NTNU – Trondheim
Norwegian University of
Science and Technology

Parameterization and Multiobjective Optimization

Espen Nilsen

Mechanical Engineering
Submission date: June 2013
Supervisor: Terje Rølvåg, IPM

Norwegian University of Science and Technology
Department of Engineering Design and Materials

THE NORWEGIAN UNIVERSITY
OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF ENGINEERING DESIGN
AND MATERIALS

MASTER THESIS SPRING 2013
FOR
STUD.TECHN. ESPEN NILSEN

PARAMETERIZATION AND MULTIOBJECTIVE OPTIMIZATION
Parameterisering og multiobjektiv optimalisering

Modeling and multiobjective optimization is one of the main tasks in the EC project SupLight. In SupLight two industrial cases are selected for design/parameterization and optimization based on a fully integrated multiobjective optimization loop. These are one control arm from Raufoss Technology (RT) and an aircraft door connection arm from Hellenic Aerospace Industry (HAI).

The main goal is to document the process as well as benchmarking how single- and multiobjective design optimization can be applied and implemented to improve products.

The following tasks must be completed:

1. Find an optimal strategy for model parameterization of the control arm in NX based on the following criteria:
    a. Identify product requirements and optimization criteria for the control arm
    b. Design flexibility for robust design perturbations(selection of design variables and optimal modeling strategy for the largest impact on optimization criteria and design requirements)
    c. A minimum number of design variables (short simulation times)
    d. Smart selection of linked design expressions (minimum number of user inputs)
2. Parameterize the control arm and perform manual or automatic optimization to verify if the above criteria are met.

3. Perform multiobjective optimization based on Mode Frontier and NX of the parameterized control arm:
    a. Study optimization theory and identify the best algorithms for the design optimization considering the given requirements (from point 1a)
    b. Implement the multiobjective optimization loop in Mode Frontier for the control arm
    c. Perform design optimization and evaluate the results wrt. ease of use and final product requirements

The thesis should include the signed problem text, and be written as a research report with summary both in English and Norwegian, conclusion, literature references, table of contents, etc. During preparation of the text, the candidate should make efforts to create a well arranged and well written report. To ease the evaluation of the thesis, it is important to cross-reference text, tables and figures. For evaluation of the work a thorough discussion of results is appreciated.

Three weeks after start of the thesis work, an A3 sheet illustrating the work is to be handed in. A template for this presentation is available on the IPM's web site under the menu "Masteroppgave" (http://www.ntnu.no/ipm/masteroppgave). This sheet should be updated one week before the Master's thesis is submitted.

The thesis shall be submitted electronically via DAIM, NTNU's system for Digital Archiving and Submission of Master's thesis.

NTNU
Norges teknisk-
naturvitenskapelige universitet
Institutt for produktutvikling
og materialer

Torgeir Welo
Head of Division

Terje Rølvåg
Professor/Supervisor

Preface

This thesis is written as the final work that completes a masters degree in Product Development and Materials Engineering at NTNU (Norwegian University of Science and Technology).

It contains detailed descriptions and illustrations that are intended to provide the reader with a complete recipe for performing a similar task. This is underpinned by a number of A3 sheets in the appendix to give an extra good understanding of the specific steps needed to carry out this task. This guide has been produced in addition to solving the problems specified in the thesis.

This thesis is one of three master theses written as a contribution to the EC research project SuPLight here at NTNU this semester. I have worked closely with the two other authors who has had a positive impact on the thesis and contributed to a great ending of the master program. Some of the material contained in this thesis such as one of the appendix and some small sections is contributed by the other authors and vice versa.

I would like to thank Terje Rølvåg, one of the SuPLight contributors that initiated this thesis for a lot of help and guidance. Application engineer Adam Thorp at Esteco Nordic AB has been providing essential files needed to configure modeFRONTIER in the correct way with NX.

Espen Nilsen

Contents

List of Figures

List of Tables

Abstract

This thesis has dealt with the overall process regarding optimization of a control arm, with a focus on providing a practical guide. The first chapter explains how to approach these kinds of issues and outlines a strategy for model parameterization. The product requirements has been identified. The overall goal is to reduce weight by 10 % while maintaining the same stiffness. Then the most suited design expressions has been chosen to ensure the best possible outcome of the following optimizations.

When the optimal parameterization strategy has been chosen the next chapter deals with an automatic optimization in the internal optimization module in NX to verify if the criteria related to model parameterization was met. This generated weight savings of 2,67 % which is a quite good result.

The final chapter regarding multiobjective optimization contains all the background with configurations, theory and optimizations done within modeFRONTIER. Five different algorithms was tested in modeFRONTIER with default preferences and benchmarked against each other. This was done as a way to determine their efficiency with no prior information that could help the efficiency of the algorithms. Two of the best algorithms were chosen for a more extensive optimization with a subsequent local search to find out if they could generate better results. The best result generated weight savings of 3,56 %, which was produced by the Hybrid algorithm.

Sammendrag

Denne masteroppgaven har tatt for seg hele prosessen vedrørende optimalisering av en bærearm, med fokus på å lage en praktisk guide. Det første kapitlet beskriver hvordan en angriper denne typen problemstilling og tar for seg en mulig strategi for å gjennomføre modell parameterisering. Kravspesifikasjonen ble redegjort for, og innebærer et mål om 10 % vektreduksjon samtidig som dagens stivhet blir ivaretatt. De beste design variablene ble valgt for å sikre seg det beste utfallet av optimaliseringene.

Når den mest optimale optimaliseringsstrategien ble kartlagt, ble det neste kapitlet viet til en automatisert optimalisering i NX sin interne optimaliseringsmodul. Dette verifiserte at kriteriene vedrørende parametriseringen ble tilfredsstilt. Optimaliseringen genererte et resultat på 2,67 %, som er vurdert som et godt resultat.

Det siste kapitlet som omhandler multiobjektiv optimalisering inneholder bakgrunnen med konfigurasjonen, teorien og optimaliseringene gjort i modeFRONTIER. Fem ulike algoritmer ble testet med standard innstillinger og sammenliknet med hverandre. Dette ble gjort for å avgjøre effektiviteten til algoritmene uten tilleggsinformasjon som kunne favorisere noen av de. De to beste algoritmene ble valgt for å gjøre nye grundigere optimaliseringer med et påfølgende lokalt søk for å bestemme om disse kunne generere bedre resultater. Det beste resultatet gav en vektreduksjon på 3,56 % og ble utført med Hybrid algoritmen.

Nomenclature

ARSM	Adaptive Response Surface Method
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CTETRA(10)	Four sided solid element with ten grid points
DOE	Design of Experiments
FEA	Finite Element Analysis
FEM	Finite Element Method
ISF	Incremental Space Filler
MCDM	Multi Criteria Decition Making
MOGA – II	Multi-Objective Genetic Algorith
MOGT	Multi-Objective Game Theory
MOSA	Multi-Objective Simulated Annealing
NX	NX Unigraphics, a CAD and FEM Software
RSM	Response Surface Models
SQP	Sequential Quadratic Programming

Introduction

0.1 SuPLight

SuPLight is a multidisciplinary EC research project that involves participants from the industry as well as academic environments. The project is a multidisciplinary research project combining metallurgy, continuum mechanics, structural mechanics, optimization algorithms, tolerance analysis and life cycle analysis. This multidisciplinary represents a challenge, but is also necessary to yield result that exceeds todays knowledge on the topic. SuPLight stands for Sustainable and efficient production of light weight solutions. As the worlds energy needs get higher every day, one need to find sustainable solutions that reduces todays energy consumption. Production of virgin aluminium is very energy consuming and more extensive use of recycled aluminium in addition to lightweight optimized solutions can reduce overall energy consumption. The main objective of the project is to provide sustainable lightweight industry solutions based on wrought alloy aluminium. Some of the goals are:

- Gain a 50 % increased weight/performance ratio through optimization.
- More than 75 % post consumer recycled wrought aluminium alloy is to be used.
- New methodologies and tools for holistic eco-design of products, processes and manufacturing

These goals are SuPLight overall goals. This thesis aims to develop new methods and concepts that can be used by the industry. The objective is to generate a guide that may simplify the optimization process for future parts and that might highlight some of the most common questions. This thesis will start from scratch with respect to necessary model preparations and practical concerns regarding optimization. To help in making the thesis a real life case, a car part is chosen with its associated requirements. It will in the following be dealt with requirements listed in section 1.1. These product requirements was provided by Raufoss Technology AS.

The SuPLight project was initiated some time before this thesis was made. Supervisor Terje Rølvåg had been one of the contributors in this project from the start. Terje handed down the support arm model in addition to some documentation regarding the SuPLight project [1, 20].

0.2 Introduction to Optimization

Computer Aided Engineering (CAE) has grown rapidly the last decades and have become essential for engineers today. A wide range of different tools for design and analysis helps to streamline the product development process. The structural design process may contain aspects from various fields of engineering, the best decisions are often made with respect to different factors like stiffness, strength, constructability and aesthetics.

The method used to achieve this weight goal is design optimization. This is an expression that is part of a category called structural optimization. Structural is understood as "*any assemblage of materials which is intended to sustain loads*"[12]. Optimization refers to the process of making something as good as possible. Structural optimization then means the process of making an assemblage sustain loads as good as possible. The word good can mean different things based on different goals. Sometimes this means as light as possible or as stiff as possible. This goal would be very easy to achieve if constraints were not introduced. With both goal and a constraint it is possible to search after an optimal solution. Structural optimization may deal with one or more types of optimization problems.

- Sizing optimization involves different size parameters of the structure to be optimized (for example, thickness of a beam). It is common to relate this kind of optimization to a problem where you have a truss containing beams, and you change the dimension of each beam and letting the cross section of these beams be constant.

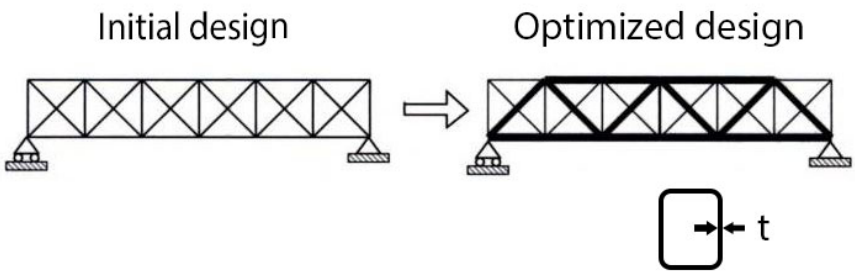


Figure 1: Sizing optimization problem

- Shape optimization optimizes a structure without changing the models topology. The entire model keeps solid without creating additional holes in it. Shape optimization has an interdisciplinary character, meaning it can be used on a wide arrange of problems. This kind of optimization is more complex than the sizing optimization. It involves mathematical disciplines as partial differential equations, approximations of these and theory of nonlinear mathematical programming. In terms of three dimensional models and finite element methods, advanced software are required.

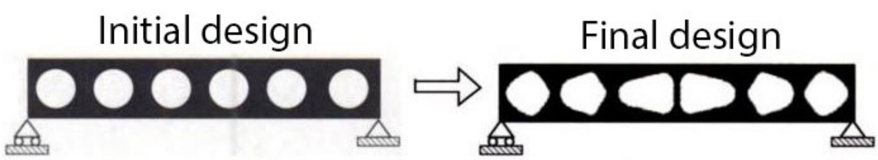


Figure 2: Shape optimization problem

- Topology optimization optimize the topology by, for example, making holes in the component. The algorithm changes the density of elements, controlling the stiffness contribution from that particular element. The result from the optimization must be interpreted and smoothed by the engineer, as output geometries are highly organic shapes that must be processed before production. Today, gradient-based algorithms are mostly implemented in commercial software, however new algorithms are continuously developed. For a more detailed description, see [10].

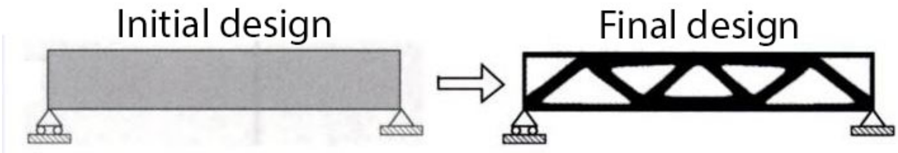


Figure 3: Topology optimization problem

Shape optimization is often done by geometry parametrization coupled with result evaluation by a solver. There are similarities between size and shape optimization which makes them difficult to distinguish from each other. In terms of this project shape optimization will be referred to as geometry optimization, though the literature equate geometry optimization with shape optimization. This match the terms used by NX. This thesis will address this type of optimization as both geometry and design optimization [12][14].

0.3 Introduction to the Control Arm Part

Figure 4: Support arm

Figure 4 support arm, is a front lower control arm produced by Raufoss technology and assembled onto cars such as an Opel insignia [16].

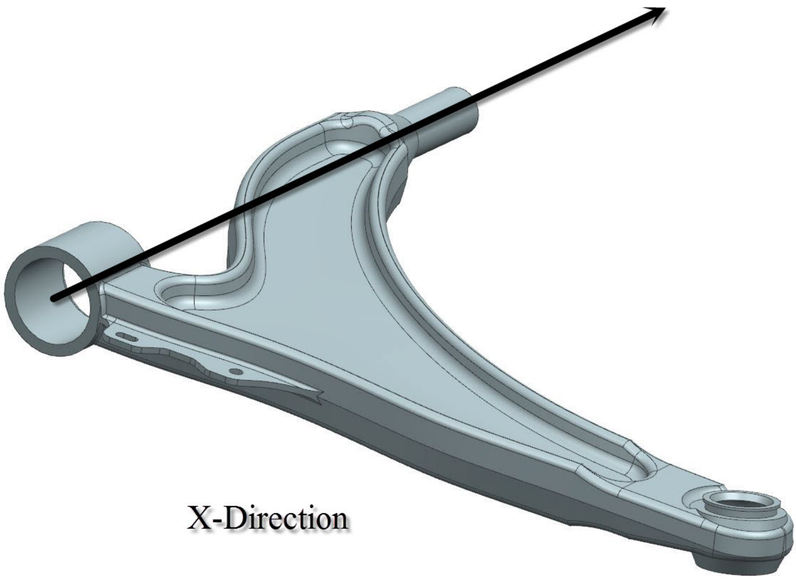


Figure 5: Definition of x-direction

Figure 5 shows positive x-direction to avoid any misunderstandings.

0.4 Material Data

Aluminium 6082 T6		
Mass Density	2700	kg/m^3
Mechanical		
Young's Modulus	70000	N/mm^2(MPa)
Poisson's Ratio	0,33	-
Shear Modulus	26315	N/mm^2(MPa)
Strength		
Yield Strength	320	N/mm^2(MPa)
Ultimate Tensile Strength	350	N/mm^2(MPa)

Table 1: Material data aluminium[13]

Nylon_massless		
Mass Density	1,2	kg/m^3
Mechanical		
Young's Modulus	4000	N/mm^2(MPa)
Poisson's Ratio	0,4	-
Strength		
Yield Strength	58	N/mm^2(MPa)

Table 2: Material data bushings

Table 1 and 2 contains the material data obtained from the supervisor, Edupack and Raufoss Technology AS for use in all future simulations [13, 1].

0.5 Approach

Figure 6: Overall workflow

Figure 6 shows the upcoming approach used in this thesis. This shows a good procedure to ensure the best possible optimization results. The CAD model needs to be parameterized before performing an optimization. Then a sensitivity analysis can detect important parameters and possible errors before a multiobjective optimization is executed. All steps are carried out in this order in the following chapters.

Chapter 1

Model Parameterization and Base Line

1.1 Product Requirements

- The stiffness in x-direction has to be the same as todays design
- 10 % weight reduction

To meet the product requirement and to get quantified data for further use in simulations it is necessary to establish a base line for the current part. The base line will form a standardized basis for future simulation throughout the project. This is done by analyzing the current part by use of a load range of 13 kN. The output values (displacement) obtained by these simulations will function as requirement that has to be met during future optimization of the part (Fx = 0,5 ± 6,5 kN)(stress range of 220 Mpa). Load range was specified by Raufoss Technology [1].

The thesis will return to this issue regarding use of expressions that ensures rigid model updates.

1.2 Design Flexibility for Robust Design Perturbation

The base line simulation is done on a simplified edition of a model provided by the supervisor. The original model had a lot of complicated geometry that caused error to design updates. The type and amount of simplifications has been chosen on the basis of manual experimental design changes. This has been done to minimize errors to future model updates and optimization simulations. The experiments involve changing expressions that seems convenient to be able to edit. In practice this means changing one by one expression as much as one would think is appropriate with regard to future potential for improvement. In this case the simplifications were quite substantial to ensure robust design perturbations.

Stiffness

It is previously stated that the stiffness to the optimized control arm has to be equal to or greater than todays model. The task is then to establish data regarding stiffness in x-direction and von mises stresses with the given load range on todays model. Some bushings with different material data (see table 2) is added to the control arm to allow axial movement. This has been done to reduce peak stresses (especially in notches in the transition between the hydro bushing and the rest of the model).

