulation and the design optimization tool will be applied. 13 ringstiffness test results are available. Seven of them were conducted by KWH and six by Asset. It will be

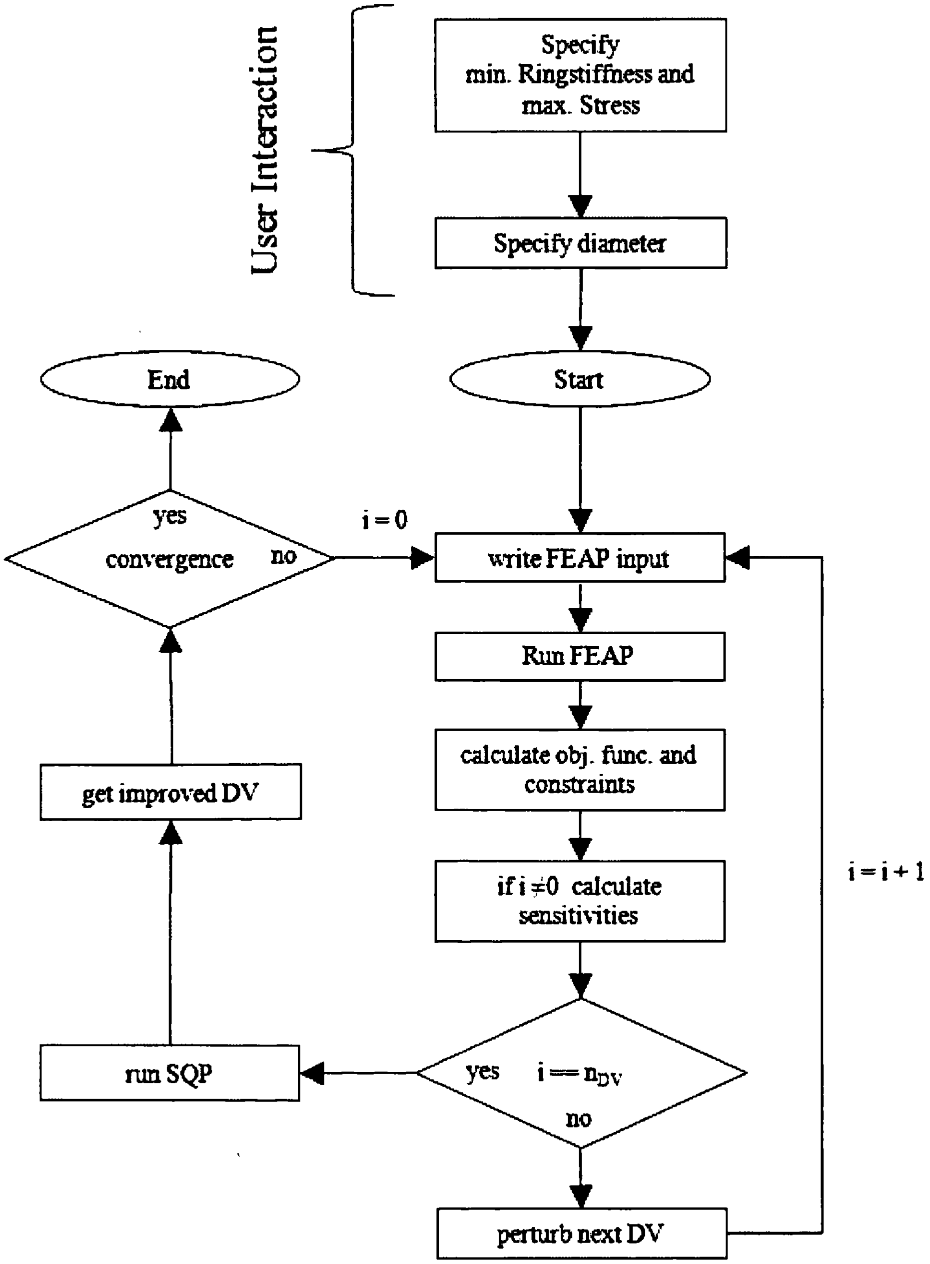


Figure 5.62: Flow of the developed optimization procedure.

attempted to reproduce these results. Subsequently, two of the original profiles will be improved by means of the optimization tools.

To this end profiles have been cut from the corresponding pipes and scanned using the laser scanner. The obtained data has been processed to generate the profile's shell element representations (see Section 5.5.1.1). Using the material model discussed in Section 5.4 the appropriate Young's modulus is calculated for the corresponding test conditions. In a second step two of the simulations are rerun using rectangular box sections. The profiles are then redesigned using the optimization feature and compared to the original ones.

5.5.4.1. Simulation using Scanned Profiles (KWH)

Results are available from seven ringstiffness tests that were conducted by KWH. The corresponding test conditions as well as the resulting ringstiffnesses of these tests are listed in Table 5.6. It can be seen that in all the tests the ambient and the pipe temperature are 23°C. This is because the pipes are stored at that temperature for a sufficiently long time to ensure that the pipes have a uniform temperature across their cross-section.

Table 5.6: The test conditions and the ringstiffnesses of the seven ringstiffness tests conducted by KWH.

		1	2	3	4	5	6	7
d_i	[mm]	1407	2402	2200	1600	1600	1000	600
d_o	[mm]	1577	2596	2399	1795	1737	1121	671
profile	[-]	4K	2K	2K	4K	2K	4K	8K
date	[-]	Jan 08	Nov 07	N/A	N/A	N/A	N/A	N/A
T_{pipe}	[°C]	23	23	23	23	23	23	23
T_{amb}	[°C]	23	23	23	23	23	23	23
T_{avg}	[°C]	23	23	23	23	23	23	23
y	[mm]	42.2	72.1	66.0	48.0	48.0	30.0	18.0
speed	[mm/min]	50	50	50	50	50	50	20
t_{test}	[s]	50.7	86.5	79.2	57.6	57.6	36.0	54.0
L_{sample}	[mm]	673	598	N/A	N/A	N/A	N/A	N/A
SN	[kN/m^2]	6.3	2.1	2.2	7.3	2.7	6.9	8.1

Two exemplary profiles are depicted in Figure 5.63. These are the 8kN profile produced for a 600mm diameter pipe and tested in Test 7 as well as the 2kN profile produced for a 2200mm diameter pipe and tested in Test 3. One can see that both profiles exhibit only a small deviation from the theoretical rectangular box-section.

The parameters, that are necessary to calculate the Young's modulus according to the developed material model discussed in Section 5.4, are listed in Table 5.7. The representative strain level $\bar{\varepsilon}$ has been calculated with the expected ringstiffness and Equation 5.26. The strain rate $\dot{\bar{\varepsilon}}$ is taken as $\bar{\varepsilon}$ divided by the duration of each test. This

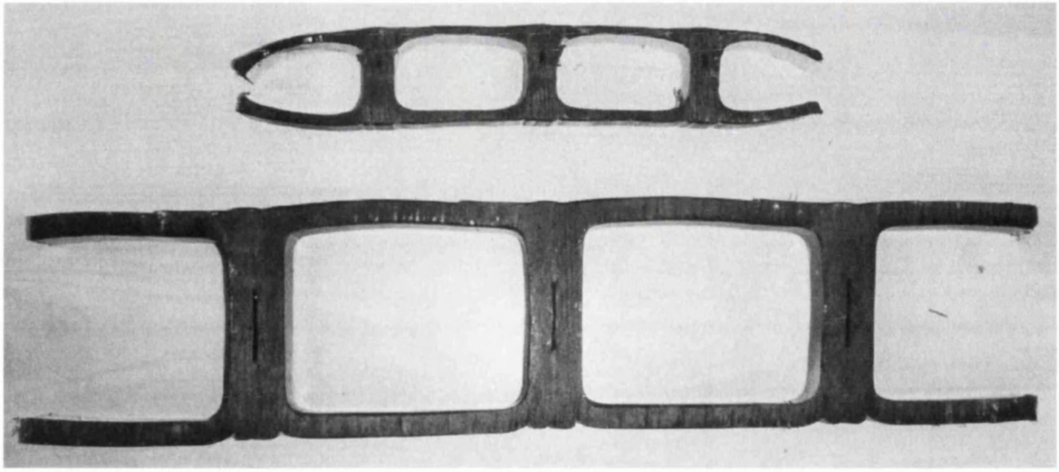


Figure 5.63: Exemplary profiles from KWH experiments: a) Test 7 (8kN profile for a 600mm diameter pipe) and b) Test 3 (2kN profile for a 2200mm diameter pipe).

Table 5.7: The parameters for the calculation of the Young's modulus in the simulations of the seven ringstiffness tests conducted by KWH.

		1	**2**	**3**	**4**	**5**	**6**	**7**
$\bar{\varepsilon}$	*[-]*	*4.8E-3*	*3.6E-3*	*3.7E-3*	*5.0E-3*	*3.8E-3*	*4.9E-3*	*5.3E-3*
$\dot{\bar{\varepsilon}}$	$[s^{-1}]$	*9.4E-5*	*4.2E-5*	*4.6E-5*	*8.7E-5*	*6.6E-5*	*1.4E-4*	*9.7E-5*
f	*[s]*	*0.128*	*0.057*	*0.063*	*0.119*	*0.090*	*0.186*	*0.132*
E'	*[MPa]*	*1194.3*	*1104.8*	*1114.1*	*1188.4*	*1158.6*	*1232.5*	*1197.2*
$\tan\delta$	*[-]*	*0.137*	*0.156*	*0.154*	*0.139*	*0.144*	*0.130*	*0.137*
fac_{visco}	*[-]*	*1.137*	*1.156*	*1.154*	*1.139*	*1.144*	*1.130*	*1.137*
fac_{level}	*[-]*	*0.669*	*0.669*	*0.672*	*0.658*	*0.686*	*0.683*	*0.656*
E_{FE}	*[MPa]*	*908.7*	*853.6*	*863.5*	*890.7*	*909.4*	*950.5*	*893.3*

in return yields the DMTA frequency when inserted into Equation 5.20. From the DMTA test data the storage modulus E' and the loss tangent are interpolated using the frequency and the average temperature T_{avg}. The Young's modulus for the FE simulations E_{FE} is then calculated with Equation 5.23.

With the Young's modulus at hand, the remaining parameters can be directly read from Table 5.6. These are the sample length L_{exp} and the internal diameter d_i. The sample length in the simulation L_{FE}, the resulting ringstiffness SN_{FE} and the error between SN_{FE} and SN_{exp} are listed in Table 5.8. The results are also illustrated in Figure 5.64. Because the profiles should be closed on the outside, the sample length in the experiment L_{FE} will not be exactly the same as in the simulation L_{exp} (see Section 5.5.1.4).

Table 5.8: The parameters for the conducted simulations to reproduce the test results of the seven ringstiffness tests conducted by KWH.

		1	2	3	4	5	6	7
L_{FE}	*[mm]*	*770*	*600*	*782*	*815*	*769*	*748*	*659*
SN_{exp}	*[kN/m²]*	*6.3*	*2.1*	*2.2*	*7.3*	*2.7*	*6.9*	*8.1*
SN_{FE}	*[kN/m²]*	*7.0*	*2.4*	*2.2*	*7.5*	*2.9*	*7.6*	*8.6*
error	*[%]*	*12*	*13*	*2*	*3*	*8*	*10*	*6*

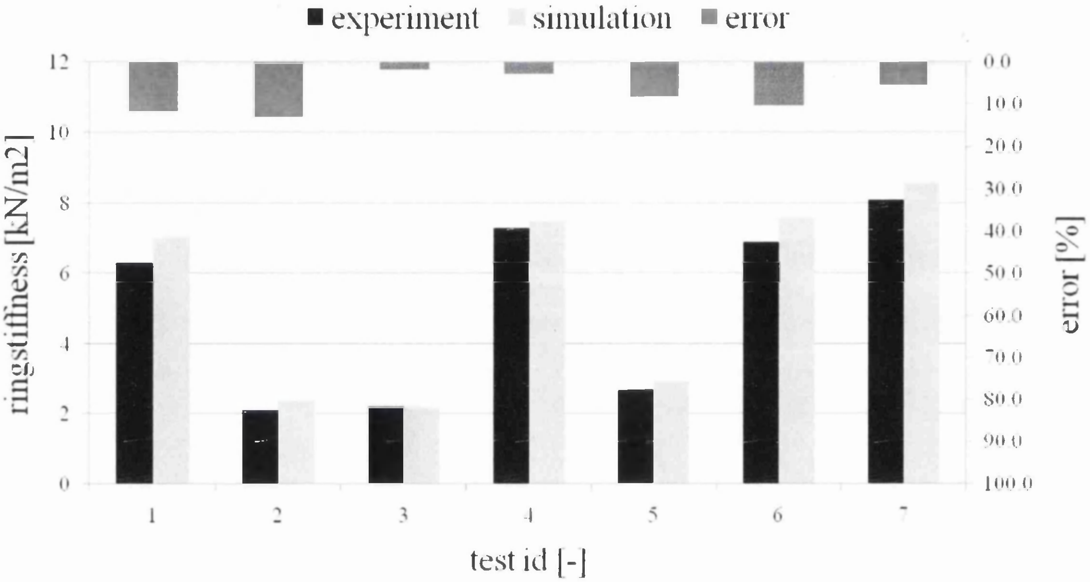


Figure 5.64: Resulting ringstiffness from the seven experiments conducted by KWH and from the corresponding simulations. The error ranges between 13 and 2%.

The differences between the experiment and the simulation ringstiffnesses range from 2 to 13%. In all cases, apart from Test 3, where the result from the simulation actually coincided with the experiment, the simulations overestimate the experiment ringstiffnesses. The reason for this could be the fact that, as explained in Section 5.5.2.2, the pipe profiles on the sides in the experiments are partly cut open which naturally leads to a pipe that is less stiff than one where the profiles are fully intact. This possibility of an overestimation of the ringstiffness should be kept in mind when a simulation is performed. Nevertheless, considering the sources for variability of test results discussed in Section 5.3.4.2, the high degree of automation as well as the resulting effectiveness and ease of use of the simulation tool, the results can be considered satisfactory.

5.5.4.2. Simulation using Scanned Profiles (Asset)

Results exist from six ringstiffness tests that were conducted by Asset Ltd. The corresponding test conditions as well as the resulting ringstiffnesses of these tests are listed in Table 5.9.

Table 5.9: The test conditions and the ringstiffnesses of the six ringstiffness tests conducted by Asset.

		8	9	10	11	12	13
d_i	[mm]	750	1500	1500	1800	1800	2100
d_o	[mm]	840	1651	1640	1964	1970	2270
profile	[-]	4K	2K	2K	2K	2K	2K
date	[-]	Aug 06	May 05	May 06	Feb 05	Feb 06	Feb 07
T_{pipe}	[°C]	15	12	20	10	12	18
T_{amb}	[°C]	19	10	19	8	14	14
T_{avg}	[°C]	17	11	19.5	9	13	16
y	[mm]	22.5	45	45	54	54	63
speed	[mm/min]	20	50	50	20	50	50
t_{test}	[s]	67.5	54	54	162	64.8	75.6
L_{sample}	[mm]	900	950	900	950	900	920
SN	[kN/m^2]	8.9	3.74	3.6	3.6	3.1	2.68

Contrary to the tests conducted by KWH, here the pipe and ambient temperatures are not identical. In some occasions the pipes are tested directly after production when the inside material is still hot and soft. On other occasions, the pipes are stored outside where they might be exposed to sunlight which leads to a high surface temperature with a low temperature on the inside. These variations naturally make the reproduction of the ringstiffness a more difficult task and higher differences between simulation and experiment results have to be expected.

Two exemplary profiles are depicted in Figure 5.65. These are the 4kN profile produced for a 750mm diameter pipe and tested in Test 8 as well as a 2kN profile produced for a 2100mm pipe and tested in Test 13. Especially the latter profile exhibits a significant curvature of the top and bottom walls. That is to say the difference between the theoretical rectangular box-section and the produced profile is considerable which indicates that a higher variability of the test results has to be expected.

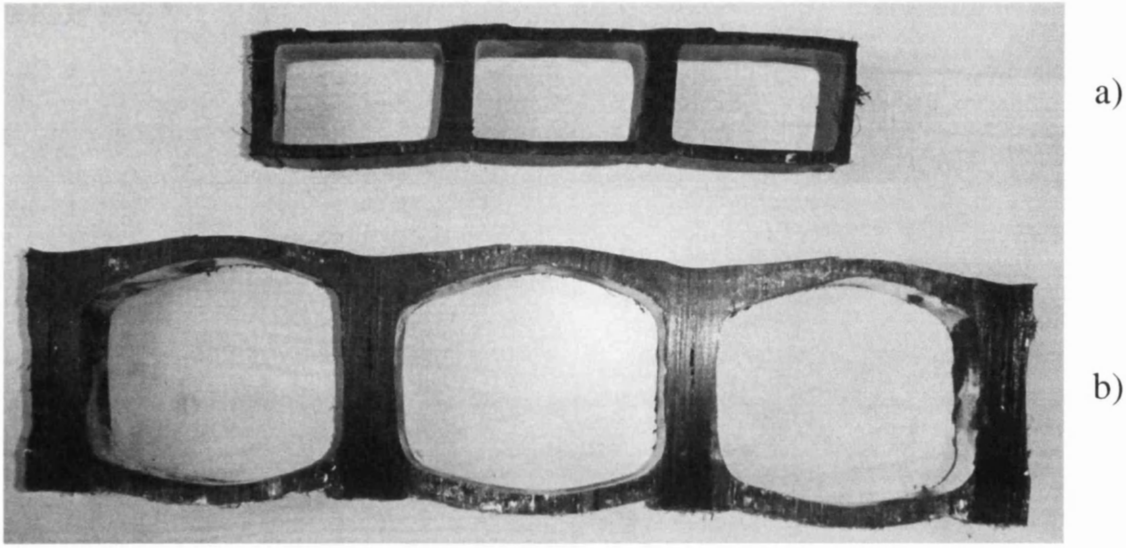


Figure 5.65: Exemplary profiles from Asset experiments: a) Test 8 (4kN profile for a 750mm diameter pipe) and b) Test 13 (2kN profile for a 2100mm diameter pipe).

The Young's modulus is calculated in the same manner as described in Section 5.5.4.1. The material test data for the Asset material is shown in Appendix B. The parameters used to calculate E_{FE} are listed in Table 5.10. The sample length in the simulation L_{FE}, the resulting ringstiffness SN_{FE} and the error between SN_{FE} and SN_{exp} are listed in Table 5.11. These results are also illustrated in Figure 5.66.

Table 5.10: The parameters for the conducted simulations to reproduce the test results of the six ringstiffness tests conducted by Asset.

		8	**9**	**10**	**11**	**12**	**13**
$\bar{\varepsilon}$	*[-]*	*5.5E-3*	*4.1E-3*	*4.0E-3*	*4.0E-3*	*3.9E-3*	*3.8E-3*
$\dot{\bar{\varepsilon}}$	*[s-1]*	*8.1E-5*	*7.5E-5*	*7.5E-5*	*2.5E-5*	*6.0E-5*	*5.0E-5*
f	*[s]*	*0.110*	*0.103*	*0.102*	*0.034*	*0.082*	*0.068*
$\tan\delta$	*[-]*	*0.112*	*0.904*	*0.124*	*0.113*	*0.106*	*0.125*
fac_{visco}	*[-]*	*1.112*	*1.090*	*1.124*	*1.113*	*1.106*	*1.125*
fac_{level}	*[-]*	*0.733*	*0.811*	*0.742*	*0.765*	*0.783*	*0.748*
E'	*[MPa]*	*1562.4*	*1768.6*	*1464.6*	*1579.4*	*1629.7*	*1472.8*
E_{FE}	*[MPa]*	*1274.4*	*1564.6*	*1221.1*	*1345.1*	*1411.2*	*1239.3*

In Test 12 the ringstiffness is overestimated by nearly 50%. Comparison of the experiment results from Test 11 and Test 12 suggests that an error occurred during the second test. While both tests are performed with the same pipe the first one has been performed at a deflection speed of 20mm/min and the second at 50mm/min. Even

Table 5.11: The parameters for the conducted simulations to reproduce the test results of the six ringstiffness tests conducted by Asset.

		8	**9**	**10**	**11**	**12**	**13**
L_{FE}	*[mm]*	*725*	*960*	*960*	*1000*	*780*	*960*
SN_{exp}	*[kN/m^2]*	*8.9*	*3.7*	*3.6*	*3.6*	*3.1*	*2.7*
SN_{FE}	*[kN/m^2]*	*10.7*	*4.6*	*3.6*	*4.4*	*4.6*	*2.4*
error	*[%]*	*20*	*19*	*1*	*23*	*48*	*9*

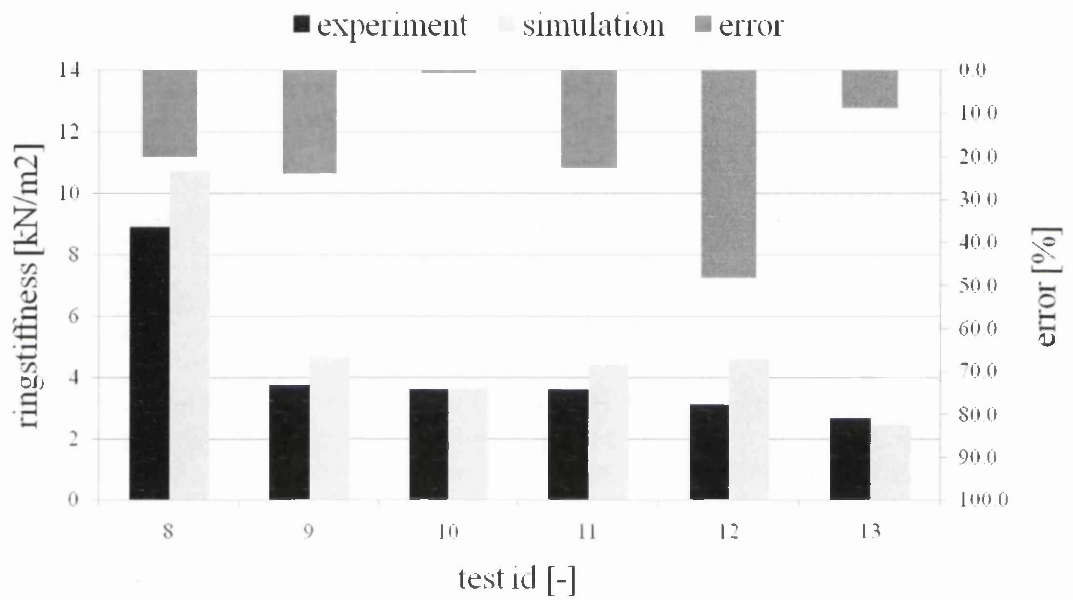


Figure 5.66: Resulting ringstiffness from the six experiments conducted by Asset and from the corresponding simulations. The error ranges between 48 and 1%.

though the second average temperature is 4°C higher, the second pipe should be stiffer than the first since the test is 2.5 times as fast. The remaining differences between the experiment and the simulation ringstiffnesses range from 1 to 23%. As expected the errors are larger for these simulations where the pipes have not been stored at constant temperature, in preparation for the experimental stiffness test.

5.5.4.3. Simulation with Rectangular Profiles and Profile Optimization

In the following the developed profile optimization feature will be applied to two pipes and the resulting profiles will be compared to the original ones. The experiments that will be considered here are Examples 3 and 7 discussed in Section 5.5.4.1.

Since rectangular box profiles are used in the optimization procedure, at first those dimensions for the rectangular profiles are found that lead to the same ringstiffness as the scanned versions. The rectangular box sections replacing their scanned counterparts are illustrated in Figure 5.67 and Figure 5.68. Their dimensions are listed in Table 5.13.

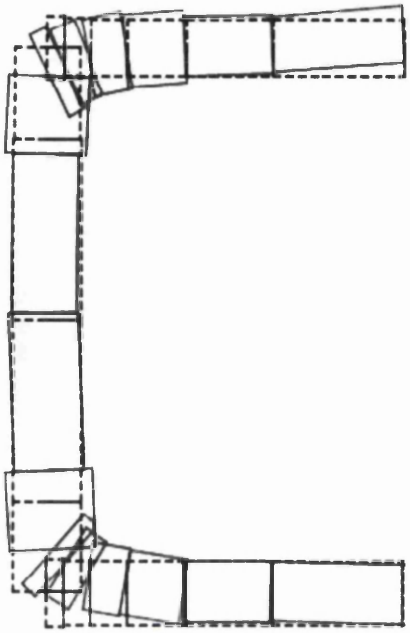


Figure 5.67: Test 3: The scanned profile and the corresponding rectangular box-section that leads to the same ringstiffness.

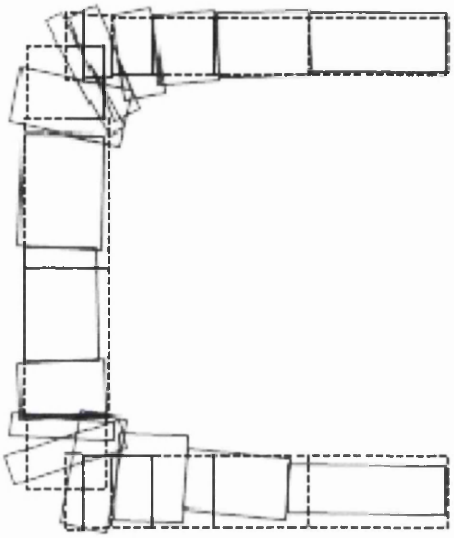


Figure 5.68: Test 7: The scanned profile and the corresponding rectangular box-section that leads to the same ringstiffness.

The sharp corners in the rectangular box-sections as shown in Figure 5.67 and Figure 5.68 obviously do not reflect reality and could lead to inaccuracies due to stress concentrations. However, due to the nature of the model creation, where the C-sections are mirrored and merged, these sharp corners disappear in the FE model wherever two profiles are merged and hence, comparable stresses are obtained (see Table 5.12). The FE models for both tests and both profile types are shown in Figure 5.69 and Figure 5.70.

Table 5.12: Maximum and minimum stresses for FE simulations based on the scanned and the rectangular cross-sections for test 3 and test 7.

		σ_{xx}		σ_{yy}		σ_{xy}	
		min	*max*	*min*	*max*	*min*	*max*
Test 3	*Scan*	*-41.43*	*39.87*	*-21.03*	*5.62*	*-1.99*	*6.88*
	Rectangular	*-41.07*	*34.58*	*-20.79*	*4.89*	*-1.14*	*6.98*
Test 7	*Scan*	*-36.39*	*37.64*	*-26.31*	*13.90*	*-2.36*	*9.12*
	Rectangular	*-36.00*	*29.58*	*-21.44*	*8.37*	*-1.31*	*8.79*

The optimization problem has been summarized in Equations 5.32 to 5.39. The desired minimum ringstiffness has been set to that obtained in the simulations, i.e. 2.2 and 8.6kN/m^2 and the maximum allowable stress has been set to 35MPa.

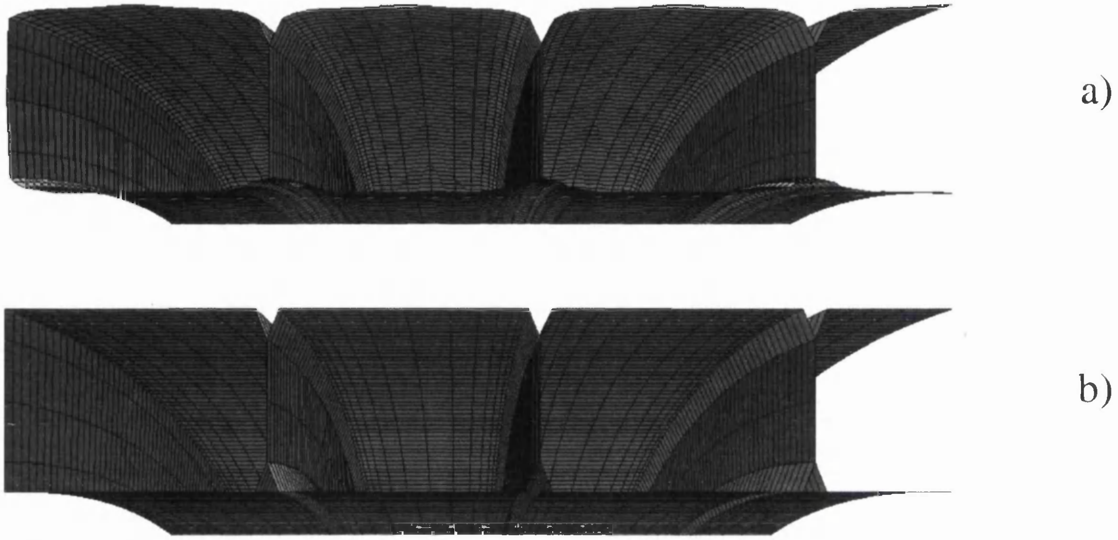


Figure 5.69: FE model for Test 3 from a) scanned cross-section and b) rectangular profile.

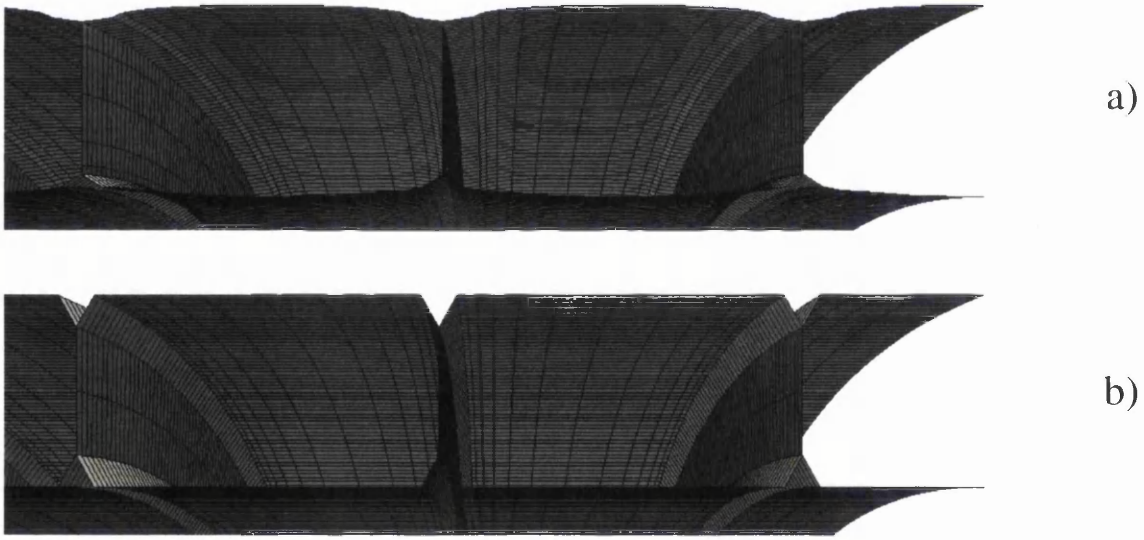


Figure 5.70: FE model for Test 7 from a) scanned cross-section and b) rectangular profile.

The results from the optimization runs are also listed in Table 5.13. These are the optimum profile dimensions as well as the corresponding objective function and the improvement with respect to the original profiles. In both cases material savings of more than 20% are possible.

In the first case the resulting optimum profile is both higher and wider than the original one while the web thickness is slightly reduced. In the second case only the height is

Table 5.13: The dimensions of the original and the optimized profile. Also, the cross-sectional area, objective function and its improvement, i.e. the saved material per unit pipe length, are listed.

		w	*h*	t_1	t_2	t_3	*lps*	*area*	*obj.*	*imp*
Test	*profile*			*[mm]*			*[-]*	*[mm^2]*	*[mm]*	*[%]*
3	*orig.*	*114.0*	*85.0*	*8.0*	*10.0*	*9.0*	*N/A*	*3638*	*31.9*	*22*
	opt.	*150.9*	*94.3*	*6.8*	*7.7*	*8.5*	*2000*	*3761*	*24.8*	
7	*orig.*	*61.0*	*36*	*3.5*	*6.0*	*4.5*	*N/A*	*920*	*15.1*	*23*
	opt.	*60.2*	*42.0*	*3.0*	*3.4*	*3.8*	*12000*	*696*	*11.6*	

increased while the width remains the same. Here, the web thickness is almost reduced by a factor of 2. The optimum profiles are illustrated in Figure 5.71 and Figure 5.72.

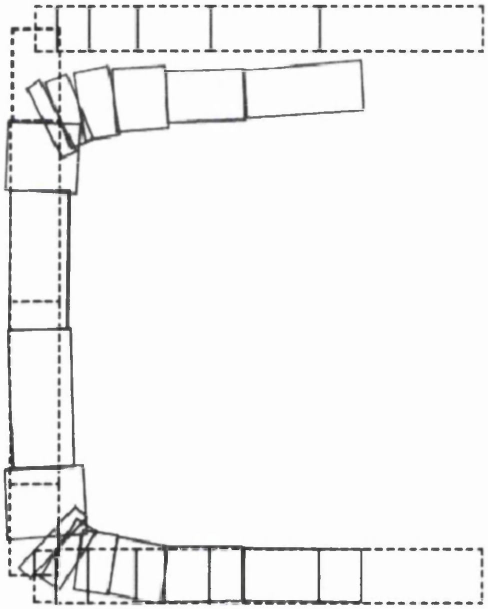


Figure 5.71: Test 3: The scanned and the optimized profile.

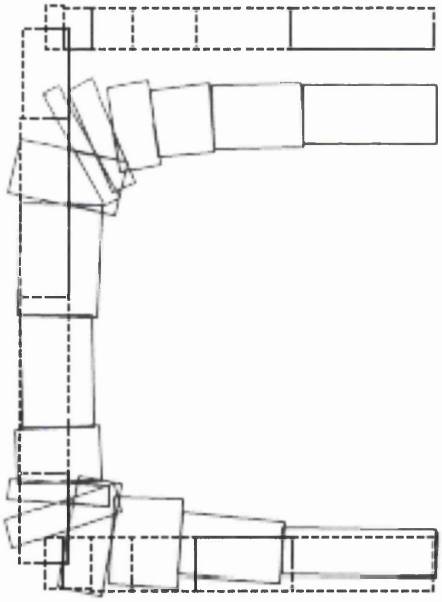


Figure 5.72: Test 7: The scanned and the optimized profile.

As discussed in Section 5.3.1 the theoretical and the produced profiles often differ significantly which led to the decision to use a laser scanner in order to accurately describe their geometries (see Section 5.5.1.1). Naturally, this problem applies in the other direction as well, i.e. the profile dimensions that the optimization yielded are those that the pipe should exhibit after production. Hence, the manufacturer has to aim to produce a profile meeting these values and not use them as the theoretical dimensions. Otherwise, this would lead to a different profile after production.

The *lps* for Test 3 was found to be 2000 and for Test 7, 12000. See Section 5.5.3.1 for a discussion on initial values for *lps*. The development of the objective function and the ringstiffness for the two examples are shown in Figure 5.73.

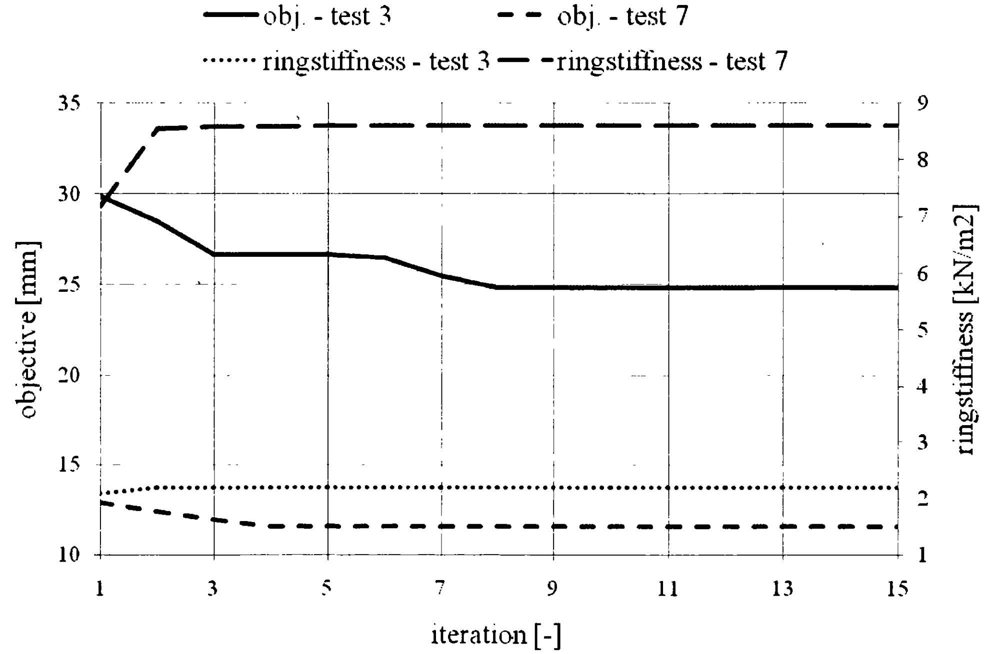


Figure 5.73: Development of the objective function and the ringstiffness during the optimization process for the two examples.

The optimum profile for Test 3 has been found after eight and for Test 7 after four iterations. Due to the manufacturing constraints stated in Equations 5.30 and 5.31 only the profile's height, width and its bottom thickness have to be specified as initial values for the optimization. Since the produced profiles often differ from the theoretical dimensions (see Section 5.3.1) they might not satisfy these constraints. Hence, the objective function and the ringstiffness in the first iteration do not match those found using the rectangular profiles in Table 5.13, even though their dimensions have been specified as the initial values. The development of the maximum absolute stress during the optimization is shown in Figure 5.74. At no time during the optimization have the stress constraints been active.

5.6. Buried Pipe Simulation

5.6.1. CANDE 2007

CANDE is an acronym that stands for Culvert ANalysis and DEsign. It is a public-domain FE software available at (Transportation Research Board, 2007). It allows for

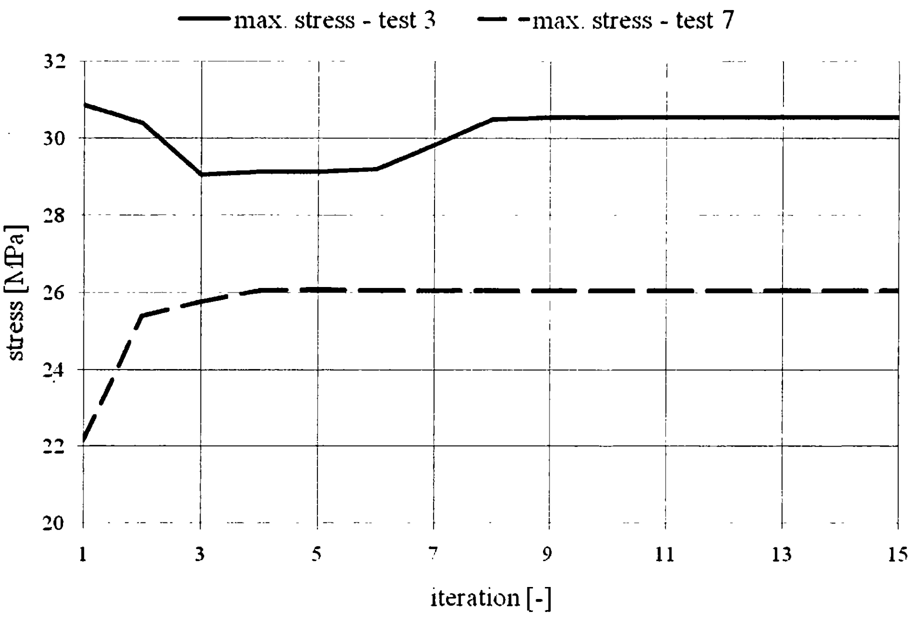


Figure 5.74: Development of the maximum stress during the optimization process for the two examples.

the analysis of the soil-structure interaction in a 2D slice of a culvert installation. It offers the possibility to simulate the installation process by means of an incremental loading schedule (Mlynarski et al., 2007). The pipes are discretized with 1D beam and the soil with 2D shell elements. The interaction between the two can be modeled by means of 1D interface elements. The results of a simulation can be viewed in the built-in post-processor.

The software offers three solution levels (Mlynarski et al., 2007): Level 1 is an approximation which is based on an analytical formula solving idealized conditions. It is suitable for single pipes buried deep in homogenous soil, subjected to gravity loading only. Level 2 is the automated FE solution level. It allows for an efficient solution of buried pipe problems with symmetrical trenches with one pipe only. The number of parameters is very limited, i.e. only a small number of scenarios can be solved in this level. Level 3 allows for the use of user-defined FE meshes. That means, arbitrary geometries and any number of pipes can be analyzed on this level. Since CANDE 2007 does not offer a meshing feature, the geometry and the corresponding FE mesh have to be generated using other tools.

A first version of the software has been developed in 1976 and has been updated several times after that. CANDE 2007 is its latest version. A private company developed a commercial tool called CandeCAD (SSIS Marketing International (Pty) Ltd, n.d.) that uses a CANDE version from 1989 as an FE solver only and offers its own pre- and post-processor to simplify these two steps. There is no link between CANDE 2007 and the company marketing CandeCAD (Mlynarski et al., 2007).

5.6.2. Automated Simulation

This work aims at developing a pre-processor that allows for a parameterized and automated pre-processing and thus, a very efficient simulation of common pipe installation scenarios. The solving and post-processing will be performed by means of CANDE 2007. The simulation tool will be developed in a way that it can be used to efficiently estimate deflections, stresses and strains in pipes and soils for various different installation and loading scenarios. Furthermore, it will be possible to choose different model complexities to improve the efficiency or the accuracy.

5.6.2.1. Parameterization

Setting up a problem with the developed buried pipe simulation tool is divided into two main steps: Specifying the geometry and the simulation parameters. The parameters that need to be specified in each of the two steps will be discussed in the following two sections.

Geometry Parameterization

The parameters governing the geometry of the problem to be solved are:

- Number of pipes and their diameters
- Choose if the whole trench is to be simulated or only the symmetric half
- Trench wall geometry (vertical rigid, see Figure 5.75 or inclined insitu side support, see Figure 5.76)
- In case of a vertical rigid wall the possibility exists to specify an inclined unsupported top part of the side
- In case of an inclined insitu side support the angle of the trench wall needs to be specified
- Height and number of increments of installation steps (primary, secondary and final backfill)
- The existence and size of a concrete slab
- Mesh density
- The distance between the pipes and the workspace next to the outer pipes

The parameters for a symmetric vertical wall installation are illustrated in Figure 5.75 and for an inclined insitu side support in Figure 5.76. The former is used when the sides of the trench are supported by rigid walls that are fixed in the trench during excavation and removed upon refilling.

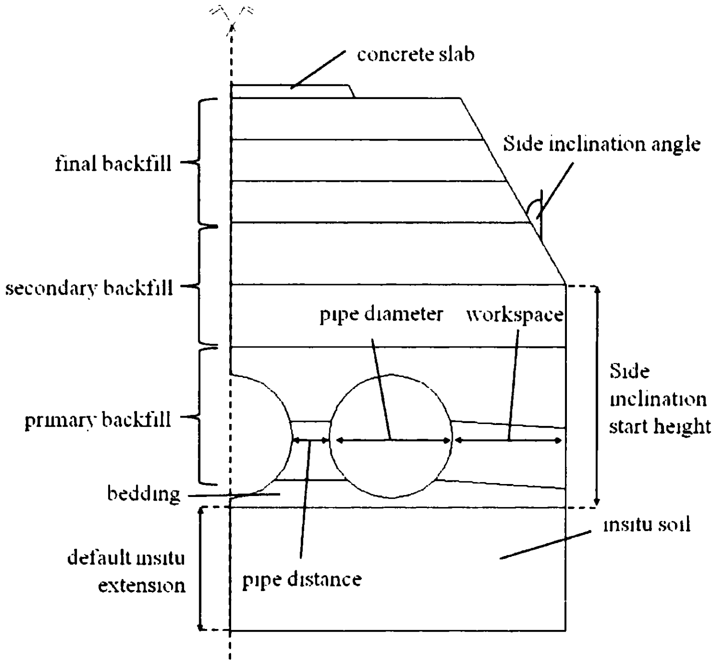


Figure 5.75: Available parameters for a typical rigid wall pipe installation. A part of the wall can be chosen to be inclined and unsupported.

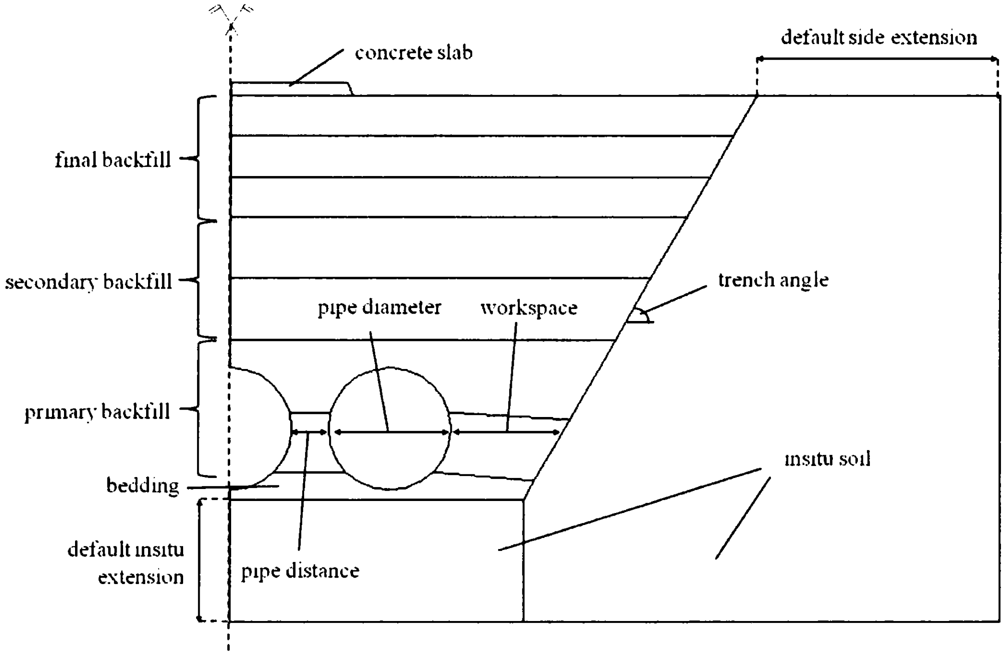


Figure 5.76: Available parameters for a typical trench pipe installation with an inclined insitu side support.

The latter is the case when the trench is not supported by rigid walls. The walls are inclined at an angle and do not need additional support. In this case the insitu soil is extended on the sides by a default value. This value is sufficiently large so that it does not affect the simulation results inside the trench. In both cases the insitu soil below the trench is extended for in a similar manner. Following the recommendations in CANDE 2007 (Mlynarski et al., 2007) these values are taken as 2.0m (for the side extension) and 1.0m (for the bottom extension).

Simulation Parameterization

After the geometry has been generated the simulation parameters need to be specified. These parameters are:

- FE theory (small or large displacement)
- Short or long term pipe properties
- Check for global and/or local buckling
- Use of interface elements for the accurate simulation of the pipe soil interaction
- Geometry of the pipe profiles (height, width and wall thicknesses)
- Material model and canned type or isotropic elastic properties for the different soil materials (insitu soil, bedding soil, primary, secondary and final backfill) as well as the concrete slab
- Specification of external loads

CANDE2007 uses 1D beam elements to model the pipes. However, it allows for the specification of the dimensions of structured wall pipes which it uses internally to calculate the stiffness properties of the elements. Here, the user needs to specify the dimensions of the produced profile. As described in Section 5.3.1, the theoretical dimensions often significantly differ from the ones that are actually produced. Hence, the most accurate way is to scan the profile and to find the corresponding rectangular box-section as shown in Section 5.5.4.3.

The short and long term pipe properties are taken as default values in accordance with CANDE 2007 (Mlynarski et al., 2007) but can be changed if desired.

The material models the user can choose from are linear isotropic, overburden dependent, Duncan and Duncan/Selig. For the linear isotropic model, the user has to specify the Young's modulus, the Poisson's ratio as well as the material density. For the latter three models so called canned materials from CANDE 2007 can be chosen. That means no material properties need to be specified but the type of soil, i.e. for instance granular or cohesive soil as well as its compaction. More information on the available material models can be found in (Mlynarski et al., 2007).

5.6.2.2. Simulation Process

To run a simulation, first a geometry needs to be specified. The necessary input parameters for that are specified in Section 5.6.2.1. The next step is to specify the simulation parameters. These are also discussed in Section 5.6.2.1. After this point no further user interaction is required but the simulation runs automatically. Internally, a geometry is created which is meshed using the meshing software Swan (Sienz & Hinton, 1993). With the mesh and the simulation parameters an input file for CANDE 2007 (Mlynarski et al., 2007) is created. Now CANDE is called in batch mode to solve the model specified in the file. After the run is completed the results can be viewed using the GUI in CANDE. The flow of such an automated simulation is illustrated in Figure 5.77.