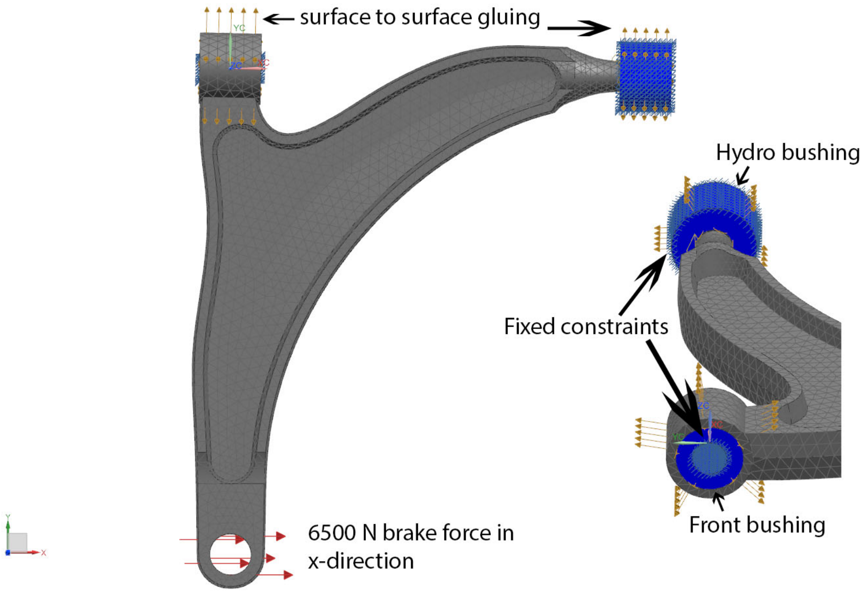


Figure 1.1: Load case, x-direction

Figure 1.1 shows the attached load of 6,5 kN in x-direction. The surface to surface gluing is attached to prevent relative motion between bodies. The bushings is also shown with the fixed constraints attached to them. The mesh is made with 8 mm CTETRA(10) elements, a reasonable size considering the amount of time each simulation would take. A function called mesh control has been added to the conical end where the hydrobushing is placed to prevent the mesh from resizing thus preventing possible peak stress concentrations.

From this simulation displacement and stress is extracted as can be seen in figure 1.2 and 1.3.

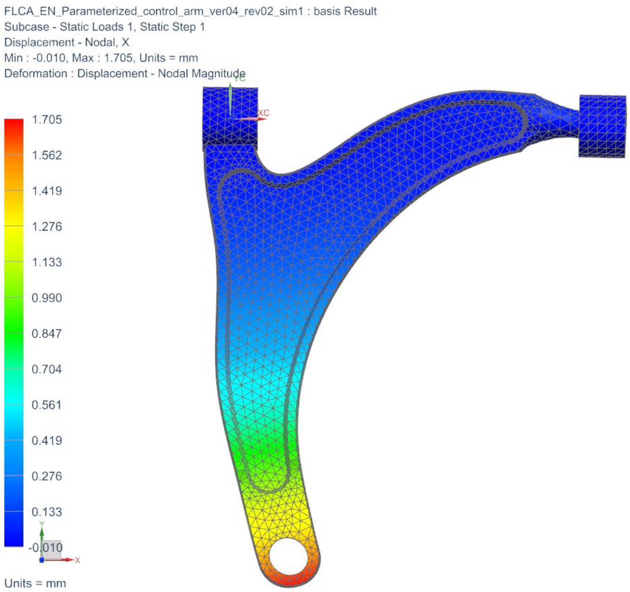


Figure 1.2: Displacement in x-direction

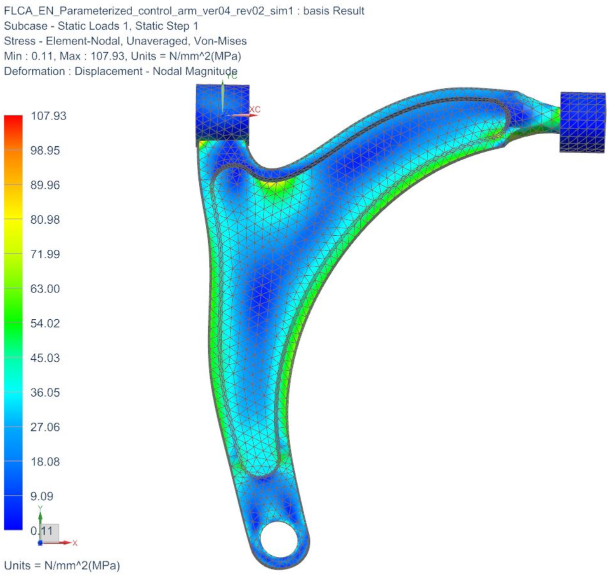


Figure 1.3: von mises stress

	Load case x-direction	Weight
Displacement [mm]	1,705	1,448 kg (14,2048 N)
Von mises stress [Mpa]	107,86	[kg]

Table 1.1: Displacement and stress data

These are the stress and displacement results with the chosen material listed in table 1.1.

Comparison

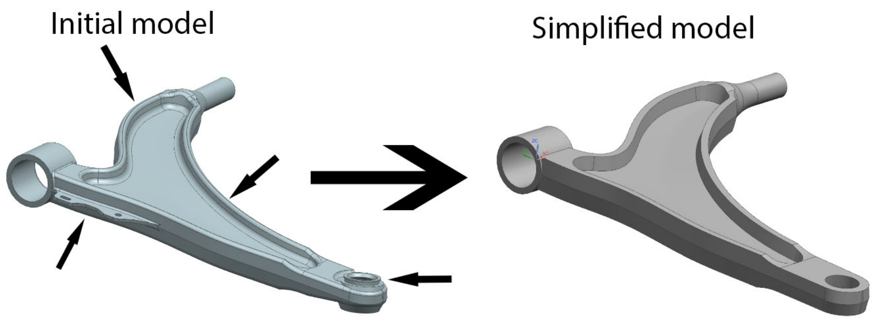


Figure 1.4: Illustration of the simplifications done on the model

The simplified model is capable of design changes that does not corrupt the model update after changes to the expressions. In table 1.2 the consequences of changes to the model has been listed. Simplifications has been a necessity to ensure design variables that may have the largest impact on the optimization criteria.

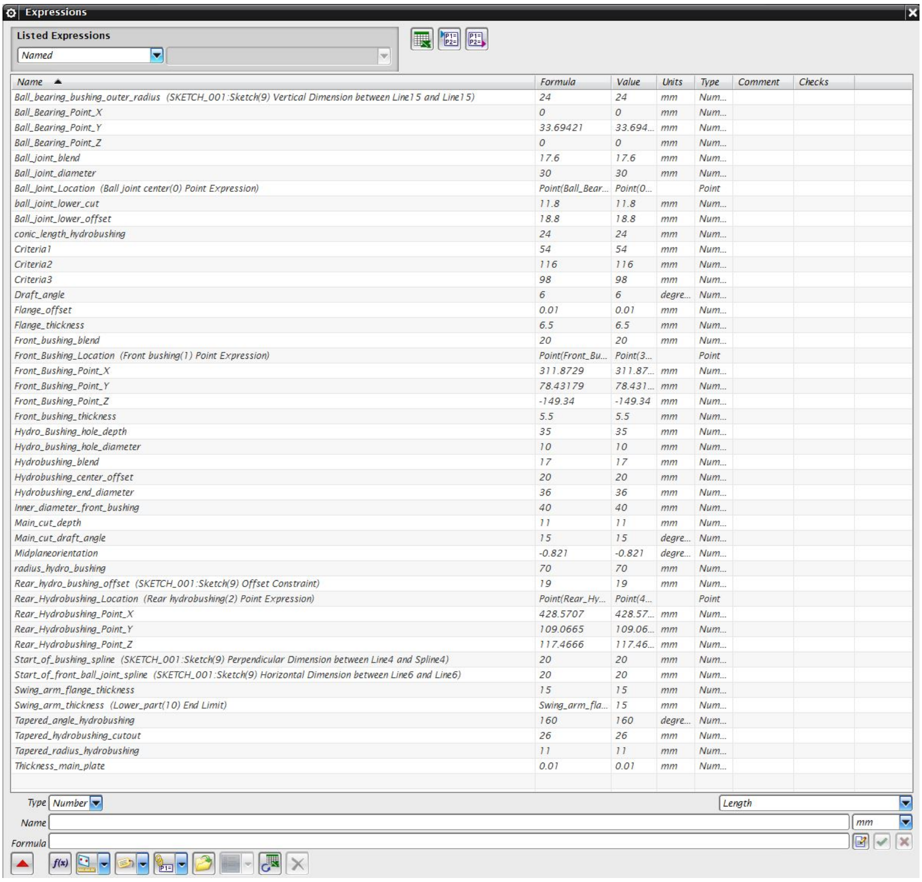
	Original model	Simplified model	Relative changes in displacement
Displacement x- direction [mm]	1,416	1,705	16,95 % worsening (deterioration)
Weight [kg]	1,4539	1,448	0,40 % improvement

Table 1.2: Relative change in displacement due to simplifications

The simplifications does not affect the overall goal achievements as the main goal is a relative improvement in weight loss. The overall weight goal is 10 % weight reduction.

1.3 Design Variables

By the time this thesis was made, the supervisor was already involved in this project and had a working model of the support arm. The part was modeled from the ground up by creating expressions on all dimensions as seen below. Most of the expressions existed when the model was handed down. In addition to expressions associated with sketches, some expressions were added using NX synchronous modeling as well.



Name	Formula	Value	Units	Type	Comment	Checks
Ball_bearing_bushing_outer_radius (SKETCH_001:Sketch(9) Vertical Dimension between Line15 and Line15)	24	24	mm	Num...		
Ball_Bearing_Point_X	0	0	mm	Num...		
Ball_Bearing_Point_Y	33.69421	33.694...	mm	Num...		
Ball_Bearing_Point_Z	0	0	mm	Num...		
Ball_joint_blend	17.6	17.6	mm	Num...		
Ball_joint_diameter	30	30	mm	Num...		
Ball_Joint_Location (Ball joint center(0) Point Expression)	Point(Ball_Bear...	Point(0...		Point		
ball_joint_lower_cut	11.8	11.8	mm	Num...		
Ball_joint_lower_offset	18.8	18.8	mm	Num...		
conic_length_hydrobushing	24	24	mm	Num...		
Criteria1	54	54	mm	Num...		
Criteria2	116	116	mm	Num...		
Criteria3	98	98	mm	Num...		
Draft_angle	6	6	degre...	Num...		
Flange_offset	0.01	0.01	mm	Num...		
Flange_thickness	6.5	6.5	mm	Num...		
Front_bushing_blend	20	20	mm	Num...		
Front_Bushing_Location (Front bushing(1) Point Expression)	Point(Front_Bu...	Point(3...		Point		
Front_Bushing_Point_X	311.8729	311.87...	mm	Num...		
Front_Bushing_Point_Y	78.43179	78.431...	mm	Num...		
Front_Bushing_Point_Z	-149.34	-149.34	mm	Num...		
Front_bushing_thickness	5.5	5.5	mm	Num...		
Hydro_Bushing_hole_depth	35	35	mm	Num...		
Hydro_bushing_hole_diameter	10	10	mm	Num...		
Hydrobushing_blend	17	17	mm	Num...		
Hydrobushing_center_offset	20	20	mm	Num...		
Hydrobushing_end_diameter	36	36	mm	Num...		
Inner_diameter_front_bushing	40	40	mm	Num...		
Main_cut_depth	11	11	mm	Num...		
Main_cut_draft_angle	15	15	degre...	Num...		
Midplaneorientation	-0.821	-0.821	degre...	Num...		
radius_hydro_bushing	70	70	mm	Num...		
Rear_hydro_bushing_offset (SKETCH_001:Sketch(9) Offset Constraint)	19	19	mm	Num...		
Rear_Hydrobushing_Location (Rear hydrobushing(2) Point Expression)	Point(Rear_Hy...	Point(4...		Point		
Rear_Hydrobushing_Point_X	428.5707	428.57...	mm	Num...		
Rear_Hydrobushing_Point_Y	109.0665	109.06...	mm	Num...		
Rear_Hydrobushing_Point_Z	117.4666	117.46...	mm	Num...		
Start_of_bushing_spline (SKETCH_001:Sketch(9) Perpendicular Dimension between Line4 and Spline4)	20	20	mm	Num...		
Start_of_front_ball_joint_spline (SKETCH_001:Sketch(9) Horizontal Dimension between Line6 and Line6)	20	20	mm	Num...		
Swing_arm_flange_thickness	15	15	mm	Num...		
Swing_arm_thickness (Lower_part(10) End Limit)	Swing_arm_fla...	15	mm	Num...		
Tapered_angle_hydrobushing	160	160	degre...	Num...		
Tapered_hydrobushing_cutout	26	26	mm	Num...		
Tapered_radius_hydrobushing	11	11	mm	Num...		
Thickness_main_plate	0.01	0.01	mm	Num...		

Figure 1.5: All dimensions

Here all of the expressions associated to this part is listed. This list gets long after modeling a part because you may not know exactly what kind of expressions you want for the optimization in the beginning. This results in naming a lot of expressions just in case you need them later. From these expression it is essential to choose the most suited variables that ensures a great impact on the optimization criteria.

1.4 Smart Selection of Linked Design Expressions

In the optimization process one will have to figure out which of the dimensions (from figure1.5) that might have the most beneficiary impact on the model with regards to the weight objective. This has been done intuitively and experimental by adding expressions in sketches and make sure that all of the new and existing expressions will allow design changes. The range of which the expressions would allow change had to be identified by trial and error. Both the synchronous modeling expressions exist of two surfaces merged in one expression to make the optimization configuration easier, and to make the expression more effective with respect to the objective.

Number	1	2	3	4	5	6	7
Name	Flange_thickness	Ball_joint_blend	Flange_offset	Front_bushing_blend	Thickness_main_plate	Hydrobushing_blend	Swing_arm_flange_thickness
Value[mm]	6,5	17,6	0,01	20	0,01	17	15
Lower limit[mm]	6	17,1	0,01	18	0,01	16	12
Upper limit[mm]	7	18,1	3	22	1	18	16
Type	Sketch based	Sketch based	Synchronous modeling	Sketch based	Synchronous modeling	Sketch based	Sketch based

Figure 1.6: Selected design expressions

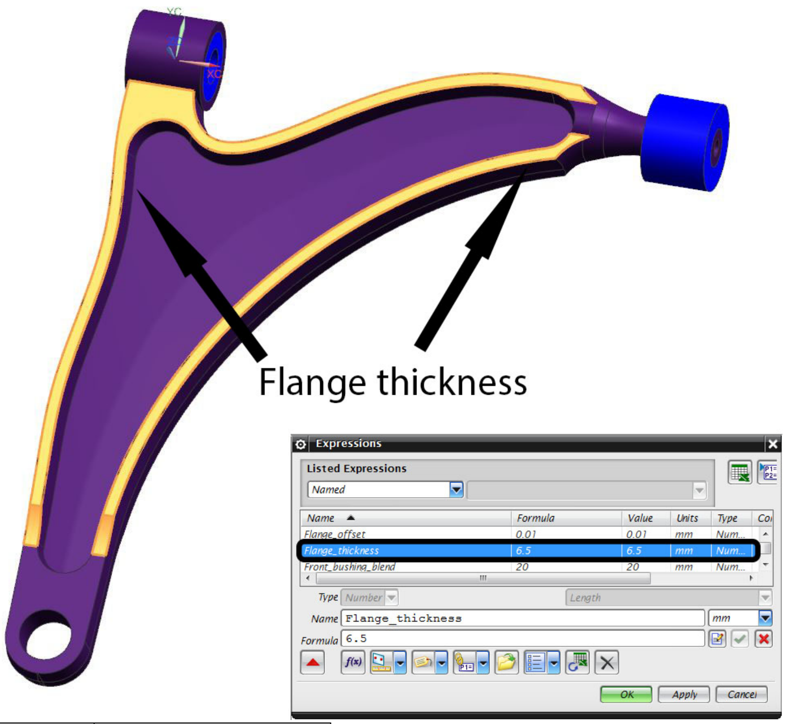
In the table above the chosen key dimensions is shown. The maximum variation is also listed. These are the main dimensions that has been considered important and most sensible. This selection of expressions represents the parameterization needed for an optimization. The following optimization simulation in NX will make use of the data listed above.

1.5 Design Expressions

In NX the variable designs has to be linked to a certain expression in the model. This is done by creating expressions (in: tools -> expressions (CTRL+E)). When this is done the created expression has to be defined in the model. This method is shown in detail in section: Creating Expressions. In the following sections the part is shown with an indication of what the expression is related to with the expression window highlighted.

Two of the parameters is linked to expressions made with synchronous modeling, in this case the synchronous modeling tool offset region window is shown.

1 Flange_thickness

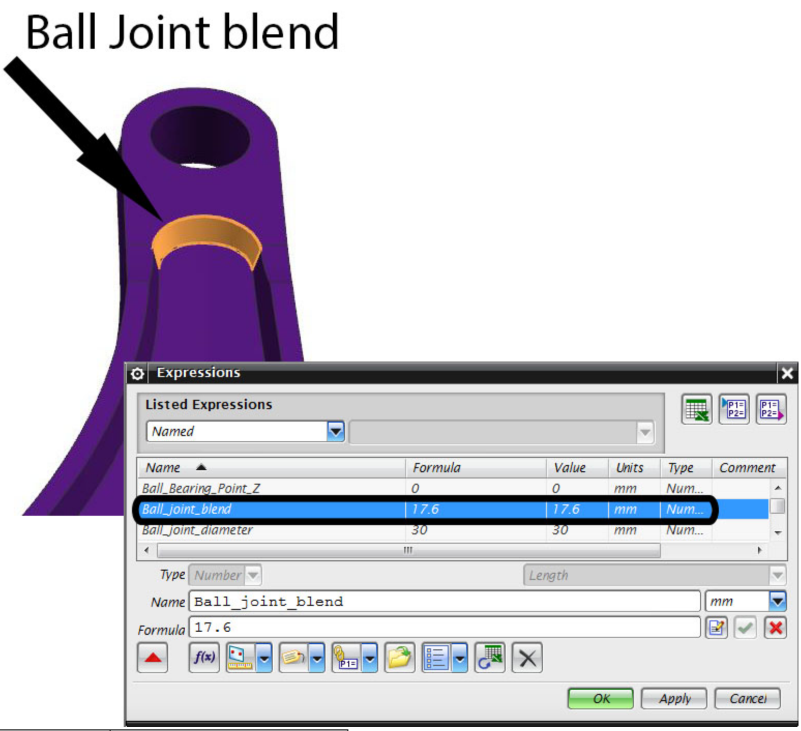


1	Flange_thickness
Lower limit[mm]	6
Upper limit[mm]	7

Figure 1.7: Flange_thickness

This expression is mirrored over XC - YC plane which means that both sides of the model is controlled by this expression.

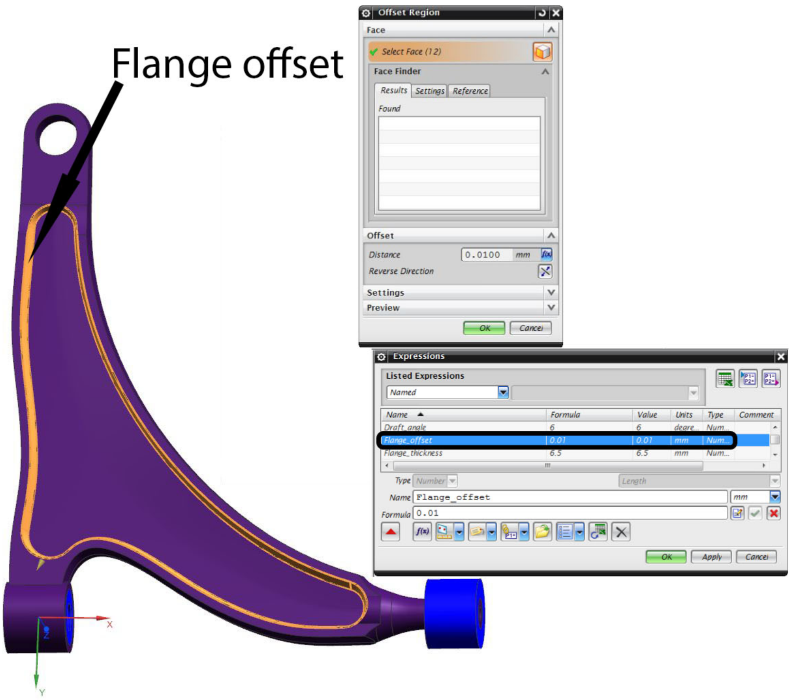
2 Ball_joint_blend



2	Ball_joint_blend
Lower limit[mm]	17,1
Upper limit[mm]	18,1

Figure 1.8: Ball_joint_blend

3 Flange_offset

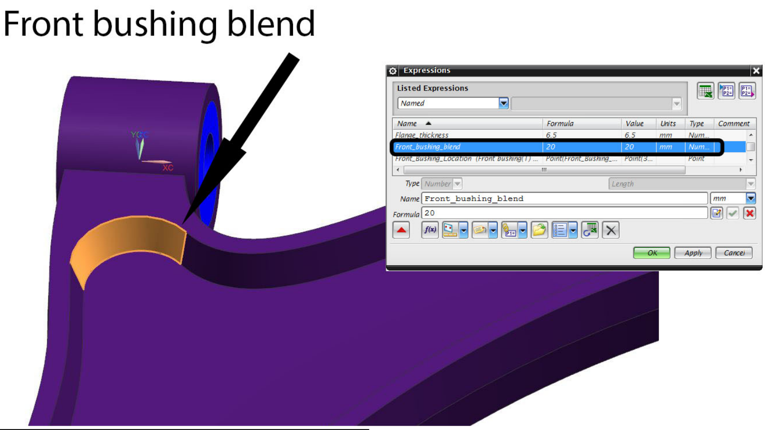


3	Flange_offset
Lower limit[mm]	0,01
Upper limit[mm]	3

Figure 1.9: Flange_offset

This expression is made with NX synchronous modeling function offset region. This is also a symmetric expression that moves the flange on both sides of the XC-YC plane. This expression moves the highlighted surface perpendicular to the paper plane.

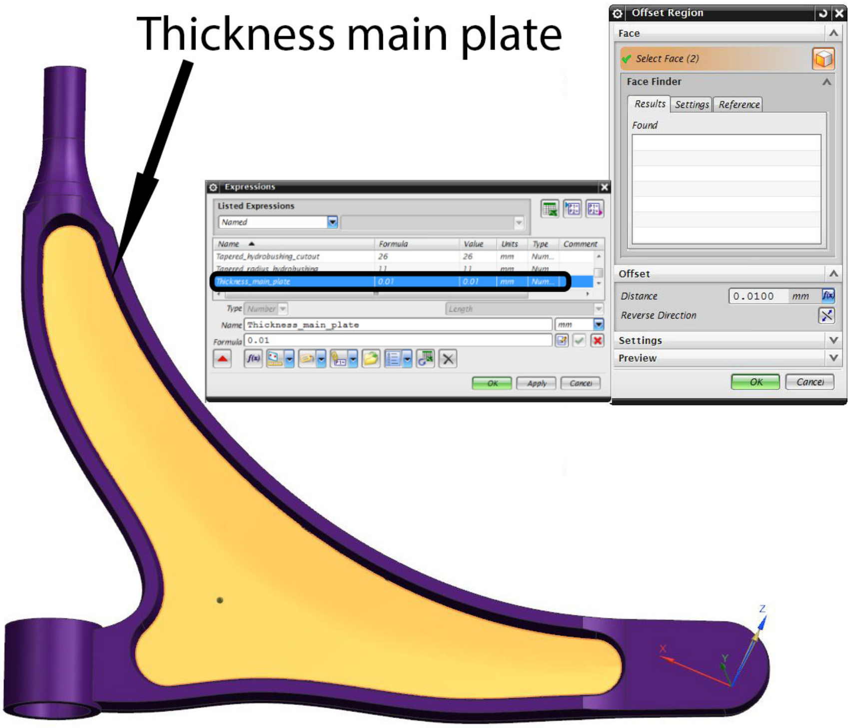
4 Front_bushing_blend



4	Front_bushing_blend
Lower limit[mm]	18
Upper limit[mm]	22

Figure 1.10: Front_bushing_blend

5 Thickness_main_plate



5	Thickness_main_plate
Lower limit[mm]	0,01
Upper limit[mm]	1

Figure 1.11: Thickness_main_plate

This expression is also made with synchronous modeling. The expression controls the face on both sides of the model.

Creating Expressions

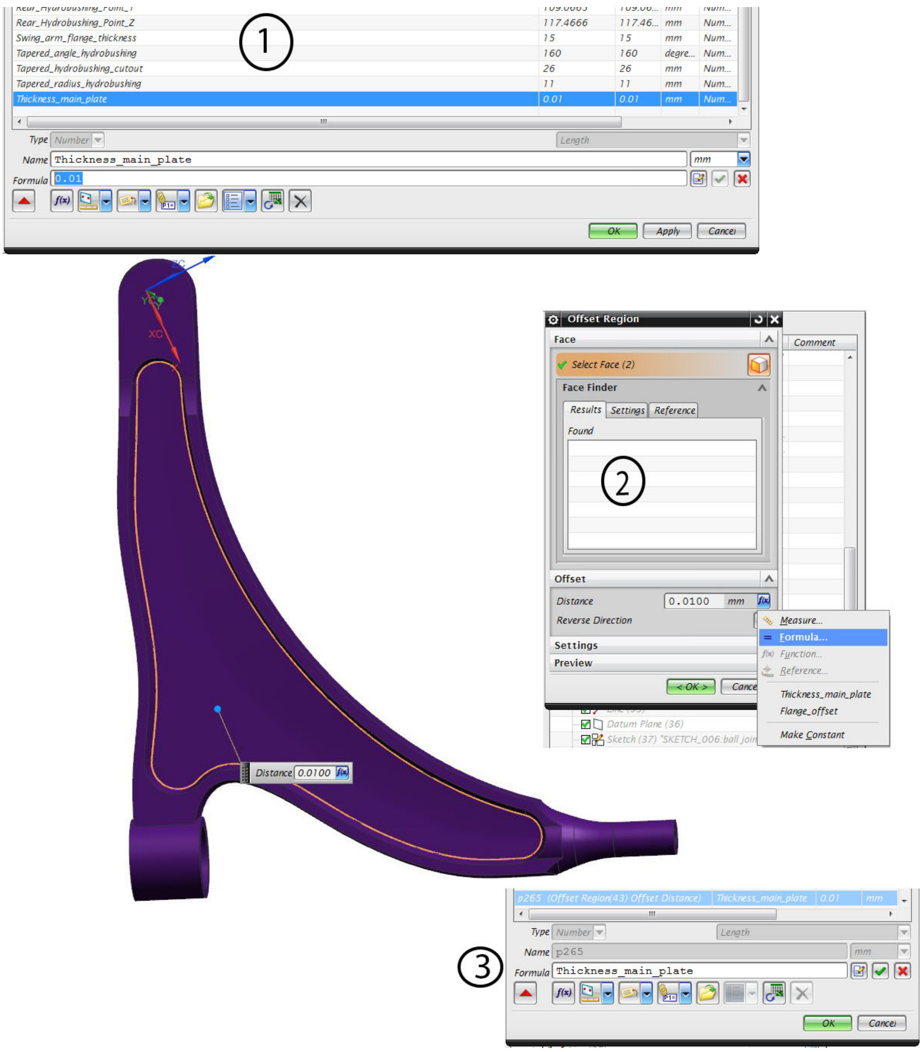
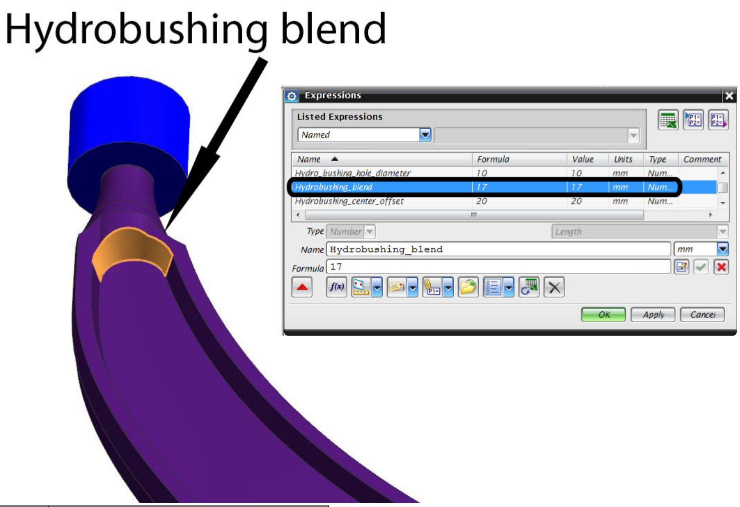


Figure 1.12: Creating expressions

The first step is to create an expression. This is done by opening the expressions menu and create a user defined expression. In this particular case synchronous modeling is used. The offset region menu is shown under step two. Here one have to define the length with a formula. In the last step it is necessary to find the user defined expression and double click on it. The expression should now be linked and editable in the expressions list.

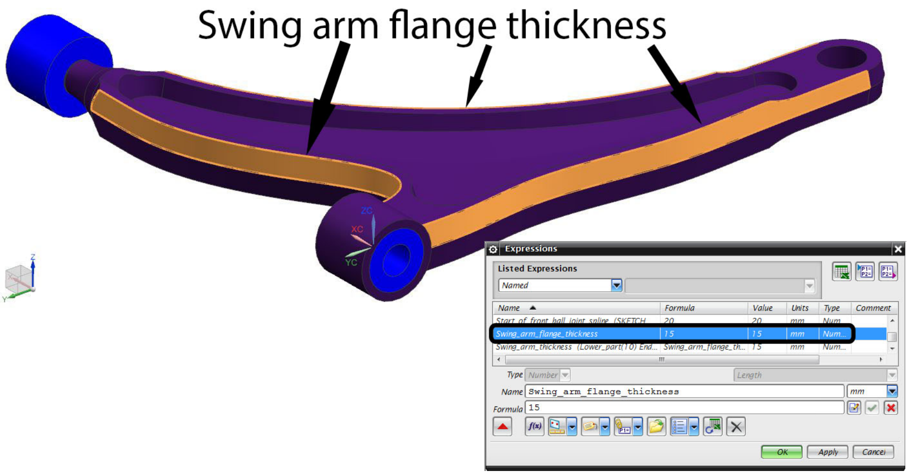
6 Hydrobushing_blend



6	Hydrobushing_blend
Lower limit[mm]	16
Upper limit[mm]	18

Figure 1.13: Hydrobushing_blend

7 Swing_arm_flange_thickness



g

7	Swing_arm_flange_thickness
Lower limit[mm]	12
Upper limit[mm]	16

Figure 1.14: Swing_arm_flange_thickness

Here the expression controls the height of the flange. This expression is mirrored over the mid plane in YC - XC plane so it controls the flange in both direction of this plane.

These are the expressions that were chosen based on intuition and manual sensitivity analysis before any optimization runs were initiated. The strategy presented in this chapter is deemed the most optimal in terms of this control arm part.

1.6 Discussion

In the above section precautions has been taken to ensure the optimal basis for further optimization. But there might still be raised concerns to sources of error in the decisions made. In the previous subsection 1.2 the modifications to the initial design is shown. This is of course not ideal but a necessity for further optimization. As previously mentioned it is not a serious error in this context as the main goal is a relative improvement in weight in addition to comparison and documentation of software and methods. Furthermore, it can also be associated uncertainty to the choice of design parameters. Since most of them are chosen with the background of intuitive thinking combined with trial and error, all of them might not be the most suited parameters. Some of them are made symmetric to control two sides of the part at the same time. This might make the variables less flexible than if they were controlled separately. It could become a problem if the two sides responds different to the same design change, however this does not seem to be a problem in this case.

There could also have been some good parameters (from 1.5) left unused. Some of the design parameters upper and lower limits are limited by the models ability to update overall designs without getting update errors in the history tree. It is conceivable that this could have been avoided to some extent by another initial design structure that allows bigger variation in the design space.

Some uncertainties is also associated with the load case. There are more considerations to make, such as buckling and multiaxial stresses. This has however been disregarded because of little relevance to the overall objective in this thesis.

Chapter 2

NX Optimization

2.1 Introduction

The thesis states that a manual or automatic optimization is to be run. The latter has been chosen because of curiosity and the fact that the latest available software provides this as a built-in feature. Based on the previous preparatory work in chapter 1, the following chapter will deal with an optimization run in the NX geometry optimization module as well as a sensitivity analysis. The optimization module runs a constrained single objective optimization, which means that it work towards a goal while maintaining its constraints requirements.

Once all of the preparations regarding base line, model updates and robust expressions is done, it is time to set up a geometry optimization to verify if the criteria regarding parameterization is met. Prior to this it is necessary to run a standard linear simulation with desired loads and constraints. This is because the geometry optimization will use this simulation as a basis for further optimization. Once the geometry optimization is chosen (within NX geometry module) it is possible to choose between Altair Hyperopt and global sensitivity analysis. The global sensitivity analysis will be discussed later on. The Altair Hyperopt is an adaptive response surface method (ARSM). Hyperopt uses a quadratic polynomial that is found and updated for each of the iterations, these are based on current and previous iterations. It uses a least square algorithm to define the polynomial (see Altair documentation for more detailed information about the algorithm [5]).

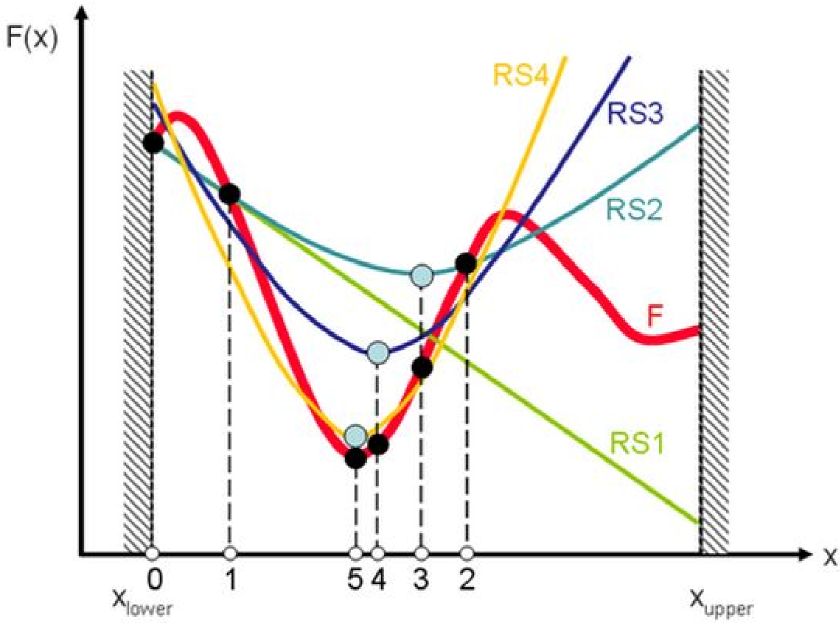


Figure 2.1: Altair Hyperopt

In figure 2.1, the algorithm tries to estimate a quadratic polynomial which will fit the actual function curve (F). As new designs are found along F, the response surface curve is updated (RS1, RS2, ..., RSx) until convergence is reached. In case the last quadratic polynomial curve does not converge sufficiently, the process is restarted from the first linear RS1 and quadratic response surfaces is generated for RS2, RS3 and so on [4, 18].

Since the curves are a quadratic polynomial, they are susceptible to find local minimum points.