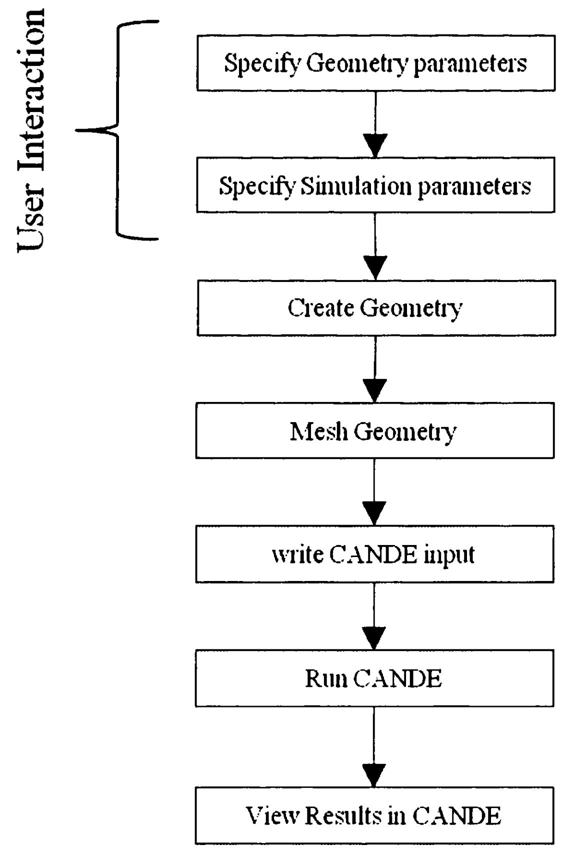


Figure 5.77: Flow of the automated buried pipe simulation.

5.6.3. Applications

In order to demonstrate the capabilities of the buried pipe simulation feature and to investigate the behaviour of flexible pipes in different installation scenarios, three examples will be discussed next. The first example is concerned with the influence of the pipe's side support, followed by an example investigating the effect of different degrees of pipe cover. The third example is based on real measurements taken during the installation of a system containing 15 pipes.

5.6.3.1. Investigation of the Importance of Pipe Side Support

As stated in Section 5.3.5 the horizontal support of the soil surrounding the pipe plays a crucial role in reducing pipe deflection. This importance will be investigated by means of an example where the pipe deflection of different installation scenarios will be compared. To this end, seven scenarios with a varying workspace size will be analyzed.

The chosen material models, as well as the material properties or the canned soil type are listed in Table 5.14.

Table 5.14: The material models, as well as the material properties or the canned soil type for the example investigating the influence of pipe cover.

	Material model	**Canned soil type**	**Compaction**	**Young's modulus [MPa]**	**Poisson's ratio [-]**	**density [kg/m^3]**
Insitu soil	*Linear elastic isotropic*	-	-	*20.0*	*0.3*	*0.0*
Bedding soil	*Duncan*	*Coarse aggregates*	*105%*	-	-	*1922.0*
Primary backfill	*Overburden dependent*	*granular*	*Fair*	-	-	*1922.0*
Secondary backfill	-	-	-	-	--	-
Final backfill	*Overburden dependent*	*mixed*	*Fair*	-	-	*1922.0*
Concrete slab	-	-	-	-	-	-

The general outline of the problem is given in Figure 5.78. Two pipes with a diameter of 2.0m are placed in a trench with a vertical fully supported left side and a by 20° inclined unsupported right side. The primary backfill height ends 300mm above the pipe crown. The final backfill is 1.0m. A load of 500kN/m is distributed over a length of 1.2m above the right pipe.

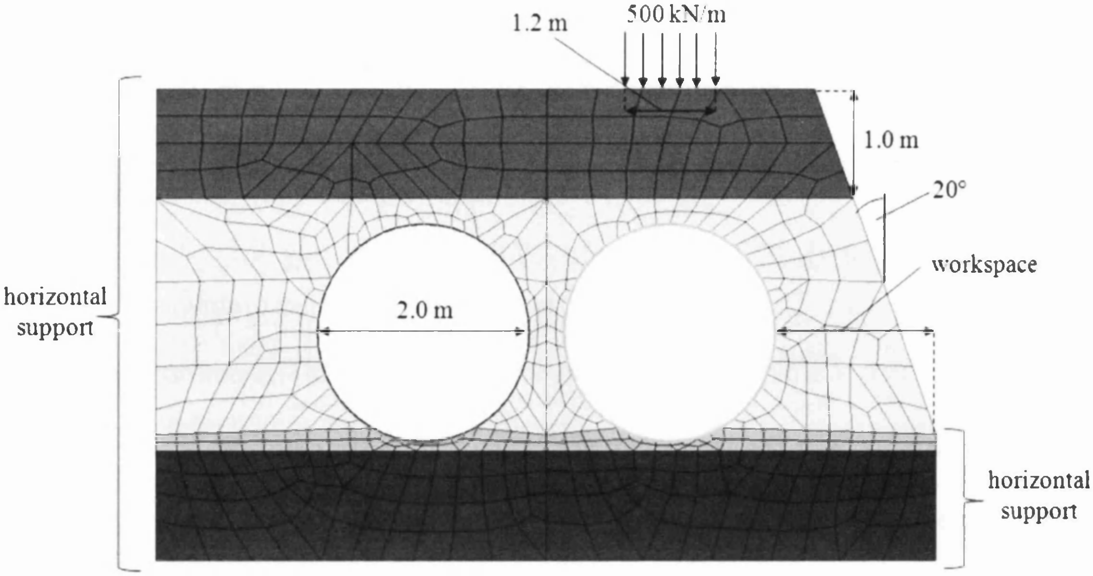


Figure 5.78: Problem definition for the example investigating the importance of the side support. Two pipes with a diameter of 2.0m are placed in a trench with a vertical fully supported left side and a by 20°inclined unsupported right side. The primary backfill height ends 300mm above the pipe crown. The final backfill is 1.0m. A load of 500kN/m is distributed over a length of 1.2m above the right pipe.

The workspaces investigated are 0.45, 0.5, 1.0, 1.5, 2.0, 3.0 and 4.0m. The geometry of the trench for 0.45, 1.5 and 4.0m are illustrated in Figure 5.79, Figure 5.80 and Figure 5.81, respectively.

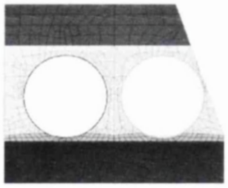


Figure 5.79: 0.45m of workspace

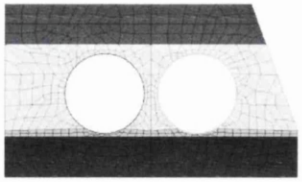


Figure 5.80: 1.5m of workspace.

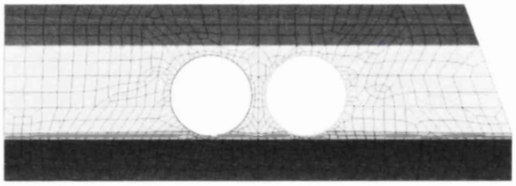


Figure 5.81: 4.0m of workspace.

The maximum pipe deflection for the right one of the two pipes is plotted against the value for the workspace in Figure 5.82. Here, the importance of the workspace is clearly observable. For the first two examples, effectively without side support very large deflections of 10% and 9% occur. The deflection drops sharply at a workspace of 1.0m to around 5.5% and converges towards 5% for workspaces higher than that.

This example shows the importance of sufficient side support of pipes in controlling pipe displacement. In the extreme case, high deflections can lead to pipe buckling and a failure of the installation. This case also stresses the importance of a thorough soil compaction of the soil surrounding the pipe, since a poorly compacted soil around the pipe does not provide a good side support and can lead to a similar increase in pipe deflections. In order to illustrate this further, three variations regarding the primary backfill of the example with 3.0m of workspace are discussed. In the original example, granular material of fair compaction has been used for the primary backfill. The three variations use a granular material of good compaction as well as a cohesive material of fair and good compaction, respectively. The resulting maximum pipe displacements are also shown in Figure 5.82. As expected, the pipe deflections are reduced further for the well compacted granular material and significantly increased for the two cohesive materials.

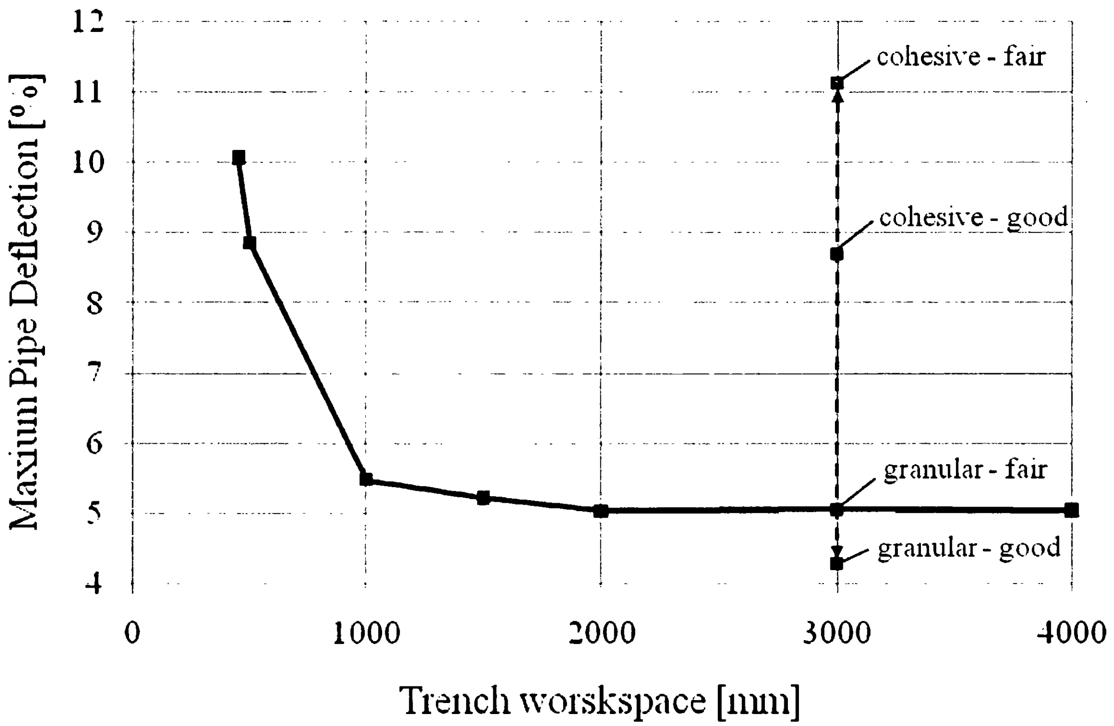


Figure 5.82: The maximum pipe deflection for the right pipe plotted against the workspace value.

In Figure 5.83 the total displacements are plotted for the pipe nodes along the circumference of the right pipe. It can be seen that for the two smallest workspaces, i.e. for the lowest side support, an increase in pipe deflection occurs near an angle of 170° which corresponds to the part of the pipe that is closest to the right trench wall. This displacement is mainly in horizontal direction. As discussed in Section 5.3.5 the vertical displacement is governed to some extent by the horizontal displacement. In this sense the vertical displacement of the pipe crown (at 90°) is reduced with increasing workspaces. For workspaces higher than 2.0m virtually no difference in the maximum

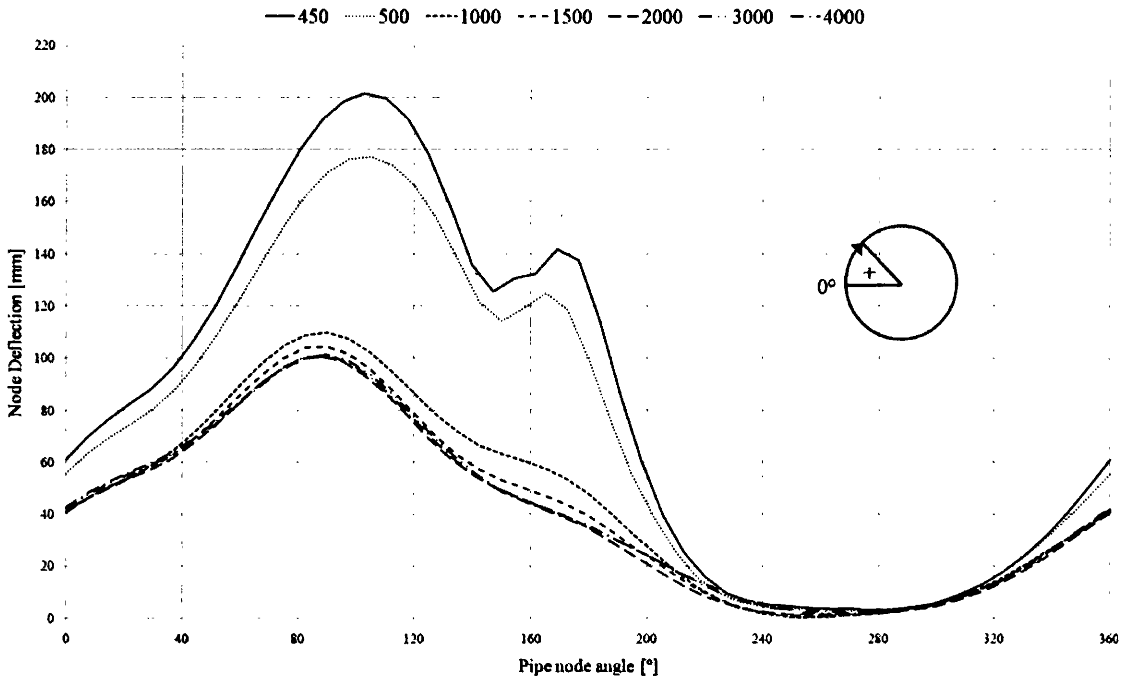


Figure 5.83: The total displacements plotted for the pipe nodes along the circumference of the right pipe.

pipe deflection occurs, i.e. the vertical deflection is not reduced any further with increasing workspace value.

5.6.3.2. Investigation of the Importance of Pipe Cover

The installation guidelines discussed in Section 5.3.5 recommend trafficking the soil above the pipe crown until a cover of at least 800mm has been reached. This is due to the fact that the arching effect, i.e. the load carrying capabilities of the surrounding soil, requires a certain amount of soil to work. That means if this is not the case, too much of the load would be carried directly by the pipe which would, due to its flexibility, exhibit high deformations. At the same time an increase of the pipe cover means an increase in the pipe loading, which leads to higher pipe deformation. Nevertheless, as stated in Section 5.3.2, the study 'Design of Buried Thermoplastic Pipes' found that the relative contribution of the pipe cover to the pipe deflection is only 15%.

The influence of the pipe cover on pipe deflections due to both, external and soil loading will be investigated by means of the following example. Here, a trench with vertical walls containing three pipes with a diameter of 1.0m will be analyzed while varying the pipe cover. Due to symmetry of the conditions only half of the trench will be simulated. The general outline of the problem is illustrated in Figure 5.84.

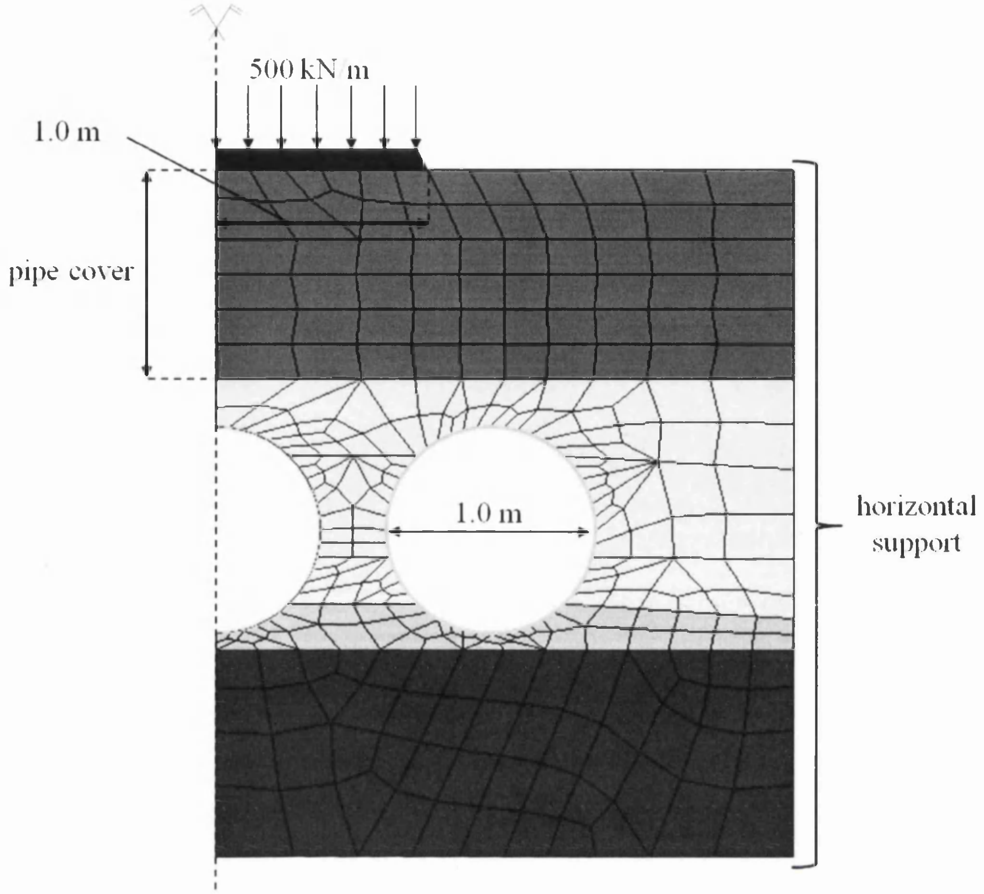


Figure 5.84: Problem definition for the example investigating the influence of pipe cover. Three pipes with a diameter of 1.0m are placed in a trench with vertical walls. The primary backfill height ends 300mm above the pipe crown. The final backfill varied. A load of 1000kN/m is distributed over a 2.0m wide concrete slab located over the middle pipe. Due to symmetry, only half of the trench is modeled.

A concrete slab is located above the middle pipe with a total width of 2.0m and loaded with a distributed load of 1000kN/m. Note that in Figure 5.84 only half of the loading is displayed. The reason that the unit of the loading is given in 'kN/m' is due to the fact that only a 2D slice of the trench is modeled.

The chosen material models, as well as the material properties or the canned soil type are listed in Table 5.15.

The problem will be analyzed for eight different pipe covers: 0.2, 0.5, 0.75, 1.0, 2.0, 4.0, 6.0 and 8.0m. The models for a pipe cover of 0.2 and 1.0m are illustrated in Figure 5.85 and Figure 5.86, respectively.

The maximum total pipe deflection for each of the eight configurations is depicted in Figure 5.87. It can be seen that the pipe deflection decreases with increasing pipe cover up to a value of 4.0m of cover. That is to say, the loading on the concrete slab is more

Table 5.15: The material models, as well as the material properties or the canned soil type for the example investigating the influence of pipe cover.

	Material model	**Canned soil type**	**Compaction**	**Young's modulus [MPa]**	**Poisson's ratio [-]**	**density [kg/m^3]**
Insitu soil	*Linear elastic isotropic*	-	-	*20.0*	*0.3*	*0.0*
Bedding soil	*Overburden dependent*	*granular*	*good*	-	-	*1922.0*
Primary backfill	*Overburden dependent*	*granular*	*good*	-	-	*1922.0*
Secondary backfill	-	-	-	-	-	-
Final backfill	*Overburden dependent*	*mixed*	*good*	-	-	*1922.0*
Concrete slab	*Linear elastic isotropic*	-	-	*30000.0*	*0.2*	*2400.0*

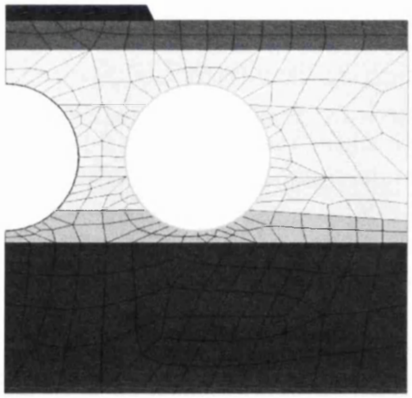


Figure 5.85: Model for 0.2m of pipe cover.

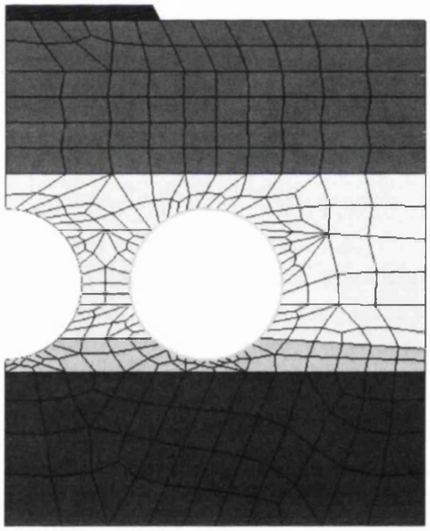


Figure 5.86: Model for 1.0m of pipe cover.

and more carried by the surrounding soil or in other words the 'arching' effect becomes more and more effective. For pipe covers of more than 4.0m, the pipe deflection starts to increase. Naturally, the loading due to the soil increases which increases the loading on the pipe but due to the shear resistance of the soil this increase is fairly slow, i.e. for an additional pipe cover of 2.0m the pipe deflection increases by approximately 0.3%.

In order to illustrate the interaction of the various parameters one of the scenarios is rerun with a varied pipe distance. The previously used pipe distance of 300mm (as recommended in Section 5.3.5) is now changed to 50 and 550mm. As discussed before,

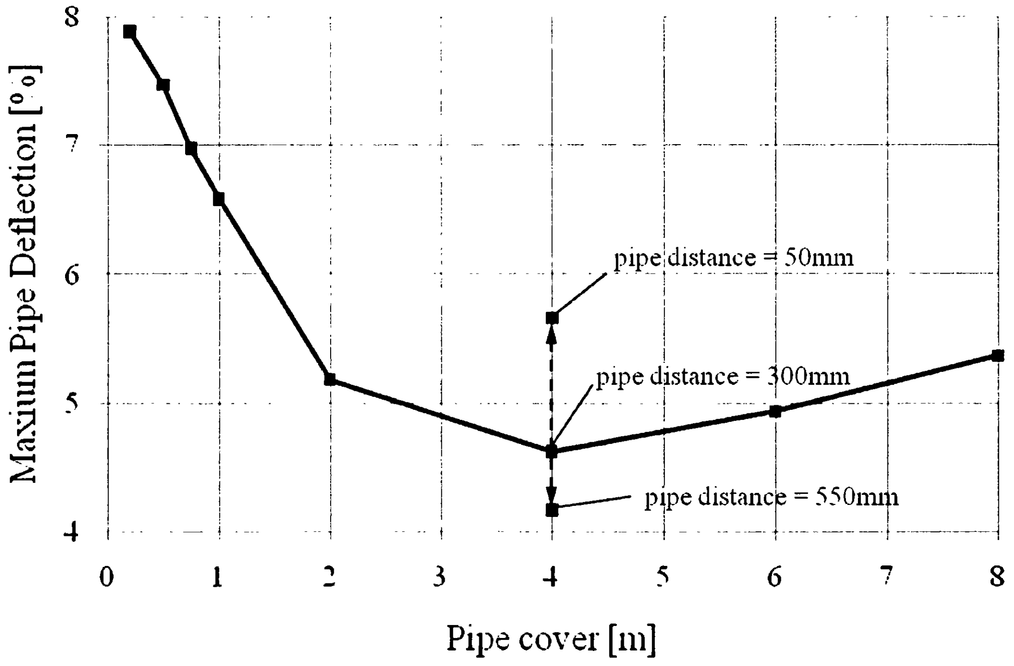


Figure 5.87: The maximum total pipe deflection for pipe covers of 0.2, 0.5, 0.75, 1.0, 2.0, 4.0, 6.0 and 8.0m for the example investigating the influence of pipe cover. Additionally, the example with 4.0m pipe cover has been rerun with a pipe distance of 50 and 550mm.

the side support is crucial for the arching effect. That means without sufficient side support the pipe deflects more since less load is carried by the surrounding soil. Obviously, for an increased pipe distance, the side support is improved and the pipe displacements are decreased. The two additional results are also shown in Figure 5.87.

In Figure 5.88 the total displacements are plotted for the pipe nodes along the circumference of the middle pipe. It can be seen that the relative difference in pipe deflection is highest at the crown of the pipe while the bottom of the pipe deflects to similar amounts. The qualitative shape of the curves is identical for all pipe cover configurations.

This example shows the importance of sufficient pipe cover when external loads are applied to the trench surface. For increasing pipe cover the maximum pipe deflection reduces at a very high rate. Once a certain pipe cover has been reached, the influence of the increased soil loading becomes more dominant than the external loading and the pipe deflection starts to increase. This increase is less steep than the decrease for small pipe covers.

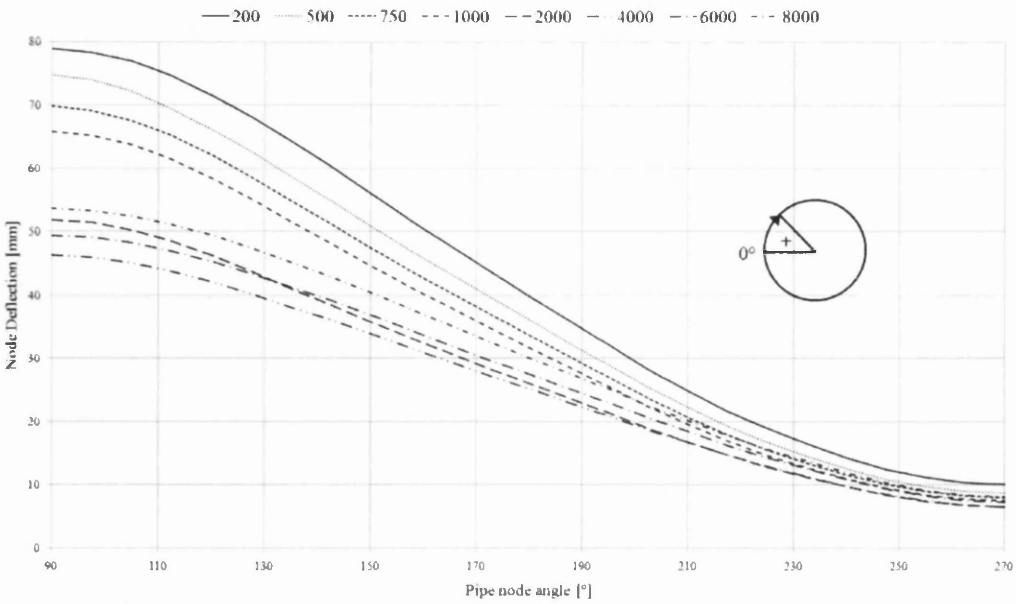


Figure 5.88: The total displacements plotted for the pipe nodes along the circumference of the middle pipe for the example investigating the influence of pipe cover.

5.6.3.3. Industry Example

The study 'Design of Buried Thermoplastic Pipes' found that the most important factor in controlling pipe deflection is the quality of the installation (The European Plastics Pipe and Fitting Association, 1999), i.e. a pipe surrounded by thoroughly compacted soil will deflect significantly less than one that has been poorly installed. In order to investigate this relationship and to reproduce actual measurements, a real life example is considered now. Here, 15 pipes with a diameter of 2.1m and a leg length of 114m were installed. The construction site is shown in Figure 5.89.

Figure 5.89: Construction site for the installation of 15 pipes prior to burying. The spacing between them is 0.5m, their diameter is 2.1m and the overall length of each leg is 114m.

The distance between the pipes is 0.5m and the cover above the top end varies from 0.89m to 2.33m. The dimensions of the end conditions can be found in Figure 5.90. The primary backfill, i.e. the installation soil up to 0.3m above the pipe is a granular and the remaining backfill is a cohesive soil. Additional loads are induced by heavy machinery used for installation. According to (British Standards Institution, 1998) this loading can be accounted for by a load of 150kN/m.

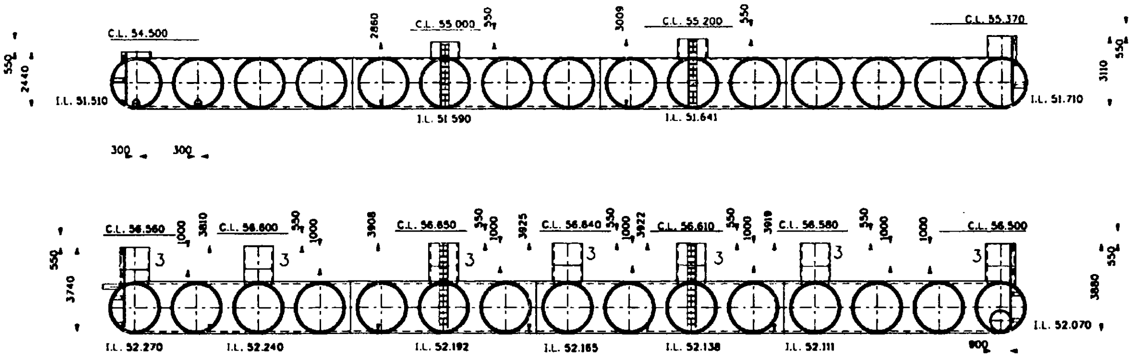


Figure 5.90: CAD drawing of end conditions. The cover above the pipes varies from 0.89m to 2.33 m.

In the course of the installation the vertical displacement of every pipe has been measured at nine equidistant locations along their length. The maximum recorded displacement is 3.8% of the pipe diameter and the lowest is -0.7%. This means in some areas the pipe crown actually moved upwards. This occurs when the compaction of the soil on the sides of the pipe is performed so thoroughly that due to the soil pressure, the sides of the pipes move inwards, which effectively forces the pipe crown to move upwards. The original displacements recorded on the site for each of the 15 legs are shown in Figure 5.91.

The objective of this application is to simulate the presented installation situation and predict the maximum pipe displacements. To this end simulations for several different scenarios are performed. The altered conditions are the height of cover, the location of the pipe within the trench, the quality of the soil compaction and the presence of additional loading. To estimate the maximum displacements, additional loading due to installation machinery will be applied with fair compaction scenarios. This loading will be omitted for those scenarios having good compaction. To predict the response of a pipe on the inside of the installation a vertical wall model is used with two more pipes on either side. The problem definition is shown in Figure 5.92. Here the model for the simulation of an interior pipe for a pipe cover of 2.33m is shown.

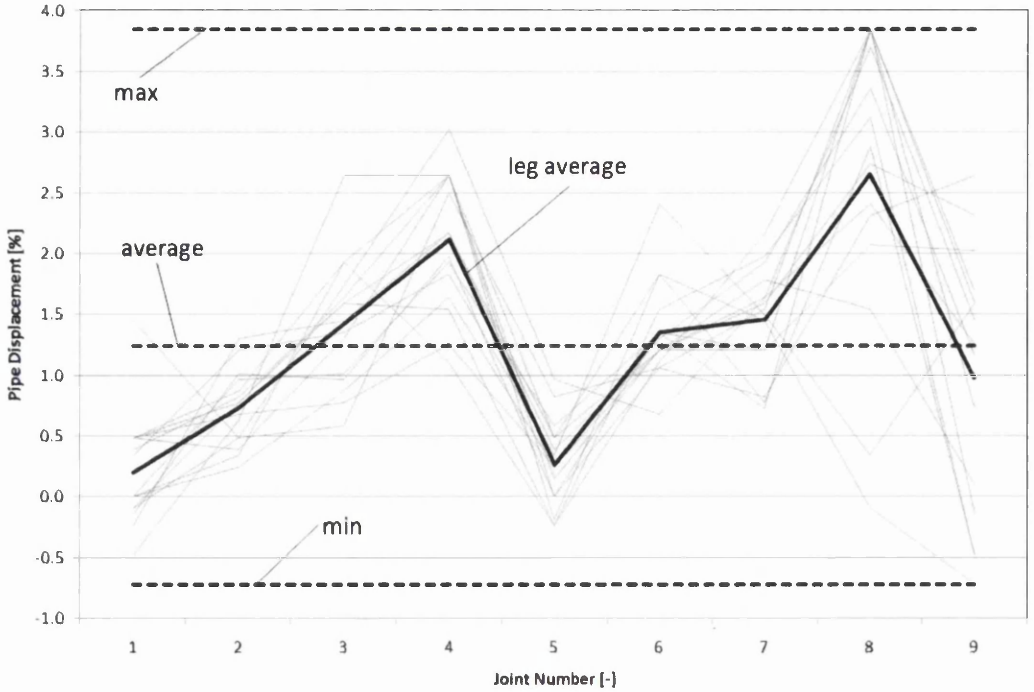


Figure 5.91: Recorded pipe deflections in percent along their lengths. The maximum deflection is 3.8%, the minimum is -0.7% and the average is 1.2%

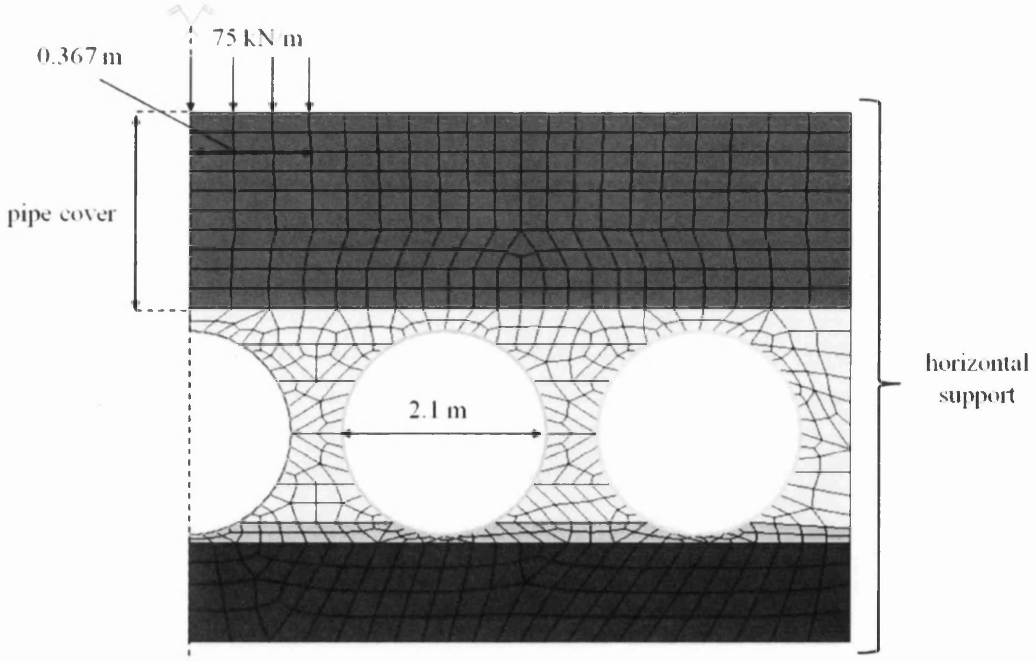


Figure 5.92: Problem definition for the industry example investigating Here the model for the simulation of an interior pipe for a pipe cover of 2.33m is shown. The loading is omitted for a good compaction scenario.

For covers of 0.89m and 2.33m the models are illustrated in Figure 5.93 and Figure 5.95, respectively. Pipes on the outside of the installation require the insitu side support

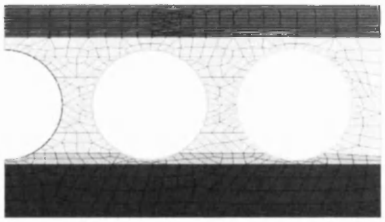


Figure 5.93: Illustration of FE model used to simulate a pipe in the interior with a pipe cover of 0.89m.

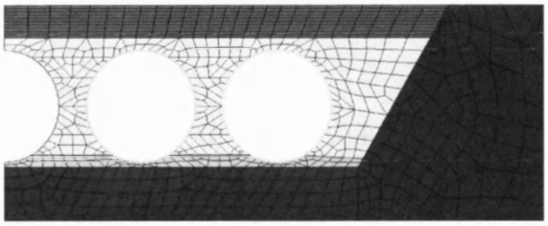


Figure 5.94: Illustration of FE model used to simulate a pipe at the exterior with a pipe cover of 0.89m.

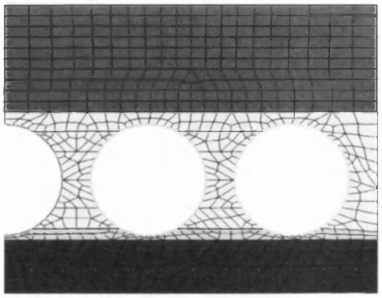


Figure 5.95: Illustration of FE model used to simulate a pipe in the interior with a pipe cover of 2.33m.

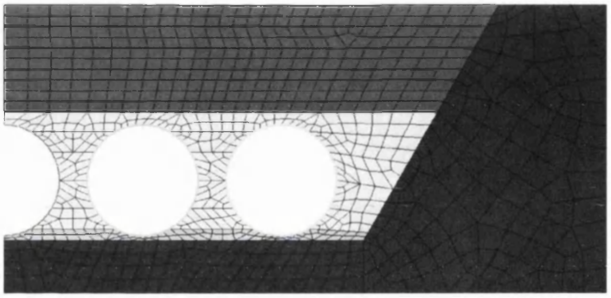


Figure 5.96: Illustration of FE model used to simulate a pipe at the exterior with a pipe cover of 2.33m.

condition. The corresponding models are shown in Figure 5.94 and Figure 5.96, respectively. The obtained displacements for the different scenarios are listed in Table 5.16.

Table 5.16: Results from simulations for different locations, installation qualities and cover heights. The maximum deflection is 3.6% and the minimum is 0.9%.

height of cover [m]	**0.89**				**2.33**			
location	*exterior*		*interior*		*exterior*		*interior*	
compaction	*good*	*fair*	*good*	*fair*	*good*	*fair*	*good*	*fair*
additional loading	*no*	*yes*	*no*	*yes*	*no*	*yes*	*no*	*yes*
max. vertical deflection [%]	*1.1*	*3.3*	*1.1*	*3.3*	*1.7*	*2.8*	*1.6*	*2.8*

The maximum displacement corresponds to an internal pipe with fair compaction quality, a cover height of 0.89m and the additional external loading. This deflection is 3.3% and hence, given the approximations in the model, coincides sufficiently well with the observed data. As to be expected the lowest deflection is found for a good quality compaction with 0.89m of cover for both an internal and external pipe without any external loads. The upwards deflection as found in the real observation could not be captured, since the compaction loading is not modeled. It can also be seen that almost no difference exists between an interior and an exterior pipe.

5.7. Concluding Remarks

A Java based GUI has been developed that uses in-house Fortran and public domain software to provide a hands-on decision guidance for many occurring design situations for the studied structured wall HDPE pipes. It allows for the automated simulation of the pipes in the ringstiffness test according to the standard BS EN 1446: 1996 (British Standards Institution, 1996) as well as pipes in an installation process. These simulations require minimal user interaction and are easy to set up. Furthermore, an optimization feature has been incorporated that allows for the generation of optimum cross-sections in the sense of minimal material expenditure. Here a simplified model using only one box-section and a horizontal spring support is used which significantly reduces the simulation time and makes the optimization computationally feasible in a manufacturing environment. All the underlying computationally expensive tasks are solved by either in-house, open-source or public-domain programs. This means that no licensing fees have to be paid to use the software.

Material tests in the form of DMTA and tension tests have been conducted to investigate the influence of temperature, strain and strain rate on the material behaviour. Furthermore, a guideline has been developed that allows for the extraction of a secant modulus from the obtained data. This secant modulus can be used directly in the FE simulations. Material data has been obtained for the material used by two companies: KWH Pipe Ltd. and Asset International Ltd.

For the accurate capture of the profile geometry the possibility has been developed to process laser-scan data for the creation of shell element representation. This process can also be used for the import and automatic processing of profiles created in DXF format. A less accurate yet more efficient option for the creation of rectangular box sections is implemented as well.

Application of the various features and the comparison of the obtained results to real life examples showed both the usability and validity. Results from numerous conducted ringstiffness tests were available. The simulation of the experiments conducted by KWH showed that when tests are performed carefully an accuracy of the simulation results of approximately 10% can be expected. The maximum error found was 13%.

Given the assumptions made and the sources for variability of test results this accuracy can be viewed as satisfactory. The tests performed by Asset however, where especially the pipe temperature control is not as thorough as with KWH showed a lower accuracy. In one case the difference amounted to 23%. In another case the predicted ringstiffness was 48% higher than the one in the experiment but an experiment conducted with the same pipe and profile size suggested that the experiment result was erroneous.

The application of the profile optimization tool to two of the previously analyzed profiles showed that a material saving of more than 20% is possible.

The usability and validity of the pipe installation simulation feature has been shown by means of three examples. The first two investigated the influence of the pipe's side support and pipe cover on pipe deflection. Here, two main principles of the load bearing capabilities of flexible pipes could be validated. The last of the three examples is based on data from a real life pipe installation. Displacement measurements taken during the installation of a system of 15 pipes were sought to be reproduced. Eight different simulations were run varying the height of cover, the location of the pipe within the trench, the quality of the soil compaction and the existence of additional external loading. The recorded maximum displacement was 3.8%. The simulations yielded a maximum of 3.3%. Given the approximations in the model, this can be considered as sufficiently accurate.

Overall it was shown that an efficient and easy to use software tool has been developed which addresses real life design problems in a manufacturing environment. More importantly this has been achieved by coupling public domain and in-house code which effectively led to a software that requires no licensing fees.

6. A Decomposed Optimization Framework for the Design of Carbon-fibre Competition Bicycle Frames

Summary: This chapter presents a framework for the optimum design of carbon-fibre mountain bike frames that has been developed. An extensively parameterized simulation model has been created that allows for varying tube shapes, paths and laminate ply thicknesses as well as joint locations. The model creation has been automated by using a Fortran code which writes command files for the control of HyperMesh and for the interpretation and processing of the created FE mesh. For improved efficiency a decomposition approach has been employed that decomposes the original optimization problem into a size optimization sub problem and a shape optimization top level problem. The former is solved by the built-in optimization tool in OptiStruct and the latter by means of surrogate based optimization where each experiment in the DoE is a full size optimization. The applied framework for surrogate based optimization contains the developed scheme for the creation of OLHs in constrained design spaces, an implementation of a Kriging approximation and a continuous GA. To reduce the dimensionality of the top level problem the paths of the seat and chain stays are determined in an optimization step prior to the main optimization. Application of the developed framework to two examples shows that optimum bicycle frame's with very good characteristics can be designed in an efficient way.

6.1. Introduction

An ideal bicycle frame has to satisfy general characteristics which are applicable to most fields of use. These are firstly a low weight and a high lateral stiffness. Both characteristics contribute to its energy efficiency. While the reason for the former is obvious, the latter guarantees that a maximum amount of the rider's force applied to the pedal is transformed into momentum. Torsional stiffness of the head tube offers a good handling feel and avoids dangerous vibrations at high speeds. In spite of a high lateral stiffness, for the sake of comfort, the frame should absorb shock and vibration to a high degree (Calfee & Kelly, 2002). The ability to withstand high impact forces and reliability against fatigue failure are crucial for safety and legal liabilities (Nelson, 2003). To fit a rider's physical requirements a bicycle frame is ideally tailor made or offered in a broad range of sizes (Calfee & Kelly, 2002).

The success of carbon-fibre in modern bicycle frames is due to its material behaviour that meets the above requirements to a significantly higher degree than classic materials. Its high stiffness to weight and strength to weight ratio, its shock absorbing capabilities, the reliability against fatigue failure and corrosion as well as its variability in forming complex shapes make it the ideal material. The main disadvantages are the high costs for production and the smaller experience with the material itself (Calfee & Kelly, 2002). Especially the high costs make the target group either professionals or ambitious amateurs. In either case a very high standard and excellent characteristics are required which makes the production of optimal bicycle frames a profitable business.

(Nelson, 2003), (Schroeder & Callahan, 2005) and (Yen, n.d.) give an insight in the design and manufacturing processes of large companies producing carbon-fibre bicycle frames but little information about optimization procedures is revealed. The use of the FE method in designing composite bicycle frames is discussed by (Lessard et al., 1995) but no information on optimization is given. The only work on frame optimization has been done by (Finn, 2009). It covers the thickness optimization of a racing bike frame.

The purpose of this piece of work is the development of a novel bespoke optimization software framework for the design of optimal bicycle frames. This framework is sought to be capable of taking into account rider specific demands. The targeted frames are intended for competition cross-country mountain bikes which have comparable requirements to any other bicycle. Only the loading and the targeted characteristics vary. These characteristics can be measured by real life longevity tests, for instance by means of the EFBe standard (EFBE Prüftechnik, n.d.) which is based on the European Standard EN 14766:2005 (European Standard, 2005). It comprises two stiffness, two static high-load failure and three fatigue tests. Rider specific demands means, given personal desires and physical requirements of a customer, a tailor-made optimal frame can be designed almost instantly. This requires a thorough and extensive parameterization that will be realized by means of an in-house code due to limitations of existing programs. This in-house code is coupled with commercial software to maximize efficiency.

The present work is structured as follows: Firstly, the EFBe tests are presented followed by the FE model simulating these tests and the automated procedure that creates them. Finally the developed optimization procedure and the obtained results of two considered examples are discussed.

6.2. EFBe Tests

The EFBe standard (EFBE Prüftechnik, n.d.) is based on the European Standard EN 14766:2005 (European Standard, 2005) and consists of two stiffness, two static high-load failure and three fatigue tests. The various tests will be presented briefly in the following.

6.2.1. Rigidity Tests

Two rigidity tests are performed within the EFBe standard. Their results give an insight into the stiffness of key parts of the frame. These are the rotational stiffnesses of the head tube (Track-holding rigidity test) and of the bottom bracket (Out-of-Saddle rigidity test).

6.2.1.1. Track-Holding Rigidity Test (head tube)

This test reveals how accurate the frame feels during use. If, during design, too much attention was paid to the weight and too little to the stiffness the frame could give the rider a 'soft' feel and irritating vibrations might occur, ultimately reducing the rider's confidence and performance.

In the test, the frame is subjected to a force normal to the frame's plane acting at the end of a 450mm long extension of the head tube, i.e. near the front wheel contact point. The rear axle is fixed except for rotations about the axle's axis. The middle of the head tube is prevented from moving out of plane. The rigidity is defined as the ratio between the moment acting at the middle of the head tube measured in Nm and the corresponding rotation at that point, measured in degrees. The loading is illustrated in Figure 6.1 and the qualitative deformation in Figure 6.2.

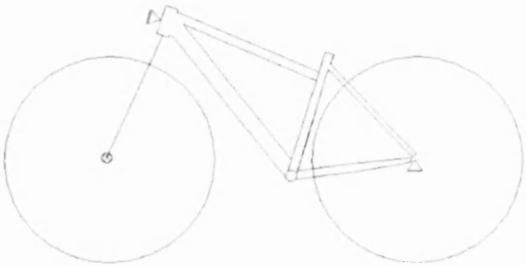


Figure 6.1: Loading and boundary conditions for Track-holding rigidity test.

Figure 6.2: Deformed shape for the Track-holding rigidity test.

6.2.1.2. Out-of-Saddle Rigidity Test (bottom bracket)

This test measures how energy efficiently a frame can be ridden, i.e. how well the rider's energy input is converted into driving power. During the design process the proportional relationship between weight and stiffness has to be taken into account since both characteristics contribute to the energy-efficiency inversely proportional. Thus a good compromise between the two has to be found.

The tip of a 150mm long horizontal extension mounted at the center of the bottom bracket is subjected to a force, which is inclined at 26° with respect to the frame plane. The rear is supported via a mount ending with a ball joint, which simulates the contact between tyre and ground. The head tube is extended with a dummy fork 450 mm of length which is simply supported. The rigidity is defined as the ratio between the force in N and the displacement in mm measured in the direction of the force. The loading is illustrated in Figure 6.3 and the qualitative deformation in Figure 6.4.

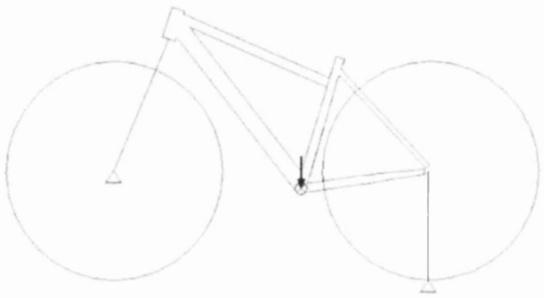


Figure 6.3: Loading and boundary conditions for Out-of-saddle rigidity test.

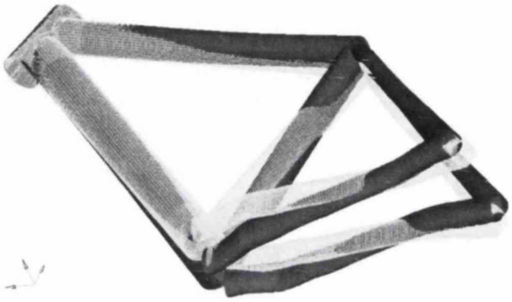


Figure 6.4: Deformed shape for the Out-of-saddle rigidity test.

6.2.2. Excessive Load Tests

The excessive load tests reveal whether the frame can withstand short-term excessive loads without brittle fracturing. Two tests of this type have to be withstood.

6.2.2.1. Drop Load Test (bottom bracket)

In this test the bottom bracket is subjected to a vertical force of 6000N. The rear is supported via a mount ending with a ball joint, simulating the contact between tyre and ground. The head tube is extended with a dummy fork, 450 mm of length, and its tip is simply supported. The loading is illustrated in Figure 6.5 and the qualitative deformation in Figure 6.6.