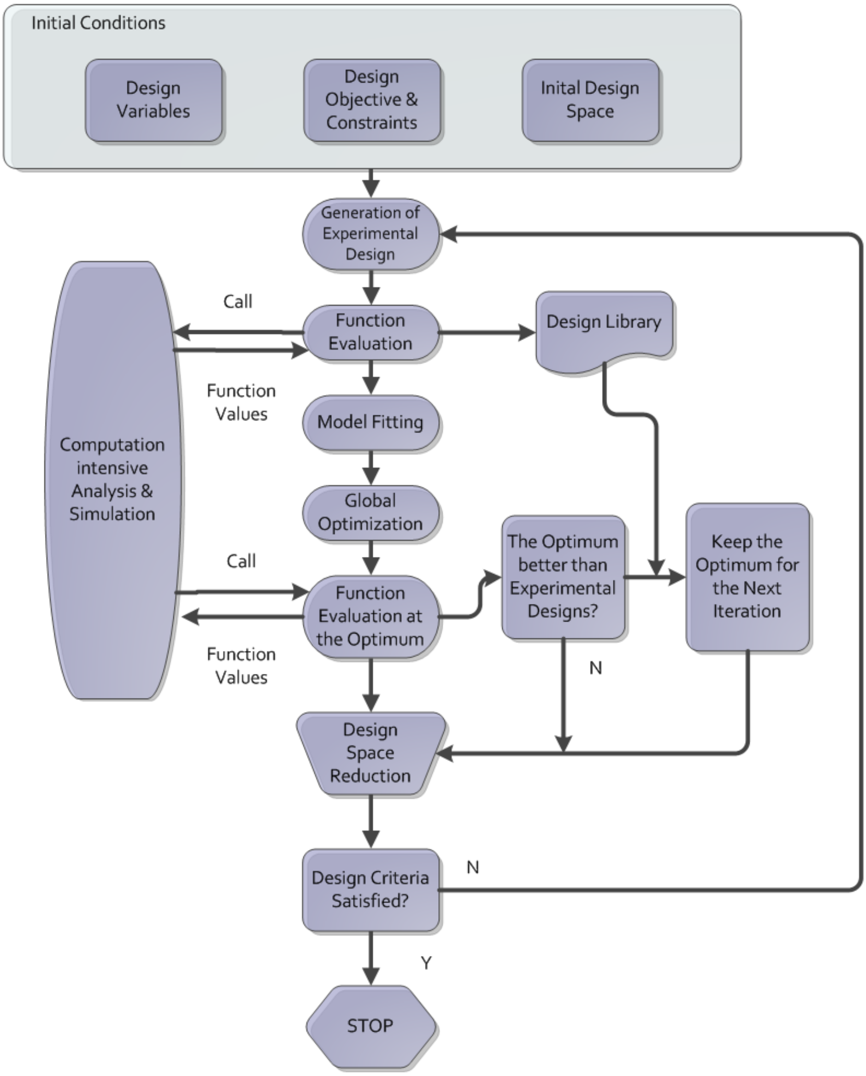


Figure 2.2: Flowchart ARSM

This chart shows the overall process of the ARSM procedure. The chart shown is a remake from [9].

2.2 Geometry Optimization

In the following section a detailed description of the optimization method is given.

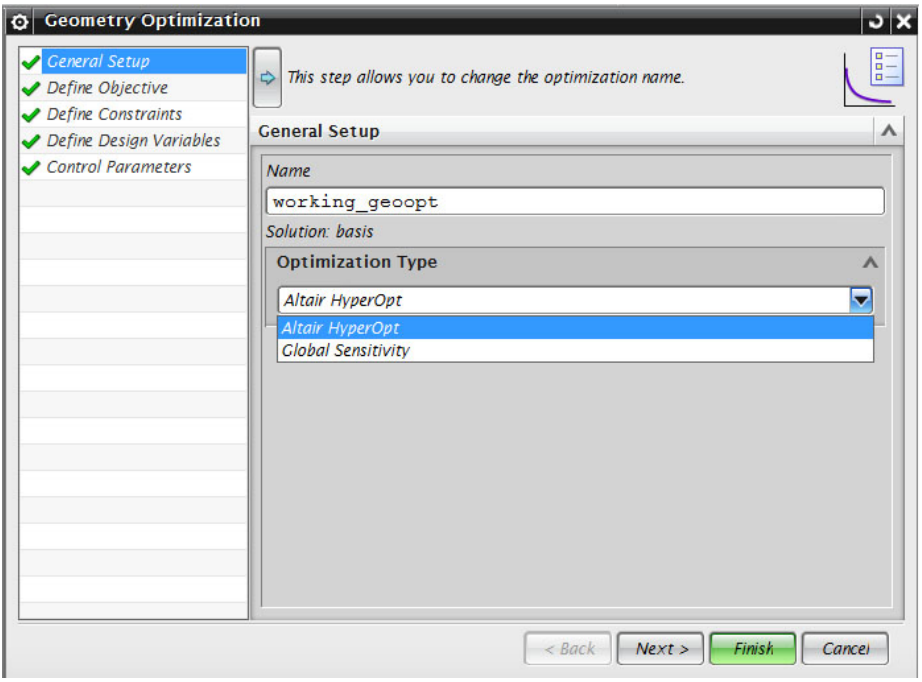


Figure 2.3: General setup

The next step is to define the objective of the optimization. The options weight, volume and result measure is available. It is also possible to apply an objective to a specific type of mesh.

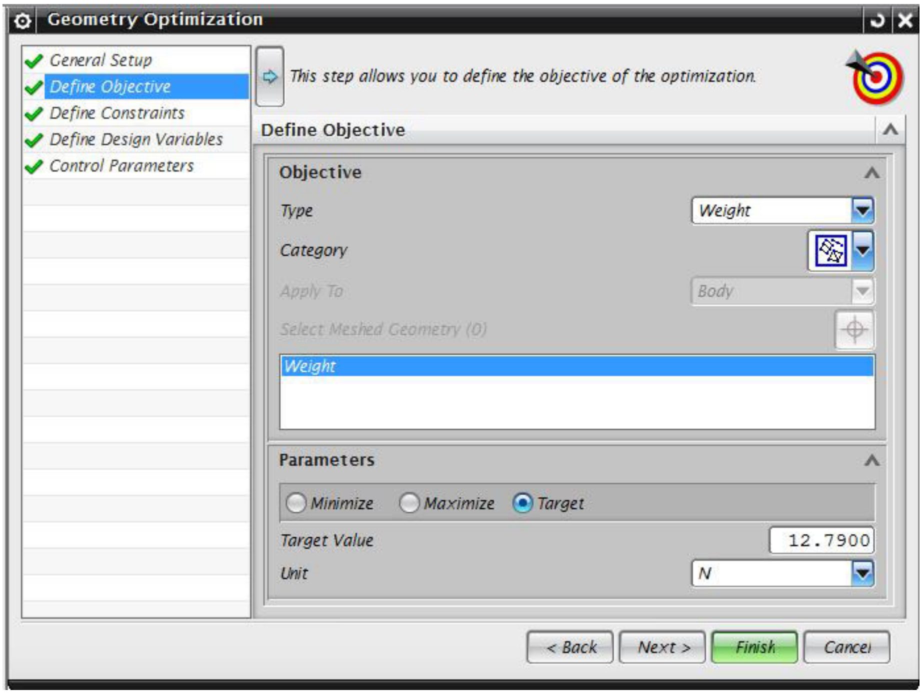


Figure 2.4: Define objective

Here it is chosen to meet a target weight given in Newton. The target of 10 % reduction is specified.

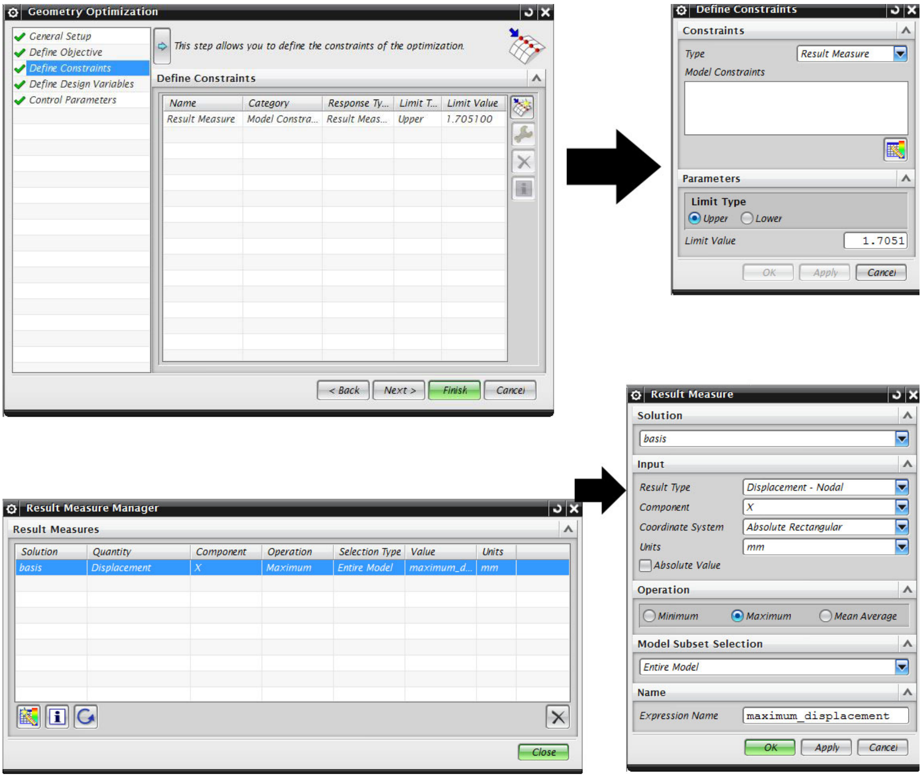


Figure 2.5: Define constraints

In the next step there are some more options to choose from. One can choose between six different constraints and define different coordinate systems. The picture shown in figure 2.5 shows how to link the necessary constraint from the initial simulation done prior to the geometry optimization initiation. The desired constraint is specified under the result measure tab.

It is also possible to choose more than one constraint if desired.

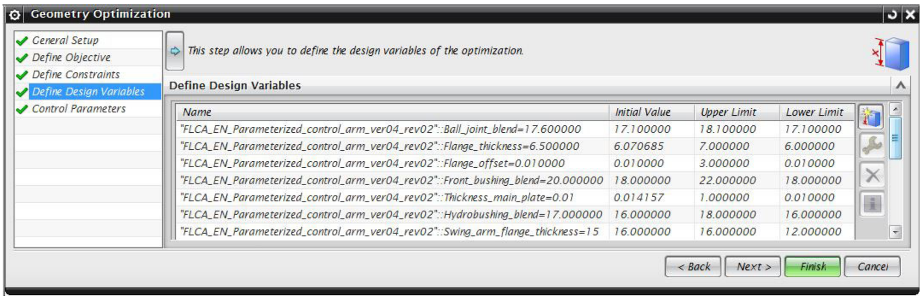


Figure 2.6: Define design variables

In this step, the design variables is defined. In the picture above the robust expressions previously chosen is displayed.

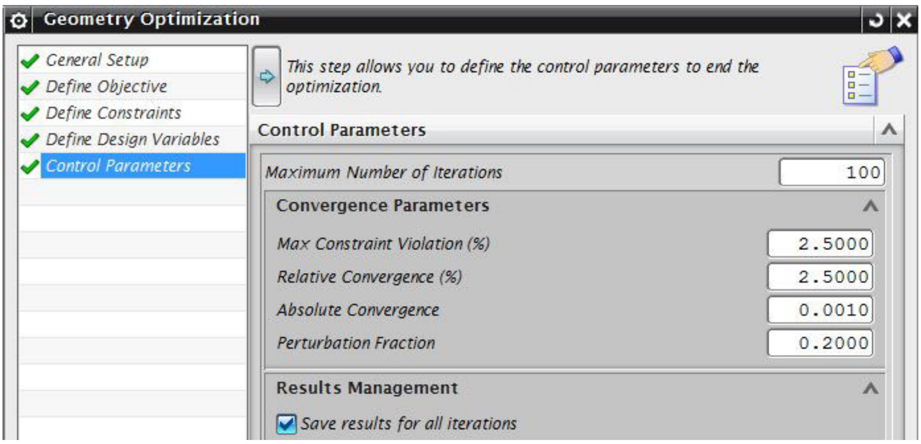


Figure 2.7: Control parameters

This is the last step containing some more options regarding goal achievement and design changes. None of these seem to have an enormous impact on the solution, they might come in handy when trying out extremities. Number of iterations can be kept high since they seldom seem to exceed about 30 iterations. Max constraint violations tells NX how much more deviation from target it can tolerate. Relative and absolute convergence decides when NX is satisfied with the results and terminates the iterations. Perturbation fraction is how big percentage of the predefined design space (proportion between upper and lower limit of the design variables) it is allowed to alter between each iteration.

2.3 Optimization Results

A number of optimization runs were executed with different settings. Table 2.1 shows relative results for each of the different combinations of goal achievement with stress and displacement. It is also compared with running the same analysis with or without specified weight target instead of minimizing it. In this table the reduction in weight is given in percentage. The trend shows that the goal achievement for runs with stress as constraint is a bit higher. The numbers in parentheses shows that this increase entails that the run violates the constraints to a greater extent. This means that the net improvement is worse than the results with displacement as constraint. From the spreadsheets provided in appendix D one can see that the fluctuations in stress results gives quite unreliable results.

	Goal achievement with weight target	Goal achievement without weight target
Stress	2,13 % (-5,28 %)	13,7 %(-17,24 %)
Displacement	1,3 %(0,84 %)	2,69 % (-0.17 %)

Table 2.1: Goal achievement, stress vs displacement

Because of this, all consecutive optimization runs uses displacement and weight as optimization criteria.

The best result from these optimization runs is shown in figure 2.2. NX generated spreadsheet data for each iteration, from this spreadsheet a graph was made showing weight savings for each iteration step. The original spreadsheet is located in appendix B.

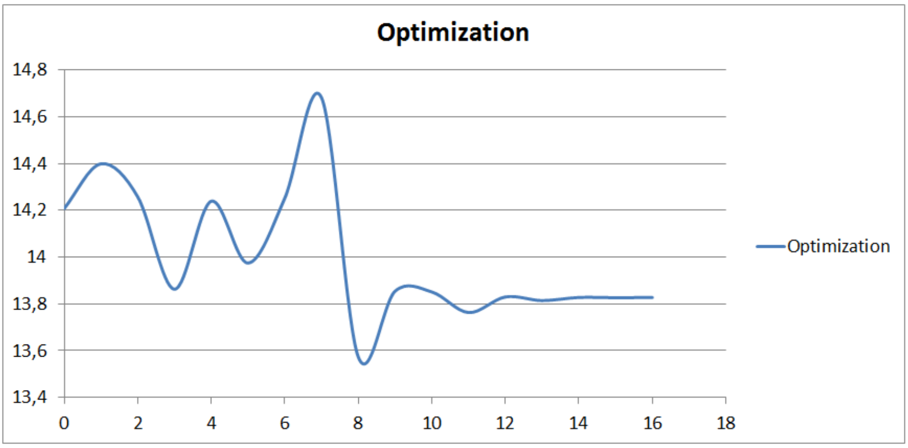


Figure 2.8: Optimization results

	Initial model [g]	Optimized model [g]	Relative improvement	Quantified improvement [g]
Weight	1448,29	1409,54	2,675 %	38,75

Table 2.2: Improvement results

Although the requirement of 10 % weight reduction were not met, the total weight reduction was still 2,67 % reached in 30 minutes.

2.4 Global Sensitivity Analysis

As mentioned earlier in section 2.1, there are two options to choose from in geometry optimization. Global sensitivity analysis is very similar to Altair Hyperworks with regard to procedure. The main changes is step 1 where type of analysis is selected. The options in step 5 is no longer relevant except for the iteration option which tell the solver how many steps the solver should divide the design space into. When the results are finished NX will write the results in an excel sheet which can be found in appendix C.

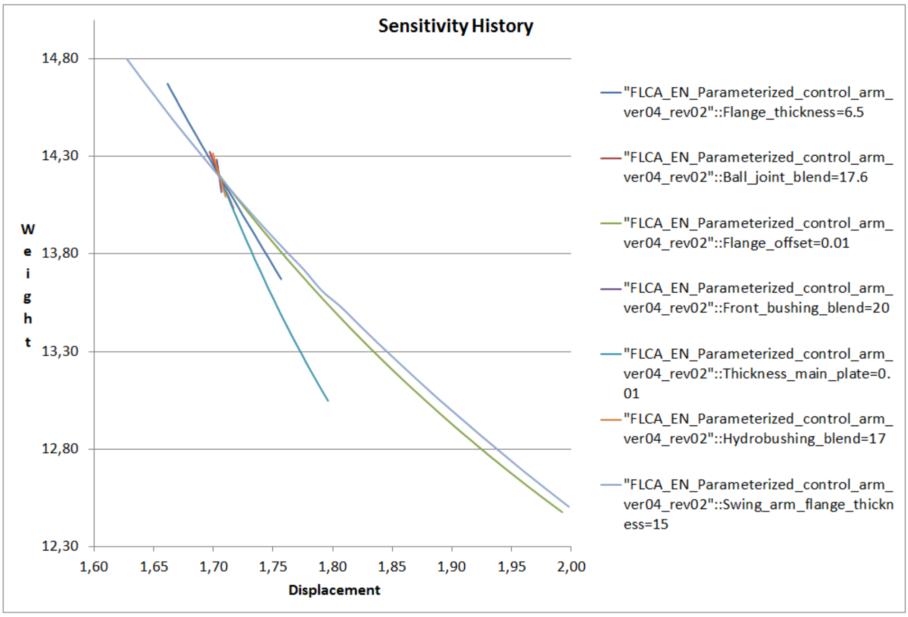


Figure 2.9: Sensitivity history

Each of the design variable has its own line and color, the length on the y- axis indicates its significance related to the objective. From this figure, it can be seen which of the parameters that has the biggest relative influence on the overall objective. The green, purple and blue line represents the design parameters that is the most important. As one can see, three of the parameters are less important to the overall goal.

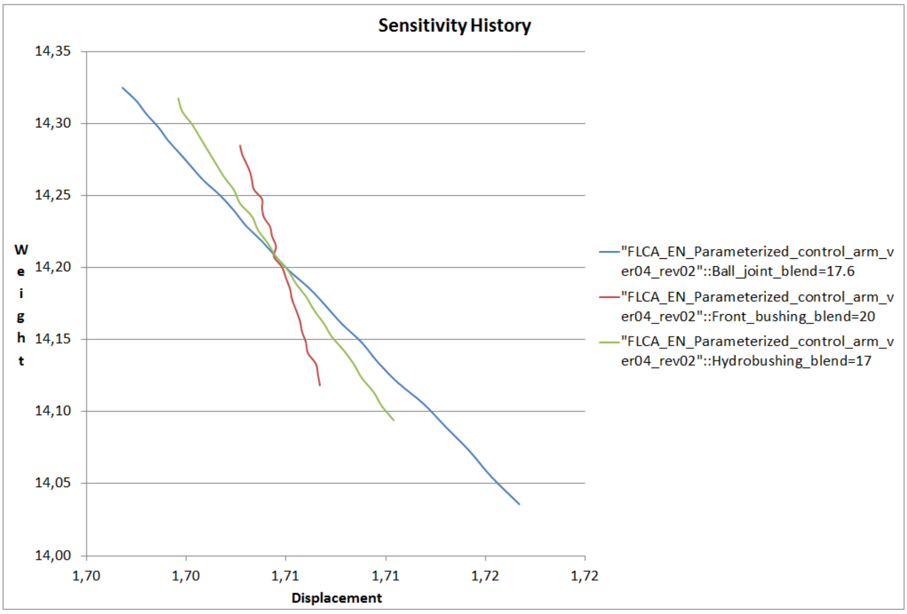


Figure 2.10: Detailed view of sensitivity history

Figure 2.10 shows the sensitivity of the least significant parameters. These parameters does only affect the goal to a lesser extent. Even though they are less significant they can contribute to the overall goal.

2.5 Discussion

The NX geometry optimization did not yield the results that were expected in advance. One of the reason that the results differs from the 10 % objective might be that the model itself is already optimized through years of manual optimization. This leaves less room for improvement on todays model. The Altair Hyperopt algorithm used by NX geometry optimization might not be the most suited algorithm for this particular optimization problem. This is related to the fact that the approximations is based on quadratic polynomial curves that can converge in local optimum.

When viewed in light of the relative ease of use and the fact that it has reduced the weight by 2,67 % on an already optimized part in 30 minutes, this is considered good results.

The results depicted in figure 2.10 and 2.9 verifies that the parameterization is robust and the model updates without failing. This is proved by no major irregularities in the lines, which would indicate design errors.

Chapter 3

Multiobjective Design Optimization

3.1 Optimization Theory

The objective of the following chapter is to study optimization and DOE algorithm theory. A short introduction is provided to identify the characteristics and manner of operation of the algorithms intended to be used for this optimization problem. This has been done with regards to modeFRONTIERs available tools and procedure. First off is a quick review off DOEs and the most common algorithms included in the software.

3.1.1 Design Of Experiments (DOE)

Design space is every possible combinations of each design variable.

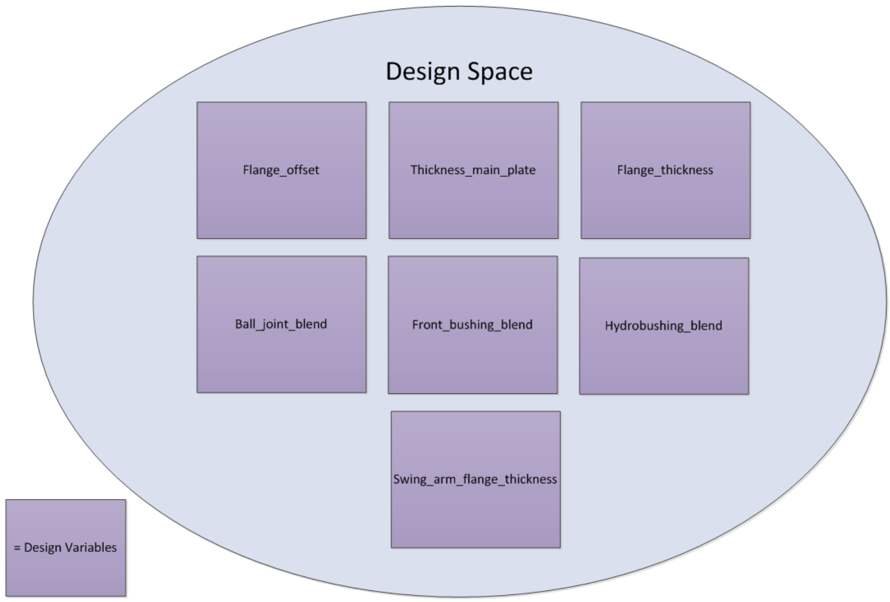


Figure 3.1: Design space

Each design variable has a given range of values to choose from.

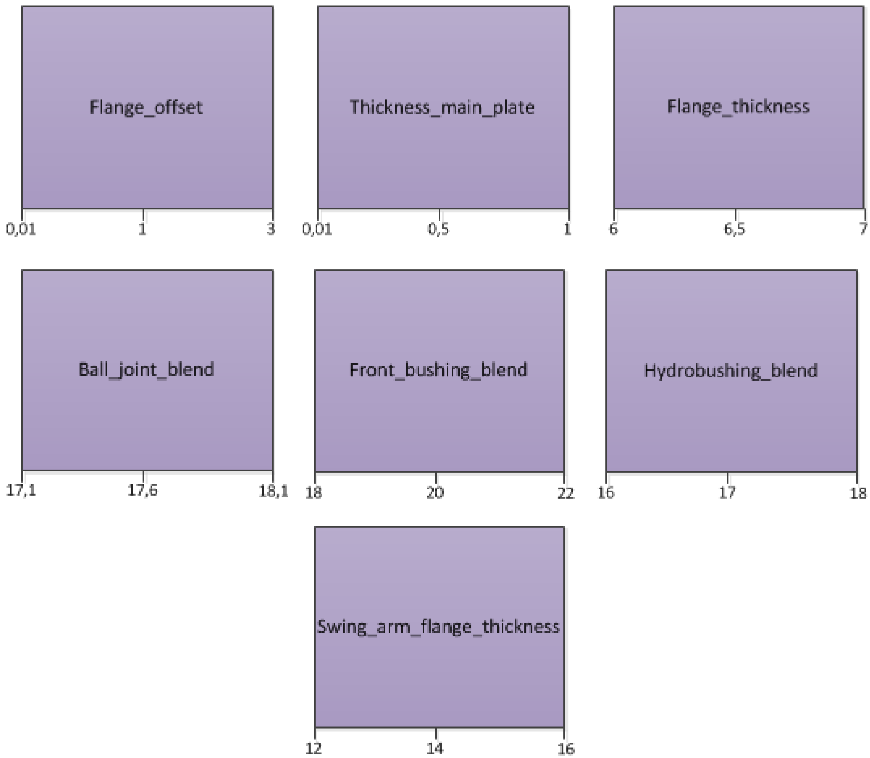


Figure 3.2: Range of Design

A sample is a combination of designs chosen from each variables possible range.

	M	CATEG...	Ball_joint_blend	Flange_offset	Flange_thickness	Front_bushing_blend	Hydrobushing_blend	Swing_arm_flange_thickness	Thickness_main_plate
0	☐	☐ ULH	1.7100E1	1.3600E0	6.5000E0	1.8750E1	1.7820E1	1.2000E1	6.3000E-1

Figure 3.3: Example of a design sample

When more designs are sampled from the design space, it constitute a sampling of the design space. This sampling is in turn run by the program to execute a DOE. In order to perform an optimization it is necessary to sample the design space to get a starting point. This involves a determination of sampling used for DOEs. The type of sampling depends on how wide the search should be, and how much that is known about the parameters. Number of test designs might also depend on the type of algorithms used for further optimizing. It is common to use some kind of sampling that suits the optimization in addition to the overall goal.

In this context DOE means a run of initial designs which can be used (the basis for further optimization by the selected schedulers algorithm) by the schedulers algorithm. Not the classic sense of the word which involves the whole process of trial and error by manual design changes. The following section explains different ways to generate DOE.

The DOE‘s can be used to explore the design space in an early stage. Some of the suited DOE‘s for this purpose are:

• **Random** sequence which spreads points random, and then fill in points uniformly between the random design space.

• **Sobol** is similar to the random DOE, but aims to reduce the clustering effect. The design will try to avoid each other as much as possible.

• **Uniform Latin Hypercube** distribution is a random generator that conforms to different statistical distributions and makes a relatively uniform DOE sampling.

Reducers can be helpful to generate good designs based on large data sets, this technique will save computational time by eliminating designs.

• **Uniform Reducer** is useful when it is necessary to distribute values uniformly within an existing input space. This input space could be defined manually by selecting a previously generated DOE data set. It could also inherit a data set from a previous run (distribute more values uniformly between already chosen values).

• **Dataset Reducer** algorithm extracts a data set from an already existing data set. Used for metamodels training. Metamodels/RSM is elaborated in chapter 3.1.2.

• **Full and Reduced Factorial** DOE. These are well suited for statistical analyzes. The full factorial generates every possible combinations of all values, while the reduced does this on a subset of a full factorial.

Some other types of DOEs is designed for more special purposes.

• **Incremental Space Filler**(**ISF**) evaluates the previous generated designs. The algorithm is used for uniformly expanding the design space. This is useful for generating data for RSM training (metamodels training)

- **Taguchi Matrix** uses orthogonal arrays. This is a special algorithms that may reduce errors in a run. It is based on the same principles as factorial DOE. The method is used to find values that make the system less sensitive to variations.
- **Constraint Satisfactory** problem works with heavy constrained optimization problems.

These are a selection of some of the DOE algorithms available in modeFRONTIER. The most important is usually the exploration DOEs. The other ones is usually applicable in more specific problems. The most used sampling is the exploration DOE, Uniform Latin Hypercube, which will give a good spread and avoid inbreeding[1] of the design parameters. If time permits it, one should use a large DOE instead of a smaller one. This will help cover most of the available design possibilities [8, 6].

[1]This means that the design will vary as much as possible, avoiding identical designs.

3.1.2 Scheduler

Introduction to modeFRONTIER Optimization Algorithms

modeFRONTIER contains a lot of different optimization algorithms,(which is specified in the scheduler) both single and multiobjective. The definition of a multiobjective optimization is an optimization that deals with two or more conflicting objectives. Most of real life problems involve improving one objective that leads to worsening of another. This typical multiobjective solution is depicted in figure 3.4. Here the designer is given the opportunity to choose which of the optimal solutions that suits the needs of the problem at hand. Most of the algorithms provided in modeFRONTIER is multiobjective.

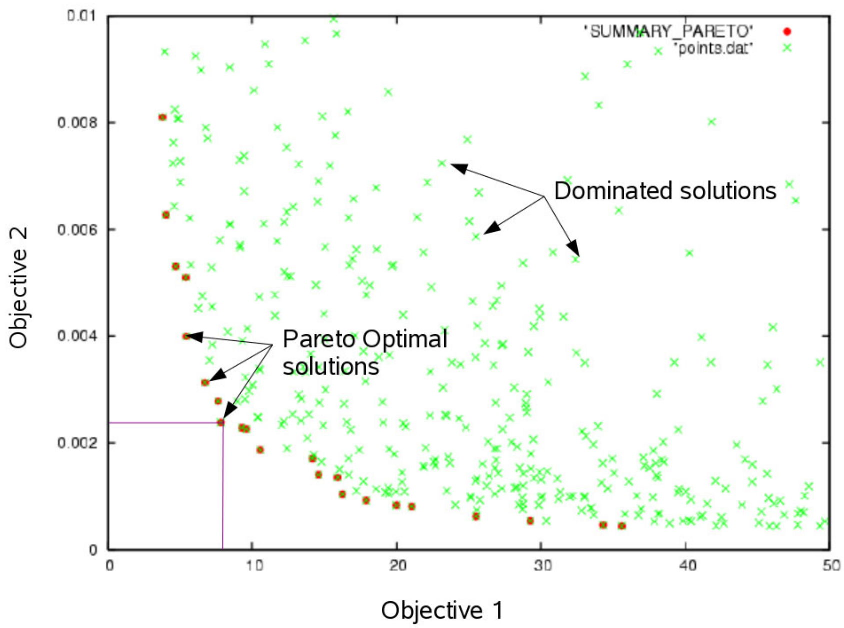


Figure 3.4: Illustration of a pareto front (curve)[2]

Figure 3.4 shows a X - Y scatter of two objectives. The designs are scattered according to their goal achievement of the two objectives. Optimal pareto solutions represents the best designs available according to the conflicting objectives. Dominated solutions are the bad designs far from the optimal solutions. From this scatter the user has to weight the two objectives and make a trade off.

Scheduler Algorithms

It is generally hard to predict how effective a specific algorithm will prove to be. This may also be dependent of the DOE sampling chosen. A general description of how some of the supported algorithms will function is stated in the following: The genetic algorithms can be compared to the natural evolution of species and uses tools such as natural selection to guide the individuals (designs) towards optimal solutions. This is why a lot of notions like parent and children is used to describe the development of the algorithm. Evolution strategies algorithms works in the same way but uses a mutation tool that produces individuals that stands out from the rest of the population. This way the algorithm can break the pattern and produce diversity in the population. These functions can be combined as well. Game theory algorithm exists as well. Here the multiobjective problem is considered a game between two or more competitors. Some of the most universal multiobjective algorithms available in modeFRONTIER is: MOGA – II, Hybrid, MOGT, and Fast which can work as a single objective as well (Simplex is solely a single objective algorithm).

A short introduction to a few of the most relevant algorithms is provided in the following section [7].

MOGA – II

MOGA - II is a multiobjective genetic algorithm. This means that it will strive for two conflicting solutions.

The algorithm utilizes four operators in its search for better designs, and will alternate between the use of each of the operators based on a defined operator probability.

Operators used are:

- Mutation
- Selection
- Elitism
- Crossover

The optimizer is encoded as a binary string. The example illustrations in this section uses the binary numeral system to show how operators make changes in this algorithm.

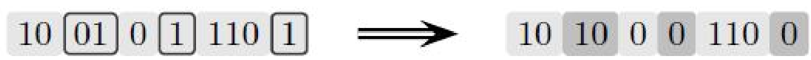


Figure 3.5: Mutation operator illustration with bit string

Mutation controls how often the program should alter a parameter randomly. This operator may help break the pattern in cases where the algorithm can get stuck.

Selection defines the probability for how often a design parameter should stay the same throughout the run.
Elitism will ensure preservation of good individuals. This means that the algorithm will assure that the new design generated is as good or better than the previous design.

The overall driving factor used to decide which individuals to choose in a genetic algorithm is the probability of being better than other individuals, called the fitness factor.

One of the algorithms strength is the use of crossover. Crossover can be done in two ways within the same optimization run; classic and directional.

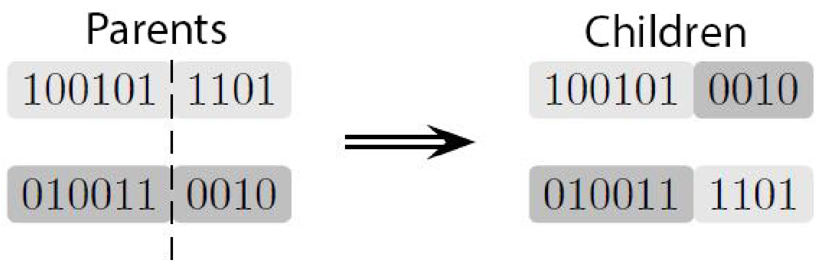


Figure 3.6: One point crossover

The classic way (one point crossover) involves dividing a bit string at a random point. The divided pieces from the parents is then put together to form a new resulting individual. The initial parent is put together by taking a random parent and combining it with the best from a tournament selection. The tournament winner is decided by the individuals fitness factor.

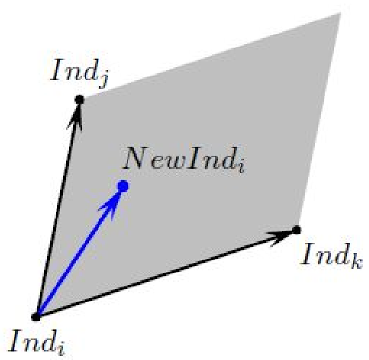


Figure 3.7: Directional crossover

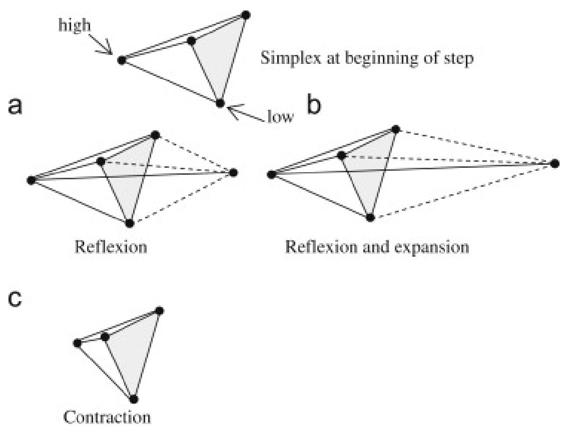
Directional crossover differentiates itself by comparing the fitness value of two reference individuals. It considers the most appropriate direction of improvement by evaluating the parents fitness factor (Indj and Indk) with respect to a weighted direction compared to the new children position(Indi). This generates the New Indi (the actual new individual). Directional crossover represents one of the most helpful properties that make this algorithm a very powerful tool.