6.2.2.2. Pedal Load Test (bottom bracket)

This test is identical to the Out-of-saddle rigidity test (see Section 6.2.1.2) except for the magnitude of the force which in this case is 2850N.

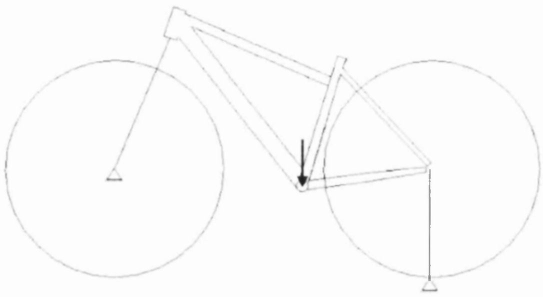


Figure 6.5: Loading and boundary conditions for Drop load test.

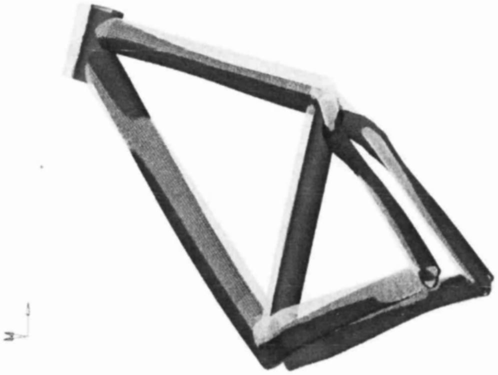


Figure 6.6: Deformed shape for the Drop load test.

6.2.3. Fatigue Tests

The fatigue tests help to measure the risk of fatigue fractures that often only appear after years of use and can lead to sudden failure and potentially serious accidents. The tests measure whether a bike component can withstand cyclic loads without showing any signs of damage. Since the theory for the FE analyses is linear static, the real dynamic forces are approximated by static ones. The dynamic effect is accounted for by dividing the allowable stresses by a safety factor specific to the nature of each test. This safety factor is discussed in Section 6.3.5.

6.2.3.1. Out-of-Saddle Test (bottom bracket)

In this test an extension inclined at 45° downwards with respect to the horizontal plane having a projected horizontal length of 150mm is mounted at the bottom bracket. Its tip is subjected to a force of 1300N, inclined at 26° with respect to the frame's plane. The force is applied on either side, being activated alternately. The rear is supported via a mount with a ball joint at the end, simulating the contact between the tyre and the ground. The head tube is extended with a dummy fork, 450 mm of length, which is simply supported. The required number of cycles the frame has to withstand is 100,000. The loading is illustrated in Figure 6.7 and the qualitative deformation in Figure 6.8.

6.2.3.2. Rider Load Test (seat tube)

In this test the frame is loaded with a vertical force of 1400N acting at the end of an extension of the seat tube ending at 250mm in the tube direction and 70mm horizontally towards the rear. The rear axle is fixed except for rotations about the axle's axis. The

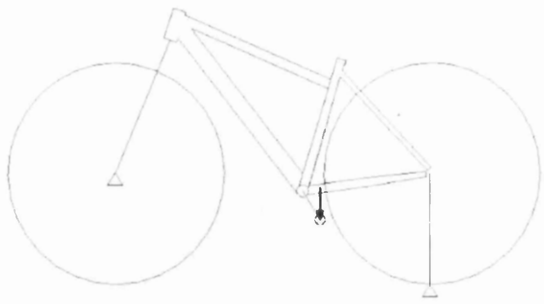


Figure 6.7: Loading and boundary conditions for Out-of-saddle test.

Figure 6.8: Deformed shape for the Out-of-saddle test.

head tube is extended with a dummy fork, 450mm of length. The tip of the extension is prevented from moving out of the frame's plane. The number of cycles is 100,000. The loading is illustrated in Figure 6.9 and the qualitative deformation in Figure 6.10.

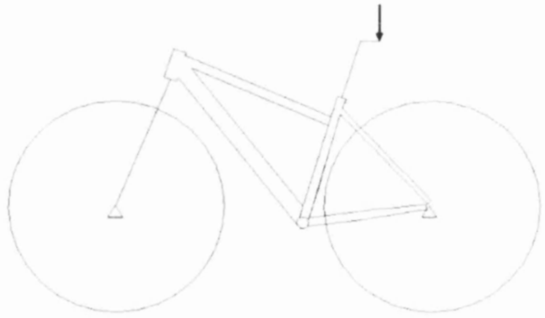


Figure 6.9: Loading and boundary conditions for Rider load test.

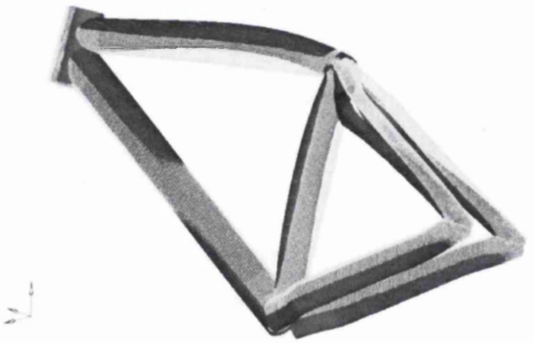


Figure 6.10: Deformed shape for the Rider load test.

6.2.3.3. Jump Test (head tube)

In this test the rear axle is fixed except for rotations about the axle's axis. The bottom bracket is secured from vertical movements. The head tube is extended with a dummy fork 450mm of length. The tip of the extension is loaded with a force perpendicular to it and parallel to the frame's plane. The force alternates between +450N and -900N. The required number of cycles is 50,000. The loading is illustrated in Figure 6.11 and the qualitative deformation in Figure 6.12.

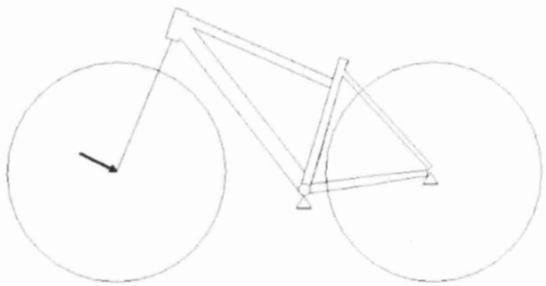


Figure 6.11: Loading and boundary conditions for Jump load test.

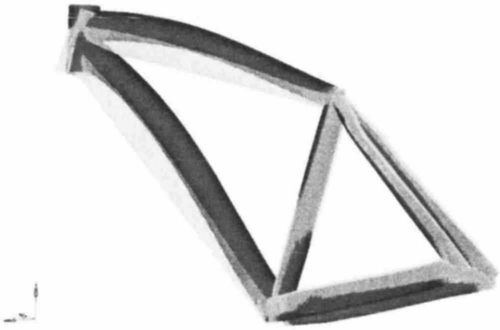


Figure 6.12: Deformed shape for the Jump load test.

6.3. Finite Element Model and Automated Model Generation

6.3.1. Composite Lay-Up

The composite is made up of five plies, an outer bi-directional ply oriented at 45° and four inner uni-directional layers at -45°, 0°, 45° and 90°. The relevant material properties are listed in Table 6.1. Theoretical background on Prepreg materials can be found in (Brent Strong, n.d.).

Table 6.1: Mechanical properties for bi-directional and uni-directional plies

		bi-directional	***uni-directional***
Longitudinal Tensile Modulus	*[MPa]*	*61800*	*130330*
Longitudinal Tensile Strength	*[MPa]*	*587.1*	*1433.6*
Longitudinal Compressive Strength	*[MPa]*	*454.2*	*984.2*
Transverse Tensile Modulus	*[MPa]*	*61800*	*7220*
Transverse Tensile Strength	*[MPa]*	*587.1*	*32.5*
Transverse Compressive Strength	*[MPa]*	*454.2*	*108.3*
Inter Laminar Shear Strength	*[MPa]*	*53.8*	*71.9*
In-plane Shear Modulus	*[MPa]*	*4081*	*4230*
In-plane Shear Strength	*[MPa]*	*62.4*	*71.9*
Poisson's ratio (Longitudinal Strain)	*[-]*	*0.037*	*0.337*
Density	*[kg/m^3]*	*1459*	*1502*
Ply Thickness	*[mm]*	*0.22*	*0.3*

6.3.2. Finite Element Model

The geometry model is made up of seven tubes (top tube, down tube, seat tube, two seat stays and two chain stays as shown in Figure 6.13). In the locations of the joints the tubes need to be merged for the frame to form a continuous structure. Merging several tubes with each other that meet at various angles and that have different diameters leads to complex geometries which are too difficult to handle within the desired automated process. However, an automated geometry and FE model generation is a prerequisite for the optimization that is aimed for. Thus, another way of joining the tubes had to be found. Here it has been realized via rigid cylinders that envelope the joints and transfer the loading from one tube to the next, as illustrated in Figure 6.14.

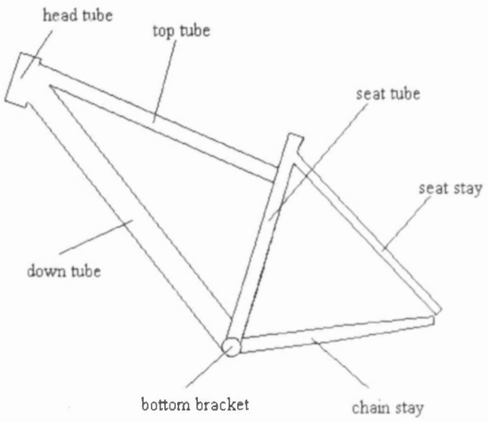


Figure 6.13: Frame member denotation.

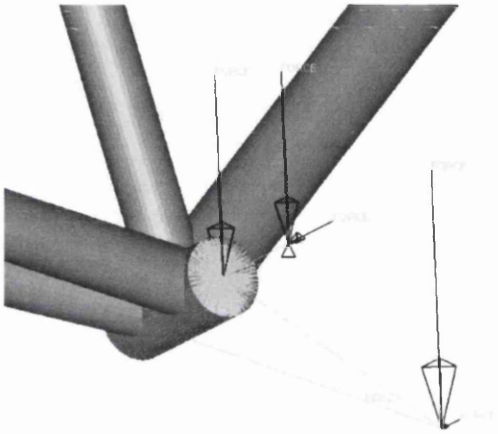


Figure 6.14: Cylinders enveloping joints and rigid members for load input.

The use of these cylinders results in only one intersection path between a tube and the cylinder. Naturally, this leads to an overestimation of the rigidities and neglects the frame's vulnerability for failure in the joints during testing. The joints are designed in a separate study which is briefly presented in Section 6.5.4.

Each cylinder has a circular rigid link at its two ends and its middle. One of the links can be seen in Figure 6.14. They are used to uniformly transfer forces and boundary conditions onto the cylinder. Additional rigid links are used to simulate the test conditions, for instance the inclined bottom bracket extension in the Out-of-saddle fatigue test (see Figure 6.14).

The elements used to mesh the created geometry are linear triangular and quadrilateral shell elements.

6.3.3. Model Parameterization

The input parameters governing the geometry of the frame can be divided into two main categories: Size and shape parameters. The shape parameters are subdivided into global and local shape parameters. The categorization of the input parameters is illustrated in Figure 6.15 and will be discussed in the following.

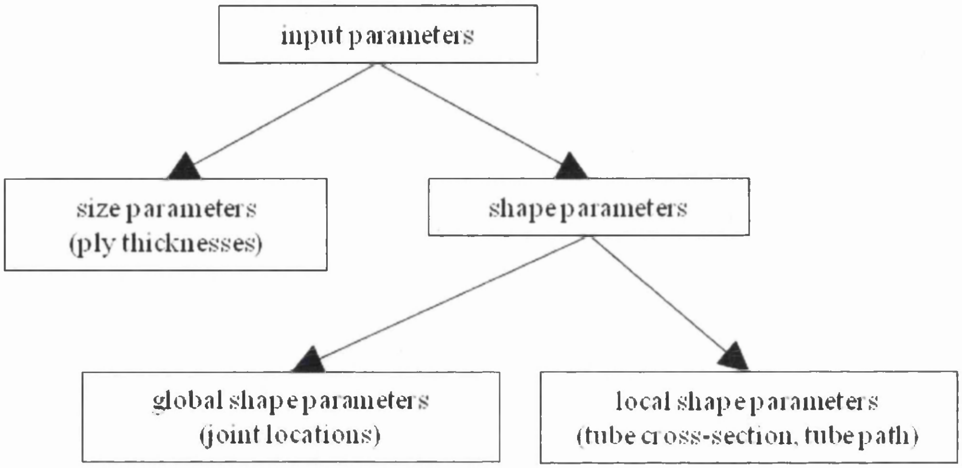


Figure 6.15: Optimization input parameter categorization.

6.3.3.1. Size Parameters

This first category contains the size parameters. These are the thicknesses of the composite plies. The orientations are fixed but their appearance is controlled implicitly via the thicknesses since four orientations are offered in the laminate. For instance if no ply with a 45° orientation is required the corresponding thickness is set to zero.

6.3.3.2. Shape Parameters

The second category contains the shape parameters. They can be subdivided into parameters that have discrete values and those governed by B-splines.

The first subcategory can be denoted as global shape parameters. These are the locations of the joints in space, i.e. the rigid cylinders enveloping the joined tubes as illustrated in Figure 6.16.

The second subcategory could be described as local shape parameters. These are the shape of the tubes and their spatial path, which are both described by means of B-splines. The shape of the cross-section is governed by two splines which are laid through each tube. They describe the semi-major and semi-minor axes of an ellipse, while the path of the tube is governed by two more B-splines which govern the horizontal and vertical distance from a straight center-line. The latter is important for the creation of seat or chain stays that avoid contact with the tyre or the pedal. Basically, the local shape parameters govern the shape and the path of the tubes which connect the joints illustrated in Figure 6.16.

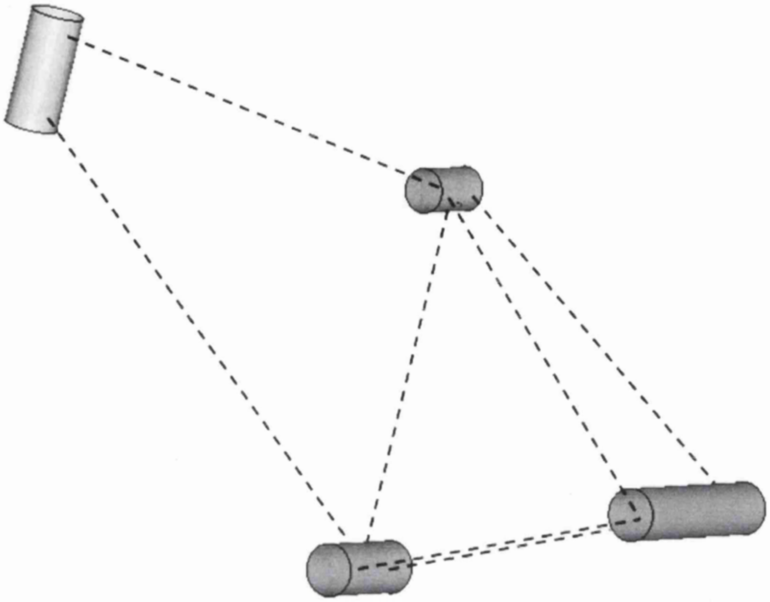


Figure 6.16: The global shape parameters determine the location of the rigid cylinders.

The tubes are created in the following manner: The global shape parameters determine the locations of the beginning and the end points of the tubes. Now, the B-splines for each tube are evaluated at a certain number of intervals n_{Eval} giving the width and the height of the ellipse that forms the cross-section as well as the horizontal and vertical deviation from a straight line. The horizontal and vertical deviation governs the location of the center-point of the ellipse. A certain number of points is then created on that ellipse. When all n_{Eval} locations of a tube have been processed, the individual nodes are connected by lines and the lines are used to create surfaces on which the FE mesh is generated in the end. With the above procedure tubes can be created that have a smoothly varying cross-section and are bent with respect to two axes. A tube, generated from the cross-sectional nodes can be seen in Figure 6.17.

In order to show the influence of a change of the control values on a tube the following example of a straight tube with a constant cross-section of width 20mm and height 40mm is considered. The number of evaluations n_{Eval} is 70. Now the middle control

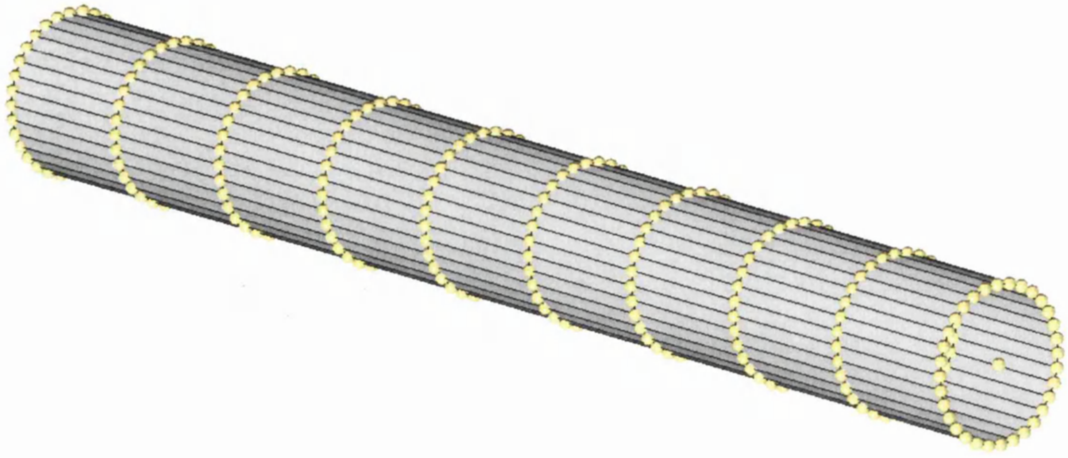


Figure 6.17: Generation of a tube: At n_{Eval} locations the B-splines are evaluated, yielding the width and height of the ellipse that forms the cross-section as well as the horizontal and vertical deviation from a straight line. A certain number of points is then created on that ellipse. Here n_{Eval}=10.

value of the B-spline governing the width and the horizontal deviation are changed. The width is set to 40mm and the horizontal deviation is set to 50mm. The resulting tubes before and after B-spline alteration are shown in Figure 6.18.

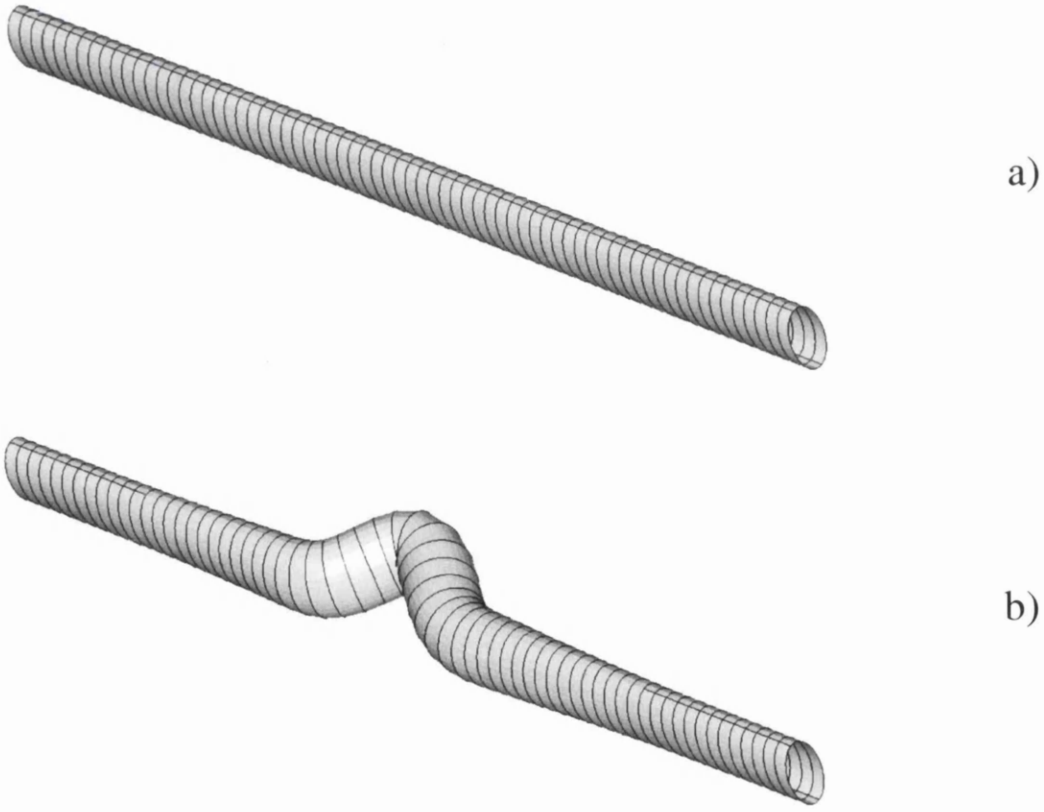


Figure 6.18: Tubes before a) and after b) changes have been made to the B-spline. Here n_{Eval}=70.

6.3.4. Automated Model Generation

Since the numerical model will be used within an optimization procedure, the models have to be created in an automated manner. The chosen approach has been implemented in accordance with Section 4.5.4. The chosen pre-processor is HyperMesh (Altair HyperWorks (a), 2009) which is used in batch mode. The tasks are conveyed via command files, which allow for the execution of the full range of actions in the program. At first a command file is written that contains commands for the creation of the geometry, the materials, the composite properties and for the mesh generation. HyperMesh is then called in batch mode, to process the command file and to output a file containing the information of the generated mesh. Next, the user developed Fortran program, which manages the procedure, reads this output file, applies the boundary conditions and writes another command file containing this information. Finally, HyperMesh is called again to process the latter and to output the final model to be solved by OptiStruct (Altair HyperWorks (b), 2009). This procedure can be implemented in the optimization procedure as a standalone FE model generator for a given set of parameters governing the geometry. The flow of the developed automated model creation scheme is displayed in Figure 6.19.

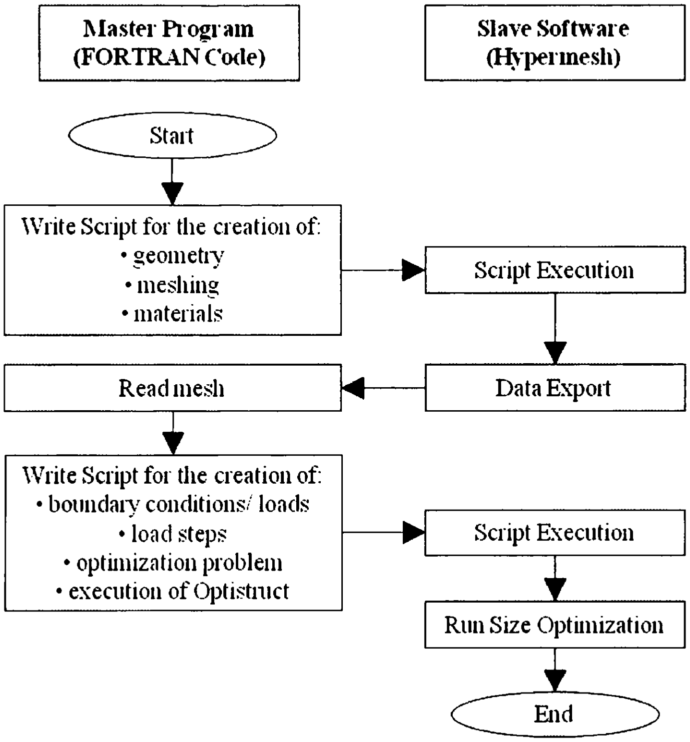


Figure 6.19: Flow of the proposed automated model generation scheme.

6.3.5. Fatigue Treatment

In order to keep the computational effort for solving the FE model feasible for the application within the developed optimization procedure, the impact on the material's strength in the fatigue context, is accounted for by solving a static problem and stating Failure Index (FI) constraints that include an additional fatigue safety factor, $SF_{Fatigue}$. An FI indicates whether or not a composite failed at a certain location. Here FIs according to Hill were chosen which are calculated according to Equation 6.1

$$FI_{Hill} = \frac{{\sigma_1}^2}{X^2} - \frac{\sigma_1 \sigma_2}{X^2} + \frac{{\sigma_2}^2}{Y^2} + \frac{{\tau_{12}}^2}{S^2}\ , \qquad 6.1$$

where X is the allowable stress in ply material direction *1*, Y the allowable stress in ply material direction *2* and S the allowable in-plane shear stress. A distinction between strength for compression and tension is made and based on the sign of the stresses. According to Equation 6.2, $SF_{Fatigue}$ is calculated as the ratio between static strength σ_{max}^{Static} and fatigue strength $\sigma_{max}^{Fatigue}$ dependent upon the test's amplitude ratio R and number of cycles n_{cycle}.

$$SF_{Fatigue} = \frac{\sigma_{max}^{Static}}{\sigma_{max}^{Fatigue}(n_{cycle}, R)} \qquad 6.2$$

(Kawai & Koizumi, 2007) provides strength ratios for zero amplitude ratios and (Michel et al., 2005) for non-zero values. n_{cycle}, R and $SF_{Fatigue}$ for the three fatigue tests are summarized in Table 6.2.

Table 6.2: Safety factors and necessary parameters for fatigue tests.

	n_{cycle}	R	$SF_{Fatigue}$
Out of Saddle test	*100,000*	*-1.0*	*3.0*
Jump Load test	*50,000*	*-0.5*	*2.2*
Rider Load test	*100,000*	*0.0*	*1.8*

For the Rider Load test the applied load ranges from no load to full load. Thus, R equals *0*. In the Jump Load test the maximum positive load is twice the negative load so that R is *-0.5* whereas in the Out of Saddle test positive and negative loads are identical which leads to an amplitude ratio of *-1.0*.

It shall be noted that the above procedure aims at taking into account the fatigue behaviour of the material in the fatigue tests. Naturally, the fatigue behaviour due to the long term use of a bicycle frame is more complex and for instance influenced by environmental factors. The degree of their impact depends on the location of the ply within the stack. Within the fatigue tests specified in the EFBe tests where the repetitive loading is simulated within a short period of time, this influence is certainly small.

Furthermore, the use of a single safety factor to account for fatigue is naturally in itself a simplified way of taking fatigue into account. For a more complex approach to fatigue treatment see for instance (Kruger & Rolfes, 2010). Naturally, such an approach would significantly increase the computational effort since repetitive simulations would be necessary which is prohibitive within the developed optimization procedure.

6.4. Optimization

6.4.1. Objective, Constraints and Design Variables

6.4.1.1. Mass versus Rigidity

One design scenario is to define the objective function as the sum of the Out-of-saddle (*OSR*) and the Track-holding-rigidity (*THR*). An increase in the frame's stiffness requires an increase in the frame's mass *M*; therefore it needs to be restricted to a maximum mass M_{Thres}.

$$\begin{aligned} &max && f = OSR + THR \\ &subject\ to && g_1 = M - M_{Thres} \leq 0 \end{aligned} \tag{6.3}$$

Alternatively, for a minimum sum of rigidities R_{Thres} the frame's mass can be minimized.

$$\begin{aligned} &min && f = M \\ &subject\ to && g_1 = R_{Thres} - (OSR + THR) \leq 0 \end{aligned} \tag{6.4}$$

6.4.1.2. Material Failure

Another constraint type assures that the material does not exceed its maximum strength by not allowing any FIs above a threshold FI_{Thres} which depends on the general safety factor SF and the fatigue safety factor $SF_{Fatigue}$. FI_{Thres} is calculated according to Equation 6.5 and $SF_{Fatigue}$ according to Equation 6.2.

$$FI_{Thres} = \left(SF \cdot SF_{Fatigue}\right)^{-1} \tag{6.5}$$

6.4.1.3. Feasible Stay Paths

Further constraints guarantee that the rear tyre, the rear disc brake, the chain ring and the pedals are not obstructed by the chain or seat stays. This is realized, by penalizing nodes intruding the defined obstruction zones. The nodes used for the creation of the tube geometries (see Section 6.3.3.2) are exploited to construct constraints that determine to which degree obstruction occurs. For each evaluation an obstruction index OI is calculated according to

$$OI = \begin{cases} -\max L_i^{obstr}, if\ penetration \\ 0, if\ no\ penetration \end{cases} \tag{6.6}$$

where L_i^{obstr} is the horizontal intrusion length of node i. The OI for each evaluation is then directly used as a constraint. For the pedal and chain ring constraint only the OI of the chain stay are considered, while for the tyre and disc brake obstruction both, the chain and the seat stay are relevant. Thus, $6 \cdot n_{eval}$ additional constraints are stated, where n_{eval} is the number of B-spline evaluations per tube. An illustration of the chain ring, pedal and tyre zone can be found in Figure 6.20. In Figure 6.21 and Figure 6.22 the penalized zones as well as the frame geometry are shown.

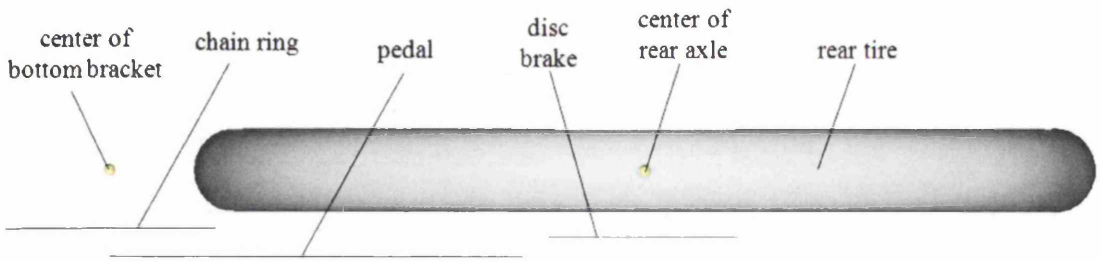


Figure 6.20: Penalized boundaries for chain ring, pedal, tyre and disc brake.

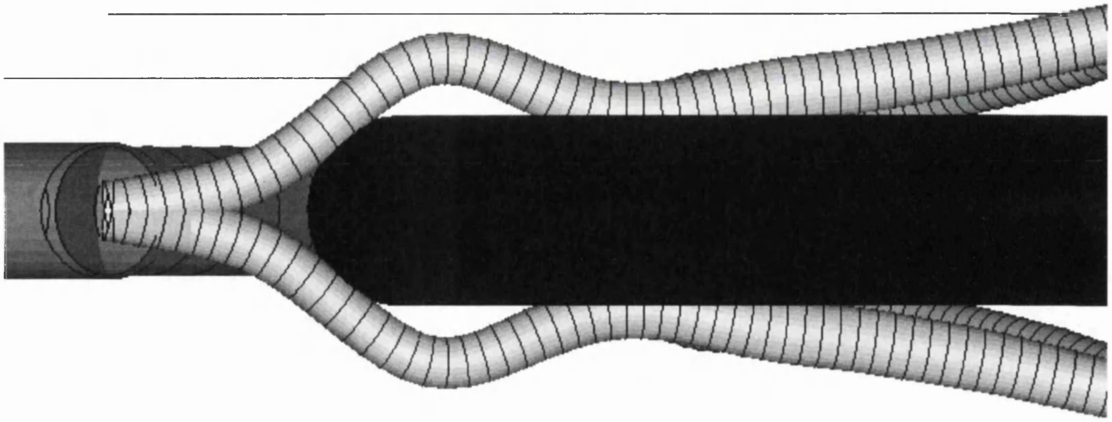


Figure 6.21: Penalized boundaries for chain ring, pedal and tyre. The top horizontal line represents the pedal, the line below the chain ring and the black surface in between the stays represents the tyre.

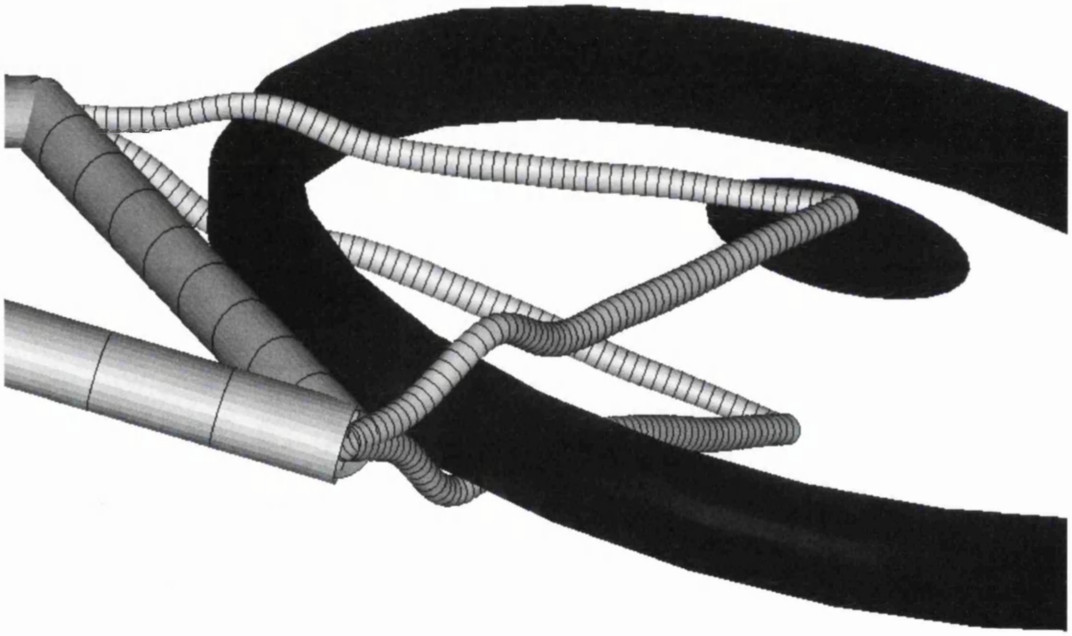


Figure 6.22: Penalized boundaries for tyre and disc brake obstruction constraint. The small circle at the rear drop out represents the disc brake and the surface in between the stays represents the tyre.

6.4.1.4. Design Variables

The variables that have to be determined in the optimization are the ply thicknesses for each tube. Their orientations are determined implicitly, by allowing plies with zero thickness and providing five plies at different angles. Furthermore, the control values governing the B-splines, which describe the distribution of the local shape parameters along each bar of the bicycle frame are determined. The global shape parameters are treated as input parameters governed by individual needs of the potential rider. They are treated as input values that need to be specified before the optimization and will not be considered as design variables in the optimization process. The parameters have been discussed in detail in Section 6.3.3.

6.4.1.5. Individual Demands

The influence of personal desires and requirements on the optimization is twofold. Firstly, the global shape parameters, i.e. the location of the frame's joints, are governed by the physical characteristics of the individual rider such as leg and arm lengths. Secondly, the constraints are governed by personal preferences and the purpose of use. For instance the desired compromise between weight and energy-efficiency or controllability determines the weight or stiffness constraint. Moreover, the field of application governs characteristic targets and input parameters. For example in endurance races the comfort of the bike plays a crucial role since the riders spend long times on the track. Other important aspects depending on personal preferences are the impact resistance of the frame against debris, influencing constraints on the ply layup and minimum thicknesses, the frame's strength in case of a crash and aesthetic requirements governing for example maximum and minimum tube dimensions as well as their relative differences.

6.4.2. Optimization Decomposition

As discussed in Section 6.3.4, OptiStruct will be used as the FE solver. This piece of software has a built-in optimization tool which solves structural optimization problems very efficiently. The declaration of size optimization design variables is straightforward and no sacrifice of design variability has to be made. This makes it very interesting to use OptiStruct to find the optimum ply thicknesses. The local shape parameters discussed in Section 6.3.3.2 on the other hand, are far more difficult to implement and for the sake of the frame's geometry variability, those parameters will be optimized separately. However, the two sets of parameters are strongly coupled and cannot be optimized individually without the use of a valid decomposition approach.

In this case the novel approach presented in Section 4.4.2 will be applied. Hence, two separate problems will be solved: A sub problem which is the size optimization that determines the optimum ply thicknesses and a top level problem which is the shape optimization that determines the optimum geometry of the frame. A Fortran tool has been developed that, given a set of shape parameters, creates the corresponding frame geometry (see Section 6.3.4). These shape parameters are optimized by means of a surrogate model. The idea behind it is that each experiment on which the meta-model is based, is a full size optimization, so that all values in the response surface are based on an optimized set of ply thicknesses.

The FI and the weight or stiffness constraint, (depending on the choice of the objective) will be considered in the size optimization. That means, as long as there is a feasible solution to the size optimization, these two constraints will be satisfied. This has the effect that no surrogate model needs to be created for these constraints which further improves the efficiency. At the same time no approximation error due to additional meta-models is made which effectively improves the accuracy. The only constraint that needs to be considered within the surrogate based optimization is the feasible stay path constraint. Since the necessary response is purely geometrical, its evaluation is inexpensive and no surrogate model needs to be created for it. Hence, only one approximation model will be built. This approximates the resulting sum of rigidities or the mass (depending on the chosen objective function). Due to the very high number of design variables in the size optimization (see Section 0) this approach is very beneficial in terms of efficiency.

When studying Figure 6.20, Figure 6.21 and Figure 6.22 it becomes apparent to which extent the obstruction constraints limit the paths of the stays, especially in between the chain ring and the tyre and in between the pedal and the disc. In fact, the determination of a feasible stay path can be considered as more the search for a feasible than an optimum solution. Additionally, to describe a feasible shape of this complexity, a significant number of control values are necessary. Hence, the design variables governing the stays' paths would add a significant computational effort since more experiments in the DoE would be required. The decision was made to exclude those paths from the main optimization problem, but to introduce a precedent optimization step that determines a path for each stay which remains constant for the entire procedure, which reduces the total number of design variables. Naturally, this extraction of the stay paths from the main optimization problem, will impair the final frame design, since the design variables governing the paths are strongly coupled with the remaining ones, but this step is necessary for the optimization problem to become sufficiently efficient.

Effectively, the optimization has been decomposed into two main problems: The determination of feasible stay paths and the determination of the tube cross-sections as well as the ply thicknesses. The latter has been subdivided into a top level shape optimization for the cross-sections and a sub level size optimization for the ply thicknesses. A summary of the optimization problem is given in Table 6.3, listing the design variables, the constraints and the objective function, in which sub problem they are considered and which load cases have to be solved. The effect of the proposed decomposition strategy on the optimization process is discussed in more detail in Section 6.4.4.

6.4.3. Optimization Problem

A summary of the optimization problem was given in Table 6.3.

Table 6.3: Specification of what objective function, constraints and design variables are, in which sub problem they are considered and in which load cases they are obtained.

		size opt.	***shape opt.***	***stay opt.***	***load cases***
objective function	*sum of rigidities (or frame's mass)*	*X*	*X*		*1,2 (geometry)*
	stay path			*X*	*geometry*
constraints	*failure indices*	*X*			*3,4,5,6,7*
	obstruction constraints		*X*	*X*	*geometry*
	max. weight (or min. rigidity)	*X*			*geometry*
design variables	*ply thicknesses*	*X*			
	tubes' semi-major axis		*X*		
	tubes' semi-minor axis		*X*		
	tubes' spatial path			*X*	

It shows that the weight and FI constraints are only considered in the size optimization and not in the top level shape optimization. This can be done because for each simulation specified in the DoE a full size optimization is performed and thus these constraints are a priori satisfied (if a feasible solution exists). The obstruction constraints on the other hand are not considered in the size optimization, since they are not influenced by the thicknesses of the tubes but they have to be considered in the shape optimization and naturally in the stay path optimization. The three optimization problems will be presented in the following.

6.4.3.1. Optimization Problem for the Determination of the Stay Paths

The design variables in this step are the control values governing the deviation of the seat and chain stays from a straight line. They are constrained by the obstruction constraints. In order to find an optimum set of control values the following optimization problem is considered: The objective function is based on the assumption that a frame increases in stiffness with decreasing deviation from a straight line of the stays. To avoid stress concentrations it should have no kinks but exhibit a smooth transition from one control value to the next. The resulting objective f from these assumptions is a weighted combination of two functions. The first function $f_{straight}$ is an indication on the deviation from the straight line, weighted by the factor w_1 and the second one f_{smooth} on the smoothness of the paths, weighted by the factor w_2. Each function is the sum of the evaluations for the seat and the chain stay. The optimization problem reads:

$$\begin{aligned} &min && f = f_{straight} \cdot w_1 + f_{smooth} \cdot w_2 \,, \\ &with && f_{straight} = \left|\sum_{i=1}^{n_{ctrl}} b_i^{str} - b_i\right|_{seat} + \left|\sum_{i=1}^{n_{ctrl}} b_i^{str} - b_i\right|_{chain} \\ & && f_{smooth} = \left|\sum_{i=2}^{n_{ctrl}} b_i - b_{i-1}\right|_{seat} + \left|\sum_{i=2}^{n_{ctrl}} b_i - b_{i-1}\right|_{chain} \,. \\ &subject\ to && g_1 = \sum_{i=1}^{6 \cdot n_{eval}} OI_i \leq 0 \end{aligned} \tag{6.7}$$

Here, b_i is the control value i of the B-spline governing the horizontal deviation of the seat or the chain stay from a straight line and b_i^{str} is the control value i in a B-spline describing a straight line. n_{Ctrl} is the number of control values in the respective B-spline.

This precedent optimization step is solved my means of a commercial optimization tool that uses the Modified Feasible Directions Algorithm (MFDA). This tool will be used in a black-box manner. For more information the reader is referred to (Vanderplaats

Research and Development Inc., 1999). For this procedure a constant width for the stays is used. This is the maximum width for which the optimizer finds a feasible solution.

The weights were chosen in a way such that the emphasis lies on stay paths that deviate as little as possible from a straight line and only major kinks are avoided. A good working set of weights to achieve this proofed to be [0.7 0.3]. In order to illustrate the influence of the weights, feasible paths for the seat and chain stays for an example frame have been determined for three different sets. These are [0.0 1.0], [0.7 0.3] and [1.0 0.0]. The resulting control values are illustrated in Figure 6.23 and Figure 6.24. One can see the strong influence of the tyre on the first three control values and the fairly mild influence of the disc brake at the end.

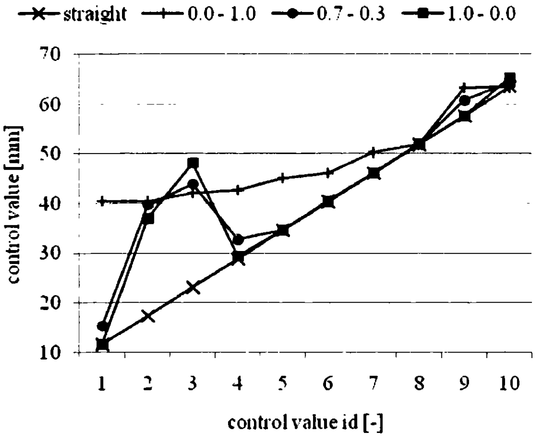


Figure 6.23: Influence of different weights on the resulting control point distribution for the seat stay.

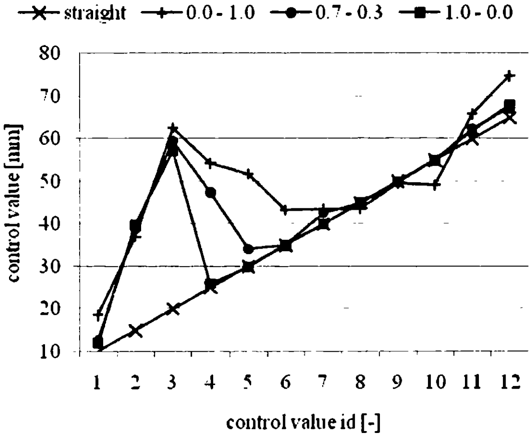


Figure 6.24: Influence of different weights on the resulting control point distribution for the chain stay.

It should be noted that the nature of this problem is multi-objective, since two objective functions exist. These are the minimization of the deviation from a straight line and the minimization of kinks. As seen before a focus on the former (weight set [1.0 0.0]) will lead to severe kinks and the focus on the latter (weight set [0.0 1.0]) leads to a high deviation from a straight line. Naturally, in order to investigate the interaction between the sums thoroughly, a Pareto front would have to be established and the best compromise between the two objectives found. This exceeds the scope of this thesis and is recommended for future work.

6.4.3.2. Size Optimization Sub Problem

In mathematical form, the size optimization sub problem for the maximization of the sum of rigidities reads:

$$\max \qquad f = OSR + THR \tag{6.8}$$

$$\text{subject to} \qquad g_1 = M - M_{Thres} \leq 0$$

$$g_2 = FI_{max} - \left(SF \cdot SF_{Fatigue}\right)^{-1} \leq 0$$

Here, M is the frame's mass and FI_{max} is the highest FI occurring in the frame. Alternatively, the objective can be stated as the minimization of the frame's mass. In that case, the problem reads:

$$\min \qquad f = M \tag{6.9}$$

$$\text{subject to} \qquad g_1 = R_{Thres} - (OSR + THR) \leq 0$$

$$g_2 = FI_{max} - \left(SF \cdot SF_{Fatigue}\right)^{-1} \leq 0$$

As mentioned before, this problem is solved by means of OptiStruct.

6.4.3.3. Shape Optimization Top Level Problem

The shape optimization top level problem is solved by means of a surrogate model, which interpolates the results from the size optimizations. Hence, the constraints g_1 and g_2, stated in the size optimization sub problem, are a priori satisfied. The only constraint that has to be considered is the obstruction constraint. In accordance with Section 2.3.5.5, this constraint is incorporated in the objective via a penalty term, which makes the problem unconstrained. The objective for the shape optimization maximizing the frame's rigidity can be formulated as:

$$\max \qquad f = OSR + THR + pen \cdot \sum_{i=1}^{6 \cdot n_{eval}} OI_i \tag{6.10}$$

where *pen* is a penalization factor. Numerical experiments showed that a value of 1000 for pen is a good working value to safely rule out any obstruction. For the minimization of the frame's mass the objective reads:

$$\min \qquad f = M - pen \cdot \sum_{i=1}^{6 \cdot n_{eval}} OI_i \tag{6.11}$$

6.4.4. Optimization Process

The developed optimization procedure incorporates three commercial software tools: The geometry and meshing software HyperMesh (Altair HyperWorks (a), 2009), the structural optimization tool OptiStruct (Altair HyperWorks (b), 2009) and an MFDA type optimization program (Vanderplaats Research and Development Inc., 1999). For the remaining tasks, in-house Fortran tools are used. One is the driver of the optimization, thus issuing the commands for the model generation, reading the analysis results, calculating the objective function and constraints as well as handling the B-splines. Others create the OLH, perform the Kriging interpolation and optimize using the response surface. The flow of the developed optimization procedure is illustrated in Figure 6.25 and will be presented in more detail in the following.

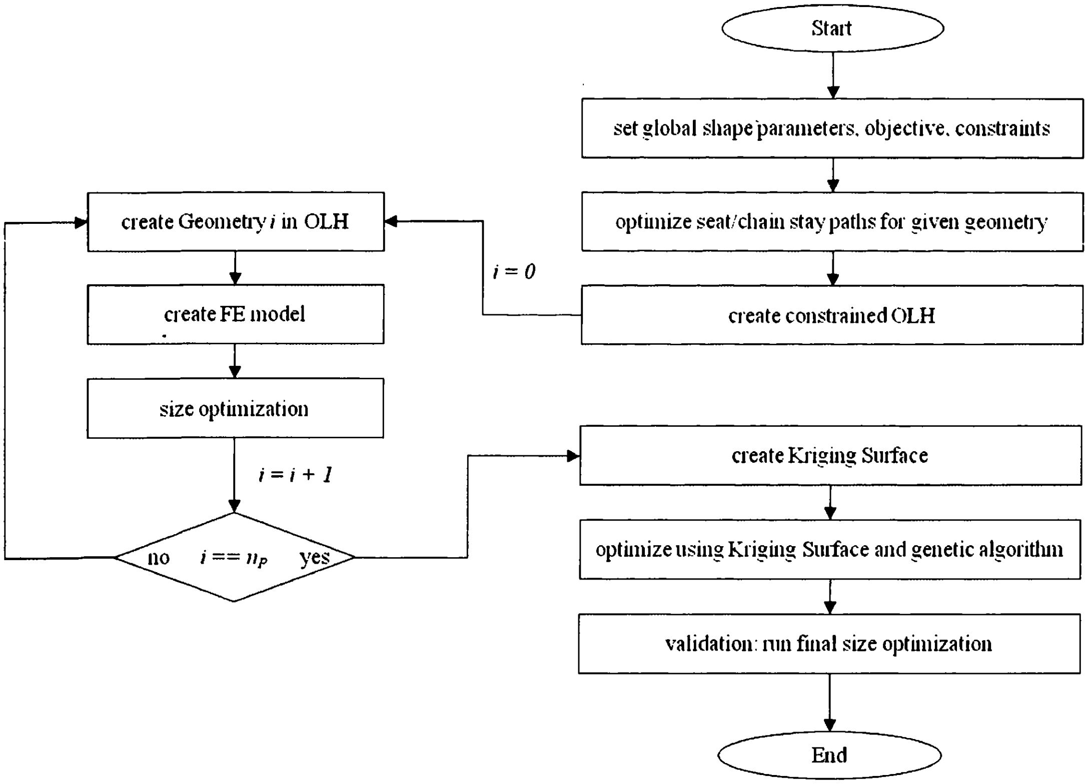


Figure 6.25: Flow of the developed optimization software framework.

Step 1: Specification of Global Shape Parameters, Objective, Constraints

In the first step, the designer has to specify the global shape parameters, i.e. the location of the joints. Also, the desired frame characteristics such as the required stiffness or the maximum weight have to be determined. This information is dependent on the potential rider. More information on this can be found in Section 6.4.1.5.

Step 2: Feasible Stay Paths

The global shape parameters specified in Step 1 determine the general outline of the frame geometry. For this geometry, the path of the seat and chain stays are determined so that, for a maximum possible stay width, the tyre, disc brake, chain ring and pedals are not obstructed. More information on this step in the optimization procedure can be found in Section 0.

Step 3: Optimum Latin Hypercube

Since the shape parameters are optimized using a Kriging model, a DoE needs to be created that determines for which shape parameter settings the experiments are to be run. That means for n_{DV}^{Shape} shape design variables an OLH is created in a n_{DV}^{Shape}-dimensional design space. In order to thoroughly parameterize the shape of all the tubes in the frame a considerable number of design variables is necessary which can result in a very high-dimensional design space. In the examples discussed in Section 6.5 this design space has 19 dimensions. However, as it can be seen in Section 0, the majority of these design variables are governing the shapes of the four stays in the rear triangle of the frame. Due to the obstruction constraints these are highly constrained (see Section 6.4.1.3). In order to reduce the size of the searched design space experiments will only be created in the feasible regions. To this end the proposed strategy for the creation of OLHs in constrained design spaces discussed in Section 3.2.3 will be applied here. The constraint considered during the creation of the OLH checks whether or not the corresponding shape parameters in a potential experiment obstruct the tyre, the disc brake, the chain ring or the pedals. Only those experiments that correspond to an unobstructed rear triangle will be considered. This reduces the design space significantly and a smaller number of experiments will be required for the creation of a sufficiently accurate approximation model.

Step 4: Run Experiments

After the OLH has been created all n_P experiments have to be conducted. Here, each experiment is a full size optimization, i.e. for the actual set of local shape parameters an optimal set of ply thicknesses is determined that maximizes the sum of the rigidities or minimizes the mass while the FI constraints and the mass or stiffness constraint are satisfied.

Step 5: Create Kriging Surface

The obtained results from the conducted experiments that have been specified in the OLH are used to build a Kriging model. The quality of the created surrogate model can be estimated by means of a cross-validation. More information on the creation and validation of Kriging models can be found in Section 3.3.4.

Step 6: Optimize using Kriging Surface

In the next step, the local shape parameters are optimized using the Kriging model and the continuous GA that has been presented in Section 2.3.5.3. Much like during the creation of the constrained OLH, during this optimization step too, only the obstruction constraint has to be considered. That means for each chromosome which represents a set of local shape parameters has to be checked whether or not the seat and chain stays obstruct the tyre, the disc brake, the chain ring or the pedals. If obstruction occurs the fitness of this chromosome is penalized as described in Section 6.4.3.3 in order to avoid infeasible designs. The FI and the mass or stiffness constraints do not have to be considered since the Kriging model is built with the responses from size optimizations, which imply satisfied constraints. As per definition, the surrogate model replaces the computationally expensive simulation which in this case is a sub level size optimization. This means the prediction from the Kriging model evaluation will approximate the response of a full size optimization.

Step 7: Final Size Optimization

The seventh and last step in the proposed optimization procedure is another size optimization. A model is created for the optimum local shape parameters that have been obtained in the previous step. For this geometry, a size optimization is performed. This is done for two reasons. Firstly, the optimum stiffness or mass given by the GA needs to be validated. If the two values do not match to a satisfactory degree more experiments need to be added to the DoE to improve the accuracy of the surrogate model. The second reason is that, at this stage, for the optimum geometry no optimum set of ply thicknesses exist, which will be obtained by means of the final size optimization.

6.5. Results

In the following, the developed optimization procedure will be applied to two examples to demonstrate its capabilities in finding optimal frame designs while meeting real life requirements. The total number of design variables in these applications is 49. These are 19 for the shape and 30 for the size optimization sub problem. They are listed in Table 6.4.

Table 6.4: Number of design variables for each tube for both examples.

	shape optimization		***size optimization***
	semi-major axis	*semi-minor axis*	*ply thickness*
top tube	*1*	*1*	*5*
down tube	*1*	*1*	*5*
seat tube	*1*	*1*	*5*
seat stay	*4*	*2*	*5*
chain stay	*5*	*2*	*5*
head tube	*0*	*0*	*5*

Their limits can be found in Table 6.5. The general safety factor for the FI constraints *SF*, the maximum allowable weight or minimum required rigidities are listed in Table 6.6. The first example covers a frame which is intended for a tall rider desiring a frame with a balanced weight to stiffness ratio by constraining the weight and maximizing the sum of the rigidities. The second example generates a frame for small sized riders demanding a low weight frame by minimizing the weight and constraining the sum of the rigidities to equal that of the first example's result.

Table 6.5: Limits for design variables for both examples.

	shape optimization				***size optimization***	
	semi-major axis [mm]		*semi-minor axis [mm]*		*ply thickness [mm]*	
	min	*max*	*min*	*max*	*min*	*max*
top tube	*15.0*	*25.0*	*22.0*	*32.0*	*0.0*	*2.0*
down tube	*15.0*	*25.0*	*22.0*	*32.0*	*0.0*	*2.0*
seat tube	*16.5*	*25.0*	*16.5*	*25.0*	*0.0*	*2.0*
seat stay	*2.0*	*16.0*	*5.0*	*20.0*	*0.0*	*2.0*
chain stay	*2.0*	*16.0*	*5.0*	*20.0*	*0.0*	*2.0*
head tube	-	-	-	-	*0.0*	*2.0*

Commonly, bicycle frames in different sizes mainly vary in the projected top tube length, whereas the rear triangle remains the same (see for instance (Kona Bikes, n.d.), (Trek Bicycle Corporation, n.d.) or (Specialized Bicycle Components, n.d.)). The

Table 6.6: Safety factor for FI constraints, maximum weight and surface area.

	SF [-]	*Max. weight [kg]*	*Min. sum of rigidities [-]*
Example 1	*1.15*	*0.7*	-
Example 2	*1.15*	-	*379*

projected top tube length of frames in size L is usually around 8% shorter than those in size S. Hence, in Example 2 it has been reduced by 8% with respect to that in Example 1. The difference in the geometrical outline of the two frames can be seen in Figure 6.26.

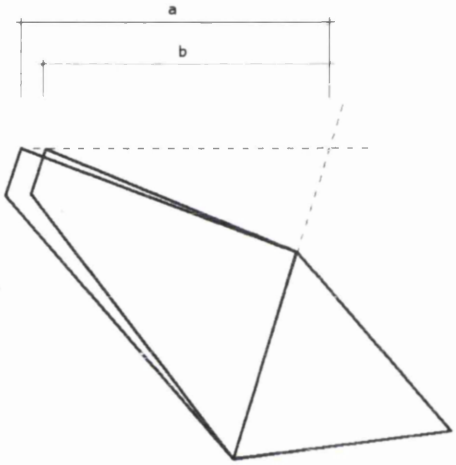


Figure 6.26: Comparison of frame geometries for Examples 1 and 2. The projected top tube length *a* for Example 1 has been reduced by 8% for Example 2.

6.5.1. Feasible Stay Paths

The dimensions and locations of the chain ring, the pedal, the rear tyre and disc brake for the studied frame are shown in Figure 6.27.

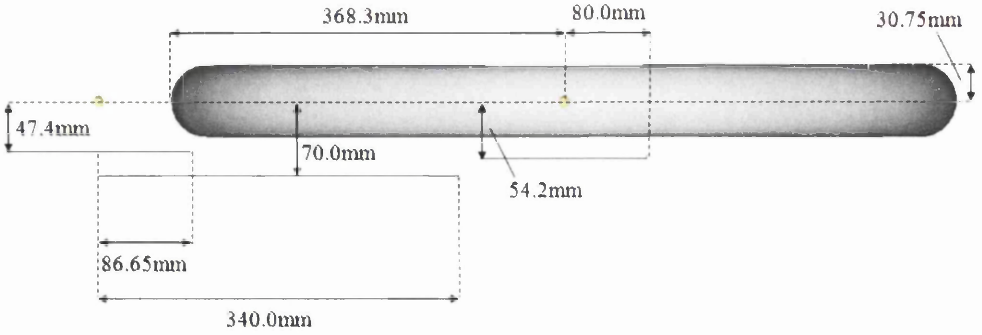


Figure 6.27: Dimensions and locations of the chain ring, the pedal, the rear tyre and disc brake.

Because the rear triangle is the same in both examples this step has to be performed only once. The maximum stay width for which the optimizer yielded a feasible solution is 16mm. Thus, the semi-major axis for both stays is taken as 8mm throughout this first optimization step. The optimization process required 56 iterations to converge. The development of the objective function and the sum of obstruction constraint violations are illustrated in Figure 6.28. The strong fluctuation of the constraint violation indicates the degree to which the stay paths are constrained. The resulting stays as well as the penalized obstruction zones are shown in Figure 6.29. One can see that the stays remain mostly straight. Only in the area of the tyre near the bottom bracket and the seat joint they are moved outward to avoid contact. The transition can be considered very smooth and no kinks occur.

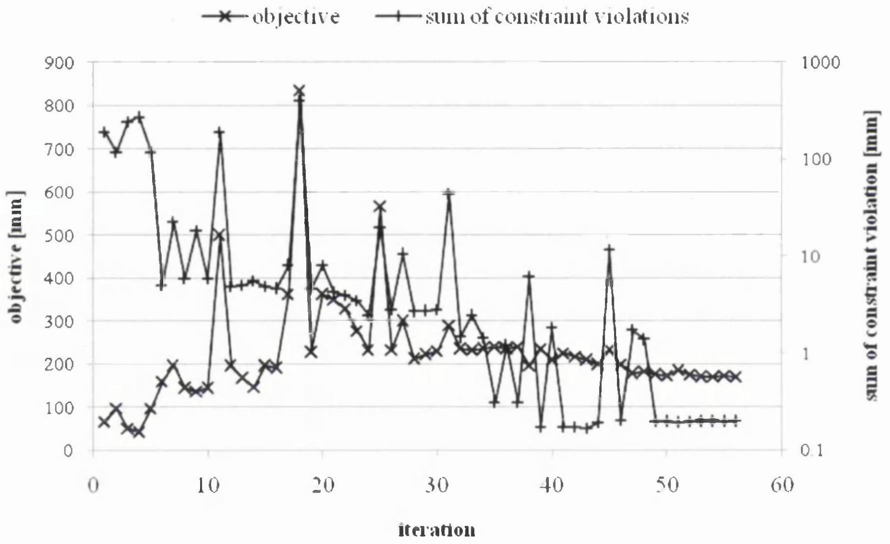


Figure 6.28: Development of objective function and sum of obstruction constraint violations during the initial stay path optimization.

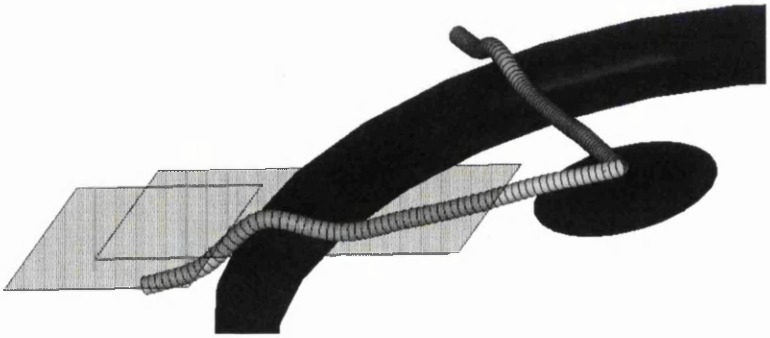


Figure 6.29: Resulting seat and chain stays as well as obstruction zones for tyre, disc brake, wheel ring and pedal. The constant diameter is 16mm.

6.5.2. Surrogate Based Optimization

6.5.2.1. Optimum Latin Hypercube

Once the paths of the seat and chain stays are determined the constrained OLH has to be created. In this case it is a set of 70 experiments distributed in a 19 dimensional design space. During the creation of the OLH, the obstruction constraint is considered and experiments will be only created in the feasible region. Affected by this constraint are

all B-spline control values governing the semi-major axes of the seat and chain stays, i.e. the widths of their cross-sections. In the present case these are four for the seat and five for the chain stays. Hence, ten of the 19-dimensions are unconstrained and nine are constrained. The potential energy of the created constrained OLH is 0.1755 and the space division vector is $\underline{n}_{Div} = [140 \quad \ldots \quad 140]$. The same OLH can be used for both examples as the rear triangle does not change. The development of the potential energy during the optimization process is shown in Figure 6.30. The OLH itself as well as the local shape parameters for each experiment are given in Appendix C.

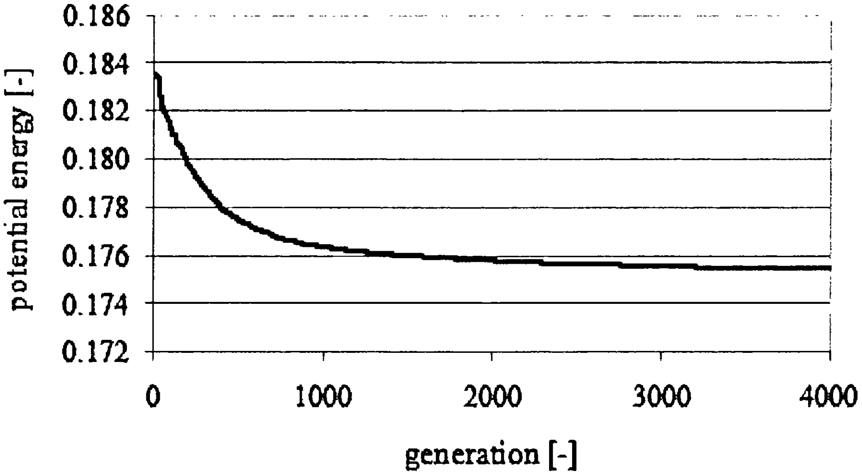


Figure 6.30: Development of the potential energy throughout the optimization process for the creation of the OLH.

6.5.2.2. Kriging Model

Now the 70 experiments for each example have to be conducted. Here each experiment is a full size optimization of the corresponding geometry which is governed by the values of the local shape design variables specified by the location of the experiment in the design space. The results from these experiments are used to create a Kriging model. They are listed in Table 6.7.

Table 6.7: Results from the 70 experiments conducted in Example 1 and Example 2.

Example 1				***Example 2***			
Exp. ID	*Sum of rigidities [-]*	***Exp. ID***	*Sum of rigidities [-]*	***Exp. ID***	*Mass [kg]*	***Exp. ID***	*Mass [kg]*
1	197.7	36	227.9	1	1.076	36	0.953
2	176.8	37	194.5	2	1.199	37	0.927
3	218.4	38	298.0	3	0.925	38	0.799
4	184.0	39	176.6	4	1.082	39	1.239
5	164.3	40	249.2	5	1.230	40	0.892
6	200.1	41	190.1	6	1.036	41	1.119
7	314.2	42	257.1	7	0.750	42	0.882
8	215.7	43	287.9	8	0.954	43	0.799
9	203.2	44	232.4	9	1.006	44	0.961
10	235.9	45	243.2	10	0.931	45	0.901
11	275.3	46	252.2	11	0.822	46	0.887
12	274.7	47	250.8	12	0.803	47	0.884
13	245.2	48	186.5	13	0.859	48	1.112
14	228.1	49	190.0	14	0.958	49	0.989
15	258.4	50	240.2	15	0.882	50	0.937
16	253.0	51	246.8	16	0.878	51	0.904
17	208.7	52	223.1	17	1.037	52	0.950
18	302.9	53	268.9	18	0.774	53	0.842
19	191.2	54	193.9	19	1.159	54	1.084
20	262.5	55	164.6	20	0.836	55	1.164
21	240.7	56	201.7	21	0.900	56	1.040
22	202.4	57	242.2	22	1.012	57	0.878
23	287.7	58	264.2	23	0.779	58	0.832
24	225.8	59	273.8	24	0.924	59	0.811
25	217.5	60	309.8	25	0.982	60	0.726
26	245.0	61	188.9	26	0.887	61	1.118
27	268.4	62	293.2	27	0.831	62	0.778
28	235.8	63	284.3	28	0.967	63	0.799
29	207.3	64	249.2	29	0.990	64	0.887
30	223.1	65	222.9	30	0.878	65	0.996
31	244.0	66	206.7	31	0.910	66	1.015
32	226.7	67	269.0	32	0.942	67	0.842
33	204.0	68	264.3	33	0.961	68	0.856
34	163.0	69	239.4	34	1.199	69	0.863
35	211.3	70	295.6	35	0.952	70	0.791

The frame geometries for the experiments in Example 1 and 2 yielding the highest and lowest response are shown in Appendix C. For Example 1 these are Experiments 7 and 34 and for Example 2, Experiments 39 and 60. The development curves for the sum of rigidities and the mass for these four size optimizations can also be found in Appendix C.

The quality of the surrogate model can be estimated by means of a cross-validation where one of the 70 responses is estimated by evaluating the Kriging model based on all experiments except for the one that is estimated. In a perfect model the estimated response equals the one from the experiment. Thus, when the estimated values are plotted against the real ones a straight line would be ideal. The cross-validations for Examples 1 and 2 are shown in Figure 6.31 and Figure 6.32 respectively. Given the relatively small number of 70 experiments in a 19 dimensional design space, the response surfaces can be considered sufficiently accurate. The maximum SCVR for Example 1 is 1.79 and in Example 2 it is 1.74. More information on cross-validation can be found in Section 3.3.4.

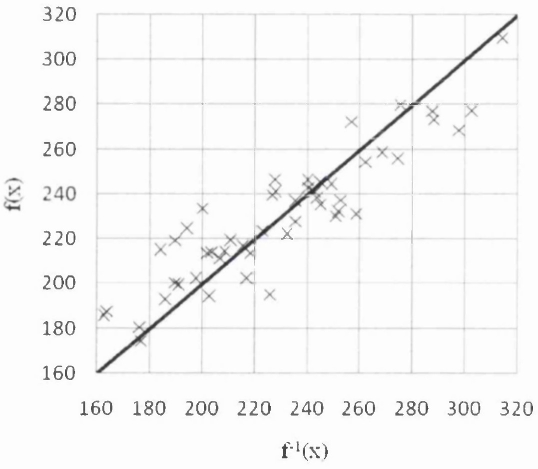


Figure 6.31: Cross-validation for Kriging response surface for Example 1. The maximum SCVR is 1.79.

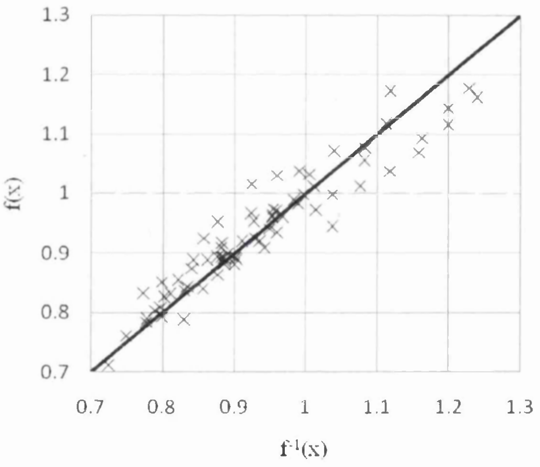


Figure 6.32: Cross-validation for Kriging response surface for Example 2. The maximum SCVR is 1.74

6.5.2.3. Optimization using Kriging Model

In the next step the response surface is used to optimize the shape of the frame by means of the continuous GA. Here too, the obstruction constraints are incorporated into the optimization. As explained in Section 6.4.3.3, the constraints are incorporated via a penalization term.

The optimization yields a maximum sum of rigidities of 348 in Example 1 (11% better than the best experiment in the DoE) and a minimum frame weight of 0.571kg in Example 2 (21% lighter than the lightest experiment in the DoE). The development of the objective functions during the optimization using the Kriging model, i.e. the sum of rigidities in case of Example 1 and the frame's weight in case of Example 2 are illustrated in Figure 6.33. The values indicate that in both cases no penalty term was added to the objective function, which means that the best chromosome was feasible throughout the optimization.

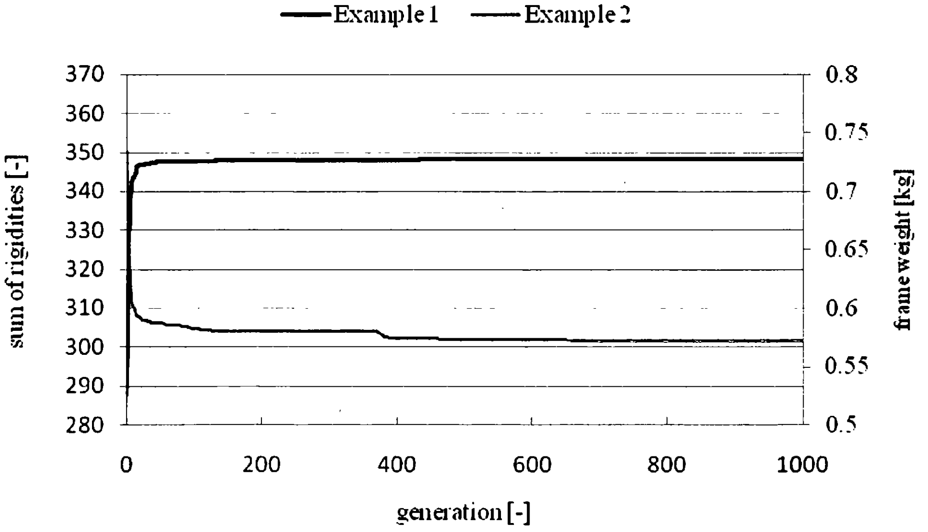


Figure 6.33: Development of objective functions during optimization using Kriging model for both examples.

The resulting frame geometries are illustrated in Figure 6.34 for Example 1 and in Figure 6.35 for Example 2. The design variables, i.e. the local shape parameters, corresponding to the optimum solutions are listed in Table 6.8 and Table 6.9.

6.5.3. Final Size Optimization

For two reasons a final size optimization has to be performed. Firstly, the result from the optimization of the surrogate model has to be validated and secondly, an optimum set of ply thicknesses has to be obtained that corresponds to this geometry. For Example 1 this step yields a sum of rigidities of 379 (348 predicted by Kriging) and for Example 2 a frame weight of 0.613kg (0.571kg predicted by Kriging). That means compared to the prediction of the Kriging model, these results have an error of 8.9% and 7.3% with respect to the Kriging interpolation. Considering the fact that only 70 experiments in a 19 dimensional design space have been used, this result is very satisfactory.

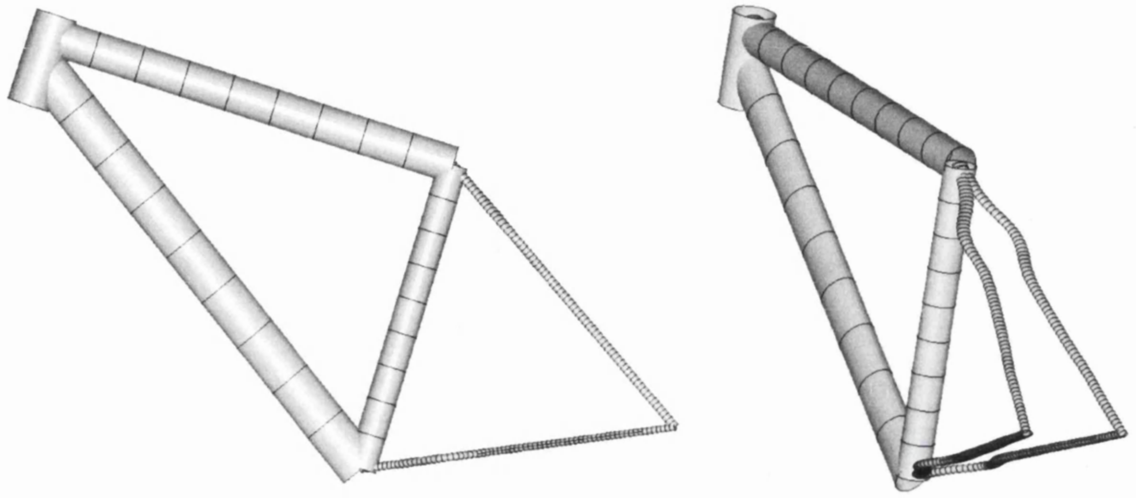


Figure 6.34: Optimal frame geometry for Example 1.

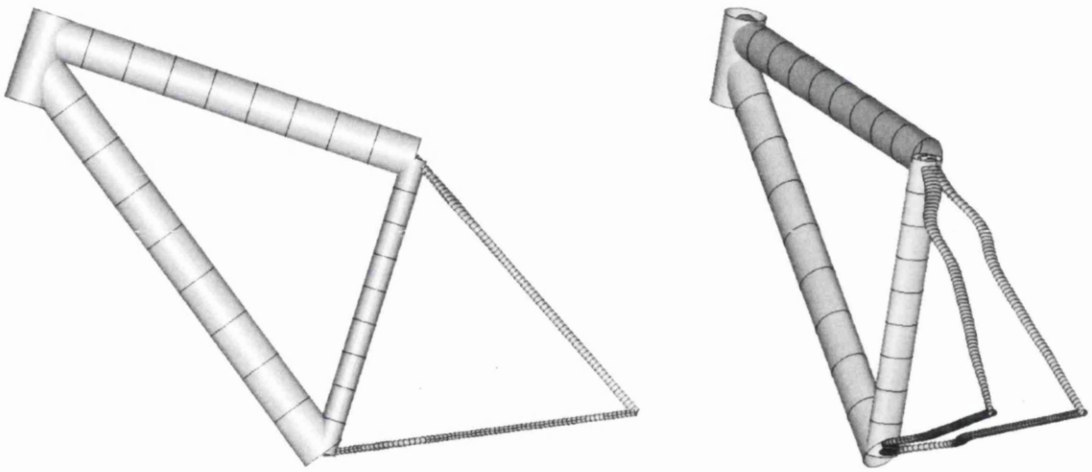


Figure 6.35: Optimal frame geometry for Example 2.

In Example 1 the maximum weight was set to 0.7kg and in Example 2 the minimum sum of rigidities to 379. Due to the nature of the size optimization, these targets are met in both cases. The ply thicknesses obtained in this final size optimization are shown in Table 6.10 and Table 6.11.

A common index to measure the performance of a bicycle frame is the ratio between the sum of the two rigidities and the frame's weight denoted as SGI. According to (Motor Presse Stuttgart GmbH & Co. KG, n.d.) typical values for high-end frames usually don't exceed a value of 250. The obtained frame designs in the two examples feature SGIs of 541 and 618, respectively. Naturally, the stiff joints caused by the enveloping cylinders lead to an overestimation of the stiffness and an underestimation of the weight. The design of the joints is discussed in the following section.

Table 6.8: Optimal semi-major and semi-minor axes for Example 1. Obtained from the optimization using the Kriging model.

	semi-major axis			***semi-minor axis***		
	Total number	*id*	*value [mm]*	*Total number*	*id*	*value [mm]*
top tube	*1*	*1*	*24.5*	*1*	*1*	*25.9*
down tube	*1*	*1*	*23.3*	*1*	*1*	*32.0*
seat tube	*1*	*1*	*23.7*	*1*	*1*	*21.3*
seat stays	*4*	*1*	*11.0*	*2*	*1*	*6.5*
	-	*2*	*8.3*	-	*2*	*6.5*
	-	*3*	*8.0*	-	-	-
	-	*4*	*5.8*	-	-	-
chain stays	*5*	*1*	*11.4*	*2*	*1*	*5.9*
	-	*2*	*6.6*	-	*2*	*5.0*
	-	*3*	*15.6*	-	-	-
	-	*4*	*11.2*	-	-	-
	-	*5*	*8.6*	-	-	-

Table 6.9: Optimal semi-major and semi-minor axes for Example 2. Obtained from the optimization using the Kriging model.

	semi-major axis			***semi-minor axis***		
	Total number	*id*	*value [mm]*	*Total number*	*id*	*value [mm]*
top tube	*1*	*1*	*25.0*	*1*	*1*	*29.7*
down tube	*1*	*1*	*25.0*	*1*	*1*	*31.9*
seat tube	*1*	*1*	*25.0*	*1*	*1*	*17.6*
seat stays	*4*	*1*	*14.8*	*2*	*1*	*5.8*
	-	*2*	*7.6*	-	*2*	*6.7*
	-	*3*	*10.2*	-	-	-
	-	*4*	*2.9*	-	-	-
chain stays	*5*	*1*	*12.0*	*2*	*1*	*5.0*
	-	*2*	*6.5*	-	*2*	*5.2*
	-	*3*	*14.8*	-	-	-
	-	*4*	*8.8*	-	-	-
	-	*5*	*10.2*	-	-	-

The development of the sum of rigidities and the frame's mass for both examples is shown in Figure 6.36.

Table 6.10: Optimal ply thicknesses in mm for Example 1. Obtained in the final size optimization.

	Bi-directional	***Uni-directional***			
	0°	*-45°*	*0°*	*45°*	*90°*
head tube	*0.22*	*0.03*	*0.03*	*0.03*	*0.02*
top tube	*0.22*	*0.19*	*0.04*	*0.16*	*0.01*
down tube	*0.22*	*0.53*	*0.38*	*0.56*	*0.01*
seat tube	*0.22*	*0.02*	*0.34*	*0.02*	*0.01*
seat stays	*0.22*	*0.07*	*0.5*	*0.06*	*0.03*
chain stays	*0.22*	*0.2*	*1.64*	*0.18*	*0.07*

Table 6.11: Optimal ply thicknesses in mm for Example 2. Obtained in the final size optimization.

	Bi-directional	***Uni-directional***			
	0°	*-45°*	*0°*	*45°*	*90°*
head tube	*0.22*	*0.03*	*0.03*	*0.03*	*0.03*
top tube	*0.22*	*0.14*	*0.03*	*0.12*	*0.02*
down tube	*0.22*	*0.4*	*0.47*	*0.44*	*0.02*
seat tube	*0.22*	*0.02*	*0.26*	*0.02*	*0.02*
seat stays	*0.22*	*0.05*	*0.39*	*0.05*	*0.03*
chain stays	*0.22*	*0.19*	*1.45*	*0.17*	*0.07*

6.5.4. Joint Design

In a separately conducted study (Sexton, 2010), the three joints of which two have been considered as rigid during the optimization process, were sought to be designed. Here, the optimal frame geometry of the final design from Example 1 was considered. The design process was the following: The finite element model obtained from the optimization was converted into a workable 3D CAD geometry in IGES format. In the CAD environment the rigid joints were removed and the concept joints were overlaid straight onto the tube layout while ensuring that the necessary clearances were provided. The bottom bracket joint is shown in Figure 6.37, the seat joint in Figure 6.38 and the head tube joint in Figure 6.39.

Now a size optimization followed. To this end the joint surfaces were divided into numerous patches and each patch was allowed to exhibit different ply thicknesses. The

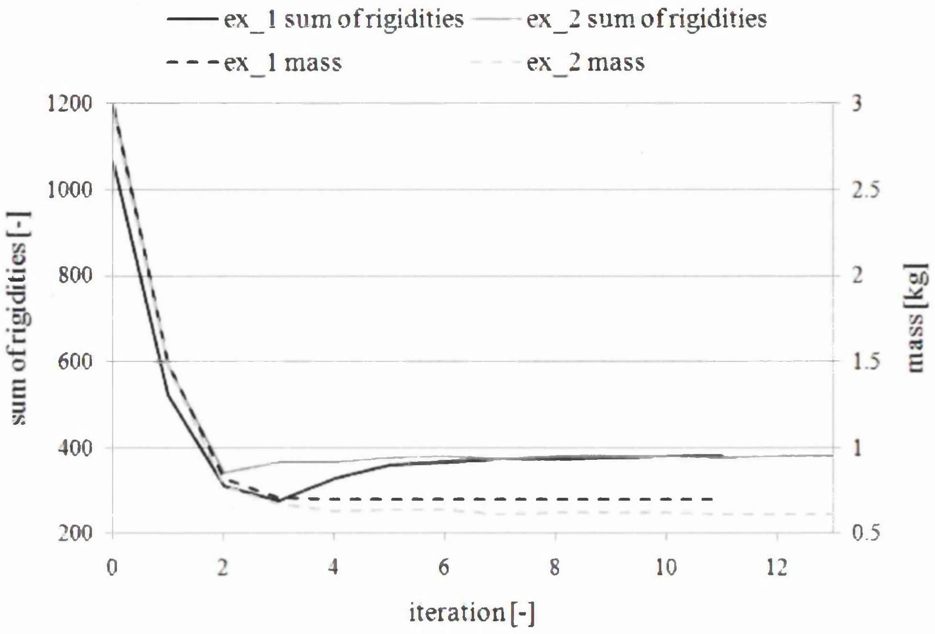


Figure 6.36: Development of the sum of rigidities and the frame's mass for Example 1 and 2.

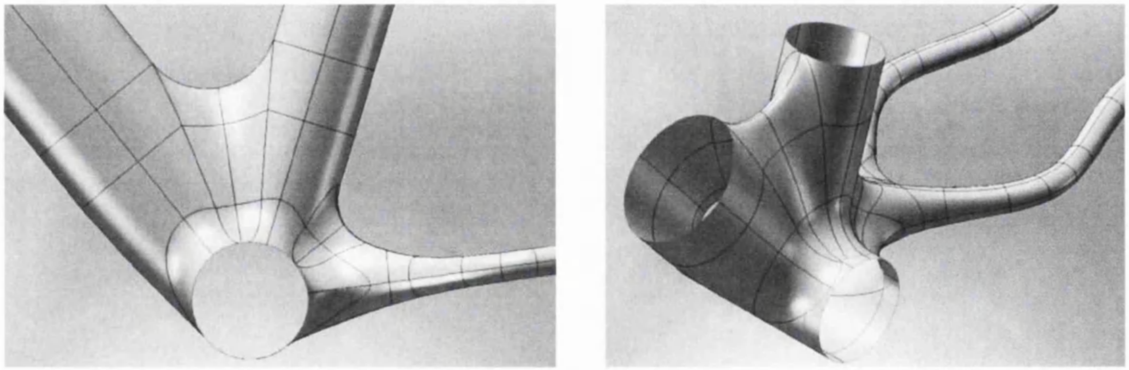


Figure 6.37: Surface model for the bottom bracket.

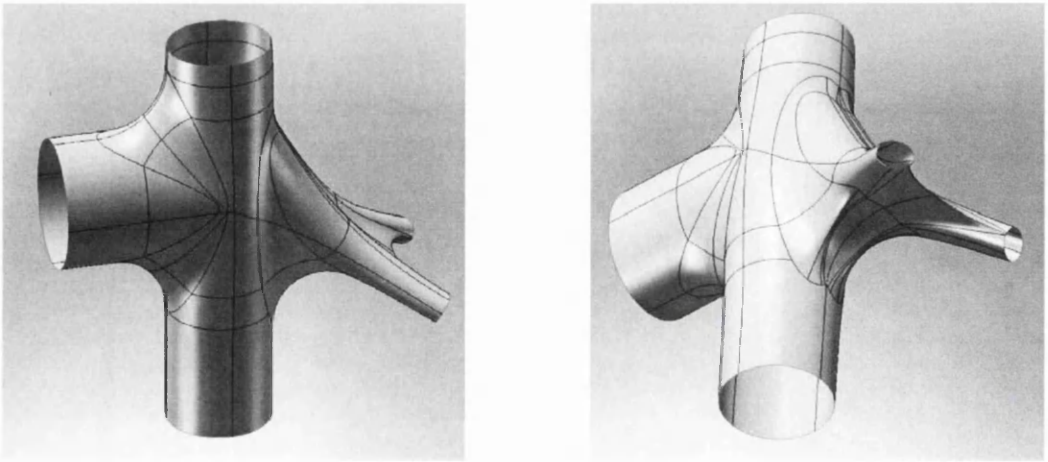


Figure 6.38: Surface model for the seat joint.

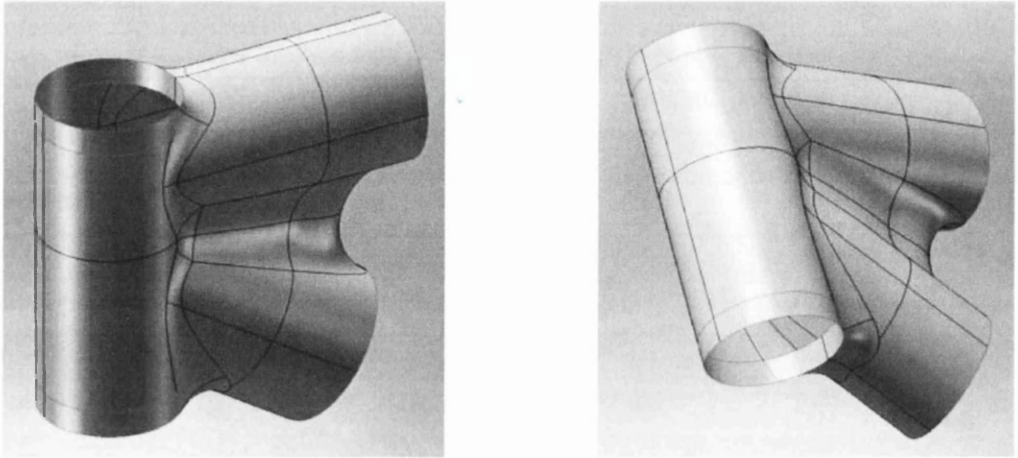


Figure 6.39: Surface model for the head tube joint.

ply thicknesses of the tubes were also defined as design variables. The objective function and the FI constraint remained the same as within this study. The maximum allowable weight was increased to 0.8kg. The optimization problem was solved using OptiStruct. The resulting sum of rigidities was 242 (379 with rigid joints) which, with a mass of 0.8kg, corresponds to an SGI of 303. This value still exceeds those of high-end frames. Furthermore, within this study, the maximum mass has been chosen very low. A value closer to existing frames could possibly yield a higher SGI. Naturally, the results are based on continuous ply thicknesses and those need to be translated into a discrete number of plies at available thicknesses which in return either increases the weight or reduces the stiffness.

6.6. Concluding Remarks

A bespoke optimization software framework for the design of optimal bicycle frames with respect to the EFBe tests (EFBE Prüftechnik, n.d.) taking into account rider specific demands was developed. To this end four software tools have been used. These are the geometry and meshing software HyperMesh (Altair HyperWorks (a), 2009), the FE solver and optimization tool OptiStruct (Altair HyperWorks (b), 2009), an MFDA type optimization program (Vanderplaats Research and Development Inc., 1999) and an in-house Fortran tool issuing the commands for the model generation, reading the analysis results, calculating the objective function and constraints.

For improved efficiency the developed decomposition approach discussed in Section 4.4.2 has been applied here. That means the optimization problem is decomposed into a size optimization sub problem solved by OptiStruct and a shape optimization top level problem solved by means of surrogate based optimization where each experiment in the DoE is a full size optimization. The developed surrogate based optimization framework presented in Chapter 3 has been used, which contains the developed in-house code for the generation of OLHs in constrained design spaces and an implementation of the Kriging interpolation method as well as a continuous GA for the optimization. To decrease the dimensionality of the problem and because they are heavily constrained by surrounding bike parts that may not be obstructed by the frame, the paths of the seat and chain stays have been extracted from the main optimization and are determined in a preliminary optimization step. This step has been solved for an objective that requires weights for their degree of deviation from a straight line and for the paths' smoothness. Within this work the weights have been chosen as [0.7 0.3]. A possible field for future work could be the investigation of these two weights by means of methods of multi-objective optimization. Within the main optimization the stay paths remain constant while the cross-sections are variable. Naturally, the exclusion of the paths from main optimization has an impairing effect on the final design since the design variables governing the paths are strongly coupled with the remaining ones, but this step was necessary for the optimization problem to become sufficiently efficient. In the future one could work on a way to include the paths in the main objective.

In order to realize a high variability of the frame geometry the model has been extensively parameterized using B-splines. This parameterization allows for smoothly varying tube shapes and paths as well as different laminate ply thicknesses for each tube. Furthermore, the global geometry of the frame can be altered by means of the joint locations. The attained degree of parameterization allows the consideration of rider specific demands through both, geometry input values and constraint boundaries.

The automation of the simulation model creation has been realized in accordance with the approach proposed in Section 4 using a Fortran tool to write command files for HyperMesh and for the interpretation and processing of the created FE mesh. The automation required a simplification of the frame's joints. Here rigid cylinders are created which envelope the joints to transfer the loads from one tube to the next.

The application of the software framework to two example problems showed that a good compromise between efficiency and a high degree of parameterization could be achieved through the use of B-splines, the decomposition of the optimization problem and the coupling of in-house code and commercial software. Furthermore, it was demonstrated how the complex geometry was incorporated in an automated geometry creation scheme. Despite a very high-dimensional design space for the local shape design variables (in the example it had 19 dimensions) a surrogate model based on 70 experiments proved to be satisfactory in terms of accuracy. The reason for this are the obstruction constraints that constrain the sizes of the seat and chain stay. This was benefitted from by using an OLH that creates feasible experiments only.

The resulting mass and rigidity of the frame designs in the application showed successfully that optimal bicycle frames with very good characteristics can be designed in an efficient way for different rider specific demands. The frame's joints of the frame obtained in Example 1 have been designed in a separate study which showed that the sum of rigidities was reduced by 13% where the maximum allowable mass has been increased by 100g.

7. C-Fec Wind Turbine Blade Optimal Design

Summary: This chapter is concerned with the optimum design of a blade for a novel vertical axis wind turbine. Due to scarce knowledge and literature for vertical axis wind turbines a design approach similar to those with horizontal axes is chosen. The altered design requirements are accounted for by creating a parameterized simulation model and performing size optimization runs for 32 models with different material settings and shear web locations where the model creation process has been automated. The continuous laminate ply thicknesses of the best design are translated into discrete numbers of plies and a ply stacking optimization is performed. The final design satisfies all the requirements specified in the Germanischer Lloyd design standard.

7.1. Introduction

C-Fec Ltd. have developed a concept design for a novel vertical axis wind turbine. This wind turbine consists of eight blades that are positioned between two circular supports as illustrated in Figure 7.1. While rotating, the blades pass through a tunnel that influences and funnels the wind blowing on the blades. To avoid obstruction the displacements of the blades' tips have to be constrained. This can be seen in Figure 7.2.

Figure 7.1: The assembled wind turbine. It consists of eight blades that are fully supported at both ends. Due to that fact bending stiffness is not as crucial as for blades in regular turbines.

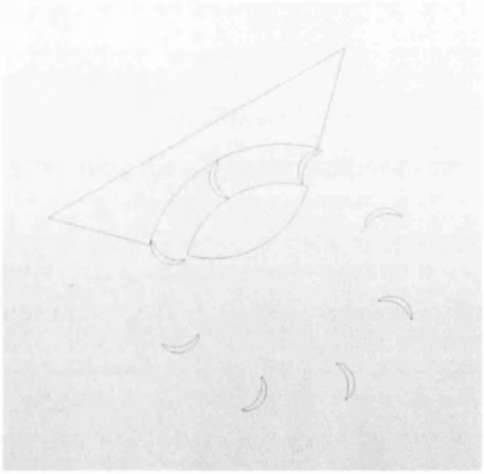


Figure 7.2: While the rotor is in motion the blades pass through a tunnel. To avoid obstruction the displacements of the blades' tips have to be constrained.

The blades themselves have an overall length of 20m and are 4.0m wide. They are symmetrical with respect to the longitudinal and transverse planes. A top view of a blade can be seen in Figure 7.3. The blades' cross-section is of a half-moon shape. The

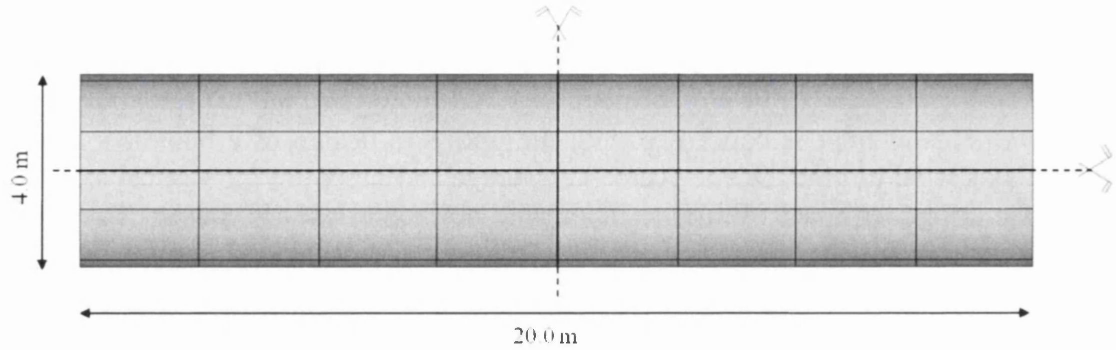


Figure 7.3: Dimensions of the full blade. It is 20m long and 4.0m wide. The blade and the loading is symmetrical with respect to two planes.

total height is 1.39m and the maximum vertical distance between the upper and lower surface is 0.64m. The cross-section is illustrated in Figure 7.4.

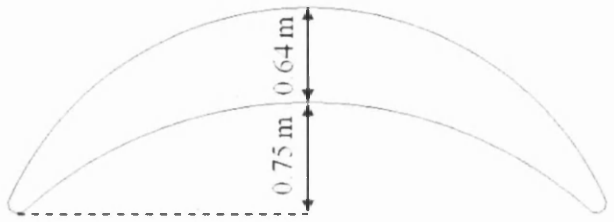


Figure 7.4: Dimensions of the blade cross-section. In total the blade is 1.39m high and 4.0m wide.

The objective of this work is the design of a blade based on the given cross-section which withstands service and extreme conditions without failure while exhibiting the lowest possible weight.

The most common type of wind turbines is the three bladed horizontal axis version shown in Figure 7.5. The blades are supported at one end only. A typical structural design for such a blade can be seen in Figure 7.6. It is mainly influenced by the required high bending stiffness to avoid contact with the tower and maintain the aerodynamic shape. This bending stiffness is mostly provided by two thick spar caps in the center of the blade that act like the flanges of an I-beam. Shear stresses are transferred by shear webs to the left and right of the spar caps. The outer skins contribute to the bending stiffness only to a small extent but govern the aerodynamic properties (Burton et al., 2001) (Gurit Holding AG (b), 2009) (Kong et al., 2005) (Veers et al., 2003).

If the blades were exclusively made from fibre reinforced polymers (FRP), the resulting structural members would be very thin. Hence, the structure would be flexible and

Figure 7.5: Common three bladed horizontal axis wind turbine. Image by dan-FreeDigitalPhotos.net

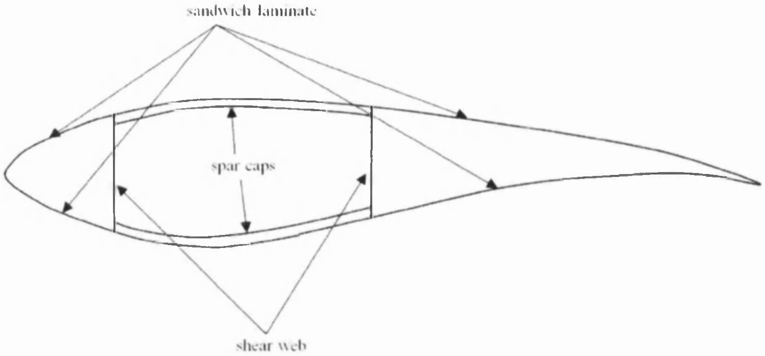


Figure 7.6: Common blade design for horizontal axis wind turbines. The bending stiffness is provided by monolithic spar caps while the aerofoil is made from sandwich laminates. Shear forces are transferred between the faces via two shear webs.

prone to buckling. To increase the buckling resistance without adding significant weight and cost, the laminates are usually made in a sandwich construction. The material for the core of the laminates is commonly low density foam or balsa wood (Gurit Holding AG (b), 2009) (Norlin & Reuterloev, 2002). Due to the high thickness of the spar caps the foam core is often omitted there (Gurit Holding AG (b), 2009). The most widely used FRP material in wind turbine blades is glass-fibre. With increasing blade dimensions though the use of carbon-fibre becomes more accepted (Gurit Holding AG (b), 2009). (Veers et al., 2003) define three fundamental ways to incorporate carbon-fibre in blade design. These are bulk replacement, selective reinforcement and total blade design. The advantages of carbon-fibre to glass-fibre are the higher elastic modulus (factor 3), the lower density (factor 0.7), higher tensile strength (factor 1.15) as well as the lower fatigue sensitivity (factor 1.6). The obvious disadvantage is the cost which is diminished to some extent since less material is required for the same structural performance and due to the resulting lower final weight which leads to smaller loading on the rest of the structure (Veers et al., 2003).

Vertical axis wind turbines have been investigated in the 1980s to a considerable extent but were considered not commercially competitive (Burton et al., 2001). Due to that fact little has been published on their design. A large Darrieus type vertical axis wind turbine is depicted in Figure 7.7.