As a rule of thumb it is recommended to use an initial number of DOE of approximately 2*number of variables*number of objectives.

Moga – II is a great tool for most uses and is less susceptible for ending up in a local maximum. The method is slower than some of the other algorithms presented, but it is very stable and rarely crashes [8, 17, 19].

Simplex

The simplex algorithm is a version of "Nelder and Mead simplex" which is updated to handle constraints and discrete variables. The scheduler utilizes an algorithm to move the initial points along with their values closer to the objective. This will continue until the scheduler exceeds its maximum number of iterations or the points converge. For two variables, a simplex is a triangle. The method searches and compares values at each vertices in a triangle. The worst vertex (where x and y is largest.) is identified and replaced with a new vertex. This results in new triangles being formed which generates smaller triangles that reveal optimal minimum coordinates.[2] The operators that control the algorithm is presented below in figure 3.8 in sequential order.



Multiobjective Design Optimization

Figure 3.8: Illustrations of simplex steps[15]

This illustration shows a three dimensional simplex and shows how the operators would work towards a converging solution.

- Reflection involves an operator that makes the function move in the opposite direction of the worst value.
- Expansion will minimize the value (objective) further by expanding the previous goal achievement.

[2]Simplex means a generalized triangle in N dimensions.

- Contraction is used if reflection gives a worse value than the previous. This means that the new point reverts back towards the initial value.

The new design generated is rounded up to the nearest discrete values defined by the initial value of each variable. Simplex iterates each variable in turn. It is not capable of iterating each variable at once [8].

Fast

The fast optimizer algorithm is creating metamodels, which is a surface based on approximations from interpolation and approximation methods. These approximations is known as RSM (Response Surface Models). The approximation methods used (RSM) are chosen by the program. See modeFRONTIER user manual for more information regarding RSM algorithms. This optimizer is fast because it estimates the best design configurations, and then tests some of the most promising designs with a solver run. Different types of virtual optimization algorithms are chosen in the scheduler configuration. Some of the algorithms to choose from is: MOGA – II, Simplex, and MOSA to name a few. This algorithm is mentioned even though it is not used in the following optimizations because it utilizes some of the same types of algorithms within its virtual run.

The optimizer runs in an iterative way as shown in figure 3.9.

- Metamodels training is a comparison of the different RSM designs, this is created randomly in the first iteration.

- Virtual exploration is used for creating additional design space in the vicinity of the pareto curve. The DOE generator ISF (Incremental Space Filler) is used for this purpose. This stage is optional and can be turned off in scheduler configuration.. The virtual optimization stage runs an optimization based on the best available metamodels by use of one of the available schedulers (as previously mentioned in the introduction).

- The validation process has to choose the best option for further optimization based on the previous two steps. The fraction of selected design from each group is defined before the optimization is carried out. This validation tests the designs with a solver run.

- Metamodels evaluation is the step where all of the metamodels created in metamodels training is evaluated compared to the designs obtained in the validation process. The best designs is used in further iterations.

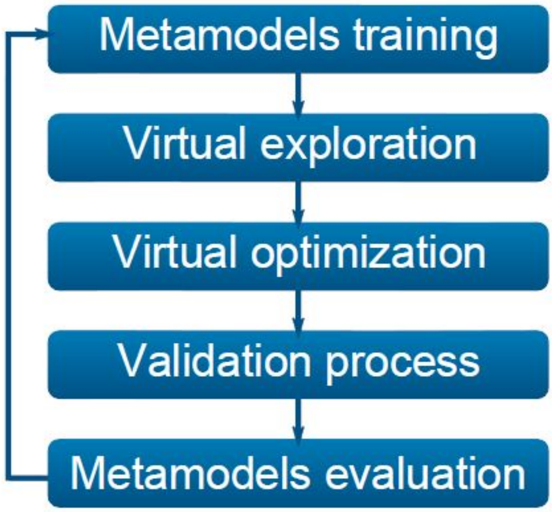


Figure 3.9: Iteration loop for fast optimizers

The type of optimizer selected within the Fast run will decide the amount of necessary DOE sampling according to recommended guidelines for the relevant scheduler (algorithm).

The main benefit from using fast is time saved by not having to do a solver run on every available design. Instead it predicts which designs is best and runs the solver in the end of each iteration loop to verify this [8].

Hybrid

This scheduler is as the name implies a mix of two different algorithms. It combines a steady state genetic algorithm and a single objective optimizer (SQP). This makes it a robust multiobjective algorithm, as well as a good single objective algorithm. The amount of robustness versus exploration can be varied by specifying this in percentage. The algorithm works by implementing SQP run as one of the operators in a genetic algorithm. For more information regarding SQP see [8]. The combination of the two algorithms makes it quick to reach the pareto front. Then the genetic algorithm fine tunes the variables in the end.

Hybrid uses Adaptive filter SQP and may also use RSM within the Hybrid algorithm. This makes it more efficient.

The main idea behind the chosen SQP solver is to use gradient information to make an approximation of the lagrangian function related to the objective function and constraints. To avoid local optimum points adaptive filters are introduced in the algorithm. This means that the old designs are stored and evaluated against the new ones. The criterion of the new design is to stand out and prevent local stagnation [8].

The overall process can be logically described as follows:

- Creation of a parent population based on an initial DOE (design of experiments) or performing a tournament selection among the population.
- The genetic algorithm work with its operators like mutation, crossover and SQP that generates offspring.
- Storing old design generated by SQP.
- If a local optimum is created, it gets sorted out as a parent in the population for further optimization.
- The design storage gets analyzed and the best designs are saved according to the elitism function.

MOGT - Multiobjective Game Theory

Game theory algorithm works by assigning two different objectives to functions called players. These players are influenced by each others choice. They in turn try to minimize each others objective based on the others move. The two players does this until each player has minimized its function, an equilibrium is now found. In this optimization only one initial DOE is required for design space sampling.

Game theory was first formulated mathematically by J.F Nash in the 1950s [8]. This has proved to be useful in economics. It is most commonly used in decision making regarding competitive fields. These strategies has been adopted by other disciplines and modified. Multiobjective game theory algorithms can be combined with different algorithms such as evolutionary algorithms to save computational time. A variety of game theory algorithms exist, one of them is a combination of Nash game theory coupled with the simplex method which is used in modeFRONTIER. This Nash simplex algorithm is a single objective algorithm that works by combining it with a competitive game theory algorithm called Nash equilibrium to make it multi objective [8].

MOSA – Multi Objective Simulated Annealing

The method is a modified SIMPLEX method. This algorithm is as the title describes multi objective and utilizes an algorithm which is associated with the annealing in metallurgy. This technique uses heat and controlled cooling of a material to reduce defects. This process is based on thermodynamic free energy principles. The algorithm works by the same principle by slowly removing bad solutions as the solution space is explored. The algorithm utilizes a probability function that determines if the new design is to be accepted or discarded. One of the most important control parameters in this algorithm is the "hot" and "cold" phases. As the algorithm iterates it is either in a hot or cold phase. The hot one implies that it explores widely the design space, avoiding local optima. The cold phase allows convergence and local exploration. These two parameters has to be specified in the scheduler properties based on what property is the most preferable. The fraction of hot iteration tells what the scheduler prioritizes (the fraction of hot iterations tells how much is hot iterated over cold ones). The total number of designs ($N_{Designs}$) necessary to complete a MOSA run is the number of initial DOEs (n) specified (in DOE properties) times numbers of iterations ($N_{Specified}$) specified in the MOSA scheduler. This yields: $N_{Designs} = N_{Specified} \times n$ [24][8].

Some General Benefits and Drawbacks

	Pros	Cons
MOGA-II	- Stable - Finds global optimum - Suited for non-linear problems	- Slow (many iterations)
Simplex	- Fast	- More sensitive than MOGA-II - Usually finds local optima - Only single objective
Fast	- Lives up to its name	- Not suited for non-linear problems
Hybrid	- Suited to cover global optimum	- Extensive search that takes advantage of two algorithms
MOGT	- Faster than MOGA-II	- Not exploratory, local optima
MOSA	- Finds global optimum - Well suited for large design spaces	- Very slow (many iterations)

Table 3.1: Benefits and drawbacks

These guidelines are very general and some of the algorithms can be configured to be quicker or more time consuming. This shows a short list of what characteristics each of the algorithms have.

No Free Lunch Theorem

It is hard to predict which of the algorithms that will yield the best results. This is stated by the "no free lunch theorem". This theorem uses an analogy about a restaurant (problem solving algorithm), a menu that combines a lunch plate (the problem) and a price(performance of the algorithm in problem solving). The menus of each restaurant are alike, except for the prices that are shuffled. A omnivore would pay the same average price for lunch because he could order any plate at any restaurant. A vegan accompanied by the omnivore that seeks economy would however pay a higher average price for lunch. To reduce the average cost, one need to know what the order will cost at each restaurant and what the order will consist of. This means that performance depends on information about the problem.

An other interpretation is that unless it is possible to make prior assumptions about the problem, it is no algorithm that can be expected to outperform any other. This will in turn mean that without assumptions no algorithm will perform better than a blind search [23].

It is therefore hard to tell which of the algorithms that will generate the best results and cost less computational time. Because it is hard to predict the most appropriate approach some kind of brute force[3] search will be used. In the design optimization chapter a set of DOE and scheduler algorithms will be tried out and evaluated [8][23, 25].

[3]Brute force search is a wide systematic search based on all possible combinations.

3.2 modeFRONTIER Configuration

This chapter describes a step by step guide for configuring collaboration between modeFRONTIER and NX.

3.2.1 Introduction to modeFRONTIER Configuration

modeFRONTIER is a multiobjective optimization software which allows you to connect several different CAD and/or FEA software together. Through the graphical interface you are able to set up a workflow consisting of nodes which are connected with each other to constitute a logical scheme of an optimization process. The overall procedure is presented in the following chapters, for an even more detailed description see appendix E.

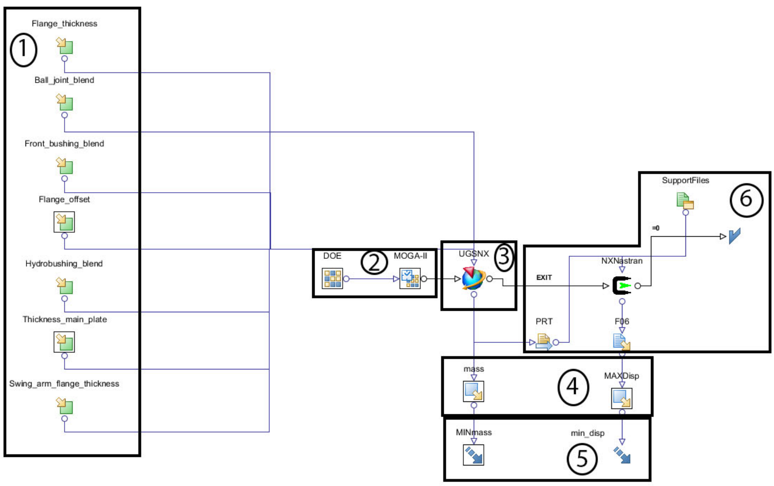


Figure 3.10: Workflow modeFRONTIER

The numbered corresponding nodes is:

1. Input variables: defines design space
2. DOE and Scheduler: DOE and algorithms provides different values for the input variables

3. NX CAD Node: Interacts with NX expressions
4. Output variables: design output variables
5. Objective: minimizing or maximizing output variables
6. Support files and Cygwin shell script

In the following sections a quick review of the different nodes is presented. Appendix E contains more specific information about modeFRONTIER configuration.

3.2.2 Input Variables

The range of each parameter has to be defined in this node. The sum of all variables defines the design space. The DOE will in the following (in turn) sample a design space based on this area.

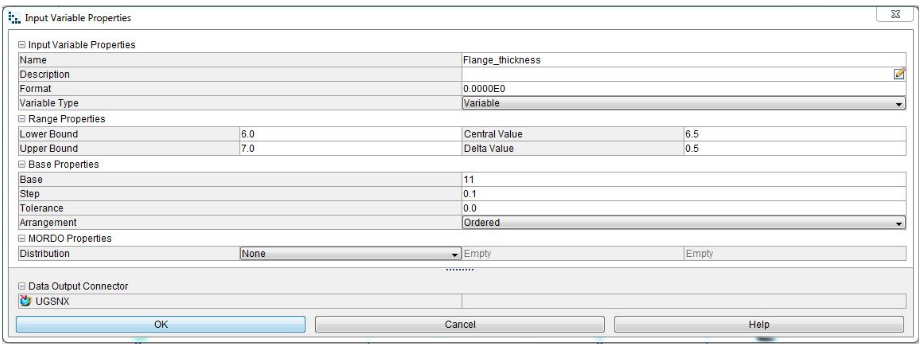


Figure 3.11: Input variable properties

The variable is defined by adding upper and lower bound in addition to initial value associated to the specific parameter. This tells the DOE node what range it can sample designs from.

3.2.3 DOE and Scheduler

Design of experiments (DOE) is a necessary sample of the design space which the scheduler algorithm can base its optimization algorithms on. These are the initial designs that the scheduler are based on.

The Scheduler contains different types of algorithms that work in various ways to reach optimization goals based on the problem at hand. The scheduler uses the initial DOE to build a population of new designs. The way it controls the evolution varies from algorithm type selected. Both DOE and scheduler provides NX with designs to test and explore.

3.2.4 NX CAD Node

The NX CAD node can only interact with input and output parameters regarding geometry. This stage will require a part file with expressions predefined as shown in appendix E.

In figure 3.12 the NX properties menu is shown. The black box to the left shows the selected variables. The one to the right is the output connected to NX that provides modeFRONTIER with weight data which is one of the goals.

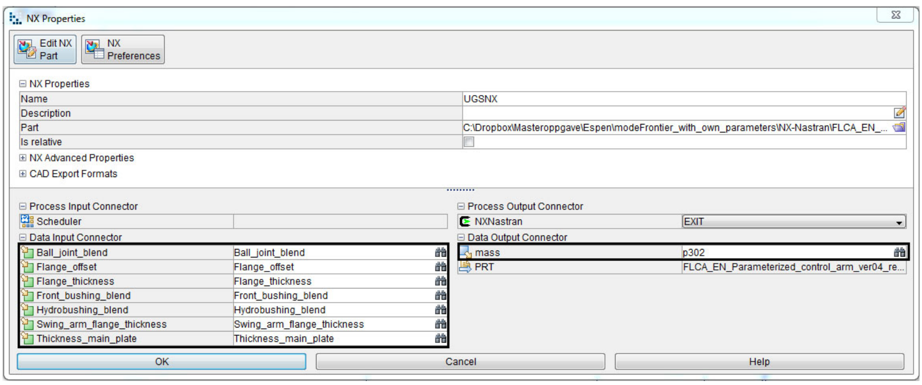


Figure 3.12: NX properties

The preferred part file has to be defined as well. This output data connector gives information regarding the design variables continuously to the part file that is exported for further simulations.

3.2.5 Output Variables and Objective

These nodes does not need much configuration. The output variables has to be connected to the corresponding node that delivers output information. Objective node needs specified that the target is minimizing, this is a formality for the program to know which goals to include in multiobjective optimization. The output data is extracted from the f06 file following an iteration run executed by the Cygwin node.

3.2.6 Support Files and Cygwin Script

In this category there are four nodes that needs to be configured.

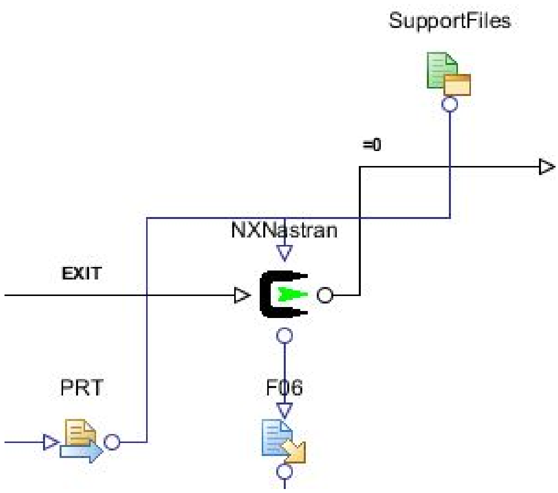


Figure 3.13: Support files and Cygwin

- Transfer file node (Named PRT)

This node needs to specify that a copy of the part file has to be transferred to the Cygwin node.

- Support files node

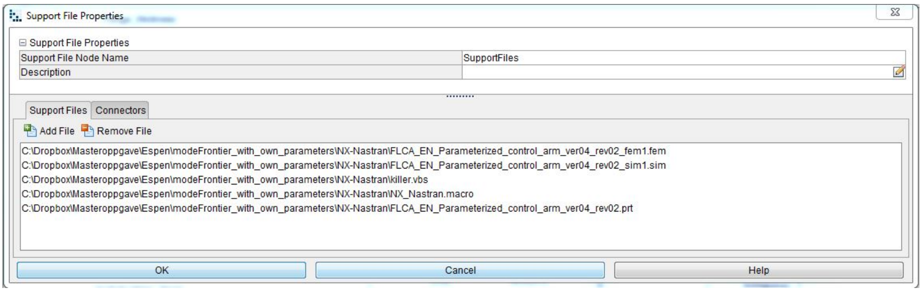


Figure 3.14: Support files

In this node the support files that the Cygwin script needs to run has to be specified with correct directory. These files are: part, sim and fem file for NX execution. Visual basic script to kill the process if any error with the run occurs. A prerecorded macro that executes a simulation run in NX. This macro is made in NX with its own macro[4] function. A guide for doing this is provided in appendix E.