Despite the differences between the studied blade and that in a common horizontal axis wind turbine, the broad knowledge available for the design of the latter will be used as a starting point in this study. In the present case the blades are supported on both sides

Figure 7.7: A Darrieus type wind turbine in Canada. Image by guillom-wikipedia.org.

and no tower contact constraint is necessary. This makes the importance of bending stiffness less significant and will influence the design accordingly. To account for the altered requirements various different material and shear web locations will be tested. This requires the creation of numerous simulation models. To reduce the required time an automated model creation will be employed.

In regular turbines the blades' contribution to the overall costs is around 10 to 15%. The benefit of an optimal design is increased by the fact that a lightweight blade means reduced loads on the remaining structure which reduces the costs of those (Veers et al., 2003). In the present case this is even more applicable since the structure contains eight blades rather than three. Thus, the use of optimization during design is a natural consequence.

This chapter is structured as follows: After this introduction, the design requirements are described, followed by some specific design guidelines for composite materials. In the next section the numerical model used to analyze the potential candidate design is presented. Some aspects about process automation of the design process are then given before the approach to the optimization of the structure is outlined. Finally, the results are presented and the conclusions drawn.

7.2. Design Requirements

Design standards for wind turbines started to evolve in the 1980s and a first set of rules was published in 1986 by Germanischer Lloyd (GL) and later refined as the knowledge increased. This led to the ‘Regulation for the Certification of Wind Energy Conversion Systems’. Furthermore, national standards were published in the Netherlands in 1988 and in Denmark in 1992. The International Electrotechnical Commission published a design standard in 1994 (Burton et al., 2001). Within this work design safety factors from the GL standard will be used.

The blade has to sustain loads in both service and extreme situations. In accordance with the GL a maximum wind speed of 30m/s for service conditions (blades are moving) and 73m/s for extreme conditions (blades are not moving) is considered. Computational Fluid Dynamics (CFD) calculations have been performed by C-Fec Ltd. for the two conditions and the three most critical pressure loadings identified. Two correspond to extreme and one to service conditions. The pressures have been averaged for the inside and outside surface, respectively. The critical pressure load for the service load case is illustrated in Figure 7.8 and those for the extreme conditions in Figure 7.9 and Figure 7.10. According to GL load safety factors have to be applied to the pressure loading. These are $SF_{LS} = 1.2$ for service conditions and $SF_{LE} = 1.5$ for extreme conditions.

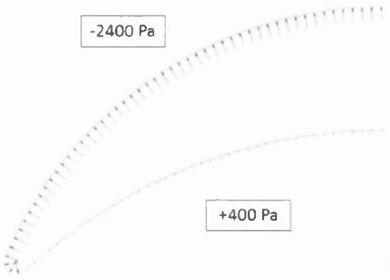


Figure 7.8: Averaged pressures for load step 1. This load step occurs during service, i.e. the blades are moving. The corresponding wind speed is 30m/s. A load safety factor of $SF_{LS} = 1.2$ will be applied.

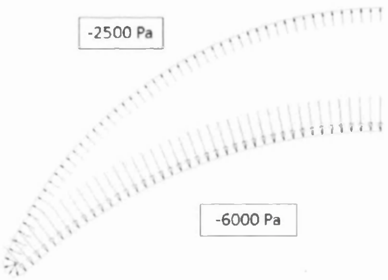


Figure 7.9: Averaged pressures for load step 2. This load step occurs during extreme conditions, i.e. the blades are not moving. The corresponding wind speed is 73m/s. A load safety factor of $SF_{LE} = 1.5$ will be applied.

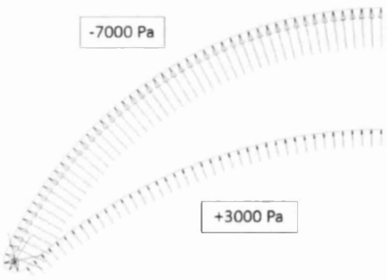


Figure 7.10: Averaged pressures for load step 3. This load step occurs during extreme conditions, i.e. the blades are not moving. The corresponding wind speed is 73m/s. A load safety factor of $SF_{LE} = 1.5$ will be applied.

In the present work a feasible design needs to fulfill the following requirements: It needs to withstand all loads without damage of the laminate or buckling of the structure. Furthermore, contact between the blades and the channel has to be avoided. The horizontal clearance between the blade and the channel wall is 79mm. Hence, the radial displacements of the blades' tip nodes have to be restricted to a value smaller than the clearance.

With regards to material failure GL requires the use of material safety factors during the design. For extreme conditions the material strength has to be reduced by the following factors: A general factor (SF_{MG} = *1.35*), an ageing factor (SF_{MA} = *1.5*), a high temperature factor (SF_{MT} = *1.1*), a manufacturing factor (SF_{MM} = *1.2* in case of hand lay-up laminates and SF_{MM} = *1.1* in case of automated manufacture) as well as a curing factor (SF_{MC} = *1.1* in case the laminate is not post cured). Thus, assuming hand lay-up and no post curing the global material safety factor for extreme loading accounts to SF_{Mextr} = *2.94*.

For service conditions the ageing factor is omitted and the manufacturing factor is replaced by one that takes the reduced material strength due to the repeated load cycle into account (Burton et al., 2001). A lifetime of a wind turbine of around 20 years corresponds to around 10 million load cycles, during which time the material strength of lower quality glass-fibre can be reduced to 20% (Gurit Holding AG (b), 2009). Hence, for service conditions a reduction of the material strength by a factor of five should be used to account for fatigue (Gurit Holding AG (b), 2009), (Burton et al., 2001). Thus, a fatigue safety factor SF_{MF} = *5.0* is added to the global safety factor. The global material safety factor for service conditions accounts to SF_{Mserv} = *8.17*.

Considering a projected surface area of $80m^2$ and a total pressure loading of 10,000Pa in load step 3 a horizontal force of 800kN is acting on the blade. An anticipated blade weight of 2t would induce a vertical force of 19.6kN mainly acting near the supports which corresponds to less than 2.5% of the pressure loading. Taking into account the high safety factors for both the loading and the material strength, the loading due to self weight can be safely neglected within this work.

7.3. Design Rules for Laminated Composite Materials

During the design of structures made from laminated composite materials the designer should follow some guidelines which help in creating safe and manufacturable designs. These are three sets of rules which affect the size optimization (sizing constraints), the translation of the continuous ply thicknesses (blending) and the ply stacking optimization (stacking rules). Commonly used rules found in the literature will be described in the following. Note that some of these rules only affect the case where the structure has been divided into individual patches.

7.3.1. Sizing Constraints

- A balanced number of +45°,-45° layers is desirable to minimize the likelihood of introducing manufacturing stresses such as torsion (Funnell, 2007) and to take damage tolerance under compressive loading into consideration (Nagendra et al., 1996)
- Large changes in thickness from patch to patch should be avoided (Patten, 2009)
- Maximum and minimum ply percentages in the overall laminate should be stated to ensure there are sufficient plies for the ply stacking rules, for instance the +45°,-45° layer cover (Patten, 2009)
- In many composite design optimizations a symmetric laminate layup. See for instance (Liu et al., 2010) or (Serestra et al., 2007)

7.3.2. Blending

One of the main benefits of composite materials is their flexibility, i.e. that they can be used in the most efficient way in a certain region of the structure. To allow for these varying laminate properties from one location to another various patches can be defined. When a size optimization is performed on such a patched laminate structure where each patch can exhibit different ply thicknesses no continuity exists between the patches, i.e. a discontinuous structure is created. This increases manufacturing costs and leads to unsafe discontinuities in the interfaces. The problem of integrating the different patches is typically referred to as blending problem. It has been first introduced by (Kristinsdottir et al., 2001) and has been subject of several recent publications. (Soremekun et al., 2002) use a two step approach. In the first step each the thicknesses in each patch are optimized without considering any blending rules. In the second step sub-laminates are defined and a ply stacking optimization is performed. (Serestra et al., 2007) propose a guide based approach. This means the optimization determines for each patch how many of the plies in the guide are used as well as the stacking sequence of the guide. Another approach has been discussed in (Liu et al., 2010). After the number of plies in each patch is obtained in a global optimization a shared layer blending method is applied to obtain plies that span over several patches.

The concept of defining plies that overlap several patch boundaries to provide the required integrity is typically referred to as the sub-laminate concept (Ghiasi et al., 2010). This step can also be performed manually in a post-processing step (Liu & Krog, 2008). This approach will be employed in this thesis. Due to the manual work involved in this method the designer can decide on the preference between a good performance in the objective function or good blending properties (Ghiasi et al., 2010).

7.3.3. Stacking Rules

- A laminate cover of a set of +45°,-45° plies should be enforced to obtain a damage tolerance and increase resistance to delamination (Patten, 2009)
- A number of successive plies that have the same angle should be smaller or equal to four to obtain damage tolerance, to increase resistance to delamination (Patten, 2009), to prevent excessive interlaminar stress (Wang et al., 2010) and to prevent matrix cracking (Kim, 2007)
- The angle difference between two adjacent plies should not exceed 45° to prevent matrix cracking and delamination (Kim, 2007)

7.4. Design Approach

The general approach to the blade design is based on that of regular horizontal axis wind turbine blades. Contrary to the asymmetrical cross-section of the latter here a symmetrical design can be assumed. Thus, the design obtained for one quarter of the blade can be mirrored along the symmetry axes.

(Kong et al., 2005) investigated the structural performance of wind turbines by altering key parameters such as skin thickness, spar flange thickness and width as well as the location of the shear webs. A similar approach will be used here. In addition to that, the use of different materials will be investigated and optimization performed.

To account for varying structural requirements within the blade, the outside surface of the quarter blade is divided into 20 sections. There are four divisions along the blade axis and five in transverse direction. The latter five can be subdivided into two divisions on the top, two on the bottom and one on the side. The widths of the top and bottom sections are governed by the location of the shear web. The width of the side section is constant with 0.13m. In addition to the outer surface sections an additional web section is created. Thus, a total of 21 sections for the quarter blade can be identified. They are illustrated in Figure 7.11. The sections are used to allow for different laminate thicknesses as well as different materials. The material of choice is a laminate with a low density foam core. Due to the fact that bending stiffness is not as crucial as for

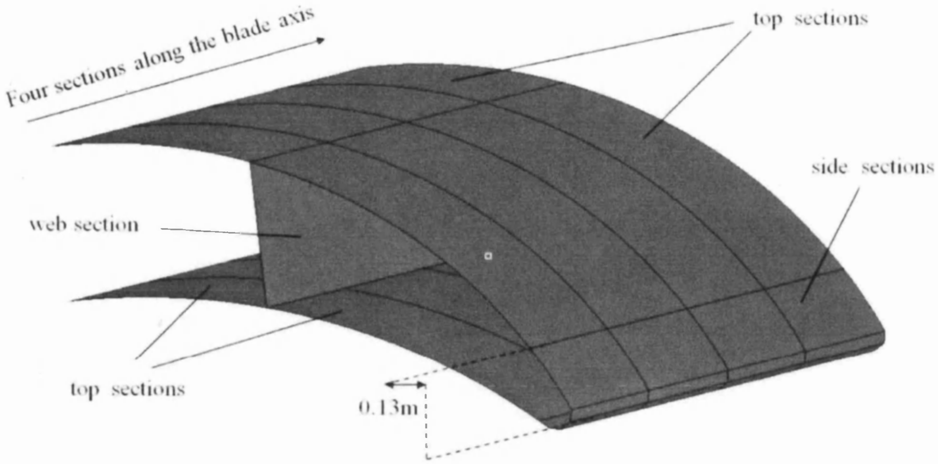


Figure 7.11: Illustration of the blade model. Due to symmetry only one quarter of the blade is modeled. The outer shell has been divided into 20 sections: Four along the blade axis and five across of which two lie on the top surface, two on the bottom and one on the side. Additionally, a web section has been created.

regular blades the ply thicknesses of the spar caps will not be as high. To provide sufficient buckling resistance in this area a foam core is used here as well. Due to the high curvature of the sides and the corresponding manufacturing difficulties, no foam will be used in these sections. Since the best location of the shear webs is unknown, four different possibilities are tried. The chosen horizontal distances are 0.2m, 0.8m, 1.6m and 2.4m. The corresponding cross-sections are shown in Figure 7.12, Figure 7.13, Figure 7.14 and Figure 7.15, respectively.

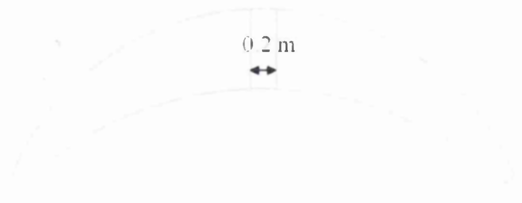


Figure 7.12: Topology of the cross-section for web setting 1. The distance between the webs is 0.2m.

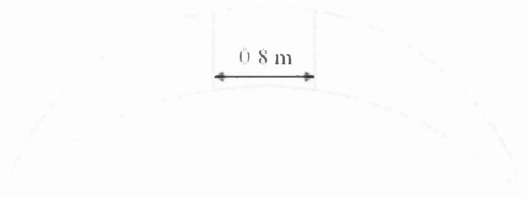


Figure 7.13: Topology of the cross-section for web setting 2. The distance between the webs is 0.8m.

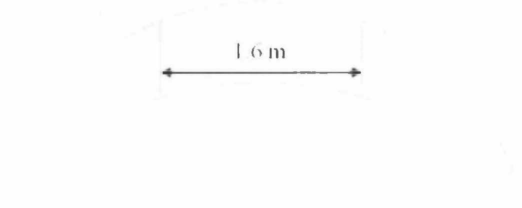


Figure 7.14: Topology of the cross-section for web setting 3. The distance between the webs is 1.6m.

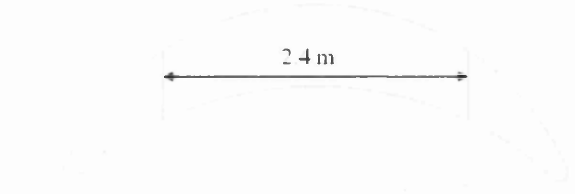


Figure 7.15: Topology of the cross-section for web setting 4. The distance between the webs is 2.4m.

In addition to the distance between the shear webs, the used materials are altered as well. First of all, two different foams will be investigated. Both are produced by Gurit Holding AG and are called T400 and T550. The density of the former is 71kg/m^3 and that of the latter is 104kg/m^3 (Gurit Holding AG (a), 2009). Their mechanical properties will be presented in Section 7.5.

Furthermore, four different settings for the FRP materials are used. In specific sections of the blade the influence of substituting the glass-fibre by carbon-fibre will be tested. The chosen FRP material for one of the transverse sections is the same along the blade axis. The different settings that will be investigated are summarized in Table 7.1. In total two different foams, four shear web settings and four FRP settings are investigated leading to a total number of 32 runs. Each run is a full size optimization which determines the optimal ply thicknesses in each of the 21 sections of the blade (see Section 7.7.1).

Table 7.1: Different FRP material settings being investigated. '*G*' indicates the use of glass-fibre and '*C*' carbon-fibre in the corresponding section. Each of the settings is run once for each of the two foams and for each of the four shear web settings. Thus, in total 32 runs will be performed, where each run is a full size optimization.

	top		**bottom**		**side**	**web**
combination	left	right	left	right		
1	*G*	*G*	*G*	*G*	*G*	*G*
2	*G*	*G*	*C*	*G*	*G*	*G*
3	*C*	*G*	*C*	*G*	*G*	*G*
4	*C*	*G*	*C*	*G*	*C*	*G*

Once these runs have been completed, the results are investigated and the best design will be chosen. In this case the best design is the lightest design. However, if two designs have comparable total masses, the price can influence the decision as well. Naturally, a design that requires less carbon-fibre is cheaper than one that does.

In the next step the continuous ply thicknesses of the chosen design have to be translated into discrete numbers of plies. Due to the large dimensions of the blade and the high loads the composites will have considerable thicknesses in which case the location of the plies within the stack have a significant influence on the structural performance. Hence, the optimum stacking sequence will be determined in the next step (see Section 7.7.3).

The flow of the proposed blade design is illustrated in Figure 7.16.

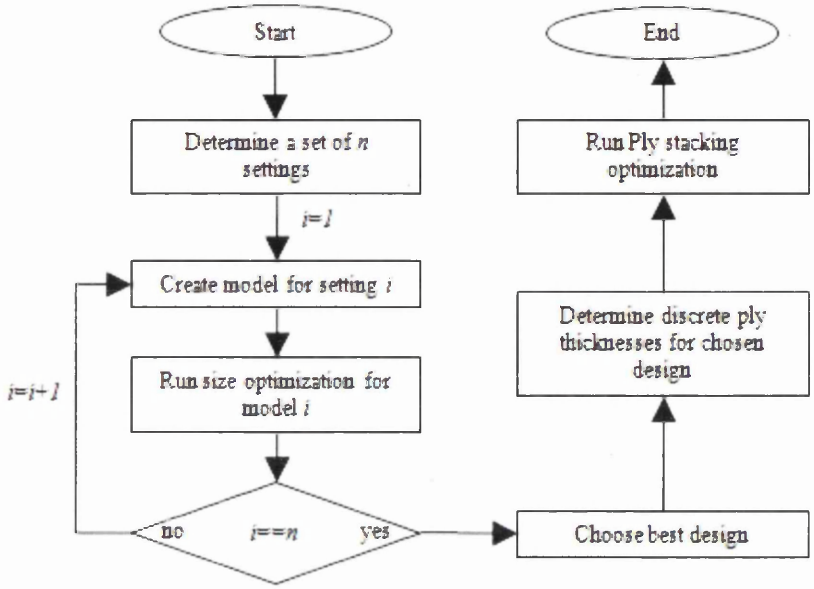


Figure 7.16:Flow of the proposed blade design.

7.5. Numerical Model

The blades are analyzed by means of the FE method. The geometry is meshed with linear triangular and quadrangular elements. Due to symmetry of the geometry and the loading only 1/4 of the blade is modeled. The model is created using Altair HyperMesh (Altair HyperWorks (a), 2009). A typical model is depicted in Figure 7.17. The model of the laminate contains four plies of either glass- or carbon-fibre oriented at -45°, 0°, 45° and 90° and one foam core ply oriented at 0°. The thicknesses of the plies can vary from section to section. The laminate structure is illustrated in Figure 7.18.

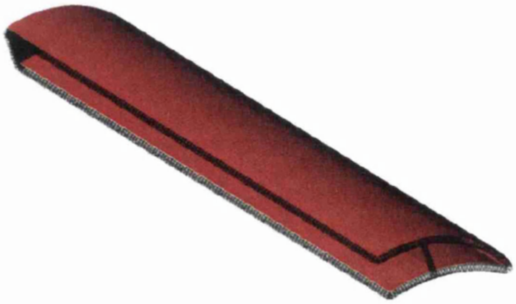


Figure 7.17: A typical FE model. Due to symmetry only 1/4 of the blade is modeled.

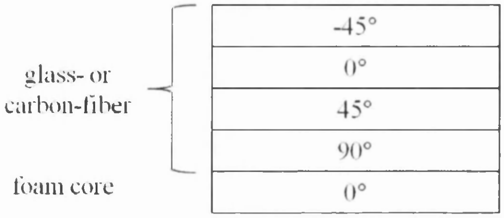


Figure 7.18: The model of the laminate contains four plies of either glass- or carbon-fibre oriented at -45°, 0°, 45° and 90°and one foam core ply oriented at 0°.

A linear isotropic material model is used for the core materials and a linear orthotropic model for the FRP materials. Two different foams are used within this study. The materials produced by Gurit Holding AG are called T400 and T550 (Gurit Holding AG (a), 2009). Their material properties are listed in Table 7.2.

Table 7.2: Material properties of the foams used as the laminate core. Properties from (Gurit Holding AG (a), 2009).Values in brackets are assumed.

	Unit	T400	T550
Density	*Kg/m³*	*71*	*104*
Compressive strength	*MPa*	*0.8*	*1.4*
Shear Strength	*MPa*	*0.7*	*1.1*
Tensile Strength	*MPa*	*1.2*	*1.8*
Compressive Modulus	*MPa*	*55*	*93*
Shear Modulus	*MPa*	*24*	*35*
Tensile Modulus	*MPa*	*94*	*147*
Poisson's ratio	-	*(0.25)*	*(0.25)*

Three different materials are used for the FRP. These are uni-directional (UD) glass–fibre reinforced polymer (GRP), UD carbon-fibre reinforced polymer (CRP) and woven GRP. The material properties are listed in Table 7.3.

The failure of the laminates can be measured by means of Failure Indices (FI). FIs indicate whether or not a composite failed at a certain location. Here FIs according to Hill were chosen which are calculated according to Equation 7.1

$$FI_{Hill} = \frac{{\sigma_1}^2}{X^2} - \frac{\sigma_1 \sigma_2}{X^2} + \frac{{\sigma_2}^2}{Y^2} + \frac{{\tau_{12}}^2}{S^2}, \qquad 7.1$$

where *X* is the allowable stress in ply material direction *1*, *Y* the allowable stress in ply material direction *2* and *S* the allowable in-plane shear stress. A distinction between strength for compression and tension is made and based on the sign of the stresses.

In order to assess whether or not buckling occurs anywhere in the structure a buckling Eigenvalue (EV) analysis is performed. Each EV represents the load factor which would lead to the corresponding buckling mode, i.e. the lowest EV for each load step can be seen as a safety factor against buckling.

Table 7.3: Material properties of the FRP used within the work. Properties for GRP from (Gay et al., 2003) and for CRP from Gurit Holding AG. Values in brackets are assumed.

	Unit	UD GRP	Woven GRP	UD CRP
Density	*Kg/m^3*	*2080*	*1900*	*1502*
Longitudinal tensile fracture strength	*MPa*	*1250*	*400*	*1430*
Longitudinal compressive fracture strength	*MPa*	*600*	*390*	*984*
Transverse tensile fracture strength	*MPa*	*35*	*400*	*32.5*
Transverse compressive fracture strength	*MPa*	*141*	*390*	*108*
In plane shear strength	*MPa*	*63*	*(30)*	*71.9*
Interlaminar shear strength	*MPa*	*(80)*	*(70)*	*90*
Longitudinal elastic modulus	*MPa*	*45,000*	*20,000*	*130,000*
Transverse elastic modulus	*MPa*	*12,000*	*20,000*	*7220*
Shear modulus	*MPa*	*4500*	*2850*	*4230*
Poisson's ratio	-	*0.3*	*0.13*	*0.337*
Available ply thickness	*mm*	*(0.3)*	*(0.22)*	*0.3*

7.6. Process Automation

Due to the scarce knowledge about the design of vertical axis wind turbines, the present design approach is based on that for horizontal axis turbines. Since the blade geometry and the blade loading exhibit significant differences, a wide variety of designs should be tested to find a good working design that accounts for those differences. If the designers were supposed to create each numerical model manually by using the GUI of the pre-processor for each variation of the design this would require a high effort. Hence, the decision on the development of an automated simulation tool has been made.

The phase that requires the testing of various different designs is the size optimization step (see Figure 7.16). Here, the various material and shear web settings are tried. In order to automate this step, first the model needs to be parameterized. The chosen parameters are the used materials, the location of the shear web as well as the ply thicknesses of each layer in each section. In terms of the material, the type of foam core and the type of FRP in each individual section of the blade can be varied. Since each run is a full size optimization the only parameters that require specification, are the shear web location and the material setting, i.e. for a certain set of parameters the run yields an optimum set of ply thicknesses. The approach to process automation as described in Section 4.5.4 is adopted here. To this end a FORTAN program has been developed that, given the required parameters, uses HyperMesh to automatically create the model and OptiStruct to solve the corresponding size optimization. The process of this automated procedure is illustrated in Figure 7.19 and can be described as follows: After the user specified the material setting and the shear web location, the program writes a script that contains commands for the creation of the geometry, the mesh and the materials in each section. In the next step HyperMesh is called in batch mode and processes this file. At the end of this process the mesh is exported and read by the tool. Now the nodes containing boundary conditions and the elements with pressure loading are identified. The commands for the boundary condition and load creation are written to another script. Furthermore, the load steps and the optimization problem are specified. Again, HyperMesh is called to process the file. This time, the complete FE model is exported which also contains the optimization problem to be solved by OptiStruct.

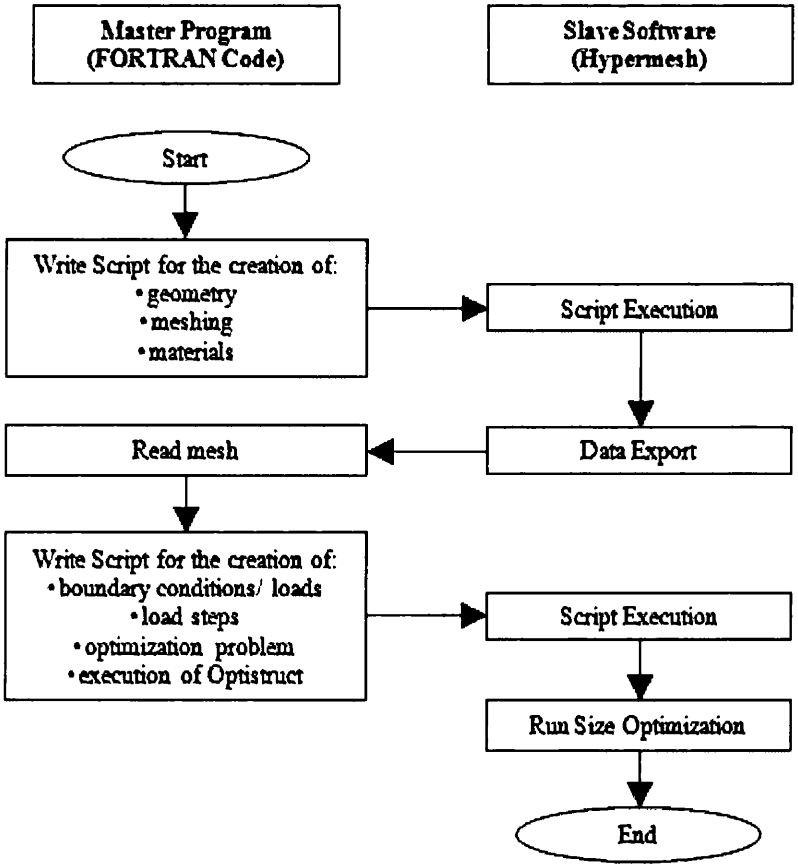


Figure 7.19:Flow of the Fortran tool that allows for the automated creation of size optimization models for the turbine blade depending on a set of material settings and the shear web location.

7.7. Optimization

The optimization can be divided into two steps: A size optimization and a ply stacking optimization. In the former the optimum laminate thicknesses are determined for each of the 21 sections in the blade. Here, a smear option is used, where the plies in the laminate are smeared to one continuous material which makes the stacking sequence of the plies irrelevant. In the latter, the optimum stacking sequence is determined. Both problems are solved by means of the built-in optimization tool in Altair OptiStruct (Altair HyperWorks (b), 2009).

7.7.1. Size Optimization

In the size optimization the design variables are the ply thicknesses. The thicknesses of the FRP are allowed to change for every section while the core thicknesses are forced to be constant along the blade axis. In case of the web section, all five ply thicknesses are constant along the blade axis. For the sides no foam is used. Hence, the total number of design variables is 89. They are summarized in Table 7.4.

Table 7.4: Design variables for the size optimization. For the top, bottom and side sections the FRP ply thicknesses can vary along the blade axis while the core thicknesses are constant. For the web section all ply thicknesses are constant along the blade axis. For the sides no foam is used. Hence, the total number of design variables is 89.

	top		**bottom**		**side**	**web**
ply	left	right	left	right		
FRP @ -45°	*4*	*4*	*4*	*4*	*4*	*1*
FRP @ 0°	*4*	*4*	*4*	*4*	*4*	*1*
FRP @ +45°	*4*	*4*	*4*	*4*	*4*	*1*
FRP @ +90°	*4*	*4*	*4*	*4*	*4*	*1*
Foam @ 0°	*1*	*1*	*1*	*1*	*0*	*1*

The objective function to be minimized in the size optimizations is the total mass of the blade. The constraints restrict the FIs to prevent material failure of the laminate. An FI greater than 1.0 indicates that the laminate failed. Using the service condition material safety factor $SF_{Mserv} = 8.17$ (see Section 7.2) the maximum FI is set to 0.122 while that in the extreme conditions is set to 0.34 due to the extreme condition material safety factor $SF_{Mextr} = 2.94$ (see Section 7.2).

To avoid buckling of the structure the minimum EV for each load step is constrained to 1.5 (safety factors of 1.5). Finally, the radial displacement of the blade's tip nodes is constrained to 53mm (safety factors of 1.5) to avoid contact between the blade and the channel wall. The constraints are summarized in Table 7.5.

Table 7.5: Constraints for the size optimization.

		Load step 1	**Load step 2**	**Load step 3**
FIs	*max*	*0.122*	*0.34*	*0.34*
buckling EVs	*min*	*1.5*	*1.5*	*1.5*
radial tip displacements	*max*	*53mm*	-	-

7.7.2. Translation of Continuous Ply Thicknesses

After the size optimization, the designer is faced with two problems. The first one is that of structural integrity. That is if a design is chosen that uses different FRP materials in different sections of the blade, no overlap of plies in transverse direction occurs (see Section 7.3.2). The other problem stems from the fact that the continuous thicknesses determined in the size optimization have to be translated into a finite number of plies, since the plies are only available at a specific discrete thickness. This translation leads to round off errors. If one would simply round up each time the blade would be overdesigned and vice versa. Both problems can be tackled by adding woven GRP wherever a thickness was rounded down. This woven GRP then provides the necessary structural integration and supports the structure in two perpendicular directions which eases the problem of filling the missing material due to the round off error.

7.7.3. Ply Stacking Optimization

When the laminates in a structure become thick, i.e. when they consist of a considerable number of plies, the stacking sequence of the plies has a significant influence on the structure's behaviour such as bending and torsional stiffness, buckling resistance and stresses and strains (Patten, 2009). In OptiStruct a 'smear-option' can be used during the sizing optimization. This option smears the composite into a homogenous material and thus makes the stacking sequence irrelevant. After the size optimization a separate ply stacking optimization can be performed which identifies the optimal stacking of the laminate. Hence, the design variables for this second optimization step are the individual plies and their stacking order. Since the stacking sequence has no influence on the blade's mass, in this step the objective is changed from minimizing the total mass to minimizing the maximum FI. The constraints stated in the size optimization (see Table 7.5) are considered here too. Furthermore, OptiStruct allows for the consideration of several stacking constraints. These are: Specification of a laminate core, specification of a laminate cover and a maximum number of successive plies with the same orientation. In order to follow the design guidelines presented in Section 7.3.3 a laminate cover of +45° (the woven fabric acts in +45° and -45° direction) and the maximum number of successive plies constraint will be added. It should be noted that due to the fact that in OptiStruct the core is specified via the ply orientation (Altair HyperWorks (b), 2009) the foam plies had to be distinguished from the FRP plies. Since for the foam an isotropic material model is used, the orientation of the corresponding plies has no influence on its structural performance. Hence a ply orientation of 15° has been chosen for the four foam material plies.

In (Liu et al., 2010) a smeared stiffness-based approach is employed. After the number of plies in each patch have been determined in the top level size optimization the detailed ply stacking is obtained in a local level optimization. Here the difference between the values of the computed lamination parameters from the top level optimization and those in the local level optimization are minimized. The advantage is that there is no need to check whether the strain or buckling constraints are satisfied in the local level as long as the lamination parameters match. The use of lamination parameters makes the ply stacking optimization very efficient and is proposed for future work.

7.8. Results

The proposed design approach will now be applied to the blade in question. The results from the size optimization step as well as the chosen design are discussed in Section 7.8.1. The translation of the continuous ply thicknesses obtained in the first step into discrete plies can be found in Section 7.8.2 and the optimization of the stacking sequence in Section 7.8.3.

7.8.1. Size Optimization

The resulting blade masses and the shares of the different materials from the 32 performed size optimizations are displayed in Figure 7.20. As expected, a general trend

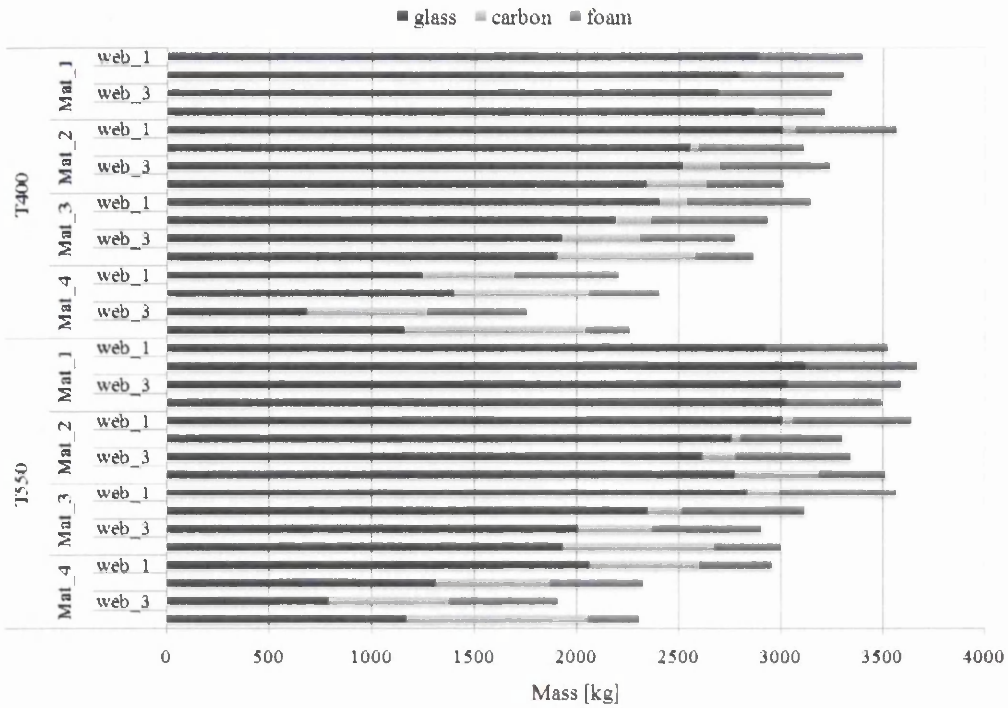


Figure 7.20: Resulting blade masses and their shares of glass, carbon and foam for the 32 size optimization runs. The final total mass varies between 1.76 and 3.68 tons.

can be observed that the total mass decreases with an increasing share of carbon-fibre. Furthermore, the lower density foam core generally leads to lighter designs.

The final total mass varies between 1.76 and 3.68 tons. The latter uses only glass-fibre for the FRP, uses the T550 foam core and has a horizontal distance between the webs of 0.8m. The former and lightest blade design substitutes the glass- by carbon-fibre in the left top and bottom sections as well as in the side sections. The used foam core is made from T400 and the horizontal distance between the shear webs is 1.6m.

Naturally, the use of carbon-fibre makes the design much more expensive. For instance a woven CRP can be 7.3 times as expensive as a woven GRP (Gay et al., 2003). Hence, if there was a slightly heavier design that uses less carbon, the latter could be the better choice. In Figure 7.21 the mass of the required carbon is plotted against the total mass of the blade for all 32 size optimizations. The two circled results represent the lightest and the second lightest design which has a lower carbon content than the first. If the latter was chosen, 127kg of carbon were saved but the total mass would increase by 443kg with respect to the former. Clearly, this is not an acceptable trade-off. Thus, the lightest design will be chosen. The chosen design is shown in Figure 7.22 and the corresponding continuous thicknesses obtained during the size optimization are listed in Table 7.6.

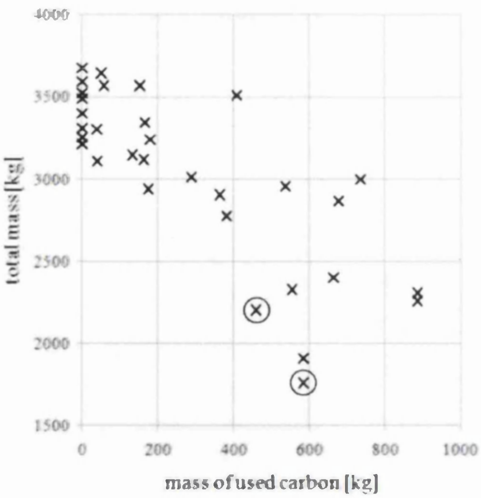


Figure 7.21: Mass of used carbon plotted against the total mass of the blade. The two circled results represent the lightest and the second lightest design which has a lower carbon content.

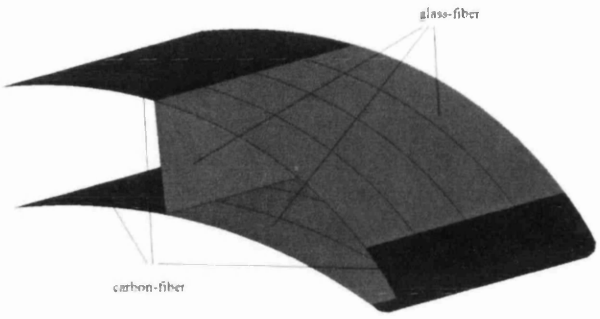


Figure 7.22: The chosen design uses carbon-fibre in the left top and bottom sections as well as in the side sections. The used foam core is made from T400 and the horizontal distance between the webs is 1.6m. Its total weight is approximately 1.76 tons.

The development of the objective function (the blade's total mass) as well as the maximum constraint violation are displayed in Figure 7.23. The optimizer required 31 iterations to converge. Due to the high initial thicknesses all constraints were satisfied in the first iteration. As the thicknesses were decreased the constrain violation rose up to its maximum in iteration 8. The blade's mass is further reduced after this point and the constraints eventually satisfied. This plot corresponds to the optimization leading to the chosen design.

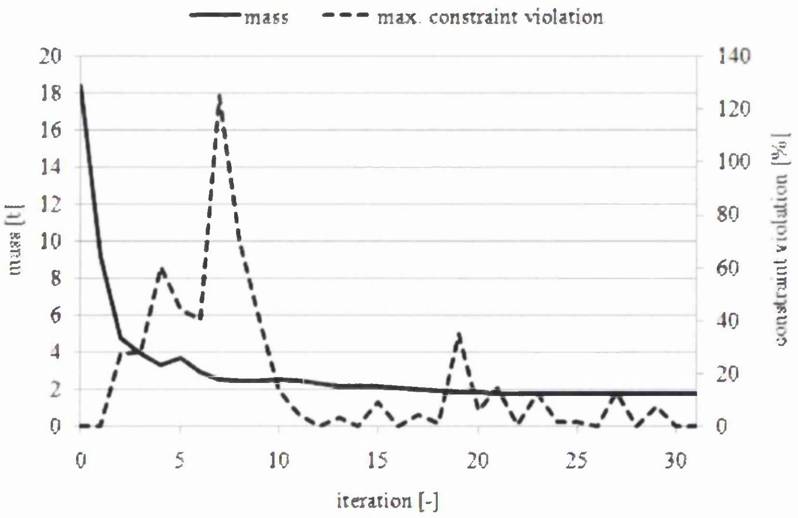


Figure 7.23: Development of objective function (the blade's total mass) and maximum constraint violation during size optimization of the chosen design.

Table 7.6: Continuous thicknesses [mm] for chosen design obtained in size optimization.

				FRP				Foam
	transv.	**longit.**	**material**	**-45°**	**0°**	**+45°**	**+90°**	**0°**
top	**left**	*1*	*CRP*	*0.18*	*1.18*	*0.25*	*5.08*	*53.34*
		2	*CRP*	*0.29*	*0.59*	*0.51*	*1.86*	*53.34*
		3	*CRP*	*0.31*	*0.63*	*0.14*	*0.23*	*53.34*
		4	*CRP*	*0.3*	*2.35*	*0.15*	*0.15*	*53.34*
	right	*1*	*GRP*	*0.91*	*0.09*	*1.45*	*2.05*	*36.4*
		2	*GRP*	*0.24*	*0.1*	*2.68*	*0.18*	*36.4*
		3	*GRP*	*0.46*	*0.1*	*2.64*	*0.09*	*36.4*
		4	*GRP*	*0.61*	*0.24*	*2.86*	*0.11*	*36.4*
bottom	**left**	*1*	*CRP*	*0.55*	*0.22*	*1.26*	*0.81*	*33.52*
		2	*CRP*	*0.17*	*0.13*	*0.1*	*0.68*	*33.52*
		3	*CRP*	*0.15*	*0.1*	*0.11*	*0.33*	*33.52*
		4	*CRP*	*0.48*	*0.42*	*0.6*	*0.45*	*33.52*
	right	*1*	*GRP*	*0.17*	*0.06*	*0.14*	*1.68*	*36.68*
		2	*GRP*	*0.15*	*0.07*	*0.21*	*2.02*	*36.68*
		3	*GRP*	*0.11*	*0.22*	*0.19*	*1.68*	*36.68*
		4	*GRP*	*1.34*	*0.08*	*0.14*	*0.12*	*36.68*
side	**-**	*1*	*CRP*	*0.43*	*3.77*	*0.36*	*3.13*	-
		2	*CRP*	*0.48*	*2.34*	*1.2*	*1.34*	-
		3	*CRP*	*0.49*	*5.4*	*0.69*	*0.18*	-
		4	*CRP*	*1.59*	*20.47*	*0.75*	*0.59*	-
web		-	*GRP*	*0.18*	*0.02*	*0.14*	*0.05*	*7.97*

7.8.2. Translation of Continuous Ply Thicknesses

After selecting the best design from the size optimization results the corresponding continuous ply thicknesses have to be translated to a multiple of plies at available thicknesses. As a rule of thumb, a number of 0.7 and higher is rounded up while numbers smaller than 0.7 are rounded down. The missing material in case of rounding down is complemented by woven GRP. As shown in Table 7.3 the UD plies are available at 0.3mm and the woven GRP at 0.22mm. That means a UD GRP ply with a thickness of 2mm would be translated to 6 plies at 0.3mm and complemented by one ply of woven GRP to fill up the missing material. If the thickness was 2.1mm the translation would yield 7 discrete plies and no woven GRP would be required. During the decision of how many plies of woven GRP are used to complement the UD plies the designer should keep in mind the smaller thickness and the inferior properties of the woven material.

This thickness translation is not necessary for the foam cores as the foam is available at custom thicknesses and hence can take any desirable value within accuracy limits. The final translated numbers of plies for each section are listed in Table 7.7.

Table 7.7: Discrete number of plies used for each section after translation. Note that the laminate layup is symmetrical, i.e. twice as many plies are used.

			Woven GRP		**UD FRP**			
	transverse	**longitudinal**	**+45°**	**0°**	**-45°**	**0°**	**+45°**	**+90°**
top	**left**	*1*	*4*	*4*	*0*	*2*	*0*	*8*
		2	*4*	*1*	*0*	*1*	*1*	*3*
		3	*5*	*3*	*0*	*1*	*0*	*0*
		4	*4*	*2*	*0*	*4*	*0*	*0*
	right	*1*	*3*	*3*	*2*	*0*	*2*	*3*
		2	*1*	*1*	*0*	*0*	*4*	*0*
		3	*1*	*2*	*1*	*0*	*4*	*0*
		4	*0*	*1*	*1*	*0*	*5*	*0*
bottom	**left**	*1*	*1*	*3*	*1*	*0*	*2*	*1*
		2	*3*	*2*	*0*	*0*	*0*	*1*
		3	*2*	*5*	*0*	*0*	*0*	*0*
		4	*0*	*6*	*1*	*0*	*1*	*1*
	right	*1*	*1*	*0*	*0*	*0*	*0*	*3*
		2	*1*	*1*	*0*	*0*	*0*	*3*
		3	*1*	*1*	*0*	*0*	*0*	*3*
		4	*1*	*1*	*2*	*0*	*0*	*0*
side	**-**	*1*	*5*	*3*	*1*	*6*	*0*	*5*
		2	*0*	*2*	*1*	*4*	*2*	*2*
		3	*1*	*3*	*1*	*9*	*1*	*0*
		4	*6*	*1*	*2*	*34*	*1*	*1*
web		-	*2*	*1*	*0*	*0*	*0*	*0*

It should be noted that due to software limitations, the ply-stacking optimization with a core ply can only be performed for a symmetrical layup (Altair HyperWorks (b), 2009), hence these numbers represent a laminate cover on both sides of the core.

The mass of the blade after translation from continuous thicknesses to discrete numbers of plies is 2344 kg. Before the translation the blade's mass was 1762 kg. This corresponds to an increase of 33%. This increase occurs for two reasons. Firstly, as explained before, during the translation some ply thicknesses are rounded up which adds a considerable mass to the structure. Secondly, whenever the ply thicknesses are rounded down, the missing material is filled by means of woven GRP. This has inferior properties to both UD CRP and UD GRP. Especially, when missing UD CRP is filled by the woven material more mass is required for the same structural performance.

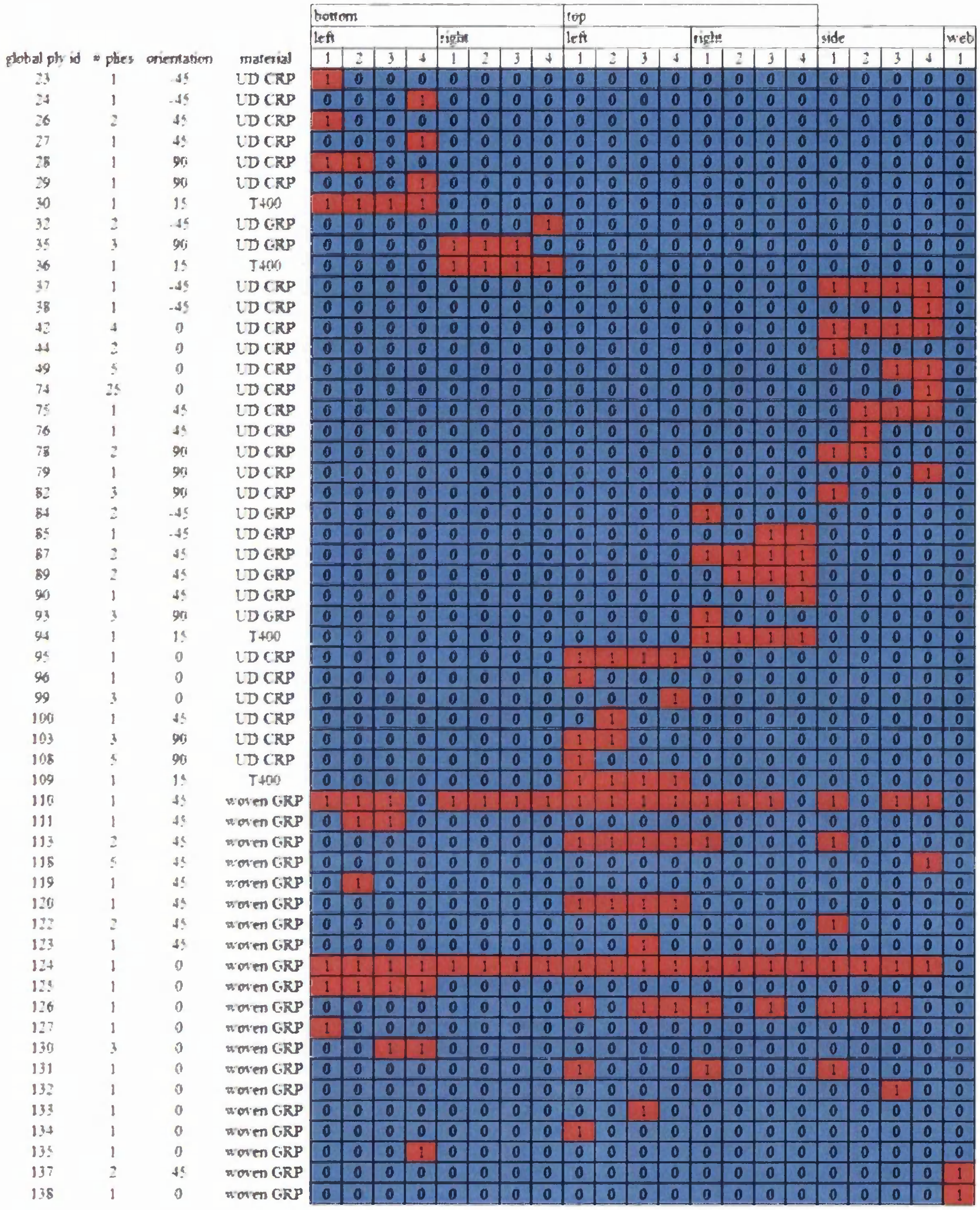
7.8.3. Ply Stacking Optimization

As mentioned before (see Section 7.3.2), for structural integrity, it is very important that some of the plies are continued across the patch boundaries. Hence, the concept of super plies is adopted here, i.e. plies are defined globally and can be larger than one section. Plies of similar material and orientation that share the same section borders are merged into one. This can lead to plies that span over numerous sections. For instance, the foam core was forced to be of constant thickness along the blade axis. Thus, for the foam cores, five global plies are defined that span along all four longitudinal sections. If one would not define one global ply, the foam layers for each section would end up in different locations along the blade's axis because the solver is not aware that they are connected. The defined global plies are listed in Table 7.8.

The optimization terminates after 34 iterations. The development of the objective function, i.e. the maximum FI and the minimum buckling EV for each load step, can be found in Figure 7.24. One can see that in the first iteration the buckling constraint was violated for load step 2 and that the FI constraint was violated up to iteration 25. After this step only the stacking rules are considered in the optimization. The fact that the FI constraint has been violated at a degree of around 100% in the first iteration highlights the importance of the correct stacking sequence of composite laminate materials.

The ply stacking for the 20 sections on the outside surface of the blade after each iteration of the ply stacking optimization is shown in Figure 7.25 and Figure 7.26. One can see that a +45° ply has been placed on the outside surface but this ply (ply id 25)

Table 7.8: The defined global plies. The red entries indicate that the ply occupies the corresponding section. In total 138 global plies have been defined. In the ply stacking optimization their location in the sequence are the design variables being optimized.



				bottom								top												
				left				right				left				right				side				web
global ply id	# plies	orientation	material	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1
23	1	-45	UD CRP	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	1	-45	UD CRP	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	2	45	UD CRP	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	1	45	UD CRP	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	1	90	UD CRP	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	1	90	UD CRP	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	1	15	T400	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	2	-45	UD GRP	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
35	3	90	UD GRP	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36	1	15	T400	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
37	1	-45	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
38	1	-45	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
42	4	0	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
44	2	0	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
49	5	0	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
74	25	0	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
75	1	45	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
76	1	45	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
78	2	90	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
79	1	90	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
82	3	90	UD CRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
84	2	-45	UD GRP	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
85	1	-45	UD GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
87	2	45	UD GRP	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
89	2	45	UD GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
90	1	45	UD GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
93	3	90	UD GRP	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
94	1	15	T400	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
95	1	0	UD CRP	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
96	1	0	UD CRP	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
99	3	0	UD CRP	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
100	1	45	UD CRP	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
103	3	90	UD CRP	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
108	5	90	UD CRP	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
109	1	15	T400	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
110	1	45	woven GRP	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	0
111	1	45	woven GRP	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
113	2	45	woven GRP	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	1	0	0	0	0
118	5	45	woven GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
119	1	45	woven GRP	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
120	1	45	woven GRP	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
122	2	45	woven GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
123	1	45	woven GRP	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
124	1	0	woven GRP	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
125	1	0	woven GRP	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126	1	0	woven GRP	0	0	0	0	0	0	0	0	1	0	1	1	1	0	1	0	1	1	1	0	0
127	1	0	woven GRP	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
130	3	0	woven GRP	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
131	1	0	woven GRP	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0
132	1	0	woven GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
133	1	0	woven GRP	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
134	1	0	woven GRP	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
135	1	0	woven GRP	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
137	2	45	woven GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
138	1	0	woven GRP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

occurs only in one patch. Furthermore, this patch is made from UD CRP while the woven ply that spans the entire outer surface (ply id 124) has been anticipated here, which would act in +45° and -45° orientation as required in the stacking rules described in Section 7.3.3.

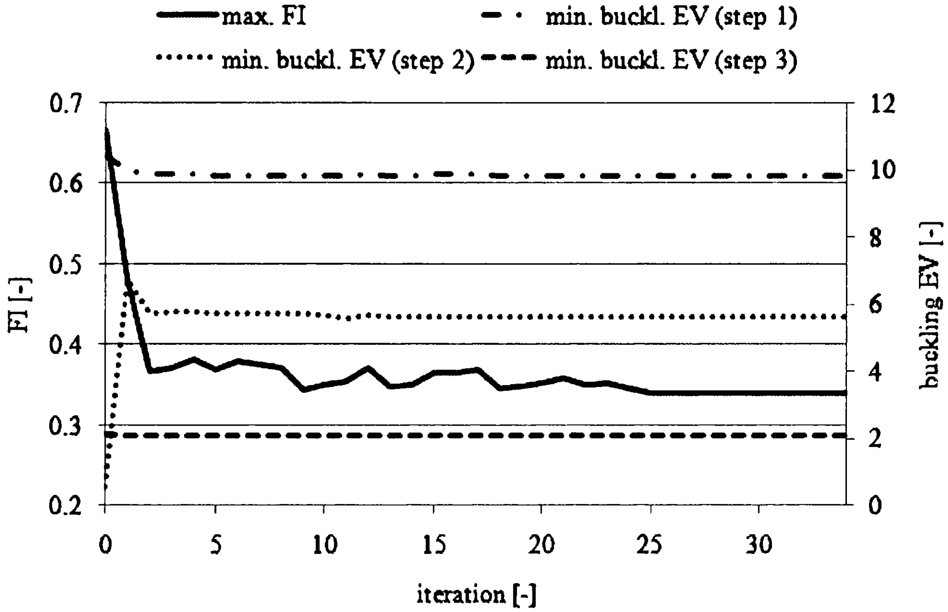


Figure 7.24: Development of objective function (minimization of the maximum FI) and the three buckling EVs during ply stacking optimization of the chosen design.

Hence, none of the patches exhibits the desired +/-45° cover. Obviously, in OptiStruct, the ply stacking rules are enforced for the global stack rather than the individual patches which is why for the solver the rule has been satisfied. The same applies to the rule that no more than four successive plies should have the same orientation. Studying the final stacking sequence in iteration 34 shown in Figure 7.26 OptiStruct has attempted to split up larger bundles in the global stack. But a satisfied rule for the global stack does not mean that the rule is satisfied on patch level. OptiStruct has not issued any warnings at the end of the optimization that the stacking rules have been violated. It should be noted that the stacking optimization is a feature in OptiStruct that has been added only in version 9.0 and that documentation is still scarce.

In order to obtain a design that fulfills the requirements stated in Section 7.3.3 it is recommended to place the woven ply oriented at 45° that spans the entire outside surface (ply id 124) on the outside and that, wherever more than four successive plies of the same orientation occur in a patch, the bundle is split by inserting a ply of different orientation. It is needless to say that this measure will add more weight to the final design. Contour plots of the total displacement after the ply stacking optimization for the three load cases are shown in Figure 7.27, Figure 7.28 and Figure 7.29.

Figure 7.25: The ply stacking for the outer surface for iterations 1 through 17. The different colors represent different ply orientations and each box represents one global ply.

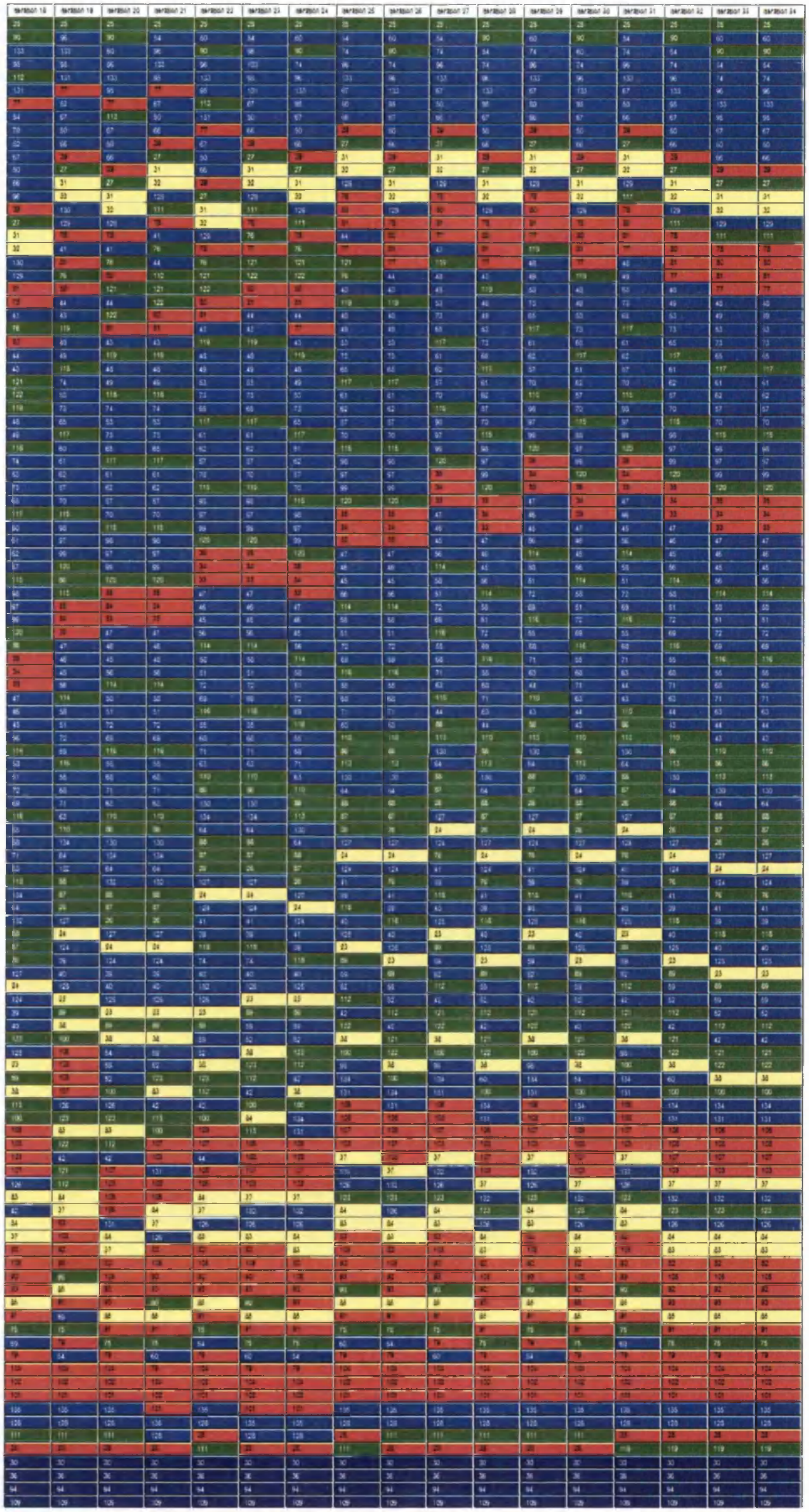


Figure 7.26: The ply stacking for the outer surface for iterations 18 through 34. The different colors represent different ply orientations and each box represents one global ply.

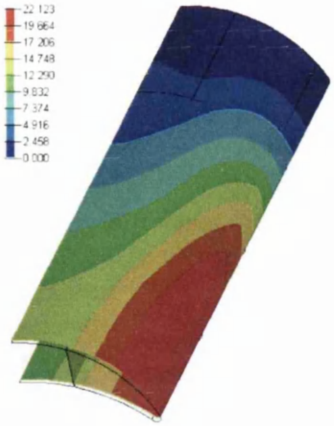


Figure 7.27: Contour plot of total displacements for load step 1. The maximum displacement is 22.1mm and the limit on radial displacement is 53mm. The deformed shape is exaggerated by the factor 3.

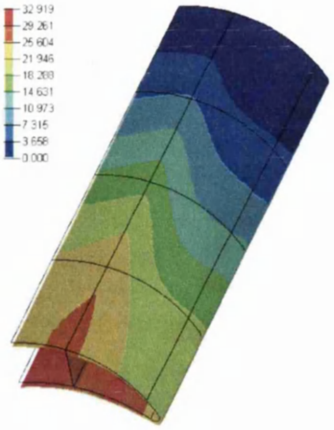


Figure 7.28: Contour plot of total displacements for load step 2. The maximum displacement is 32.9mm and there is no limit on radial displacement for this load step. The deformed shape is exaggerated by the factor 3.

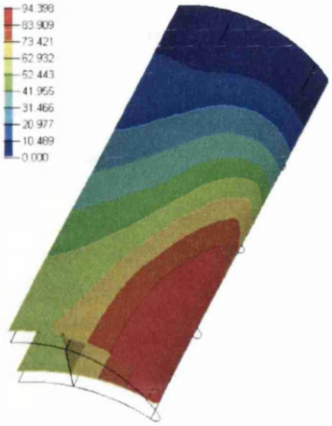


Figure 7.29: Contour plot of total displacements for load step 3. The maximum displacement is 94.4mm and there is no limit on radial displacement for this load step. The deformed shape is exaggerated by the factor 3.

The maximum total displacement for load step 1 equals to 22.1mm which is significantly lower than the maximum allowable value of 53mm in radial direction. Thus, no contact between blade and channel wall will occur during service conditions. The maximum total displacement for load step 2 is 32.9mm and that for load step 3 is 94.4mm. No displacement constraint had been stated for the latter two conditions. Considering a blade length of 20m, the maximum total displacement in extreme conditions, amounts to less than 0.5%.

The buckling modes for the first EV of the three load steps are illustrated in Figure 7.30, Figure 7.31 and Figure 7.32, respectively. The corresponding EVs are 9.83, 5.64 and 2.05. The lowest allowable EV was set to 1.5, thus the design can be considered as safe against buckling failure. Contour plots of the FIs for each load step are illustrated in Figure 7.33, Figure 7.34 and Figure 7.35. For service conditions the maximum occurring FI equals to 0.035. The upper limit for this load step was set to 0.122. For the extreme conditions, i.e. load steps 2 and 3, the maximum FIs are 0.337 and 0.34, respectively. The limit for these conditions was set to 0.34. Hence, the design can be considered as safe against material failure. Obviously, the FI constraint has been the driving constraint during the ply stacking optimization since this is the only active one (load step 3).

Figure 7.30: Contour plot of the first buckling mode for load step 1. The corresponding EV is 9.83 and the minimum allowable EV is 1.5. The buckling occurs in the web near the support. In case of buckling the maximum displacement is 143.5mm for this mode.

Figure 7.31: Contour plot of the first buckling mode for load step 2. The corresponding EV is 5.64 and the minimum allowable EV is 1.5. The buckling occurs on the top surface on the side of the blade. In case of buckling the maximum displacement is 104.1mm for this mode.

Figure 7.32: Contour plot of the first buckling mode for load step 3. The corresponding EV is 2.05 and the minimum allowable EV is 1.5. The buckling occurs in the web near the support. In case of buckling the maximum displacement is 145.6mm for this mode.

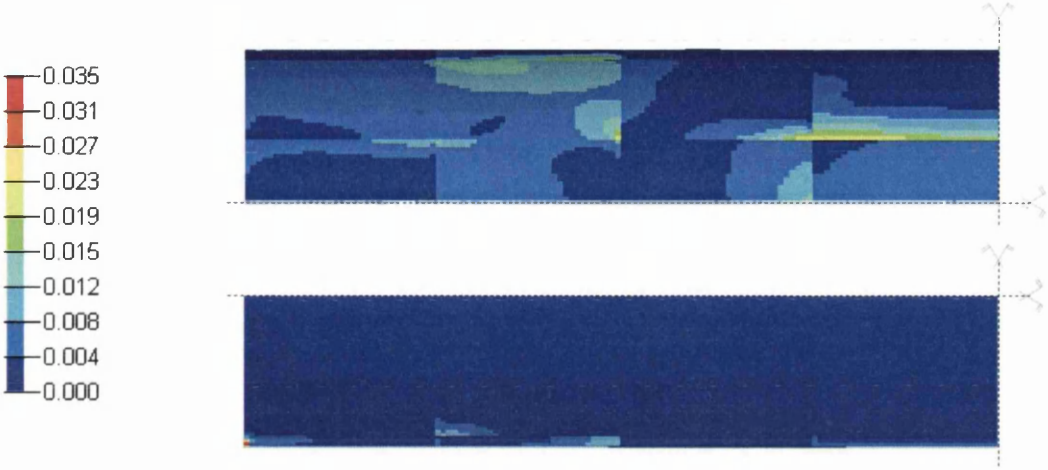


Figure 7.33: Contour plot of FIs for load step 1. The maximum FI equals to 0.035 and thus is lower than the limit of 0.122. The upper graph is the top and the lower one the bottom view.

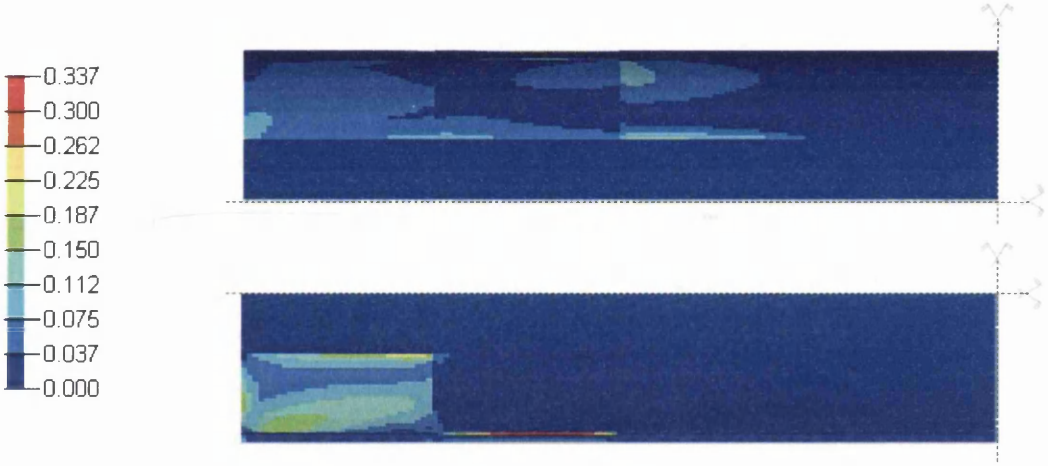


Figure 7.34: Contour plot of FIs for load step 2. The maximum FI equals to 0.199 and thus is lower than the limit of 0.34. The upper graph is the top and the lower one the bottom view.

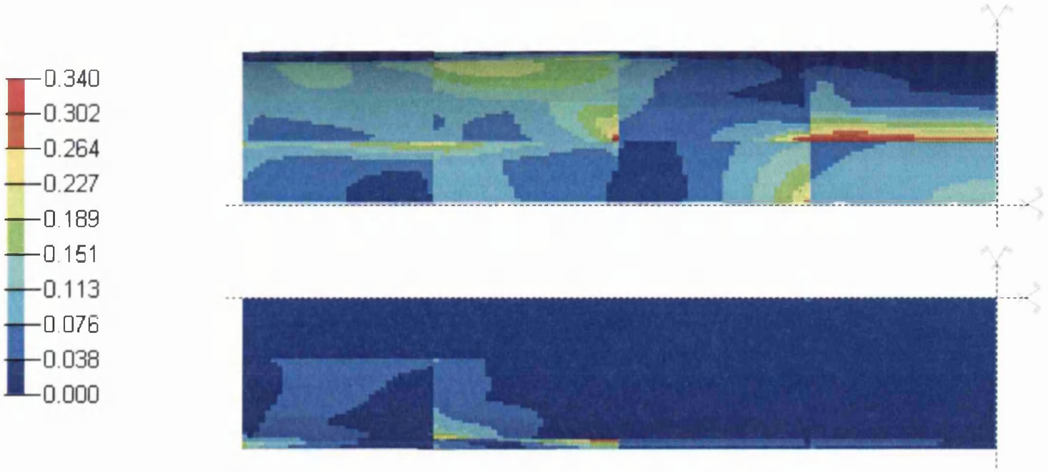


Figure 7.35: Contour plot of FIs for load step 3. The maximum FI equals to 0.34 and equals to the limit of 0.34. The upper graph is the top and the lower one the bottom view.

7.9. Concluding Remarks

A blade for a novel vertical axis wind turbine was sought to be designed. Due to scarce knowledge and literature for vertical axis wind turbines a design approach similar to those with horizontal axes was chosen. Material and load safety factors according to the GL were used during design. Preliminary CFD simulations were performed and three governing load cases identified (one during service conditions and two in extreme conditions).

Since the blade geometry and the blade loading exhibit significant differences, a wide variety of designs was tested to find a good working design that accounts for those differences. In order to limit the time for the model creation an automated simulation tool has been developed where each simulation is a full size optimization which determines an optimum set of ply thicknesses. It uses HyperMesh as the pre-processor and OptiStruct as the optimizer.

Using the developed tool, 32 size optimizations for different material settings and shear web locations were performed. The lightest blade satisfying all imposed constraints is a blade that uses a foam core made from T400 and uses carbon-fibre in the side and middle sections and glass-fibre in the other sections. The horizontal distance between the shear webs is 1.6m. The overall blade weight at this stage is 1762kg.

The continuous ply thicknesses obtained in the size optimization were then translated into multiples of available plies. For structural integrity and to reduce the impact of the round off error woven GRP was used to fill missing material when a ply thickness has been rounded down. The total weight of the blade after this step is 2344 kg which corresponds to an increase of 33%.

In a final step a ply stacking optimization has been performed. To this end, global plies have been defined that merge plies of similar material and orientation which share the same section boundaries. During the optimization the maximum constraint violation was eliminated from being close to 100% in the first iteration which highlights the importance of the correct stacking sequence of laminate materials. The final blade after the ply stacking optimization satisfies all requirements stated by the GL standard. Difficulties arose with the stated stacking rules. In OptiStruct, the ply stacking rules are enforced for the global stack rather than the individual patches which means the rule to have a laminate cover of a set of +45°,-45° plies and to avoid a number of successive plies that have the same angle could not be satisfied. For the design to fulfill the requirements it has been recommended to place the woven ply oriented at 45° that spans the entire outside surface (ply id 124) on the outside and to split bundles that contain more than four successive plies of the same orientation by inserting a ply of different orientation.

With the launch of OptiStruct 9.0 the so called 'ply-based-laminate-modeling' and 'ply-based-optimization' has been introduced. This concept allows for a much easier optimization process of composite structures. This process includes the following steps: Free element sizing, size optimization and ply stacking optimization. In the first step, i.e. free element sizing, each element in the FE model can exhibit different ply thicknesses. Based on the obtained results patches are identified automatically. The following size optimization optimizes the number of plies at the defined commercially available thickness. This procedure has been illustrated by (Patten, 2009). The application of this strategy would have several benefits in this case: The time-consuming manual translation of continuous ply thicknesses would be redundant, the increase in weight due to the rounding of the continuous values would be reduced and the automatically created patches would suit the requirements better than the defined rectangular ones. A disadvantage would be that the testing of varying materials in the form it has been done here would not be applicable.

8. Concluding Remarks

Summary: This chapter concludes this thesis by listing achievements, giving the main conclusions and proposing possible future work.

8.1. Achievements

Surrogate Based Optimization

- A formulation for the creation of OLHs for constrained design spaces has been developed. It uses an amended version of the permutation GA, that is the selective permutation GA
- 12 examples of OLHs with varying constraints and experiment numbers have been conducted
- The Kriging approximation method and a continuous GA have been implemented and were, together with the developed method for the creation of OLHs, formed into a framework for surrogate based optimization
- Two unconstrained and one constrained benchmark problems have been solved by means of the proposed framework for surrogate based optimization

Structured-Wall PE Pipes

- Numerous material tests in the form of DMTA and tension tests have been conducted to obtain data for the HDPE materials used by Asset Intl. Ltd. and KWH Pipe Ltd.
- A guideline has been developed that allows for the extraction of a secant modulus from the obtained material data based on the input parameters temperature, strain and strain rate
- A procedure has been implemented that allows for the extraction of a shell element representation of the pipe profiles from laser scan data or from profile geometries in dxf format. Another option for the creation of rectangular box sections has been implemented as well
- An automated simulation of the ringstiffness test has been set up that requires minimum user-interaction using FEAP as the FE solver. The results can be post-processed with Paraview

- An automated simulation has been set up that allows for the investigation of the most common pipe installation scenarios. For the simulation and the post-processing CANDE2007 is used
- An optimization feature has been developed and implemented that allows for the generation of optimum pipe cross-sections in the sense of minimal material expenditure. Here a simplified model using only one box-section and a horizontal spring support is used
- A Java based GUI has been developed which simplifies the use of the developed features of the tool. All the underlying computationally expensive tasks are solved by either in-house, open-source or public-domain programs
- Example problems for each of the offered features were presented. That is, 13 ringstiffness test simulations were conducted, two profile optimizations have been performed and three examples for the simulation of pipe installations have been discussed

Carbon-fibre Mountain Bike Frames

- Using B-splines a parameterized FE model has been created that allows for smoothly varying tube shapes and paths. Furthermore, different laminate ply thicknesses for each tube can be defined and the global geometry of the frame can be altered by means of the joint locations
- The automation of the simulation model creation has been realized by using a Fortran tool to write command files for HyperMesh and for the interpretation and processing of the created FE mesh
- The optimization problem is decomposed into a size optimization sub problem solved by OptiStruct and a shape optimization top level problem solved by means of surrogate based optimization where each experiment in the DoE is a full size optimization. The paths of the seat and chain stays have been extracted from the main optimization and are determined in a preliminary step
- The OLH for the surrogate model is created only in the feasible regions of the design space

Wind Turbine Blade

- An automated model creation tool has been developed where each run is a full size optimization that determines an optimum set of ply thicknesses for each of the defined sections in the blade. It uses HyperMesh as the pre-processor and OptiStruct as the optimizer

- The automated model creation tool has been used to perform 32 size optimizations for different material settings and shear web locations
- The continuous ply thicknesses obtained in the size optimization of the chosen design were translated into multiples of available ply thicknesses
- Global plies have been defined that merge plies of similar material and orientation which share the same section boundaries
- A ply stacking optimization has been performed for the defined global plies

8.2. Conclusions

Surrogate Based Optimization

- The examples of OLHs in constrained design spaces showed the validity of the chosen approach and that the selective permutation GA is capable of solving the underlying optimization problem. One shortcoming of the formulation has been identified though. That is, in some cases, areas whose parameter projections on several dimensions are unconstrained, can exhibit a sampling density that is lower than in areas where this is not the case. Nevertheless, in most cases the uniformity of the placed points can be considered good and naturally better than that of the feasible experiments in an OLH that has been created for the entire design space
- For large infeasible regions the space divisions become so close to each other that the NC is practically violated
- The functionality of the implemented framework for surrogate based optimization has been tested and validated on three benchmark problems., where one example was subject to constraints. The results of the latter illustrated the advantage of creating OLHs only in the feasible regions rather than in the entire design space
- The three benchmark problems showed that the SCVR is not an indicator on the accuracy of the optimum solution when a Kriging approximation is used. The SCVR measures how accurately function values are predicted when the corresponding experiment is not included in the model. Hence, it does

indicate the overall reliability of the model but not necessarily the accuracy of the function value in an arbitrary location in the design space

Structured-Wall PE Pipes

- The conducted material tests showed the high sensitivity of the pipe material to temperature, strain and strain rate and suggested that great care has to be taken during the ringstiffness tests in order to yield reproducible results
- In cases where the ringstiffness test is performed thoroughly an accuracy of the simulations of approximately 10% can be expected. When this is not the case errors of more than 20% can occur
- The developed simplified FE model used in the optimization procedure significantly reduced the simulation time which made the optimization computationally feasible in a manufacturing environment
- Application of the optimization feature suggested that material savings of 20% are possible
- Two main principles of the load bearing capabilities of flexible pipes could be validated by means of the automated simulation for the pipe installation. In a real life example the recorded maximum displacement of 3.8% could be reproduced with an accuracy of 13%
- The various applications showed that an efficient and easy to use software tool has been developed which addresses real life design problems in a manufacturing environment and that this has been achieved by coupling public domain and in-house code which effectively led to a software that requires no licensing fee

Carbon-fibre Mountain Bike Frames

- The dimensionality of the optimization problem has been successfully reduced by the decomposition into a size optimization sub problem and a shape optimization top level problem. To make the problem feasible to be solved by means of surrogate based optimization the paths of the seat and chain stays had to be extracted from the main optimization and are determined in a preliminary step. This step was appropriate because the paths are heavily constrained by surrounding bike parts that may not be obstructed by the frame

- A good compromise between efficiency and a high degree of parameterization could be achieved through the use of B-splines, the decomposition of the optimization problem and the coupling of in-house code and commercial software
- The automation of the model creation required a simplification of the frame's joints. Here rigid cylinders are created which envelope the joints to transfer the loads from one tube to the next
- Two examples showed that optimal bicycle frames with very good characteristics can be designed in an efficient way for different rider specific demands. Furthermore, it was shown that despite a very high-dimensional design space for the local shape design variables (in the examples it had 19 dimensions) a surrogate model based on 70 experiments proofed to be satisfactory in terms of accuracy. The reason for this are the obstruction constraints that constrain the sizes of the seat and chain stay. This was benefitted from by using an OLH that creates feasible experiments only

Wind Turbine Blade

- Through the use of an automated simulation procedure 32 designs with different material settings and shear web locations could be tested
- The use of carbon-fibre leads to a significantly lighter blade than one that is made from glass-fibre only
- The translation of the continuous ply thicknesses obtained in the size optimization into multiples of available plies increased the weight by 33% from 1762kg to 2344 kg
- During the optimization the maximum constraint violation was eliminated from being close to 100% in the first iteration. This highlights the importance of the correct stacking sequence of laminate materials
- The final design fulfills all requirements stated in the Germanischer Lloyd standard
- Due to the fact that in OptiStruct the ply stacking rules are enforced for the global stack rather than the individual patches the rule to have a laminate cover of a set of +45°,-45° plies and to avoid a number of successive plies that have the same angle could not be satisfied. For the design to fulfill the requirements it has been recommended to place the woven ply oriented at

45° on the outside and to split bundles that contain more than four successive plies of the same orientation by inserting a ply of a different orientation

8.3. Future Work

Surrogate Based Optimization

- The identified shortcoming in the developed procedure for the creation of OLHs in constrained design spaces could possibly be diminished by employing non-equidistant space divisions with larger spacing towards the constraint boundaries and vice versa
- From an engineering point of view it might be desirable to place experiments in the corners of the feasible region of the design space. For a 2D design space as in the current example this is a seemingly simple task since it is easy to illustrate. For more dimensions though, a search algorithm which finds the corner points automatically would be necessary

Structured-Wall PE Pipes

- While the results of thoroughly conducted ringstiffness tests at 23°C could be reproduced with an accuracy of around 10% this has still to be shown for different temperatures. Asset Ltd. conducted tests at different temperatures but here the pipes have not been stored at constant temperature for sufficiently long time or were tested directly after production

Carbon-fibre Mountain Bike Frames

- The paths of the seat and chain stays have been extracted from the main optimization and are determined in a preliminary optimization step. This step has been solved for an objective that requires weights for their degree of deviation from a straight line and for the paths' smoothness. A possible field for future work could be the investigation of these two weights by means of methods of multi-objective optimization
- While the step to extract the paths of the seat and chain stays from the main optimization problem was necessary for it to be solved by means of surrogate based optimization this step naturally impairs the final design of the frame. Other solutions could be thought of that do not lead to constant stay paths during optimization

- In the study for the design of the frame's joints the maximum mass has been chosen very low. A value closer to existing frames could possibly yield a higher SGI

Wind Turbine Blade

- The translation of the continuous ply thicknesses obtained in the size optimization into multiples of available plies increased the weight by 33%. This result could be improved by performing size optimizations which are based on discrete ply thicknesses. This is offered in the so called 'ply-based-laminate-modeling' and 'ply-based-optimization' which has been introduced with the launch of OptiStruct 9.0
- While the sectioning of the blade into rectangular sections was necessary for testing different material settings a free size optimization could lead to patterns of different shape and to an ultimately better design. Hence, solving the size optimization by using the 'ply-based-optimization' procedure is another field of possible future work
- The recommendations for the amendments of the final ply stacking to obtain a design that fulfills the stacking rules will result in a heavier blade. The development of another way of incorporating these rules in the ply stacking optimization is another interesting field to extend the work that has been done
- The approach to minimize the difference between the lamination parameters in the size optimization and in the ply stacking optimization means that no FE simulations need to be performed in the latter. Hence this optimization step becomes very efficient and is an interesting field for possible future work

References

Altair HyperWorks (a), 2009. *HyperMesh Manual.*

Altair HyperWorks (b), 2009. *OptiStruct Manual.*

Alvarez, L.F., 2000. *Design optimization based on genetic programming: Approximation model building for design optimization using the response surface methodology and genetic programming (PhD Thesis).* Bradford: University of Bradford.

Arora, J.S., 2007. *Optimization of Structural and Mechanical Systems.* Singapore: World Scientific Publishing Co. Pte. Ltd.

Asset International Ltd. (a), n.d. *Pipe Installation Guide.*

Asset International Ltd. (b), n.d. *Technical Manual.*

Audze, P. & Eglais, V., 1977. New approach for planning out of experiments. *Problems of Dynamics and Strengths*, pp.35:104-107.

Bäck, T., 1996. *Evolutionary Algorithms in Theory and Practice.* New York: Oxford University Press.

Barthelemy, J.F.M. & Haftka, R.T., 1993. Approximation Concepts for Optimum Structural Design - A Review. *STRUCTURAL OPTIMIZATION*, pp.5:129-144.

Bartholomew, P. & Morris, T., 2008. Industry Perceptions of Structural and Multidisciplinary Design Optimization. In *7th ASMO UK/ ISSMO conference on Engineering Design Optimization.*, 2008.

Bates, S., Sienz, J. & Langley, D., 2003. Formulation of the Audze–Eglais Uniform latin hypercube design of experiments. *Advances in Engineering Software*, pp.8:493-506.

Bates, S., Sienz, J. & Toropov, V., 2004. Formulation of the optimal latin hypercube design of experiments using a permutation genetic algorithm. *American Institute of Aeronautics and Astronautics.*

Berkeley University, 2005. *FEAPpv Manual.* [Online] Available at: http://www.ce.berkeley.edu/~rlt/feappv.

Bilgin, Ö., Stewart, H. & O'Rourke, T., 2007. Thermal and Mechanical Properties of Polyethylene Pipes. *Journal of Materials in Civil Engineering*, pp.19:1043-1052.

Bishop, C., 1995. *Neural Networks for Pattern Recognition.* Oxford: Oxford University Press.

Box, G.E.P. & Draper, N.R., 1987. *Empirical model-building and response surfaces.* New York: John Wiley & Sons.

Brent Strong, A., n.d. *Prepreg Materials – Fibers and Resins Exactly Right?* [Online] Available at: http://strong.groups.et.byu.net/pages/articles/articles/prepregs.pdf.

British Standards Institution, 1996. *BS EN 1446 : 1996 –Plastics piping and ducting systems – Thermoplastics pipes – Determination of ring flexibility.* London: British Standards Institution.

British Standards Institution, 1997. *BS EN 1610:1997 - Construction and Testing of Drains and Sewers.* London: British Standards Institution.

British Standards Institution, 1998. *Structural design of buried pipelines under various conditions of loading*. London: British Standards Institution.

Browning, T.R., 2001. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management*, 48, p.292–306.

Broyden, C..G., 1970. The Convergence of a Class of Double-rank Minimization Algorithms. *Journal of Applied Mathematics*, 6, pp.76-90.

Burton, T., Sharpe, D., Jenkins, N. & Bossanyi, E., 2001. *Wind Energy Handbook*. Chichester: John Wiley and Sons.

Calfee, C. & Kelly, D., 2002. *Bicycle Frame Materials Comparison with a Focus on Carbon Fiber Construction Methods*. [Online] Available at: http://www.calfeedesign.com/pdf/Calfee_TWP.pdf.

Chen, L., Ding, Z. & Li, S., 2005. A formal two-phase method for decomposition of complex design problems. *Journal of Mechanical Design*, 127, p.184–195.

Christensen, P.W. & Klarbring, A., 2008. *An Introduction to Structural Optimization*. Berlin: Springer.

Cook, H. & Skadron, K., 2007. *Genetically programmed response surfaces for efficient design space exploration*. [Online] Available at: http://www.eecs.berkeley.edu/~hcook/papers/cook_gprs_tr_aug07.pdf.

DeJong, K., 1975. *An Analysis of the Behaviour of a Class of Genetic Adaptive Systems*. Michigan: University of Michigan.

Di Lorenzo, R., Ingarao, G. & Chinesta, F., 2008. A gradient-based decomposition approach to optimize pressure path and counterpunch action in Y-shaped tube hydroforming operations. *The International Journal of Advanced Manufacturing Technology*, pp.44:49-60.

d'Ippolito, R., Masri, N.E., Hack, M. & Tzannetakis, N., 2009. Reliability-based Fatigue Life Optimization using Multibody Simulation. In *8th World Congress on Structural and Multidisciplinary Optimization.*, 2009.

Dorigo, M., Maniezzo, V. & Colorni, A., 1996. Ant system: Optimization by a colony of cooperating agents. In *IEEE Transactions On Systems Man And Cybernetics Part B-Cybernetics.*, 1996.

Duddeck, F., 2008. Size and Shape Optimisation for Car Body Design. In *7th ASMO UK/ ISSMO conference on Engineering Design Optimization.*, 2008.

EFBE Prüftechnik, n.d. *Company website*. [Online] Available at: http://www.efbe.de/.

European Standard, 2005. *Mountain-bicycles – Safety requirements and test methods*. Brussels: European Committee for Standardization.

FENet, 2005. [Online] Available at: www.fe-net.org.

Ferry, J., 1980. *Viscoelastic Properties of Polymers*. New York: John Wiley and Sons.

Finn, J., 2009. Application of Optimisation Tools to the Design of Advanced Carbon Fibre Bicycle: FACTOR 001. In *The 6th Altair CAE Technology Conference.*, 2009.

Fisher, R.A., 1935. *The Design of Experiments*. Edinburgh: Oliver and Boyd.

Forrester, A.I. & Keane, A.J., 2009. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45, p.50–79.

Funnell, M., 2007. Targeting Composite Wing Performance – Optimum Location of Laminate Boundaries. In *Altair Engineering - CAE Technology Conference.*, 2007.

Gassara, F. et al., 2008. Optimization of springback in L-bending process using a coupled Abaqus/Python algorithm. *The International Journal of Advanced Manufacturing Technology*, pp.44:61-67.

Gay, D., Hoa, S.V. & Tsai, S.W., 2003. *Composite Materials - Design and Applications.* Boca Raton: CRC PRESS.

Ghiasi, H., Fayazbakhsha, K., Pasinia, D. & Lessarda, L., 2010. Optimum stacking sequence design of composite materials Part II: Variable stiffness design. *Composite Structures.*

Goldberg, D.E., 1989. *Genetic algorithms in Search, Optimization and Machine Learning.* Reading, Massachusetts: Addison-Wesley.

Grefenstette, J.J. & Fitzpatrick, J.M., 1985. Genetic search with approximate function evaluations. In *First International Conference on Genetic Algorithms and Their Applications.* New Jersey, 1985. L. Erlbaum Inc.

Gurit Holding AG (a), 2009. *Corecell T-Foam - Structural Foam Core.* [Online] Available at: http://www.gurit.com/core/core_picker/download.asp?documenttable=libraryfiles&id=880.

Gurit Holding AG (b), 2009. *Wind Energy Composite Materials Handbook.* [Online] Available at: http://www.gurit.com/page.asp?section=0001000100080019&sectionTitle=Wind+Energy+Composite+Materials+Handbook.

Haftka, R.T. & Gürdal, Z., 1992. *Elements of Structural Optimization.* Dordrecht: Kluwer.

Hall, C., 1989. *Polymer Material – An Introduction for Technologists and Scientists.* London: Macmillan Education Ltd.

Hamida, S. & Schoenauer, M., 2002. ASCHEA: New Results Using Adaptive Segregational Constraint Handling. In *IEEE Congress on Evolutionary Computation.*, 2002.

Haupt, R.L. & Haupt, S.E., 2004. *Practical Genetic Algorithms.* New Jersey: Wiley.

Hilmann, J., Paas, M. & Schindler, V., 2005. Automatic Geometry Based FE Model Generation for Optimization and Robust Design for Passive Safety of Passenger Vehicles. *PAMM*, pp.5:7-10.

Hinton, E., Sienz, J. & Özakça, M., 2003. *Analysis and Optimization of Prismatic and Axisymmetric Shell Structures.* London: Springer.

Holland, J.H., 1975. *Adaptation in Natural and Artificial Systems.* Michigan: University of Michigan Press.

Iman, R.L. & Conover, W.J., 1980. Small sample sensitivity analysis techniques for computer models, with an application to risk assessment. *Communications in Statistics Part A. Theory and Methods*, p.17:1749–82.

Janson, L.E., 2003. *Plastics Pipes for Water Supply and Sewage Disposal.* Stockholm: Borealis.

Jin, R. & Chen, W., 2001. Comparative studies of metamodelling techniques under multiple modelling criteria. *Structural And Multidisciplinary Optimization*, pp.23:1-13.

Johnson, M.E., Moore, L.M. & Ylvisaker, D., 1990. Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, pp.26:131-148.

Jones, D.R., Schonlau, M. & Welch, W.J., 1998. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization*, pp.13:455-492.

Kawai, M. & Koizumi, M., 2007. Nonlinear Constant Fatigue Life Diagrams for Carbon/Epoxy Laminates at Room Temperature. *Composites Part A: Applied Science and Manufacturing*, pp.38:2342-2353.

Keane, A.J. & Nair, P.B., 2005. *Computational approaches for aerospace design: The pursuit of excellence*. Chichester: John Wiley & Sons.

Kennedy, J. & Eberhart, R.C., 1995. Particle swarm optimization. In *IEEE International Conference on Neural Networks, Piscataway, NJ.*, 1995.

Kim, J.S., 2007. Development of a user-friendly expert system for composite laminate design. *Composite Structures*, 79(1), pp.76-83.

Kirkpatrick, S., Gelatt, C.D. & Vecchi, M.P., 1983. Optimization by Simulated Annealing. *SCIENCE*, pp.220:671-680.

Kitware Inc., 2008. *Paraview Documentation*. [Online] Available at: http://www.paraview.org/paraview/help/documentation.html.

Kodiyalam, S. & Sobieszczanski-Sobieski, J., 2000. Bilevel integrated system synthesis with response surfaces. *AIAA Journal*, 38, p.1479–1485.

Kona Bikes, n.d. *Company Website*. [Online] Available at: www.konaworld.com.

Kong, C., Bang, J. & Sugiyama, Y., 2005. Structural investigation of composite wind turbine blade considering various load cases and fatigue life. *ENERGY*, pp.30:2101-2114.

Koza, J.R., 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge: Cambridge: MIT Press.

Krishnamachari, R.S. & Papalambros, P.Y., 1997. Hierarchical decomposition synthesis in optimal systems design. *Journal of Mechanical Design*, 119, p.448–457.

Kristinsdottir, B.P., Zabinsky, Z.B., Tuttle, M.E. & Neogi, S., 2001. Optimal design of large composite panels with varying loads. *Composite Structures*, 51, pp.93-102.

Kruger, H. & Rolfes, R., 2010. A Physically Motivated and Layer-based Fatigue Concept for Fiber-Reinforced Plastics. In *10th International Conference on Computational Structures Technology*. Valencia, 2010.

Kusiak, A. & Larson, N., 1995. Decomposition and representation methods in mechanical design. *Journal of Mechanical Design*, 117, p.17–24.

Kusiak, A. & Wang, J., 1993. Decomposition of the design process. *Journal of Mechanical Design*, 115, p.687–693.

Lancaster, P. & Salkauskas, K., 1981. Surfaces generated by moving least squares methods. *Mathematics of Computation*, 37, p.141–158.

Lee, D.-C. & Lee, J.-I., 2003. A structural optimization design for an aluminum-intensive vehicle. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, pp.217:771-779.

Lessard, L., Nemes, J. & Lizotte, P., 1995. Utilization of FEA in the design of composite bicycle frames. *Composites*, pp.26:72-74.

Liefvendahl, M. & Stocki, R., 2006. A study on algorithms for optimization of latin hypercubes. *Journal of Statistical Planning and Inference*, pp.136:3231-3247.

Liu, D., Barton, D.C., Toropov, V.V. & Querin, O.M., 2010. Optimum Stacking Sequence Design of a Blended Laminated Wing Structure using Permutation Genetic Algorithms (permGA). In *Tenth International Conference on Computational Structures Technology.*, 2010.

Liu, W. & Krog, L., 2008. A Method for Composite Ply Layout Design and Stacking Sequence Optimization. In *7th ASMO UK/ ISSMO conference on Engineering Design Optimization.*, 2008.

Loui, R.P., 2008. In praise of scripting: Real programming pragmatism. *COMPUTER*, pp.41:22-26.

McKay, M.D., Beckman, R.J. & Conover, W.J., 1979. A comparison of three methods for selecting values of input variables in the analysis of output from computer code. *Technometrics*, pp.21:239-245.

Mencaglia, X., Goode, D. & Lani, F., 2009. Multi-disciplinary Optimisation of Aerospace and Aircraft Components including Draping Process Aspects. In *8th World Congress on Structural and Multidisciplinary Optimization.*, 2009.

Merkel, M. & Schumacher, A., 2003. An Automated Optimization Process for a CAE Driven Product Development. *Journal of Mechanical Design* , pp.125:694-700.

Michelena, N.F. & Papalambros, P.Y., 1995. A network reliability approach to optimal decomposition of design problems. *Journal of Mechanical Design*, 117, p.433–440.

Michelena, N.F. & Papalambros, P.Y., 1997. A hypergraph framework for optimal model-based decomposition of design problems. *Computational Optimization and Applications*, 8, p.173–196.

Michel, S., Kieselbach, R. & Martens, H., 2005. Fatigue strength of carbon fibre composites up to the gigacycle regime (gigacycle-composites). *International Journal of Fatigue*, pp.28:261-270.

Mlynarski, M., Katona, M. & McGrath, T.(., 2007. *CANDE-2007 - Solution Methods and Formulations.* National Cooperative Highway Research Project.

Mlynarski, M., Katona, M. & McGrath, T.(., 2007. *CANDE-2007 - User Manual and Guideline.* National Cooperative Highway Research Project.

Moore, I. & Hu, F., 1996. Linear Viscoelastic Modelling of Profiled Density Polyethylene Pipe. *Canadian Journal of Civil Engineering*, pp.23:395-407.

Moser, A.P., 2001. *Buried Pipe Design.* New York: McGraw-Hill.

Motor Presse Stuttgart GmbH & Co. KG, n.d. *Magazine website.* [Online] Available at: http://www.mountainbike-magazin.de.

Nagendra, S. et al., 1996. Improved genetic algorithm for the design of stiffened composite panels. *Computers & Structures*, 58(3), pp.543-55.

Nelson, R., 2003. Bike frame races carbon consumer goods forward. *Reinforced plastics*, pp.47:36-40.

Norlin, P. & Reuterloev, S., 2002. The role of sandwich composites in turbine blades. *Reinforced Plastics*, pp.46:32-34.

Ousterhout, J.K., 1998. Scripting: Higher-Level Programming for the 21st Century. *COMPUTER*, pp.31:23-30.

Panda, S. & Manohar, C., 2009. Applications of meta-models in finite element based reliability analysis of engineering structures. *Computer Modeling in Engineering and Science*, pp.814:1-24.

Patten, S., 2009. Targeting Composite Wing Performance - Optimising the Composite Lay-Up Design. In *The 6th Altair CAE Technology Conference.*, 2009.

Prechelt, L., 2000. An empirical comparison of seven programming languages. *COMPUTER*, pp.33:23-29.

Queipo, N.V. et al., 2005. Surrogate-based analysis and optimization. *Progress In Aerospace Sciences*, pp.41:1-28.

Renaud, J.E. & Gabriele, G.A., 1991. Sequential global approximation in non-hierarchic system decomposition and optimization. *Advcances in Design Automation*, 32, p.191–200.

Sacks, J., Welch, W.J., Mitchell, T.J. & Wynn, H.P., 1989. Design and analysis of computer experiments. *Statistical science*, pp.4:409-435.

Samaras, V., Andtbacka, A., Webb, M. & Sienz, J., 2008. "The Strength of Flexibility" The Use of FEM Analysis for the Determination of Performance of Large Diameter HDPE Pipes under Buried Conditions. Budapest, 2008.

Santner, T.J., Williams, B.J. & Notz, W.I., 2003. *The design and analysis of computer experiments*. London: Springer.

Schoofs, A.J.G., Houten, M.H.v., Etman, L.F.P. & Campen, D.H.v., 1997. Global and Mid-Range Function Approximation for Engineering Optimization. *Mathematical Methods of Operations Research*, pp.46:335-359.

Schramm, U., 2009. Optimization Processes for Aerospace Structures. In *8th World Congress on Structural and Multidisciplinary Optimization.*, 2009.

Schroeder, M. & Callahan, L., 2005. *Taking the Witchcraft out of Advanced Composite Bicycle Frames*. [Online] Available at: http://www.specialized.com/OA_MEDIA/pdf/Witchcraft.pdf.

Sepe, M.P., 1998. *Dynamic Mechanical Analysis for Plastics Engineering*. Norwich, USA: Plastics Design Library.

Serestra, O., Gürdal, Z., Adams, D.B. & Watson, L.T., 2007. Optimal Design of Composite Wing Structures with Blended Laminates. *Composites Part B: Engineering*, 38, pp.469-80.

Sexton, M., 2010. *Design and Optimisation of Joints in a Carbon Fibre Bicycle Frame*. Swansea: Swansea University.

Shan, S. & Wang, G.G., 2009. Survey of modeling and optimization strategies to solve high-dimensional design problems with computationally-expensive black-box functions. *Structural and Multidisciplinary Optimization*, 41, pp.219-41.

Shewry, M. & Wynn, H., 1987. Maximum entropy sampling. *Journal of Applied Statistics*, pp.14:165-170.

Sienz, J., Fuerle, F., Innocente, M. & Pittman, J., 2008. *Numerical Simulation of the Ringstiffness Test For Polymeric Pipes (Consultancy Report).*

Sienz, J. & Hinton, E., 1993. *SWAN - General User Manual.* Swansea University.