- Cygwin node

Cygwin is a Unix like environment with a command line interface which resembles a DOS environment. It emulates[5] Unix based systems like Linux and is written for Microsoft Windows[21].

In this node a Cygwin shell script is specified which runs the NX analysis by the use of support files. This is the backbone that enables the NX run, which opens NX and its solver for each iteration executed in modeFRONTIER.

- Output file node (Named F06)

This provides the goal node with necessary information about displacement written by a F06 file (standard NX output file).

In order to extract this data from the F06 file, one has to specify user defined text in NX solver properties. This is done by clicking edit properties in solver properties. From here one has to click the case control tab->create modeling object. The "Text to Insert at End of Section" must contain: MAXMIN(VMAG=1,

[4] A macro is a set of instructions that is used for generating a script. This is usually done to save time on repetitive tasks. Often a type of recorder is used to generate these macros automatically [22].

[5] An emulator imitates the functions and system of software and/or hardware.

CID=BASIC,DISP)=ALL. If other information is needed from this output file, Nastran solver provides a lot of other options as well. See NX Nastrans quick reference guide [3] for more information regarding case control and solver output. A visual guide for the method described is provided in appendix E.

3.3 Design Optimization

The following sections will contain a kind of semi brute force approach to the optimization problem. Before the optimization runs are presented, a section with an illustration of the methodology and an explanation to sensitivity analysis is provided. Then the specification of the standardized DOE sampling and extraction of results are presented to add a practical guide. Including these sections gives a more complete understanding of the overall process associated with modeFRONTIERs design optimization.

Since there is not enough enough time to evaluate all possible combinations of design sampling and scheduler algorithms within modeFRONTIER, a set has been chosen. The choice of DOE fell on ULH - Uniform Latin Hypercube consisting of 28 designs. This is because it is the most general and exploratory DOE sampler with the least amount of clustering within modeFRONTIER. The reason for this is elaborated in 3.3.3.

As for the scheduler, the choice fell on MOGA-II, MOGT, Simplex, MOSA and Hybrid. These five schedulers were chosen to make the most varied results based on the most versatile algorithms. All of the runs is using the same input variables (expressions) with the same range and increment size (steps). After five initial runs, two more is executed with the use of the best performing algorithms. In the end a local search is to be run to fine tune and search thoroughly among the best designs. The optimization runs will be compared by the NX run. This has been done to measure if modeFRONTIER works as good or better than NX. The goal is still a 10 % reduction of mass while maintaining the baseline displacement.

The definition of best designs will in this case be the ones with least weight, that still fulfill the constraint criteria.
One of the methods used for extracting results is shown in section 3.3.4. It involves selecting designs within a desirable range considering one of the two goals. Then it is up to the user to decide which of these results that fits the requirement and overall goal best. modeFRONTIER also provides a decision making tool called MCDM (Multi- Criteria Decision Making) to ease the extraction of results. The way of extracting the best results can be done in various ways and a more detailed description of MCDM module and other methods is provided on a A3 sheet in appendix F.

3.3.1 Methodology

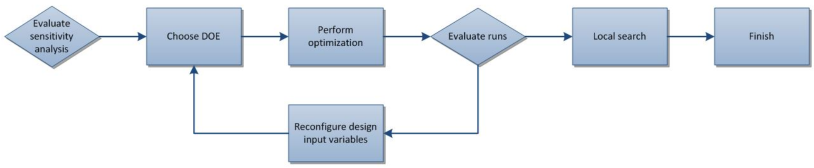


Figure 3.15: The overall optimization process

Figure 3.15 shows the overall optimization workflow. This is considered a sensible approach and represents how the process is carried out in the following chapter.

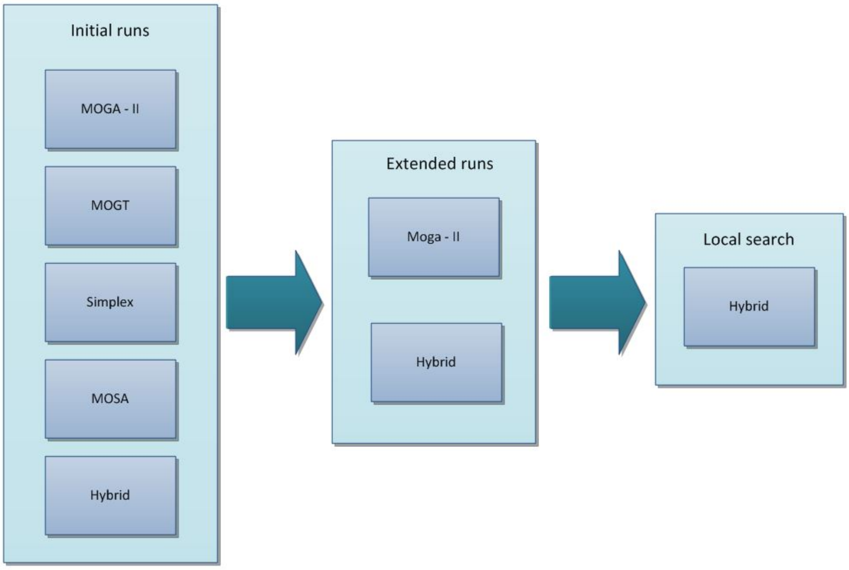


Figure 3.16: Optimization steps

Figure 3.16 illustrates how the search for the best algorithm and design results are carried out within modeFRONTIER as described in section 3.3.

3.3.2 Sensitivity Analysis

In case the already chosen design parameters had not been analyzed by NX sensitivity analysis, modeFRONTIER offers tools to do this as well. One easy way of checking this is by running a small analysis with the preferred parameters. This analysis does not have any impact on the optimizations that follow in this thesis, but illustrates a tool to detect relationships and possible errors. This is useful to identify which of the parameters that can be excluded to save computational time. The last two runs in chapter 3.3.10, however will utilize this information to generate an extended DOE sequence for a widened search.

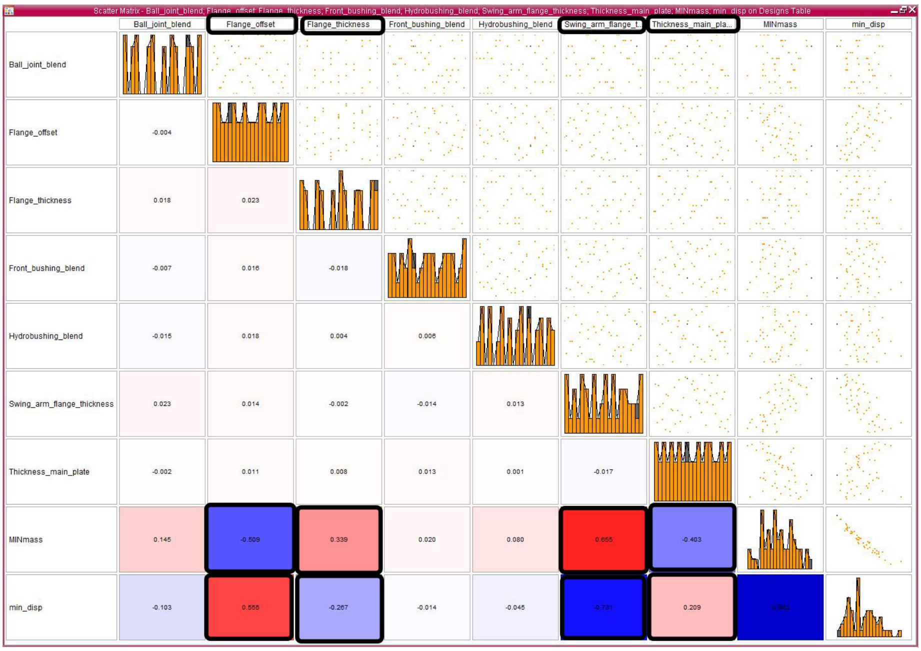


Figure 3.17: Sensitivity analysis

This picture shows which of the parameters that affect the results the most. A strong color degree indicates a strong correlation between design changes to the particular parameter and the goal. Four of the most important parameters has been highlighted with black frames. One can see from the MINmass row that Flange_offset, Flange_thickness, Swing_arm_flange_thickness and Thickness_main_plate has the biggest impact on this goal. The same is indicated in the min_disp row with the opposite color because the outcome of changing the parameters gives the opposite sign for this goal.

One can see that for instance Flange_offset has a great impact on the MINmass goal and an opposite effect on min_disp goal. This scatter matrix were made up by running a quick analysis on a DOE sample. The presented figure is the result of a run based on fifty uniform latin hypercube samples generated in the DOE scheduler. This means that no algorithm has been utilized and only a random space has been explored to estimate correlation of parameters.

Above the diagonal, scatter matrices of each variables with its designs is shown. The designs shown are distributed evenly as one would expect from using ULH (Uniform Latin Hypercube) [8].

3.3.3 DOE Sampling

The two most suitable DOE samplers for generating an exploratory design sampling is ULH and Sobol. The random DOE sampler has been omitted since Sobol is a random generator with less clustering.

The available design space with each of the variables range is defined under input variables tab in modeFRONTIERs workflow (in the bottom of the start page in modeFRONTIER). These designs are shown in figure 3.18. Base defines how many possible increments that exists for each of the variables. This defines possible selection for use in generating DOE and available designs for scheduler run.

Input Variable

	Name	Variable Type	Lower Bound	Central Value	Upper Bound	Base	Step
0	Flange_offset	Variable	0.01	1.505	3.0	300	0.01
1	Thickness_main_plate	Variable	0.01	0.505	1.0	100	0.01
2	Flange_thickness	Variable	6.0	6.5	7.0	11	0.1
3	Ball_joint_blend	Variable	17.1	17.6	18.1	11	0.1
4	Front_bushing_blend	Variable	18.0	20.0	22.0	41	0.1
5	Hydrobushing_blend	Variable	16.0	17.0	18.0	21	0.1
6	Swing_arm_flange_thickness	Variable	12.0	14.0	16.0	41	0.1

Figure 3.18: Initial design space defined in modeFRONTIER

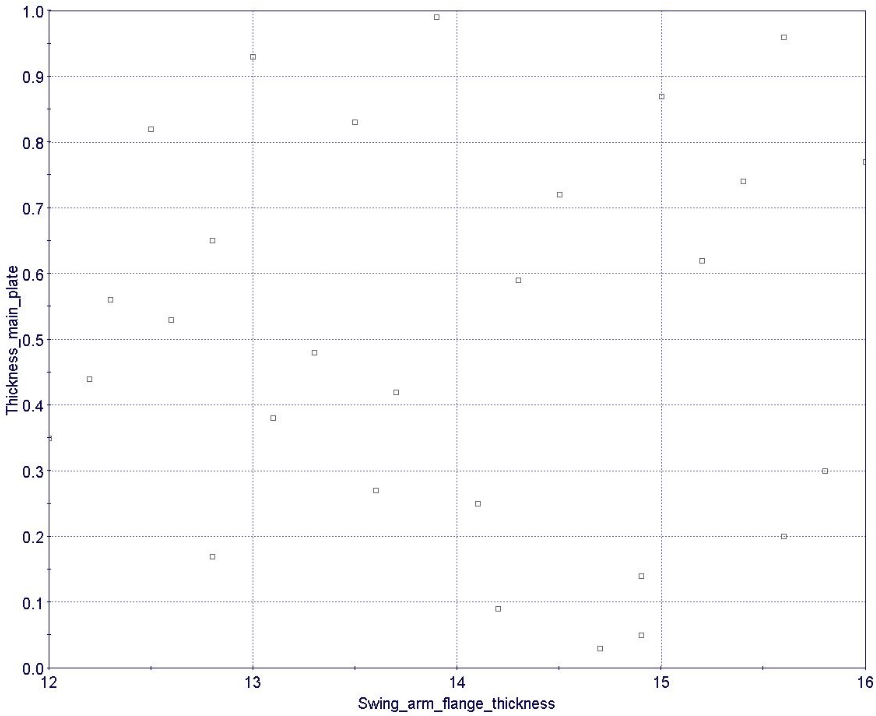


Figure 3.19: DOE with ULH distribution

The design space distribution is shown with a scatter plot of two of the design parameters. The parameters; Thickness_main_plate and Swing_arm_flange_thickness is chosen randomly to illustrate the distribution of ULH (scatter plots of the other parameters would show the same pattern). They are distributed uniformly and will thus create a wide initial search space for the algorithms.

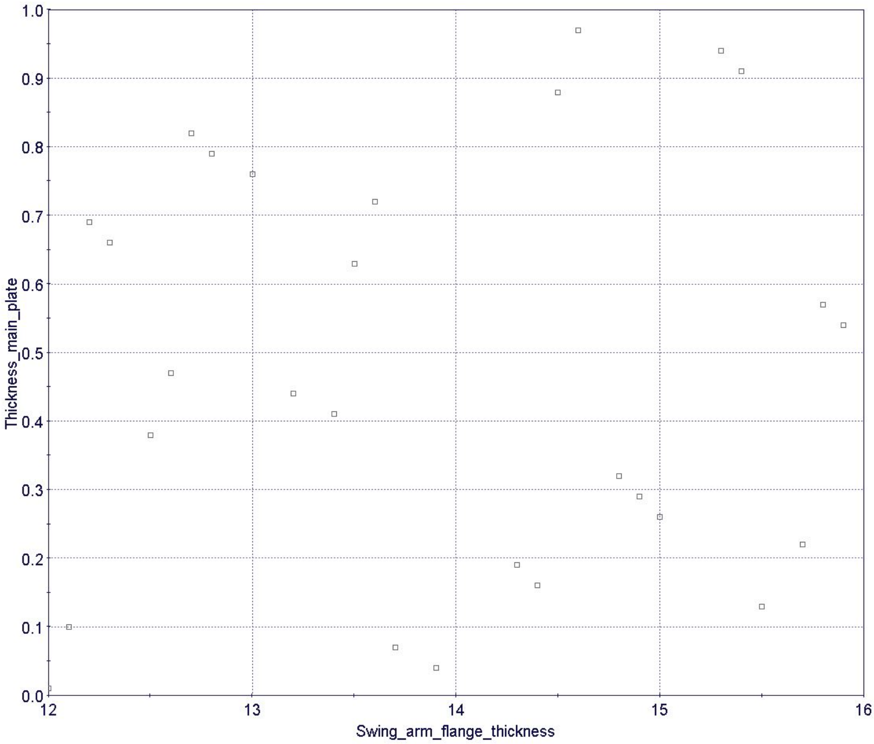


Figure 3.20: DOE with Sobol distribution

Figure 3.20 shows the same number of DOE sampling with a different distribution called Sobol. It is based on the same initial design space as the ULH sampling, but this distribution generates a bit more clustering.

As shown in the previous figures, ULH was slightly better distributed and therefore more suited for sampling the design space related to this problem. ULH was chosen as the DOE sampler for all further runs in modeFRONTIER, except for MOGT which requires only one DOE entry. This approach is used to create an equal basis for further comparison of scheduler algorithms.

3.3.4 Extraction of Results

When the optimization runs is done, one has to determine which designs is the best. modeFRONTIER offers some tools to help in this process. As previously mentioned modeFRONTIER provides a tool called multi criteria decision maker (MCDM), which is also elaborated in appendix F in addition to the upcoming method. This section intends to show the procedure for how the results can be extracted visually in modeFRONTIER.

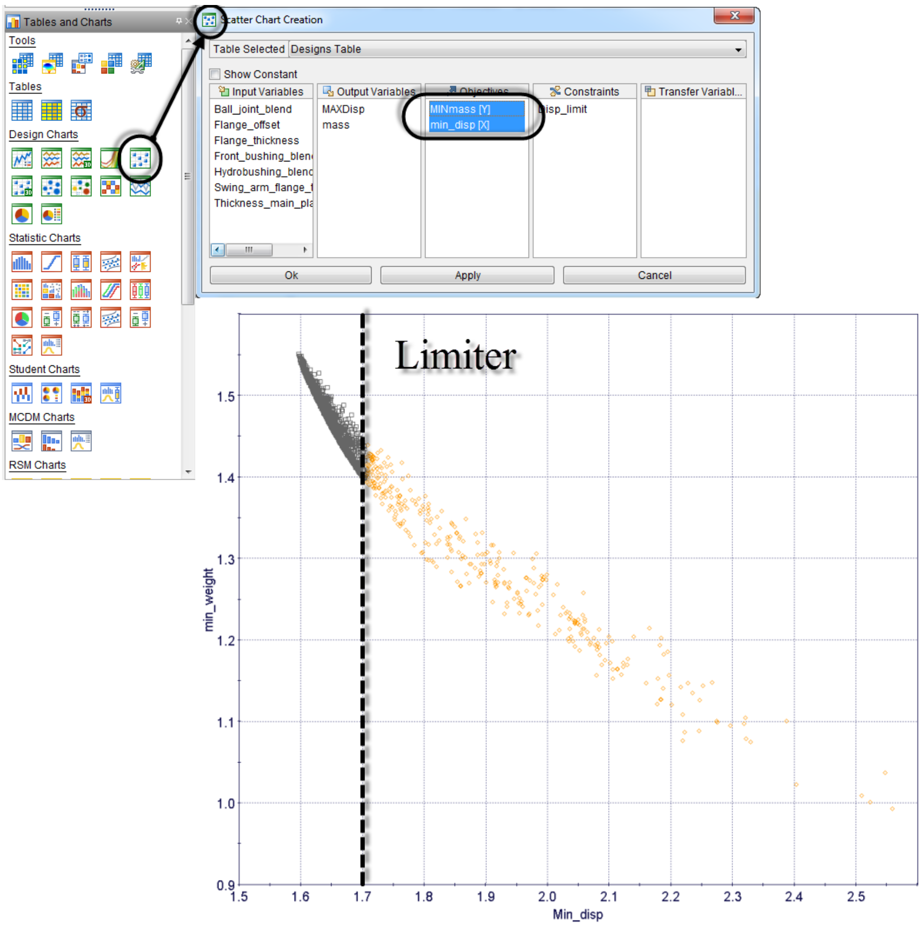


Figure 3.21: Identification of results

Figure 3.21 shows the procedure for generating a X - Y scatter of the two objectives. This generates a pareto curve with all the available designs. A limiter is specified in modeFRONTIER workflow as illustrated. These designs are barely visible in this scatter which is why the next step involves isolating the best designs in a new scatter.