Simpson, T.W. et al., 2004. Approximation methods in multidisciplinary analysis and optimization: a panel discussion. *Structural And Multidisciplinary Optimization*, pp.27:302-313.

Sobieszczanski-Sobieski, J. & Haftka, R.T., 2000. Multidisciplinary aerospace design optimization. *Structural Optimization* , 14, p.1–23.

Soremekun, G., Gurdal, Z., Kassapoglou, C. & Toni, D., 2002. Stacking sequence blending of multiple composite laminates using genetic algorithms. *COMPOSITE STRUCTURES* , 56, pp.53-62.

Specialized Bicycle Components, n.d. *Company Website.* [Online] Available at: www.specialized.com.

SSIS Marketing International (Pty) Ltd, n.d. *CandeCAD - Introduction.* [Online] Available at: http://www.ssismint.com/buynow/candecad.asp.

Stinstra, E., Stehouwer, P., Hertog, D.D. & Vestjens, A., 2003. Constrained maximin designs for computer experiments. *Technometrics*, pp.45:340-346.

The European Plastics Pipe and Fitting Association, 1999. *Design of Buried Thermoplastics Pipes.* TEPPFA.

Toropov, V.V., 1998. *Development of MARS - Multipoint Approximation method based on the Response Surface fitting.* [Online] Available at: http://www.brad.ac.uk/staff/vtoropov/burgeon/b_mars.htm.

Toropov, V.V. & Alvarez, L.F., 1998. Approximation model building for design optimization using genetic programming methodology. In *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis.*, 1998.

Toropov, V.V., Filatov, A.A. & Polynkin, A.A., 1993. Multiparameter structural optimization using FEM and multipoint explicit approximations. *STRUCTURAL OPTIMIZATION*, pp.6:7-14.

Transportation Research Board, 2007. *CANDE 2007 Documentation and Installation CD.* [Online] Available at: http://www.trb.org/NotesDocs/NCHRP15-28_CANDE-2007.iso.

Trek Bicycle Corporation, n.d. *Company Website.* [Online] Available at: www.trekbikes.com.

Trelea, I., 2003. The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection. *INFORMATION PROCESSING LETTERS*, 85(6), pp.317-25.

van Keulen, F. & Toropov, V.V., 1997. New developments in structural optimization using adaptive mesh refinement and multi-point approximations. *Engineering Optimization*, pp.29:217-234.

Vanderplaats Research and Development Inc., 1999. *Design Optimization Tools – Users Manual.* Colorado Springs.

Vanderplaats, G.N. & Moses, F., 1973. Structural optimization by methods of feasible directions. *Computers and Structures*, 3, pp.739-55.

Veers, P.S. et al., 2003. Trends in the design, manufacture and evaluation of wind turbine blades. *WIND ENERGY*, pp.6:245-259.

Wagner, T.C. & Papalambros, P.Y., 1993. A general framework for decomposition analysis in optimal design. *Advcances in Design Automation*, 2, p.315–325.

Wang, W., Guoa, S. & Chang, N., 2010. Optimum buckling design of composite stiffened panels using ant colony algorithm. *Composite Structures*, 92(3), pp.712-19.

Yen, S., n.d. *Carbon Fiber Bicycle Frames*. [Online] Available at: http://www.engin.swarthmore.edu/~syen1/Carbon%20Fiber%20Bicycle%20Frames.pdf.

Appendix A - Surrogate Based Optimization

This appendix contains the plots for the benchmark problems which were solved in Chapter 3 in order to test and validate the developed framework for surrogate based optimization.

Figure A-1 through Figure A-5 show the plots of the surrogate models approximating the Rastrigin function. The plots are based on 100x100 points that were obtained from the evaluation of the corresponding Kriging model. Figure A-6 through Figure A-10 show the cross-validation plots obtained for the accepted Kriging models.

Figure A-11 through Figure A-15 show the plots of the surrogate models approximating the Sphere function. The plots are based on 100x100 points that were obtained from the evaluation of the corresponding Kriging model. Figure A-16 through Figure A-20 show the cross-validation plots obtained for the accepted Kriging models.

Figure A-21 through Figure A-27 show the plots of the surrogate models approximating the g08 function. The plots are based on 100x100 points that were obtained from the evaluation of the corresponding Kriging model. Figure A-28 through Figure A-34 show the cross-validation plots obtained for the accepted Kriging models.

Rastrigin Function

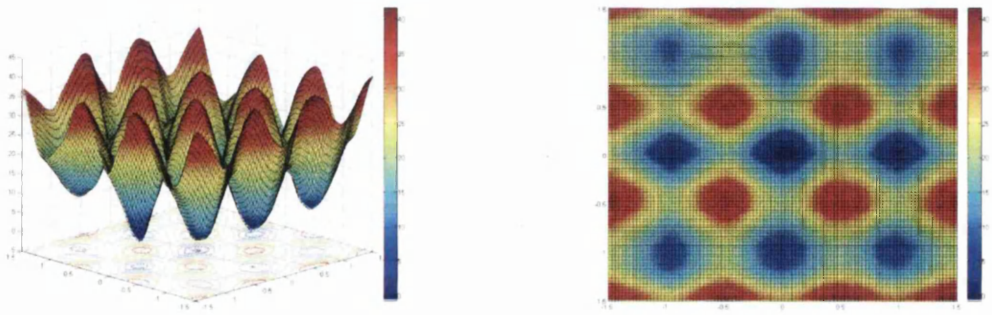


Figure A-1: Rastrigin function plotted from Kriging interpolation based on 40 experiments.

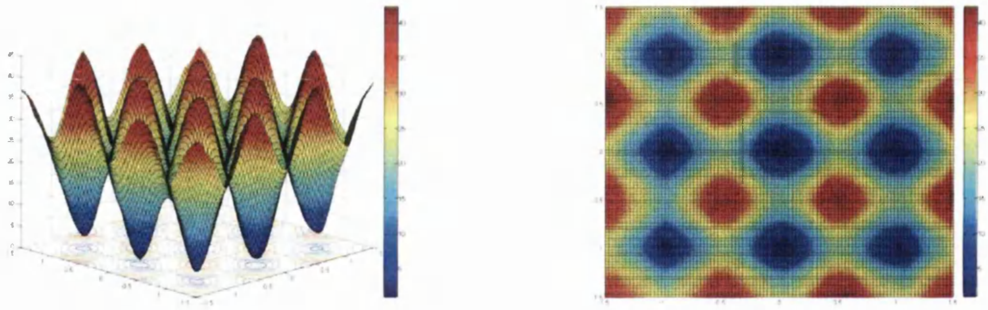


Figure A-2: Rastrigin function plotted from Kriging interpolation based on 80 experiments.

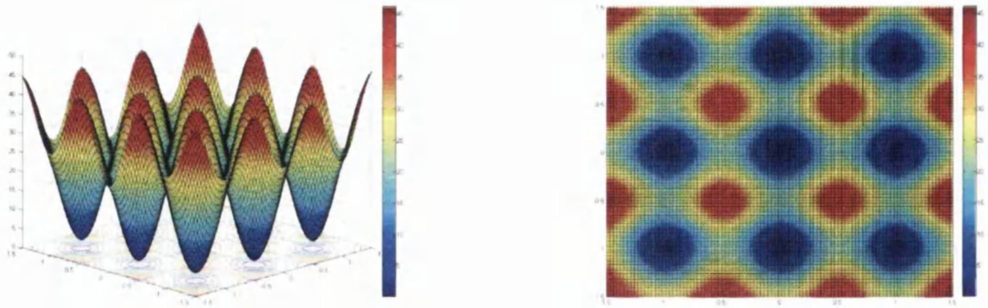


Figure A-3: Rastrigin function plotted from Kriging interpolation based on 120 experiments.

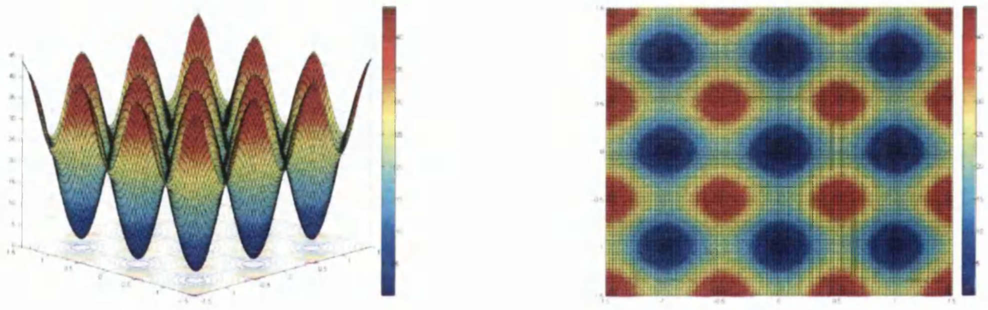


Figure A-4: Rastrigin function plotted from Kriging interpolation based on 180 experiments.

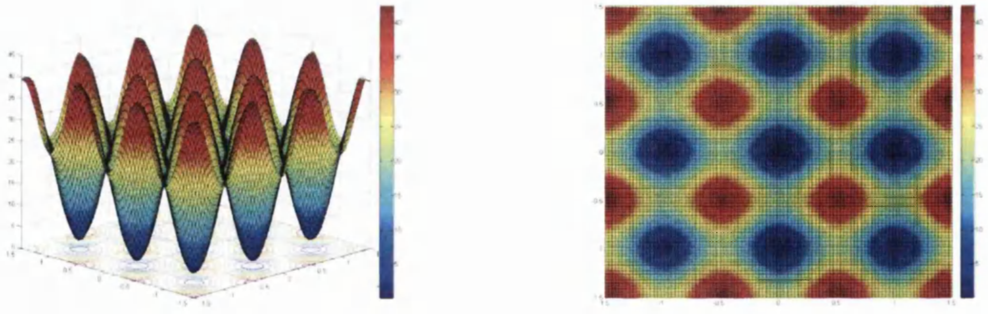


Figure A-5: Rastrigin function plotted from Kriging interpolation based on 400 experiments.

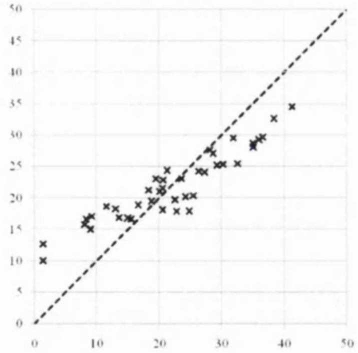


Figure A-6: Cross-validation of Kriging model approximating the Rastrigin function The number of experiments is 40.

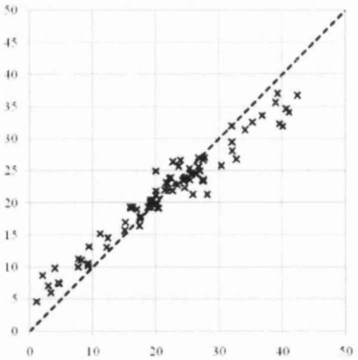


Figure A-7: Cross-validation of Kriging model approximating the Rastrigin function The number of experiments is 80.

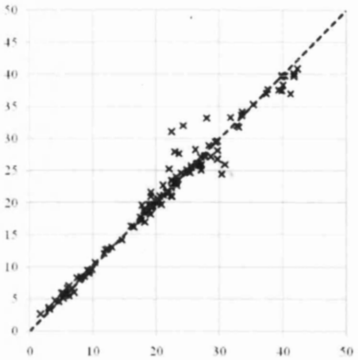


Figure A-8: Cross-validation of Kriging model approximating the Rastrigin function The number of experiments is 120.

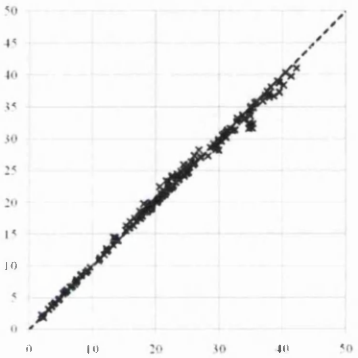


Figure A-9: Cross-validation of Kriging model approximating the Rastrigin function The number of experiments is 180.

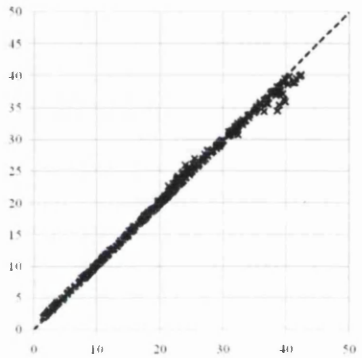


Figure A-10: Cross-validation of Kriging model approximating the Rastrigin function The number of experiments is 400.

Sphere Function

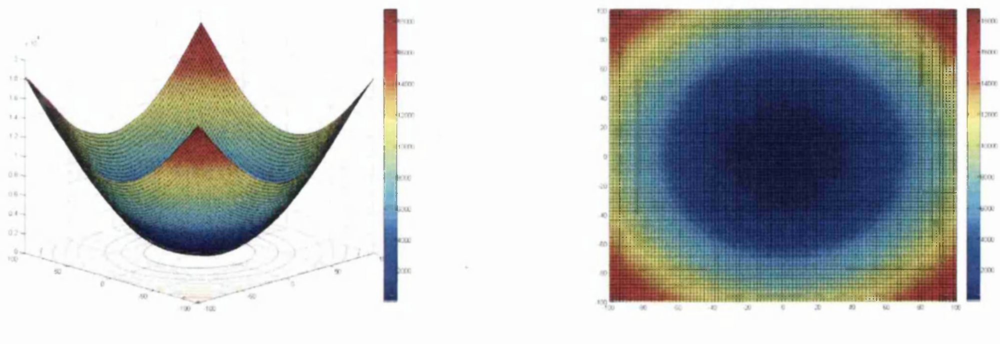


Figure A-11: Sphere function plotted from Kriging interpolation based on 40 experiments.

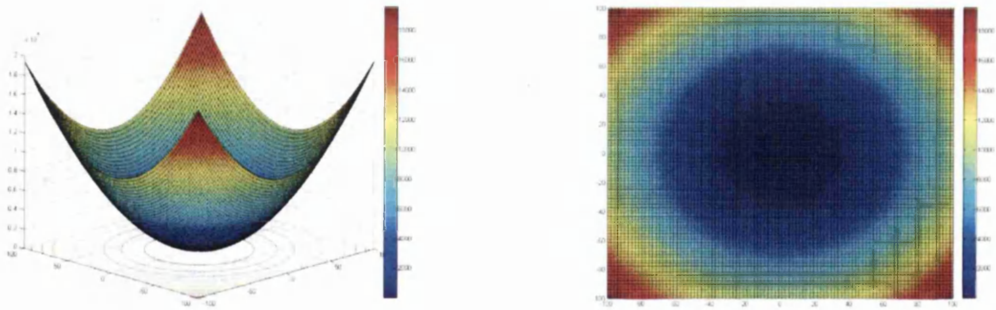


Figure A-12: Sphere function plotted from Kriging interpolation based on 80 experiments.

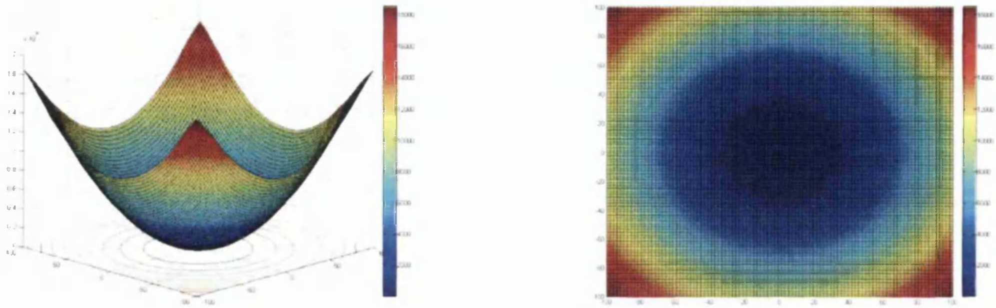


Figure A-13: Sphere function plotted from Kriging interpolation based on 120 experiments.

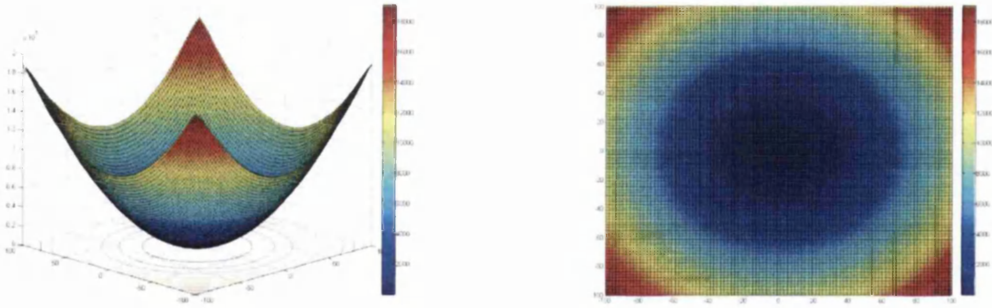


Figure A-14: Sphere function plotted from Kriging interpolation based on 180 experiments.

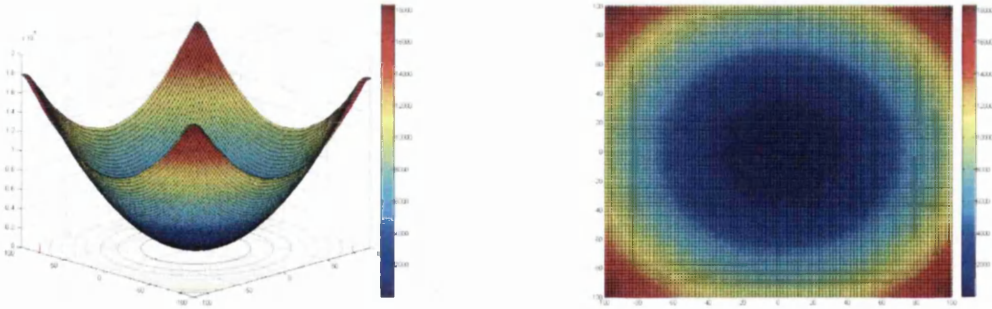


Figure A-15: Sphere function plotted from Kriging interpolation based on 400 experiments.

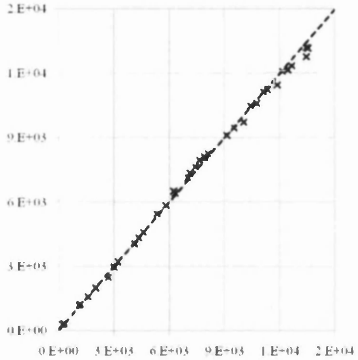


Figure A-16: Cross-validation of Kriging model approximating the sphere function The number of experiments is 40.

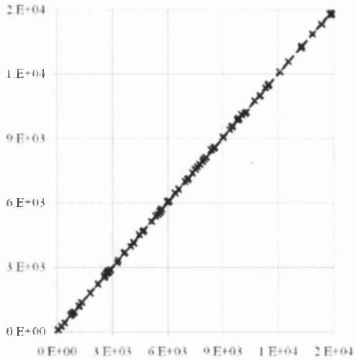


Figure A-17: Cross-validation of Kriging model approximating the sphere function The number of experiments is 80.

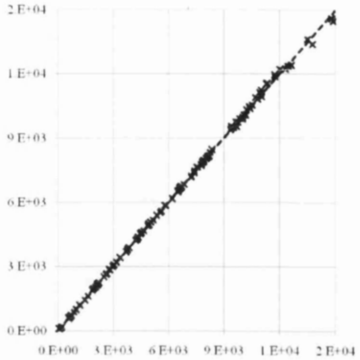


Figure A-18: Cross-validation of Kriging model approximating the sphere function The number of experiments is 120.

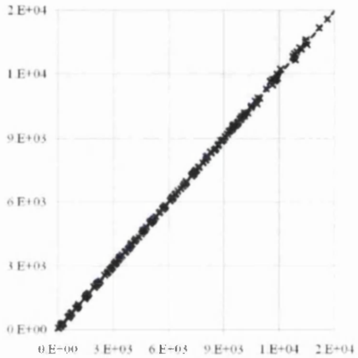


Figure A-19: Cross-validation of Kriging model approximating the sphere function The number of experiments is 180.

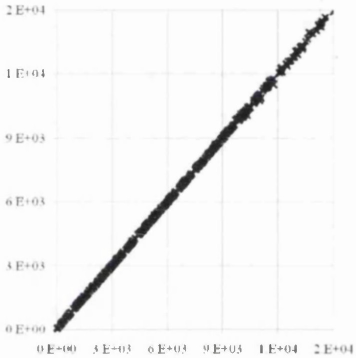


Figure A-20: Cross-validation of Kriging model approximating the sphere function The number of experiments is 400.

g08Function

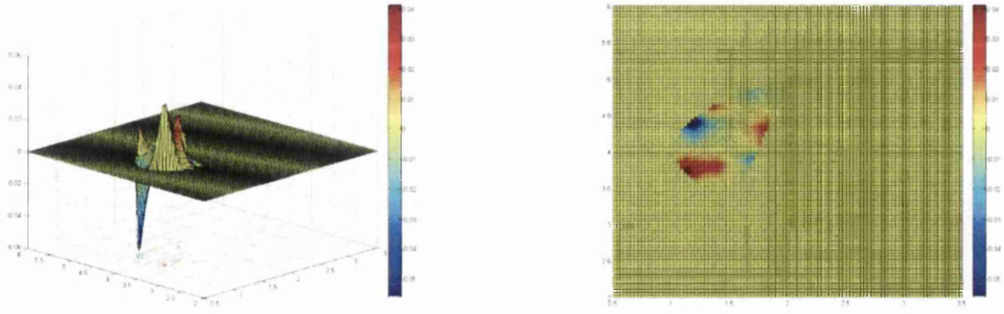


Figure A-21: g08 function plotted from Kriging interpolation based on 20 experiments created in the feasible region only.

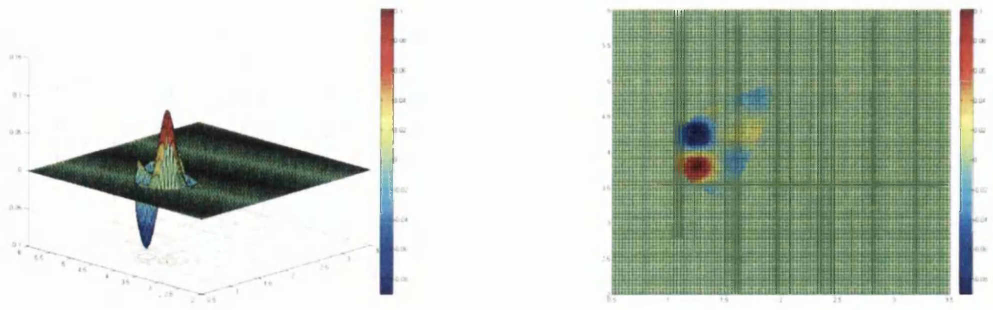


Figure A-22: g08 function plotted from Kriging interpolation based on 40 experiments created in the feasible region only.

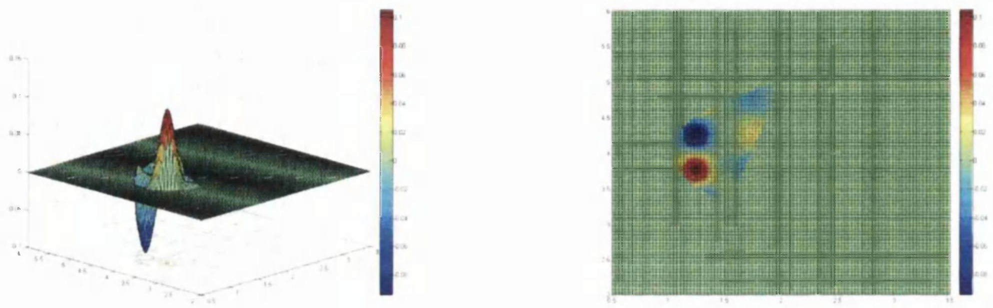


Figure A-23: g08 function plotted from Kriging interpolation based on 60 experiments created in the feasible region only.

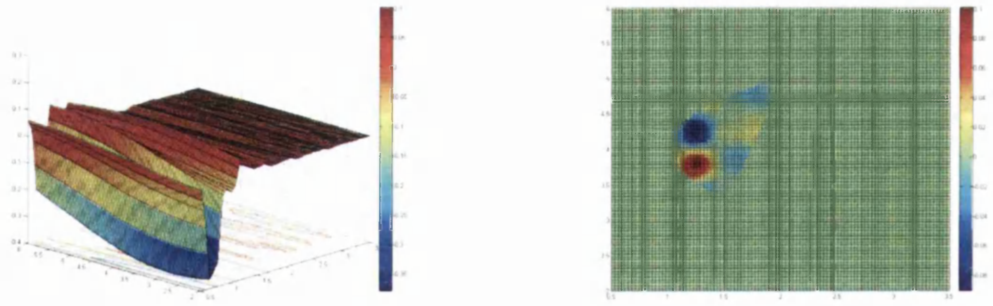


Figure A-24: g08 function plotted from Kriging interpolation based on 40 experiments created in the entire design space.

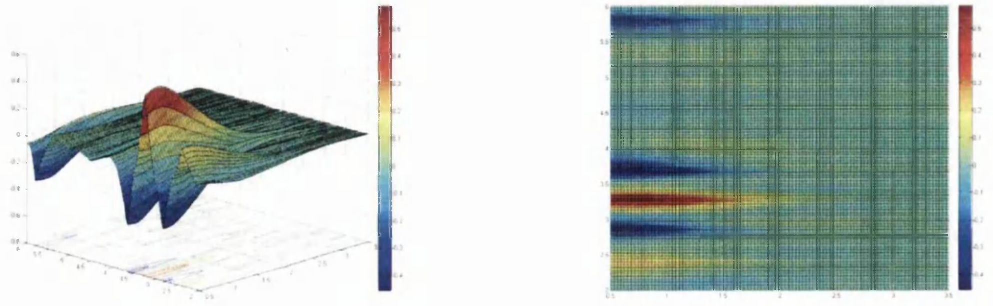


Figure A-25: g08 function plotted from Kriging interpolation based on 80 experiments created in the entire design space.

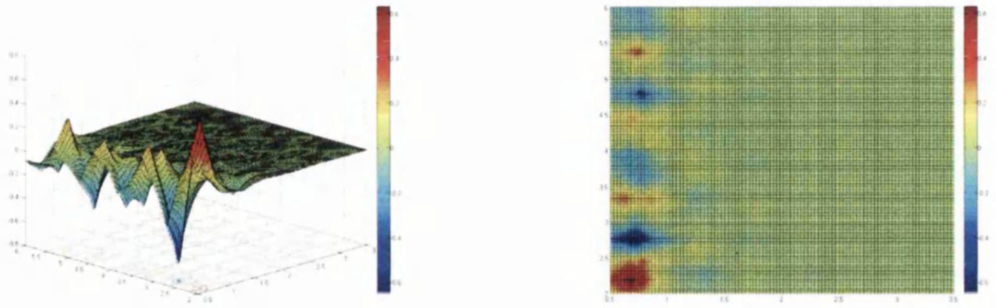


Figure A-26: g08 function plotted from Kriging interpolation based on 180 experiments created in the entire design space.

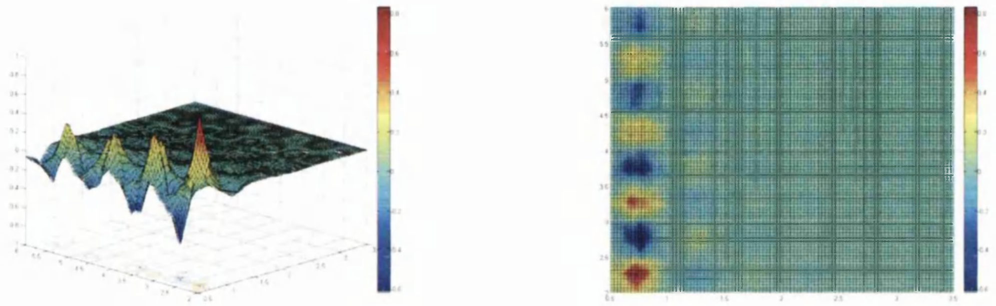


Figure A-27: g08 function plotted from Kriging interpolation based on 400 experiments created in the entire design space.

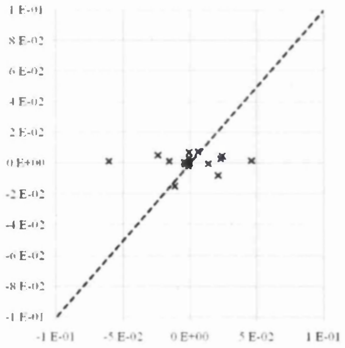


Figure A-28: Cross-validation of Kriging model approximating the g08 function The number of experiments is 20. All experiments are feasible.

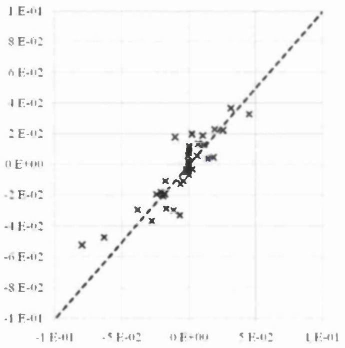


Figure A-29: Cross-validation of Kriging model approximating the g08 function The number of experiments is 40. All experiments are feasible.

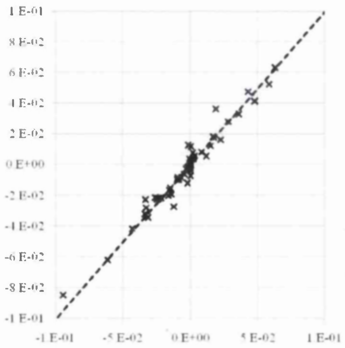


Figure A-30: Cross-validation of Kriging model approximating the g08 function The number of experiments is 60. All experiments are feasible.

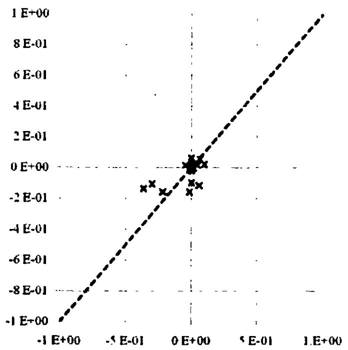


Figure A-31: Cross-validation of Kriging model approximating the g08 function The number of experiments is 40. Experiments are spread over the entire design space.

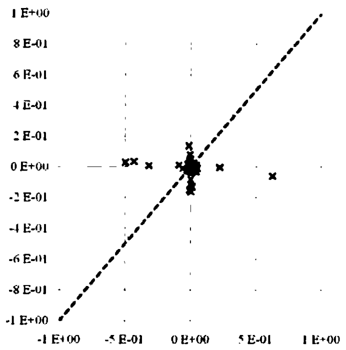


Figure A-32: Cross-validation of Kriging model approximating the g08 function The number of experiments is 80. Experiments are spread over the entire design space.

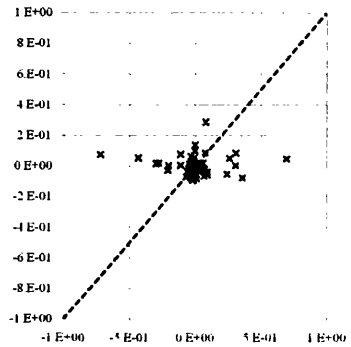


Figure A-33: Cross-validation of Kriging model approximating the g08 function The number of experiments is 180. Experiments are spread over the entire design space.

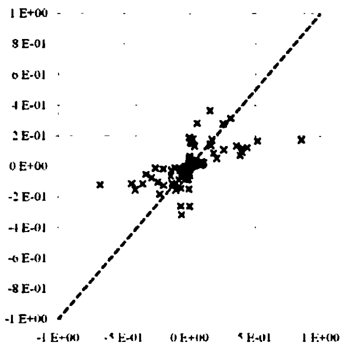


Figure A-34: Cross-validation of Kriging model approximating the g08 function The number of experiments is 400. Experiments are spread over the entire design space.

Appendix B – Structured-Wall PE Pipes

This appendix contains the material data plots for the material used by Asset Ltd. In Chapter 5 only the data for the material used by KWH Ltd. was shown. Figure B-1 through Figure B-6 show the data from the tension tests and Figure B-7 and Figure B-8 from the DMTA tests.

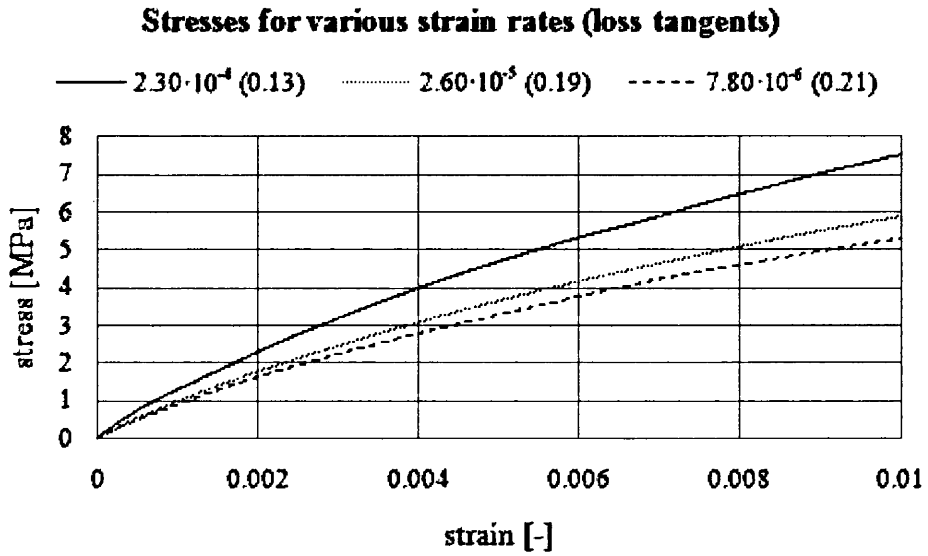


Figure B-1: Asset material. Stress plotted against strain for various strain rates (loss tangents) obtained from tension tests.

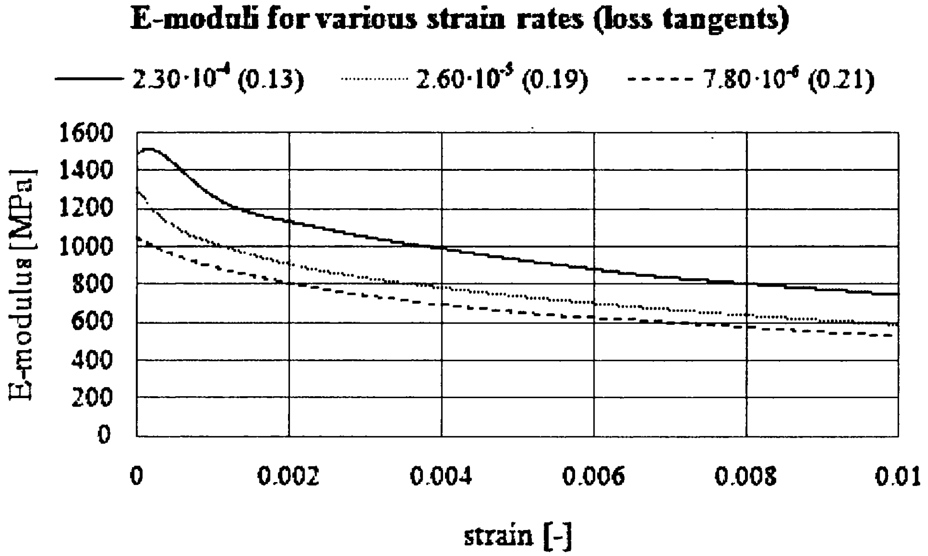


Figure B-2: Asset material. E-modulus plotted against strain for various strain rates (loss tangents) obtained from tension tests.

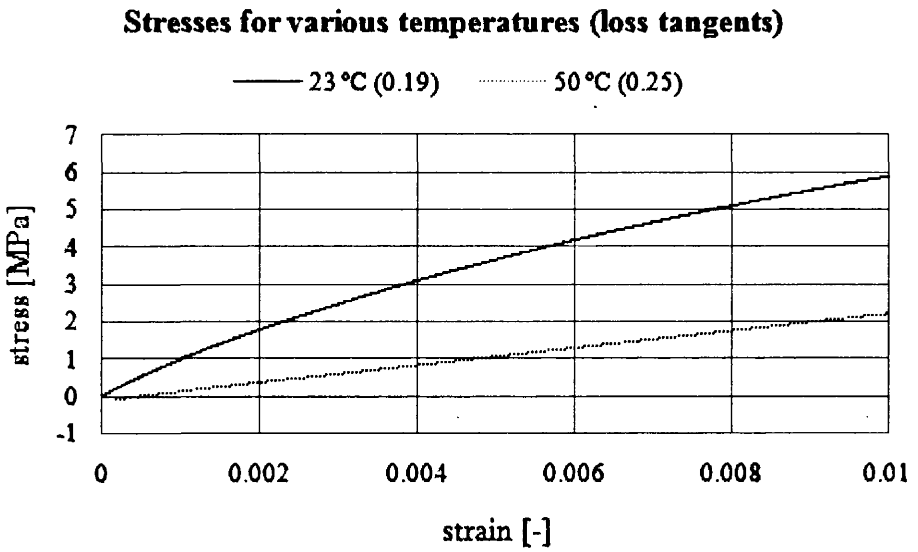


Figure B-3: Asset material. Stress plotted against strain for various temperatures (loss tangents) obtained from tension tests.

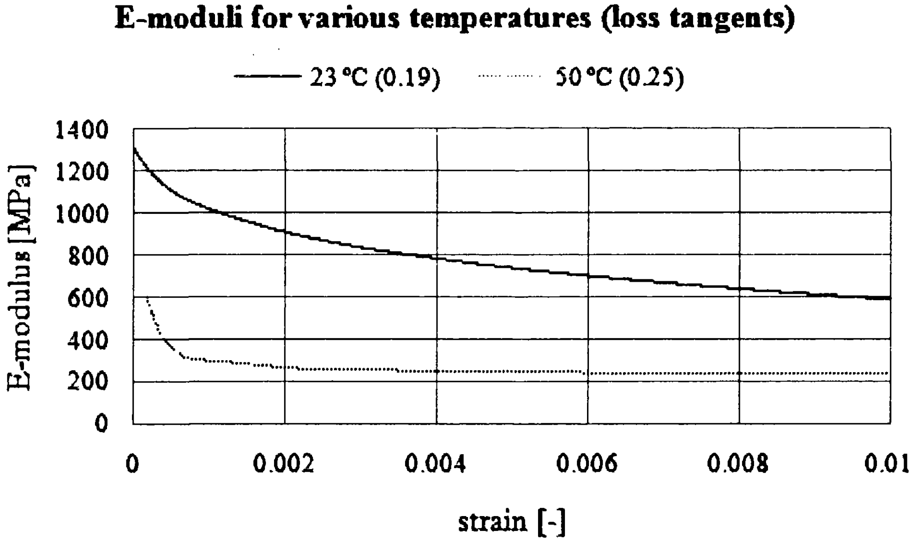


Figure B-4: Asset material. E-modulus plotted against strain for various temperatures (loss tangents) obtained from tension tests.

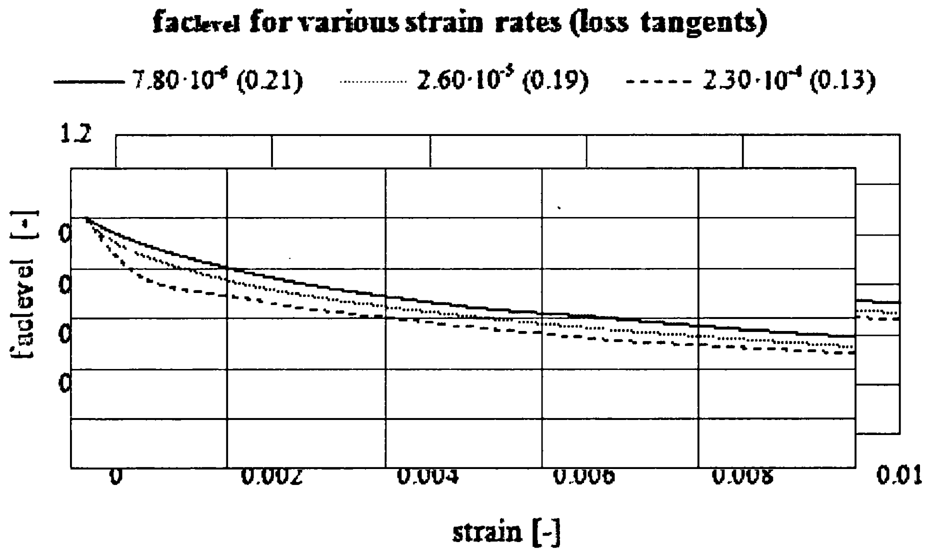


Figure B-5: Asset material. fac_{level} plotted against strain for various strain rates (loss tangents) obtained from tension tests.

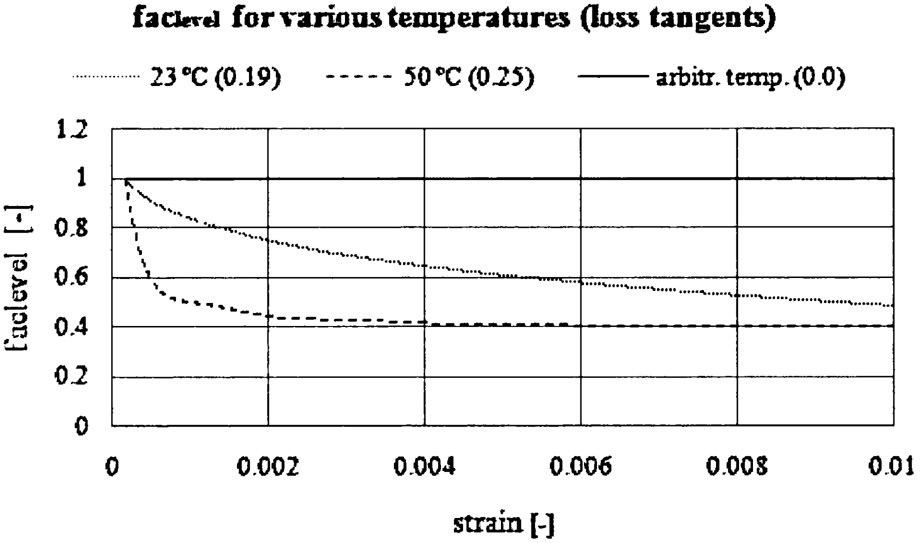


Figure B-6: Asset material. fac_{level} plotted against strain for various temperatures (loss tangents) obtained from tension tests.

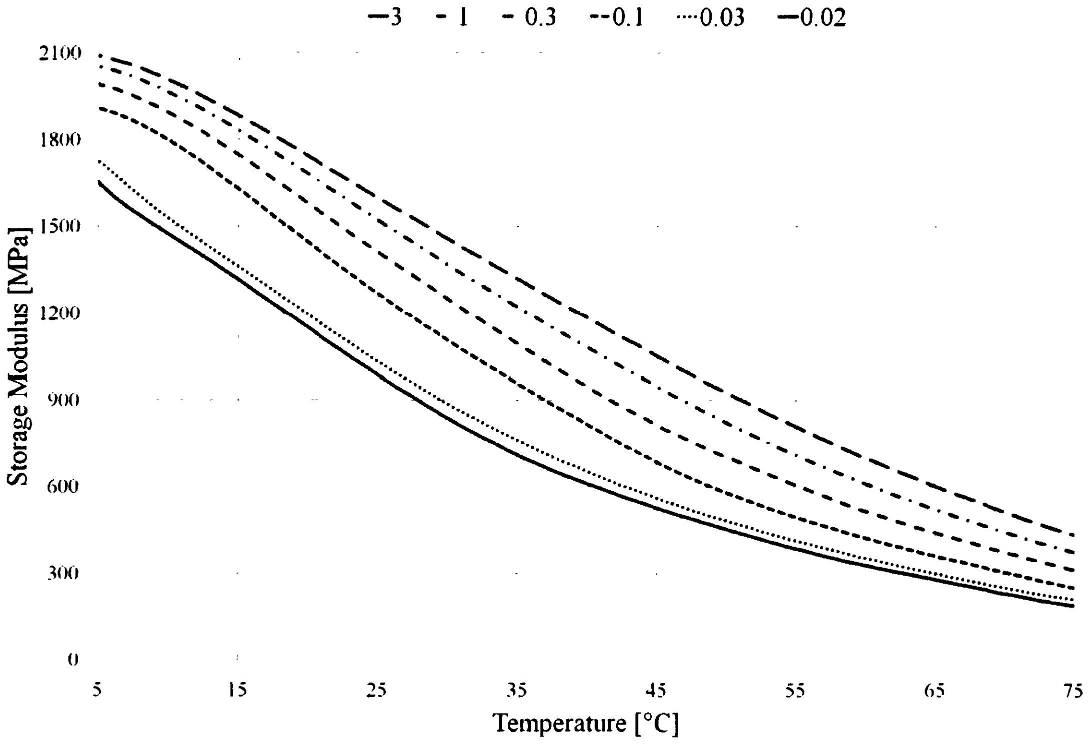


Figure B-7: Asset material. Storage modulus E' plotted against temperature for various frequencies obtained from DMTA tests.

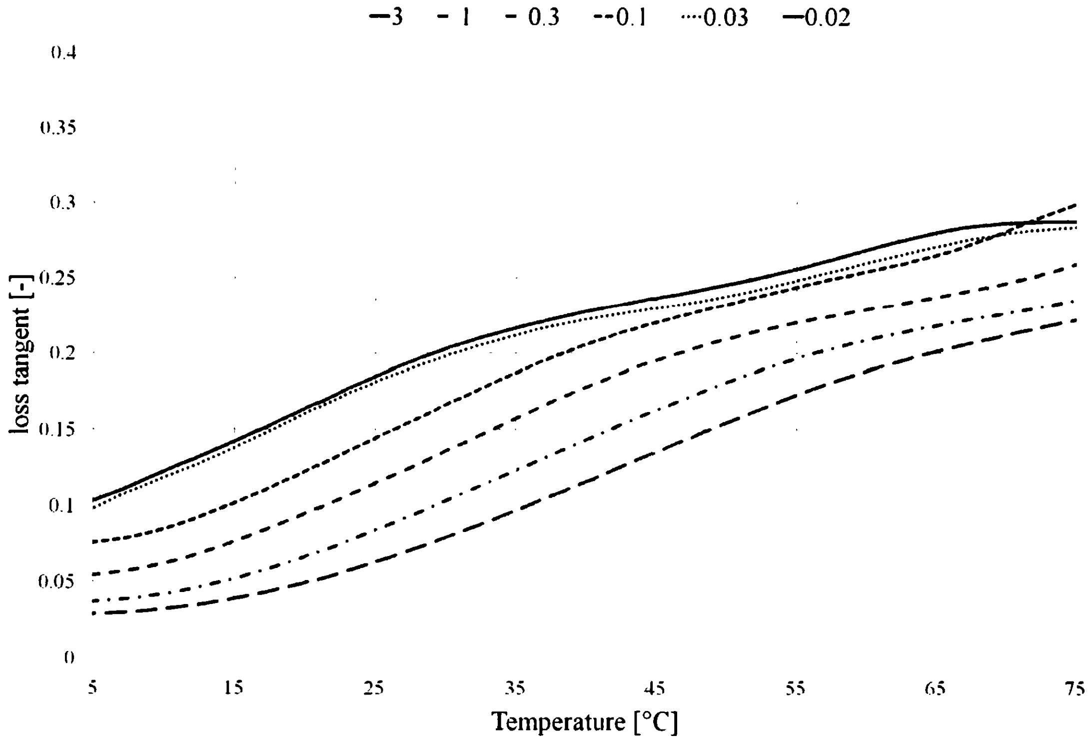


Figure B-8: Asset material. Loss tangent tan δ plotted against temperature for various frequencies obtained from DMTA tests.

Appendix C – Carbon-fibre Mountain Bike Frames

This appendix contains additional data corresponding to the design of carbon-fibre competition bicycle frames discussed in Chapter 6.

Table C-1 shows the actual OLH used in Example 1 and 2 for the creation of the Kriging model. The potential energy is 0.1755 and the space division vector is $\underline{n}_{Div} = [140 \quad \dots \quad 140]$. The corresponding fractions between 0 and 1 are given in Table C-2 and the local shape parameters resulting from the fractions can be found in Table C-3. The fractions interpolate between the maximum and minimum dimensions of the local shape parameters. In the tables, TT stands for top tube, DT for down tube, ST for seat tube, SS for seat stay and CS for chain stay.

70 experiments were conducted for each example. The two experiments in Example 1 that led to the highest and lowest sum of rigidities were Experiments 7 and 34, respectively. The corresponding frame geometries are shown in Figure C-1 and Figure C-2. The two experiments in Example 2 that led to the lowest and highest mass were Experiments 39 and 60, respectively. The corresponding frame geometries are shown in Figure C-3 and Figure C-4.

The development of the sum of rigidities and the frame's mass for the two experiments in Example 1 is shown in Figure C-5. For Example 2 it can be found in Figure C-6.

Table C-1: OLH used for the creation of the Kriging model in Example 1 and 2.

	TT	TT	DT	DT	ST	ST	SS	SS	SS	SS	SS	SS	CS	CS	CS	CS	CS	CS	CS
	A	B	A	B	A	B	A	A	A	A	B	B	A	A	A	A	A	B	B
	1	1	1	1	1	1	1	2	3	4	1	2	1	2	3	4	5	1	2
1	9	36	8	24	10	16	87	35	128	69	32	3	81	33	89	14	17	46	14
2	13	5	1	17	59	2	59	71	56	86	51	34	11	29	77	110	34	48	24
3	56	43	20	11	24	12	130	52	22	122	14	33	7	71	84	37	18	25	18
4	10	32	2	19	19	56	26	22	60	107	33	39	133	16	130	78	67	36	60
5	54	22	5	64	21	34	134	47	72	56	37	4	15	4	13	27	49	55	51
6	28	40	34	37	18	46	56	9	3	36	12	22	135	36	34	1	3	64	42
7	41	12	59	58	38	48	94	44	103	27	19	10	96	47	8	62	121	3	62
8	8	10	17	59	65	70	63	77	5	63	30	38	38	73	41	86	108	63	35
9	51	63	32	57	53	65	5	84	37	90	45	46	31	21	35	112	65	59	64
10	29	38	18	53	58	3	75	8	11	26	62	19	56	53	72	69	123	2	10
11	39	11	55	41	67	40	85	60	25	1	50	55	111	15	75	131	22	58	48
12	64	34	56	25	8	55	13	108	52	49	24	26	19	85	119	29	119	29	45
13	58	51	24	32	64	9	118	28	55	66	53	30	129	44	28	111	6	40	12
14	20	27	67	4	20	23	18	59	117	58	17	48	20	13	24	124	16	37	28
15	53	35	6	61	42	47	107	65	20	116	48	20	101	50	135	45	70	11	59
16	46	21	22	8	55	58	36	80	96	32	5	5	80	12	63	118	40	12	20
17	55	64	47	18	6	22	111	39	9	25	64	42	40	8	54	5	14	27	46
18	4	18	66	62	45	45	68	19	76	19	15	8	57	28	138	89	76	50	44
19	35	47	27	52	28	29	3	15	7	15	22	65	6	23	31	130	41	28	7
20	18	62	44	45	14	5	77	18	24	2	10	31	43	64	120	42	44	4	54
21	14	57	28	66	1	8	83	66	84	84	35	40	78	66	6	10	104	10	41
22	66	13	21	65	11	25	9	21	6	112	31	29	59	19	69	87	107	53	19
23	68	58	57	51	41	15	38	45	100	73	57	45	84	10	136	97	42	1	47
24	63	49	7	40	34	38	57	69	82	14	9	41	39	91	4	53	5	54	3
25	57	30	30	69	33	44	50	75	79	51	7	62	5	14	125	12	12	18	29
26	26	69	62	47	12	21	20	105	28	44	11	36	34	2	101	55	98	62	34
27	61	19	46	33	52	14	139	70	13	6	3	15	25	41	57	51	118	35	25
28	17	26	25	70	29	68	48	92	38	38	46	14	73	6	19	35	23	14	8
29	52	55	39	34	47	7	126	27	114	65	36	57	13	30	67	57	116	65	68
30	45	66	37	7	61	17	10	46	112	12	60	44	51	55	68	106	79	34	6
31	32	20	43	14	69	69	86	62	36	4	40	58	30	61	26	19	7	6	37
32	47	25	29	9	57	41	95	61	31	110	69	35	109	69	9	34	124	43	61
33	43	41	19	68	26	36	49	37	57	7	63	2	24	56	108	134	11	38	39
34	2	39	3	46	44	37	96	38	51	5	70	63	65	34	76	11	26	49	63
35	69	9	36	28	51	27	24	25	54	127	29	21	1	78	56	48	24	19	67
36	50	3	38	3	22	19	30	17	8	8	42	24	75	70	12	121	58	24	58
37	42	54	9	26	39	63	122	56	12	34	1	64	53	46	134	73	69	44	23
38	25	48	42	67	13	52	138	12	26	42	47	51	119	31	60	115	63	9	13
39	31	8	4	20	32	54	4	7	70	17	4	23	18	25	55	32	128	51	36
40	19	33	70	21	68	49	131	26	87	104	58	18	32	9	73	70	2	31	40
41	11	16	13	27	23	43	25	74	139	22	44	50	14	60	99	67	62	16	70
42	12	37	69	12	5	51	106	41	10	11	54	7	36	58	25	80	45	56	30
43	22	28	58	16	54	50	7	73	40	48	66	9	117	40	107	36	109	47	11
44	59	14	40	2	16	64	109	67	39	111	38	37	60	18	45	58	102	67	9
45	6	42	48	56	36	39	32	23	33	137	34	70	44	65	109	54	1	39	56
46	48	44	23	43	62	6	1	109	46	23	28	16	121	39	53	28	36	21	65
47	34	53	11	48	60	4	46	49	83	60	13	11	100	54	106	60	125	70	32
48	37	52	12	1	9	13	16	24	18	55	61	49	74	57	85	82	64	68	55
49	21	67	33	35	31	61	14	2	133	61	49	27	49	89	16	33	4	41	43
50	15	2	35	42	49	26	84	63	1	132	8	47	103	7	32	31	57	15	50
51	5	7	49	31	7	30	112	51	63	62	43	52	23	35	110	56	133	26	2
52	3	45	41	5	50	31	91	5	43	70	16	66	35	68	30	74	131	13	49
53	65	17	50	38	3	32	58	82	64	88	59	68	52	49	10	92	31	8	26
54	40	61	10	15	15	53	11	81	44	81	55	59	50	24	46	4	120	17	17
55	1	70	26	13	70	33	51	87	47	53	23	25	29	3	2	22	33	42	33
56	62	6	14	22	43	24	82	1	110	45	52	61	45	5	121	30	81	22	5
57	67	56	52	30	37	67	52	30	94	13	26	67	104	1	40	26	110	20	53
58	70	23	51	55	63	59	54	31	92	59	41	28	48	77	104	6	10	69	31
59	44	68	65	60	66	42	72	14	53	89	18	32	85	26	3	64	132	52	21
60	60	65	68	29	30	62	97	43	29	80	20	1	76	67	86	122	28	33	27
61	33	60	15	23	17	60	74	3	66	75	21	6	16	17	33	99	86	5	69
62	7	46	61	10	25	11	27	72	42	113	25	13	122	43	58	66	52	7	15
63	36	29	63	6	40	28	88	50	86	64	6	60	134	37	132	44	55	66	38
64	49	1	60	50	27	1	28	85	59	3	68	43	64	22	44	41	91	57	52
65	38	15	16	36	4	20	140	32	97	40	27	54	114	38	23	102	59	45	66
66	23	24	64	44	35	35	34	16	32	91	67	12	2	11	65	9	137	30	57
67	16	50	54	63	48	10	89	40	41	108	56	53	17	75	47	24	61	60	1
68	24	31	45	49	2	66	43	57	113	46	39	69	115	51	17	91	73	61	16
69	30	59	31	39	46	57	123	54	115	10	65	17	22	79	81	38	138	32	22
70	27	4	53	54	56	18	12	10	75	20	2	56	82	59	49	23	72	23	4

Table C-2: Fractions resulting from the OLH.

	TT	TT	DT	DT	ST	ST	SS	SS	SS	SS	SS	SS	CS	CS	CS	CS	CS	CS	CS
	A	B	A	B	A	B	A	A	A	A	B	B	A	A	A	A	A	B	B
	1	1	1	1	1	1	1	2	3	4	1	2	1	2	3	4	5	1	2
1	0.06	0.25	0.05	0.17	0.06	0.11	0.62	0.24	0.91	0.49	0.22	0.01	0.58	0.23	0.63	0.09	0.12	0.32	0.09
2	0.09	0.03	0.00	0.12	0.42	0.01	0.42	0.50	0.40	0.61	0.36	0.24	0.07	0.20	0.55	0.78	0.24	0.34	0.17
3	0.40	0.30	0.14	0.07	0.17	0.08	0.93	0.37	0.15	0.87	0.09	0.23	0.04	0.50	0.60	0.26	0.12	0.17	0.12
4	0.06	0.22	0.01	0.13	0.13	0.40	0.18	0.15	0.42	0.76	0.23	0.27	0.95	0.11	0.93	0.55	0.47	0.25	0.42
5	0.38	0.15	0.03	0.45	0.14	0.24	0.96	0.33	0.51	0.40	0.26	0.02	0.10	0.02	0.09	0.19	0.35	0.39	0.36
6	0.19	0.28	0.24	0.26	0.12	0.32	0.40	0.06	0.01	0.25	0.08	0.15	0.96	0.25	0.24	0.00	0.01	0.45	0.29
7	0.29	0.08	0.42	0.41	0.27	0.34	0.67	0.31	0.73	0.19	0.13	0.06	0.68	0.33	0.05	0.44	0.86	0.01	0.44
8	0.05	0.06	0.12	0.42	0.46	0.50	0.45	0.55	0.03	0.45	0.21	0.27	0.27	0.52	0.29	0.61	0.77	0.45	0.24
9	0.36	0.45	0.22	0.40	0.37	0.46	0.03	0.60	0.26	0.64	0.32	0.32	0.22	0.14	0.24	0.80	0.46	0.42	0.45
10	0.20	0.27	0.12	0.37	0.41	0.01	0.53	0.05	0.07	0.18	0.44	0.13	0.40	0.37	0.51	0.49	0.88	0.01	0.06
11	0.27	0.07	0.39	0.29	0.47	0.28	0.60	0.42	0.17	0.00	0.35	0.39	0.79	0.10	0.53	0.94	0.15	0.41	0.34
12	0.45	0.24	0.40	0.17	0.05	0.39	0.09	0.77	0.37	0.35	0.17	0.18	0.13	0.60	0.85	0.20	0.85	0.20	0.32
13	0.41	0.36	0.17	0.22	0.45	0.06	0.84	0.19	0.39	0.47	0.37	0.21	0.92	0.31	0.19	0.79	0.04	0.28	0.08
14	0.14	0.19	0.47	0.02	0.14	0.16	0.12	0.42	0.83	0.41	0.12	0.34	0.14	0.09	0.17	0.88	0.11	0.26	0.19
15	0.37	0.24	0.04	0.43	0.29	0.33	0.76	0.46	0.14	0.83	0.34	0.14	0.72	0.35	0.96	0.32	0.50	0.07	0.42
16	0.32	0.14	0.15	0.05	0.39	0.41	0.25	0.57	0.68	0.22	0.03	0.03	0.57	0.08	0.45	0.84	0.28	0.08	0.14
17	0.39	0.45	0.33	0.12	0.04	0.15	0.79	0.27	0.06	0.17	0.45	0.29	0.28	0.05	0.38	0.03	0.09	0.19	0.32
18	0.02	0.12	0.47	0.44	0.32	0.32	0.48	0.13	0.54	0.13	0.10	0.05	0.40	0.19	0.99	0.63	0.54	0.35	0.31
19	0.24	0.33	0.19	0.37	0.19	0.20	0.01	0.10	0.04	0.10	0.15	0.46	0.04	0.16	0.22	0.93	0.29	0.19	0.04
20	0.12	0.44	0.31	0.32	0.09	0.03	0.55	0.12	0.17	0.01	0.06	0.22	0.30	0.45	0.86	0.29	0.31	0.02	0.38
21	0.09	0.40	0.19	0.47	0.00	0.05	0.59	0.47	0.60	0.60	0.24	0.28	0.55	0.47	0.04	0.06	0.74	0.06	0.29
22	0.47	0.09	0.14	0.46	0.07	0.17	0.06	0.14	0.04	0.80	0.22	0.20	0.42	0.13	0.49	0.62	0.76	0.37	0.13
23	0.48	0.41	0.40	0.36	0.29	0.10	0.27	0.32	0.71	0.52	0.40	0.32	0.60	0.06	0.97	0.69	0.29	0.00	0.33
24	0.45	0.35	0.04	0.28	0.24	0.27	0.40	0.49	0.58	0.09	0.06	0.29	0.27	0.65	0.02	0.37	0.03	0.38	0.01
25	0.40	0.21	0.21	0.49	0.23	0.31	0.35	0.53	0.56	0.36	0.04	0.44	0.03	0.09	0.89	0.08	0.08	0.12	0.20
26	0.18	0.49	0.44	0.33	0.08	0.14	0.14	0.75	0.19	0.31	0.07	0.25	0.24	0.01	0.72	0.39	0.70	0.44	0.24
27	0.43	0.13	0.32	0.23	0.37	0.09	0.99	0.50	0.09	0.04	0.01	0.10	0.17	0.29	0.40	0.36	0.84	0.24	0.17
28	0.12	0.18	0.17	0.50	0.20	0.48	0.34	0.65	0.27	0.27	0.32	0.09	0.52	0.04	0.13	0.24	0.16	0.09	0.05
29	0.37	0.39	0.27	0.24	0.33	0.04	0.90	0.19	0.81	0.46	0.25	0.40	0.09	0.21	0.47	0.40	0.83	0.46	0.48
30	0.32	0.47	0.26	0.04	0.43	0.12	0.06	0.32	0.80	0.08	0.42	0.31	0.36	0.39	0.48	0.76	0.56	0.24	0.04
31	0.22	0.14	0.30	0.09	0.49	0.49	0.61	0.44	0.25	0.02	0.28	0.41	0.21	0.43	0.18	0.13	0.04	0.04	0.26
32	0.33	0.17	0.20	0.06	0.40	0.29	0.68	0.43	0.22	0.78	0.49	0.24	0.78	0.49	0.06	0.24	0.88	0.30	0.43
33	0.30	0.29	0.13	0.48	0.18	0.25	0.35	0.26	0.40	0.04	0.45	0.01	0.17	0.40	0.77	0.96	0.07	0.27	0.27
34	0.01	0.27	0.01	0.32	0.31	0.26	0.68	0.27	0.36	0.03	0.50	0.45	0.46	0.24	0.54	0.07	0.18	0.35	0.45
35	0.49	0.06	0.25	0.19	0.36	0.19	0.17	0.17	0.38	0.91	0.20	0.14	0.00	0.55	0.40	0.34	0.17	0.13	0.47
36	0.35	0.01	0.27	0.01	0.15	0.13	0.21	0.12	0.05	0.05	0.29	0.17	0.53	0.50	0.08	0.86	0.41	0.17	0.41
37	0.29	0.38	0.06	0.18	0.27	0.45	0.87	0.40	0.08	0.24	0.00	0.45	0.37	0.32	0.96	0.52	0.49	0.31	0.16
38	0.17	0.34	0.29	0.47	0.09	0.37	0.99	0.08	0.18	0.29	0.33	0.36	0.85	0.22	0.42	0.82	0.45	0.06	0.09
39	0.22	0.05	0.02	0.14	0.22	0.38	0.02	0.04	0.50	0.12	0.02	0.16	0.12	0.17	0.39	0.22	0.91	0.36	0.25
40	0.13	0.23	0.50	0.14	0.48	0.35	0.94	0.18	0.62	0.74	0.41	0.12	0.22	0.06	0.52	0.50	0.01	0.22	0.28
41	0.07	0.11	0.09	0.19	0.16	0.30	0.17	0.53	0.99	0.15	0.31	0.35	0.09	0.42	0.71	0.47	0.44	0.11	0.50
42	0.08	0.26	0.49	0.08	0.03	0.36	0.76	0.29	0.06	0.07	0.38	0.04	0.25	0.41	0.17	0.57	0.32	0.40	0.21
43	0.15	0.19	0.41	0.11	0.38	0.35	0.04	0.52	0.28	0.34	0.47	0.06	0.83	0.28	0.76	0.25	0.78	0.33	0.07
44	0.42	0.09	0.28	0.01	0.11	0.45	0.78	0.47	0.27	0.79	0.27	0.26	0.42	0.12	0.32	0.41	0.73	0.47	0.06
45	0.04	0.29	0.34	0.40	0.25	0.27	0.22	0.16	0.23	0.98	0.24	0.50	0.31	0.46	0.78	0.38	0.00	0.27	0.40
46	0.34	0.31	0.16	0.30	0.44	0.04	0.00	0.78	0.32	0.16	0.19	0.11	0.86	0.27	0.37	0.19	0.25	0.14	0.46
47	0.24	0.37	0.07	0.34	0.42	0.02	0.32	0.35	0.59	0.42	0.09	0.07	0.71	0.38	0.76	0.42	0.89	0.50	0.22
48	0.26	0.37	0.08	0.00	0.06	0.09	0.11	0.17	0.12	0.39	0.43	0.35	0.53	0.40	0.60	0.58	0.45	0.48	0.39
49	0.14	0.47	0.23	0.24	0.22	0.43	0.09	0.01	0.95	0.43	0.35	0.19	0.35	0.63	0.11	0.23	0.02	0.29	0.30
50	0.10	0.01	0.24	0.29	0.35	0.18	0.60	0.45	0.00	0.94	0.05	0.33	0.73	0.04	0.22	0.22	0.40	0.10	0.35
51	0.03	0.04	0.35	0.22	0.04	0.21	0.80	0.36	0.45	0.44	0.30	0.37	0.16	0.24	0.78	0.40	0.95	0.18	0.01
52	0.01	0.32	0.29	0.03	0.35	0.22	0.65	0.03	0.30	0.50	0.11	0.47	0.24	0.48	0.21	0.53	0.94	0.09	0.35
53	0.46	0.12	0.35	0.27	0.01	0.22	0.41	0.58	0.45	0.63	0.42	0.48	0.37	0.35	0.06	0.65	0.22	0.05	0.18
54	0.28	0.43	0.06	0.10	0.10	0.37	0.07	0.58	0.31	0.58	0.39	0.42	0.35	0.17	0.32	0.02	0.86	0.12	0.12
55	0.00	0.50	0.18	0.09	0.50	0.23	0.36	0.62	0.33	0.37	0.16	0.17	0.20	0.01	0.01	0.15	0.23	0.29	0.23
56	0.44	0.04	0.09	0.15	0.30	0.17	0.58	0.00	0.78	0.32	0.37	0.43	0.32	0.03	0.86	0.21	0.58	0.15	0.03
57	0.47	0.40	0.37	0.21	0.26	0.47	0.37	0.21	0.67	0.09	0.18	0.47	0.74	0.00	0.28	0.18	0.78	0.14	0.37
58	0.50	0.16	0.36	0.39	0.45	0.42	0.38	0.22	0.65	0.42	0.29	0.19	0.34	0.55	0.74	0.04	0.06	0.49	0.22
59	0.31	0.48	0.46	0.42	0.47	0.29	0.51	0.09	0.37	0.63	0.12	0.22	0.60	0.18	0.01	0.45	0.94	0.37	0.14
60	0.42	0.46	0.48	0.20	0.21	0.44	0.69	0.30	0.20	0.57	0.14	0.00	0.54	0.47	0.61	0.87	0.19	0.23	0.19
61	0.23	0.42	0.10	0.16	0.12	0.42	0.53	0.01	0.47	0.53	0.14	0.04	0.11	0.12	0.23	0.71	0.61	0.03	0.49
62	0.04	0.32	0.43	0.06	0.17	0.07	0.19	0.51	0.29	0.81	0.17	0.09	0.87	0.30	0.41	0.47	0.37	0.04	0.10
63	0.25	0.20	0.45	0.04	0.28	0.19	0.63	0.35	0.61	0.45	0.04	0.42	0.96	0.26	0.94	0.31	0.39	0.47	0.27
64	0.35	0.00	0.42	0.35	0.19	0.00	0.19	0.60	0.42	0.01	0.48	0.30	0.45	0.15	0.31	0.29	0.65	0.40	0.37
65	0.27	0.10	0.11	0.25	0.02	0.14	1.00	0.22	0.69	0.28	0.19	0.38	0.81	0.27	0.16	0.73	0.42	0.32	0.47
66	0.16	0.17	0.45	0.31	0.24	0.24	0.24	0.11	0.22	0.65	0.47	0.08	0.01	0.07	0.46	0.06	0.98	0.21	0.40
67	0.11	0.35	0.38	0.45	0.34	0.06	0.63	0.28	0.29	0.77	0.40	0.37	0.12	0.53	0.33	0.17	0.43	0.42	0.00
68	0.17	0.22	0.32	0.35	0.01	0.47	0.30	0.40	0.81	0.32	0.27	0.49	0.82	0.36	0.12	0.65	0.52	0.43	0.11
69	0.21	0.42	0.22	0.27	0.32	0.40	0.88	0.38	0.82	0.06	0.46	0.12	0.15	0.56	0.58	0.27	0.99	0.22	0.15
70	0.19	0.02	0.37	0.38	0.40	0.12	0.08	0.06	0.53	0.14	0.01	0.40	0.58	0.42	0.35	0.16	0.51	0.16	0.02

Table C-3: Actual design variable values in mm.

	TT	TT	DT	DT	ST	ST	SS	SS	SS	SS	SS	SS	CS	CS	CS	CS	CS	CS	CS
	A	B	A	B	A	B	A	A	A	A	B	B	A	A	A	A	A	B	B
	1	1	1	1	1	1	1	2	3	4	1	2	1	2	3	4	5	1	2
1	15.6	24.5	15.5	23.7	17.1	17.4	10.7	5.4	14.8	8.8	8.3	5.2	10.1	5.2	10.9	3.3	3.6	9.9	6.4
2	15.9	22.3	15.0	23.2	20.0	16.6	7.8	9.1	7.5	10.6	10.4	8.6	3.0	4.8	9.7	13.0	5.3	10.1	7.5
3	19.0	25.0	16.4	22.7	17.9	17.2	15.0	7.1	4.1	14.2	6.4	8.5	2.6	9.1	10.4	5.6	3.7	7.6	6.8
4	15.6	24.2	15.1	23.3	17.6	19.9	4.5	4.1	7.9	12.7	8.5	9.1	15.3	3.5	15.0	9.8	8.6	8.8	11.4
5	18.8	23.5	15.3	26.5	17.7	18.5	15.4	6.6	9.2	7.5	8.9	5.3	3.4	2.3	3.2	4.6	6.8	10.8	10.4
6	16.9	24.8	17.4	24.6	17.5	19.3	7.5	2.8	2.2	5.5	6.2	7.3	15.5	5.5	5.3	2.0	2.2	11.8	9.4
7	17.9	22.8	19.2	26.1	18.8	19.4	11.4	6.3	12.3	4.6	6.9	6.0	11.6	6.6	2.7	8.1	14.1	5.2	11.6
8	15.5	22.6	16.2	26.2	20.4	20.7	8.2	9.7	2.4	8.2	8.1	9.0	5.7	9.3	6.0	10.6	12.8	11.7	8.7
9	18.6	26.5	17.2	26.0	19.7	20.4	2.4	10.4	5.6	11.0	9.7	9.9	5.0	4.0	5.4	13.2	8.4	11.3	11.8
10	17.0	24.7	16.2	25.7	20.0	16.6	9.5	2.7	3.0	4.5	11.6	6.9	7.5	7.2	9.2	8.8	14.3	5.1	6.0
11	17.7	22.7	18.9	24.9	20.5	18.9	10.5	7.9	4.4	2.0	10.3	10.8	13.1	3.4	9.5	15.1	4.1	11.2	10.1
12	19.5	24.4	19.0	23.7	16.9	19.8	3.2	12.8	7.1	6.8	7.5	7.7	3.8	10.5	13.9	4.8	13.9	8.0	9.7
13	19.1	25.6	16.7	24.2	20.4	17.0	13.8	4.7	7.4	8.5	10.6	8.1	14.9	6.3	4.7	13.1	2.5	9.2	6.2
14	16.4	23.9	19.7	22.2	17.7	17.8	3.7	7.8	13.7	7.7	6.7	10.1	3.9	3.2	4.3	14.4	3.5	8.9	7.9
15	18.7	24.4	15.4	26.3	19.0	19.3	12.7	8.4	3.9	13.6	10.1	7.1	12.1	6.9	15.5	6.4	8.9	6.1	11.3
16	18.2	23.4	16.5	22.5	19.8	20.0	5.5	10.0	11.6	5.1	5.4	5.4	10.0	3.1	8.2	13.8	5.9	6.2	7.1
17	18.9	26.5	18.3	23.2	16.8	17.8	13.1	5.8	2.8	4.4	11.8	9.4	5.9	2.7	7.3	2.4	3.3	7.8	9.9
18	15.2	23.2	19.7	26.4	19.2	19.2	8.7	3.8	9.6	3.8	6.5	5.8	7.6	4.7	15.8	10.9	9.6	10.3	9.6
19	17.4	25.3	16.9	25.7	18.2	18.2	2.2	3.4	2.6	3.4	7.3	11.9	2.5	4.2	5.0	15.0	6.0	7.9	5.6
20	16.2	26.4	18.1	25.2	17.3	16.7	9.7	3.7	4.3	2.1	6.0	8.2	6.2	8.3	14.0	6.1	6.3	5.3	10.7
21	15.9	26.0	16.9	26.7	16.5	16.9	10.3	8.5	10.4	10.4	8.7	9.2	9.8	8.5	2.5	2.9	12.4	6.0	9.3
22	19.7	22.9	16.4	26.6	17.1	18.0	2.8	4.0	2.5	13.2	8.2	8.0	7.8	3.8	8.8	10.7	12.7	10.6	6.9
23	19.8	26.1	19.0	25.6	18.9	17.4	5.7	6.4	12.0	9.3	11.0	9.7	10.4	2.9	15.6	11.7	6.1	5.0	10.0
24	19.5	25.5	15.4	24.8	18.5	18.8	7.6	8.8	10.2	3.3	5.9	9.3	5.8	11.1	2.3	7.2	2.4	10.7	5.2
25	19.0	24.1	17.1	26.9	18.5	19.1	6.9	9.5	9.9	7.0	5.6	11.6	2.4	3.3	14.5	3.1	3.1	6.8	8.0
26	16.8	26.9	19.4	25.3	17.2	17.7	3.9	12.5	4.7	6.3	6.1	8.8	5.3	2.1	12.1	7.4	11.8	11.6	8.6
27	19.3	23.3	18.2	24.3	19.6	17.3	15.9	8.9	3.2	2.5	5.2	6.5	4.4	6.0	7.6	7.0	13.8	8.7	7.6
28	16.2	23.8	16.7	27.0	18.2	20.6	6.7	11.2	5.7	5.7	9.9	6.4	9.3	2.5	3.8	5.4	4.2	6.4	5.8
29	18.7	25.9	17.7	24.4	19.3	16.9	14.6	4.6	13.4	8.4	8.8	11.0	3.2	4.9	8.6	7.6	13.6	11.9	12.2
30	18.2	26.7	17.6	22.4	20.2	17.5	2.9	6.5	13.2	3.1	11.4	9.6	7.0	7.4	8.7	12.6	9.9	8.6	5.5
31	17.2	23.4	18.0	22.9	20.7	20.7	10.6	8.1	5.5	2.3	9.2	11.2	4.9	8.0	4.5	3.8	2.6	5.5	8.9
32	18.3	23.7	17.0	22.6	19.9	18.9	11.5	8.0	5.0	13.0	12.3	8.7	12.9	8.8	2.8	5.3	14.4	9.5	11.5
33	18.0	24.9	16.3	26.8	18.0	18.6	6.8	5.6	7.6	2.6	11.7	5.1	4.3	7.5	12.8	15.4	3.0	9.0	9.1
34	15.1	24.7	15.1	25.2	19.1	18.7	11.6	5.7	7.0	2.4	12.4	11.7	8.4	5.3	9.6	3.0	4.5	10.2	11.7
35	19.9	22.6	17.5	23.9	19.6	18.1	4.3	4.4	7.3	14.7	8.0	7.2	2.0	9.8	7.5	6.7	4.3	6.9	12.1
36	18.5	22.1	17.7	22.1	17.8	17.6	4.9	3.6	2.7	2.7	9.4	7.5	9.5	8.9	3.1	14.1	7.7	7.5	11.2
37	17.9	25.8	15.6	23.8	18.8	20.3	14.2	7.5	3.1	5.3	5.0	11.8	7.2	6.5	15.4	9.3	8.8	9.6	7.4
38	16.7	25.4	17.9	26.7	17.2	19.6	15.8	3.1	4.5	6.1	10.0	10.4	13.9	5.0	7.9	13.5	8.2	5.9	6.3
39	17.2	22.5	15.2	23.4	18.4	19.7	2.3	2.6	8.9	3.6	5.3	7.4	3.7	4.4	7.4	5.1	14.8	10.4	8.8
40	16.3	24.3	20.0	23.4	20.6	19.4	15.1	4.5	10.7	12.4	11.2	6.8	5.1	2.8	9.3	8.9	2.1	8.2	9.2
41	15.7	23.1	15.9	23.9	17.8	19.1	4.4	9.4	15.9	4.1	9.6	10.3	3.3	7.9	11.9	8.6	8.1	6.6	12.4
42	15.8	24.6	19.9	22.8	16.7	19.6	12.6	6.0	2.9	3.0	10.7	5.6	5.5	7.7	4.4	10.0	6.4	10.9	8.1
43	16.5	23.9	19.1	23.1	19.7	19.5	2.6	9.3	5.9	6.7	12.0	5.9	13.7	5.9	12.7	5.5	12.9	10.0	6.1
44	19.2	22.9	17.8	22.1	17.4	20.4	12.9	8.6	5.8	13.1	9.0	8.9	7.9	3.7	6.4	7.7	12.2	12.1	5.9
45	15.4	24.9	18.4	26.0	18.6	18.8	5.1	4.2	5.2	15.7	8.6	12.4	6.3	8.4	12.9	7.3	2.0	9.1	10.9
46	18.4	25.1	16.6	25.0	20.2	16.8	2.0	12.9	6.5	4.2	7.9	6.6	14.1	5.8	7.2	4.7	5.5	7.2	11.9
47	17.4	25.7	15.7	25.4	20.1	16.7	6.5	6.8	10.3	7.9	6.3	6.1	12.0	7.3	12.6	7.9	14.5	12.4	8.3
48	17.6	25.7	15.8	22.0	17.0	17.2	3.5	4.3	3.7	7.4	11.5	10.2	9.4	7.6	10.5	10.2	8.3	12.2	10.8
49	16.4	26.7	17.3	24.4	18.3	20.2	3.3	2.1	15.3	8.0	10.2	7.8	6.8	10.9	3.5	5.2	2.3	9.3	9.5
50	16.0	22.1	17.4	24.9	19.4	18.0	10.4	8.2	2.0	15.2	5.8	10.0	12.3	2.6	5.1	5.0	7.6	6.5	10.3
51	15.3	22.4	18.5	24.2	16.9	18.3	13.2	7.0	8.2	8.1	9.5	10.5	4.2	5.4	13.0	7.5	15.3	7.7	5.1
52	15.1	25.2	17.9	22.3	19.5	18.3	11.1	2.4	6.2	8.9	6.6	12.0	5.4	8.7	4.9	9.4	15.1	6.3	10.2
53	19.6	23.2	18.5	24.7	16.6	18.4	7.7	10.2	8.3	10.8	11.3	12.2	7.1	6.8	2.9	11.2	5.0	5.8	7.7
54	17.8	26.3	15.6	23.0	17.4	19.7	3.0	10.1	6.3	10.1	10.8	11.3	6.9	4.3	6.5	2.3	14.0	6.7	6.7
55	15.0	27.0	16.8	22.9	20.7	18.5	7.0	10.7	6.6	7.2	7.4	7.6	4.8	2.2	2.1	4.1	5.2	9.4	8.5
56	19.4	22.4	15.9	23.5	19.1	17.9	10.2	2.0	13.0	6.4	10.5	11.5	6.4	2.4	14.1	4.9	10.1	7.3	5.4
57	19.7	26.0	18.7	24.1	18.7	20.5	7.1	4.9	11.4	3.2	7.7	12.1	12.4	2.0	5.9	4.5	13.0	7.1	10.6
58	20.0	23.6	18.6	25.9	20.3	20.0	7.3	5.0	11.2	7.8	9.3	7.9	6.7	9.7	12.4	2.5	2.9	12.3	8.2
59	18.1	26.8	19.6	26.2	20.5	19.0	9.2	3.3	7.2	10.9	6.8	8.3	10.5	4.5	2.2	8.3	15.2	10.5	7.2
60	19.2	26.6	19.8	24.0	18.3	20.2	11.7	6.2	4.8	10.0	7.1	5.0	9.6	8.6	10.6	14.2	4.7	8.5	7.8
61	17.3	26.2	16.0	23.6	17.5	20.1	9.4	2.2	8.5	9.5	7.2	5.5	3.5	3.6	5.2	11.9	10.6	5.4	12.3
62	15.4	25.2	19.3	22.6	18.0	17.1	4.6	9.2	6.1	13.3	7.6	6.3	14.2	6.2	7.7	8.5	7.1	5.6	6.5
63	17.5	24.0	19.5	22.4	18.9	18.2	10.8	6.9	10.6	8.3	5.5	11.4	15.4	5.6	15.2	6.3	7.4	12.0	9.0
64	18.5	22.0	19.2	25.5	18.1	16.5	4.7	10.5	7.8	2.2	12.2	9.5	8.3	4.1	6.3	6.0	11.1	11.0	10.5
65	17.7	23.0	16.1	24.5	16.7	17.7	16.0	5.1	11.7	5.9	7.8	10.7	13.4	5.7	4.2	12.2	7.8	9.7	12.0
66	16.6	23.7	19.5	25.1	18.6	18.6	5.3	3.5	5.1	11.1	12.1	6.2	2.1	3.0	8.4	2.8	15.7	8.1	11.0
67	16.1	25.5	18.8	26.5	19.4	17.1	10.9	5.9	6.0	12.8	10.9	10.6	3.6	9.5	6.6	4.3	8.0	11.4	5.0
68	16.7	24.2	18.2	25.5	16.6	20.5	6.2	7.6	13.3	6.5	9.1	12.3	13.5	7.0	3.6	11.1	9.3	11.5	6.6
69	17.1	26.2	17.2	24.7	19.3	19.9	14.3	7.3	13.5	2.9	11.9	6.7	4.1	9.9	10.1	5.7	15.8	8.3	7.3
70	16.9	22.2	18.7	25.8	19.9	17.5	3.1	2.9	9.5	3.9	5.1	10.9	10.2	7.8	6.8	4.2	9.2	7.4	5.3

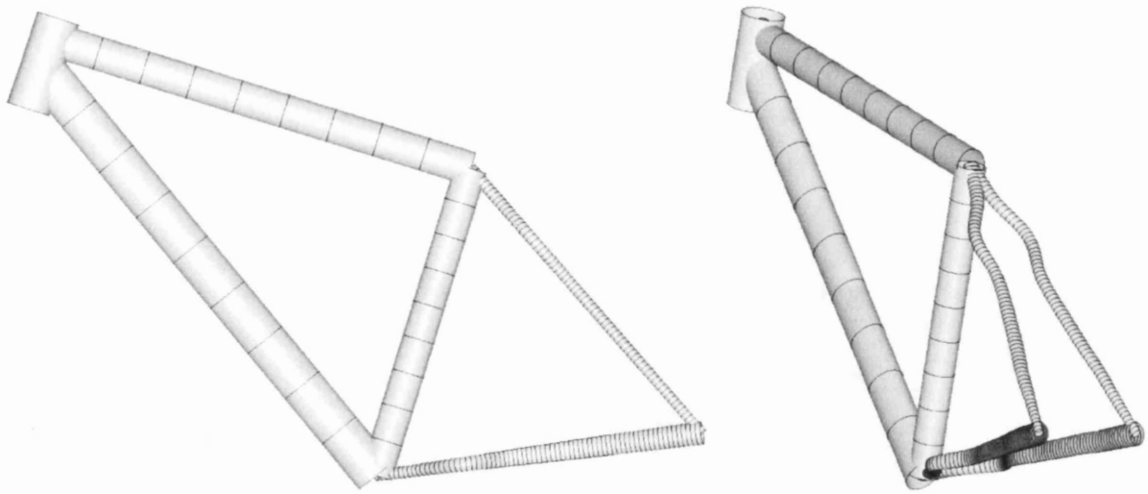


Figure C-1: Frame geometry for experiment 7 in Example 1.

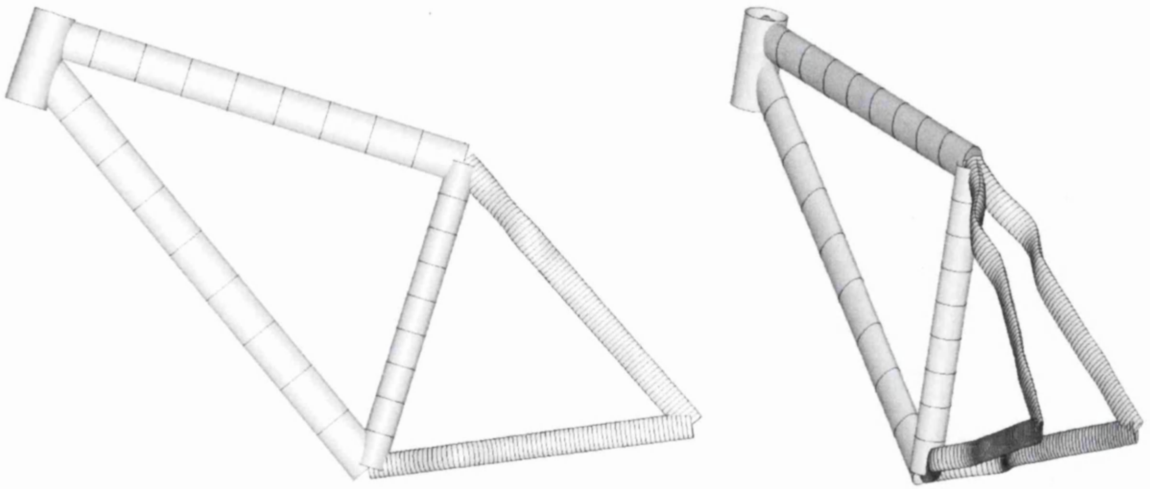


Figure C-2: Frame geometry for experiment 34 in Example 1.

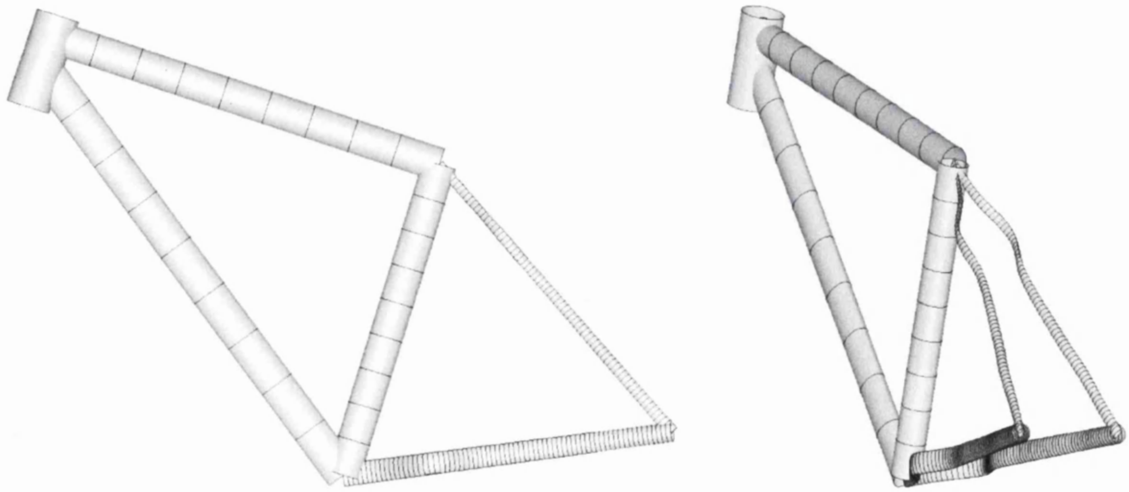


Figure C-3: Frame geometry for experiment 39 in Example 2.

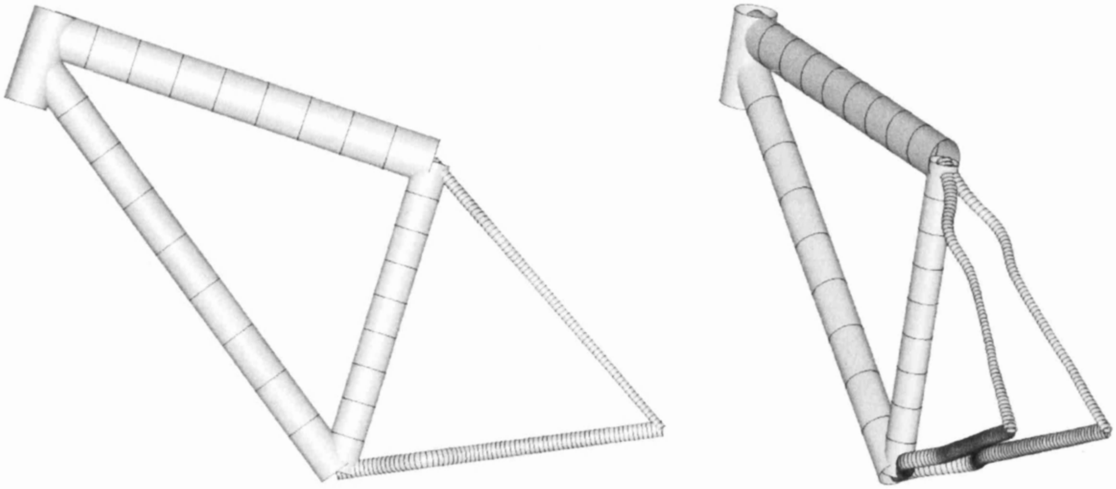


Figure C-4: Frame geometry for experiment 60 in Example 3.

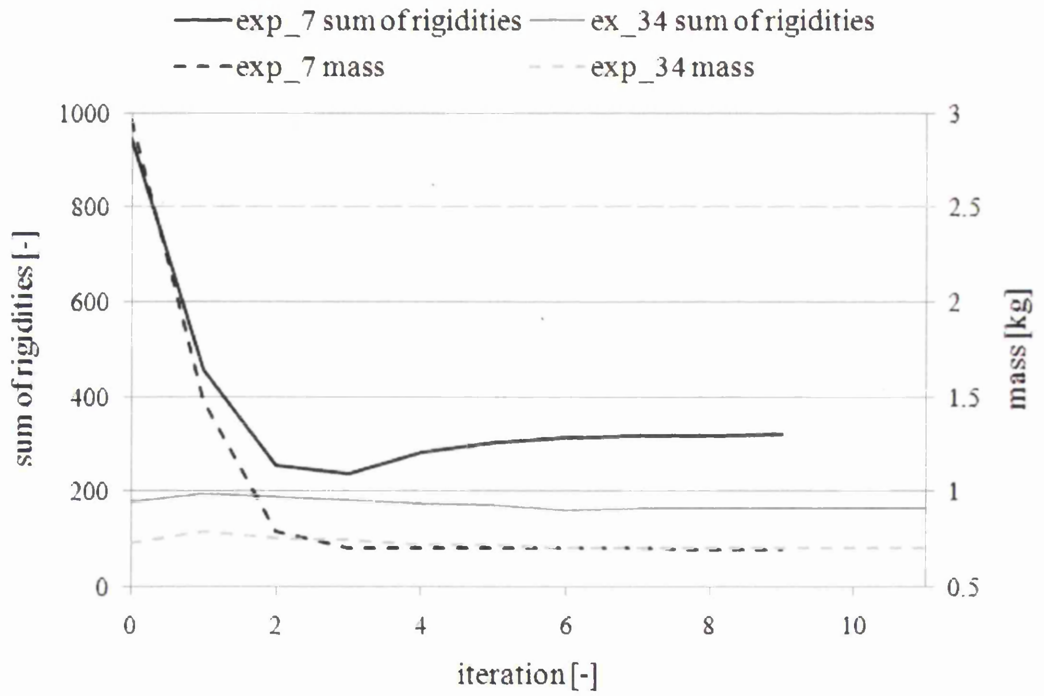


Figure C-5: Example 1, experiments 7 and 34. Development of the sum of rigidities and the mass throughout the optimization.

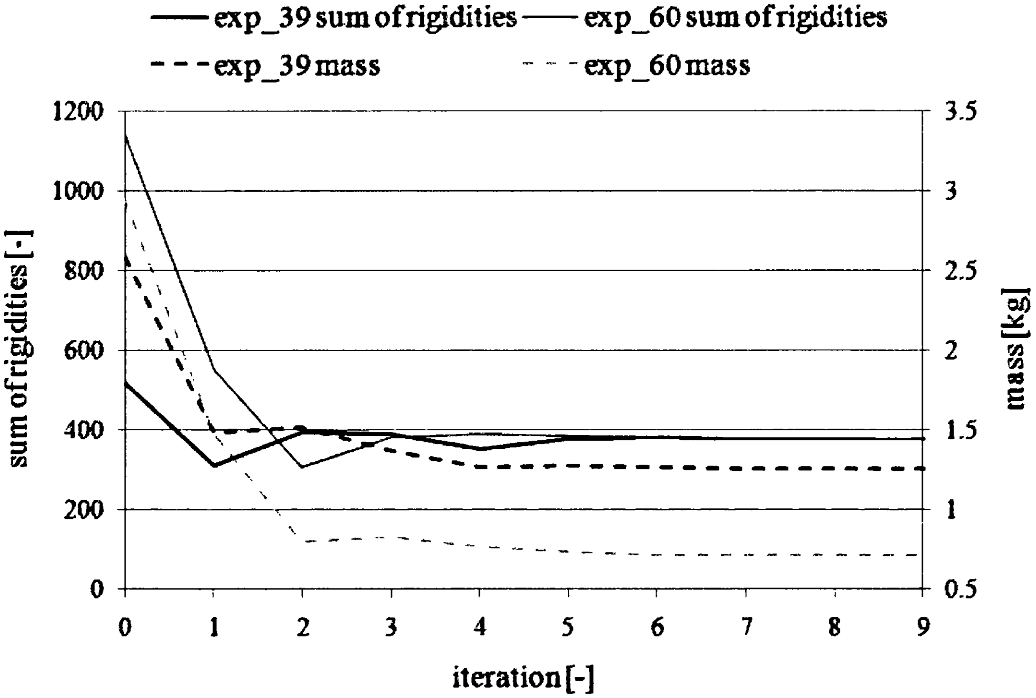


Figure C-6: Example 2, experiments 39 and 60. Development of the sum of rigidities and the mass throughout the optimization.

List of Publications

Journal Papers

Fuerle F., Sienz J., "Formulation of the Audze-Eglais Uniform Latin Hypercube Design of Experiments for Constrained Design Spaces", submitted to: Advances in Engineering Software (paper provisionally accepted), 2010

Fuerle F., Sienz J., "A Decomposed Optimization Framework for the Design of Carbon-fibre Competition Bicycle Frames", submitted to: Finite Elements in Analysis and Design (paper under review), 2010

Conference Papers

Fuerle F., Sienz J., " Determination of a Secant Modulus from Material Tests for the Efficient Simulation of Polyethylene Pipes in Ringstiffness Tests", *31st CILAMCE – Congresso Ibero-Latino-Amerciano de Metodos Computacionais em Engenharia,* Buenos Aires, Argentina, November 2010

Sienz J., Luege M., Fuerle F., " Computational Design of a Composite Race Car Seat Meeting International Regulation Standards", *31st CILAMCE – Congresso Ibero-Latino-Amerciano de Metodos Computacionais em Engenharia,* Buenos Aires, Argentina, November 2010

Fuerle F., Sienz J., Samaras V., "A Software Tool for Automated Simulation and Optimum Design of Structured-Wall PE Pipes", *ECT2010 - 7th International Conference on Engineering Computational Technology*, Valencia, Spain, September 2010

Fuerle F., Sienz J., Mowbray K., Walklett M., "Optimum Blade Design for a Novel Wind Turbine", *EngOpt2010 - 2nd International Conference on Engineering Optimization, Lisbon, Portugal*, September 2010

Pittman J., Fuerle F., Sienz J., Samaras V., "A temperature, strain and strain rate dependent linear elastic model for high density PE implemented via a secant modulus ", *Polymer Processing Society - 26th Annual Meeting*, Banff, Canada, July 2010

Fuerle F., Sienz J., " Optimal Latin Hypercubes Design of Experiments for Constrained Design Spaces", *WCSMO-8 - 8th World Congress on Structural and Multidisciplinary Optimization*, Lisbon, Portugal, June 2009

Fuerle F., Sienz J., Heuberger M., Holness J., B. Philip, "Structural Size and Shape Optimization of Carbon-Fibre Competition Bicycle Frames", *WCSMO-8 - 8th World Congress on Structural and Multidisciplinary Optimization*, Lisbon, Portugal, June 2009

Fuerle F., Sienz J., Innocente M., Pittman J., Samaras V., Thomas S., "Ringstiffness Evaluation and Optimization of Structured-Wall PE Pipes", *CST2008 - 9th International Conference on Computational Structures Technology*, Athens, Greece, September 2008

Fuerle F., Sienz J., Innocente M., "Size and Shape Optimization of Structured-Wall PE Pipes in Ringstiffness Tests", *7th ASMO-UK/ISSMO International Conference on Engineering Design Optimization*, Bath, UK, July 2008

Fuerle F., Sienz J., Innocente M., Pittman J., Samaras V., S. Thomas, " Performance Modelling and Optimization of Structured Wall PE-pipes", *Polymer Processing Society - 24th Annual Meeting*, Salerno, Italy, June 2008