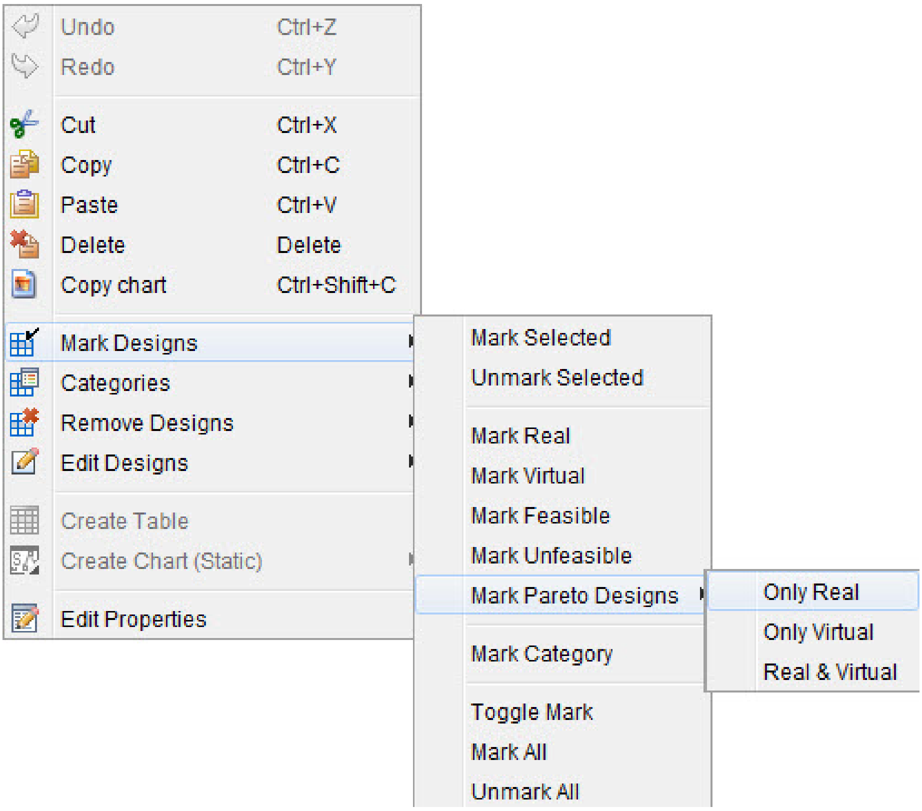


Figure 3.22: Pareto design selection

When the pareto designs is marked, the gray points to the left in figure 3.21 is highlighted green. Now the best designs is marked in the designs table.

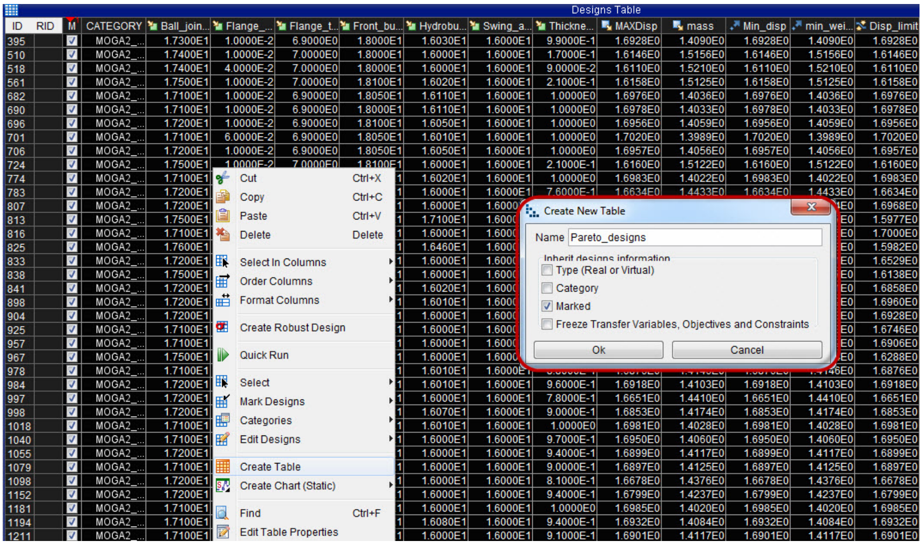


Figure 3.23: Pareto designs table

Get the best designs in order by sorting designs after whether or not they are marked (ticked off in the box to the left in designs table). Highlight these designs and create a table (in this case called Pareto_designs) as seen in figure 3.23.

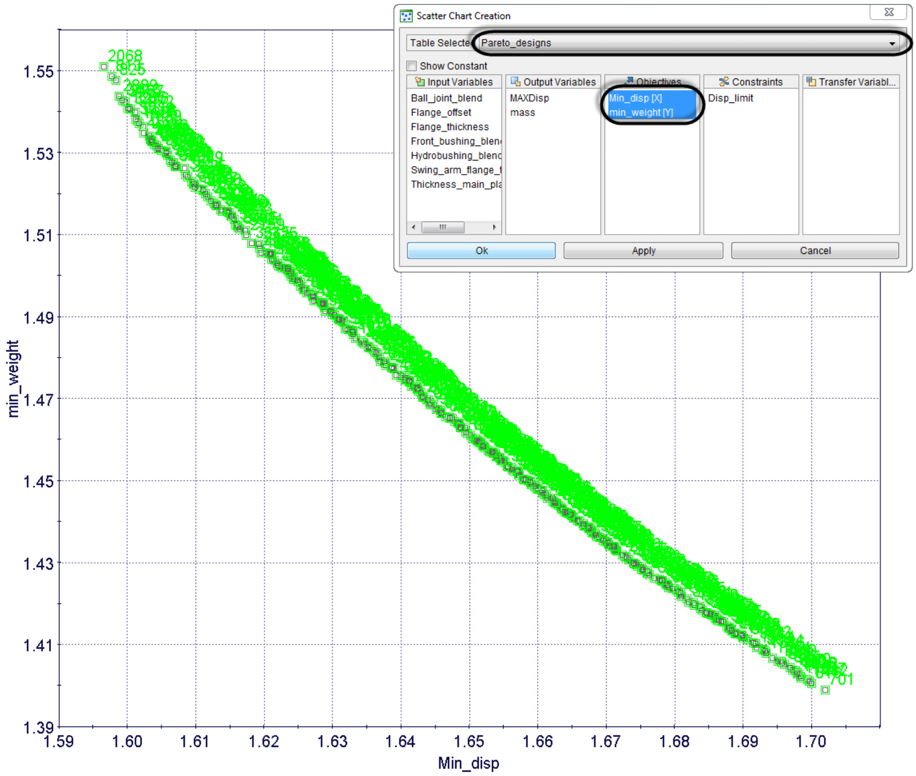


Figure 3.24: X - Y scatter of only pareto designs

Create a new X - Y scatter based on the new Pareto_designs table the same way as shown before and in figure 3.24. Now the best designs can be identified visually by holding the mouse pointer in the preferred region which will show the design ID. With this ID the corresponding design values can be identified in designs table.

3.3.5 MOGA - II

The modeFRONTIER process is shown in the picture below. This run has been given a displacement limiter which highlights the results below a certain value in the post processing values. This means that one can generate an X-Y scatter of the two objectives and mark the designs that satisfies this limiters upper value. This procedure is shown in appendix F.

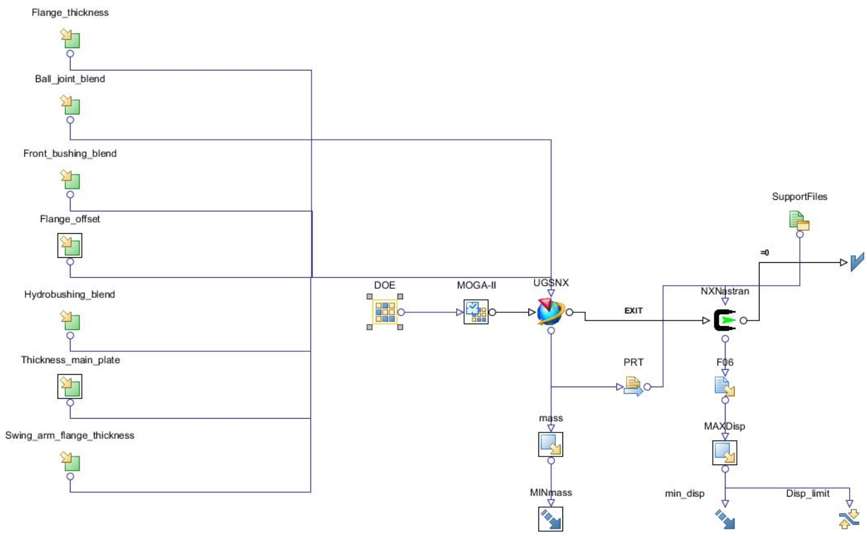


Figure 3.25: MOGA - II run with limiter

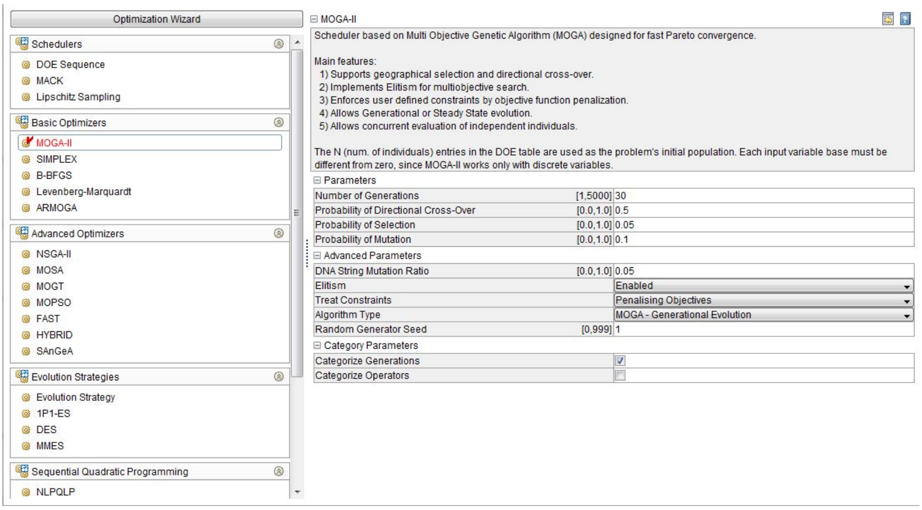


Figure 3.26: MOGA II preferences

Values chosen for this run is default configuration. The main reason for this is because it is hard to predict what parameters the optimization would benefit from having altered. The second reason is that the default values has been configured to be the most universal and suit a variety of problems in the best way.

Numbers of generations has been altered to 30 generations which adds up to 840 design iterations in total.

The pareto curve generated with the limiter shows the limiter value of 1,705 mm associated with the displacement target is shown in figure 3.27.

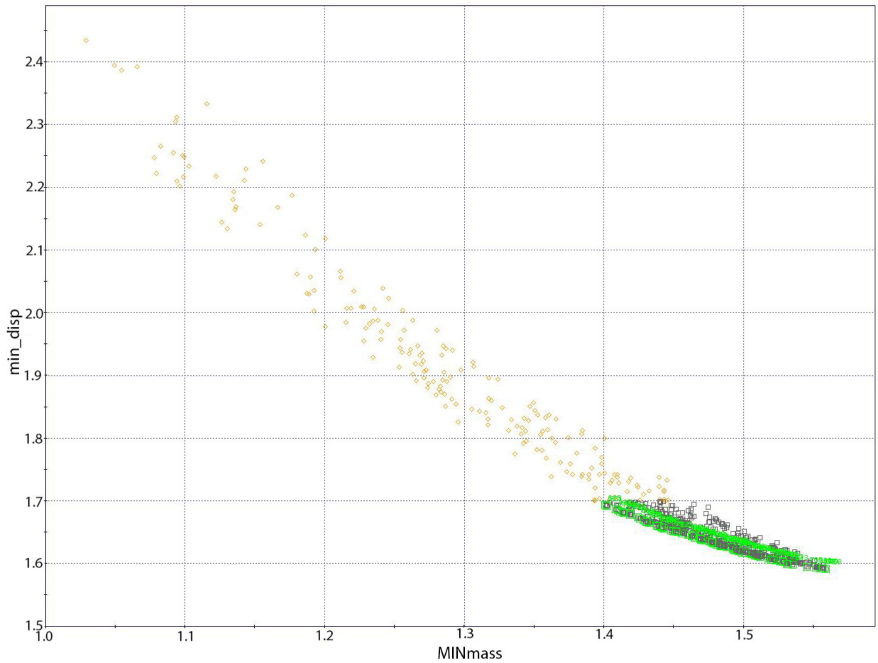


Figure 3.27: Pareto curve with limiter

From this scatter plot it is possible to mark (marking pareto designs is a function in modeFRONTIER that selects the designs satisfying limiters and designs that are located on the pareto front) feasible designs and create an additional pareto curve containing only the very best designs. modeFRONTIER is taking the limiter into account when marking the best designs which is why only the designs below 1,7 mm (min_disp) is highlighted green.

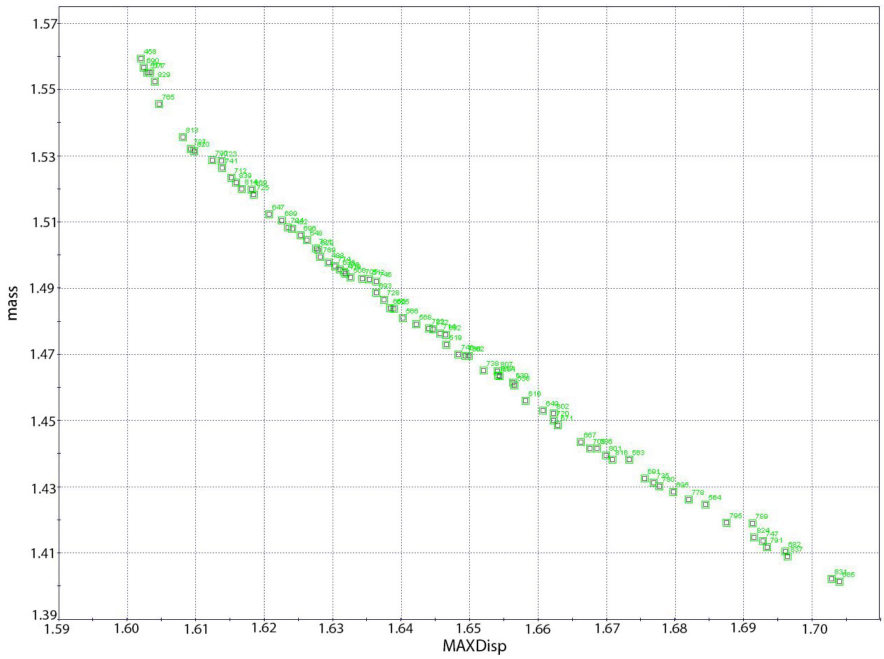


Figure 3.28: Marked pareto designs

Figure 3.28 shows the same designs as presented in the previous pareto curve with the marked designs that were saved and used to isolate and highlight these designs. The three best designs were identified and presented in table 3.2.

	Design values (ID 685)	Design values (ID 831)	Design values (ID 837)
Ball_joint_blend [mm]	17,6	17,1	17,6
Flange_offset [mm]	0,03	0,15	0,05
Flange_thickness [mm]	6,8	7	6,9
Front_bushing_blend [mm]	18	18	18,2
Hydrobushing_blend [mm]	16	16,1	16
Swing_arm_flange_thickness [mm]	16	16	16
Thickness_main_plate [mm]	0,99	1	1
MINmass [kg]	1,4040	1,4047	1,4115
Min_disp [mm]	1,6991	1,6979	1,6915

Table 3.2: Selected designs

	ID	Values	Improvement	NX Improvement
MINmass	685	1,4040 kg	3,05 %	0,39 %
Min_disp		1,6991 mm	0,36 %	0,322 %
	ID	Values	Improvement	NX Improvement
MINmass	831	1,4047 kg	3 %	0,343 %
Min_disp		1,697 mm	0,46 %	0,445 %
	ID	Values	Improvement	NX Improvement
MINmass	837	1,4115 kg	2,54 %	-0,139 %
Min_disp		1,6915 mm	0,79 %	0,76 %

Table 3.3: MOGA - II goal achievement

The best run with ID: 685 has a goal achievement of 3,05 % which is 0,39 % better than the best results produced by NX. Taken into account that the displacements are also improved by a small amount, these are considered very good results. This optimization run took approximately eight hours, which is considered a reasonable amount of time consumption compared to other algorithms.

3.3.6 MOGT

This run differs from the others by requiring only one DOE entry. This one has therefore been manually created in the DOE scheduler with a mean value to ensure a neutral starting point.

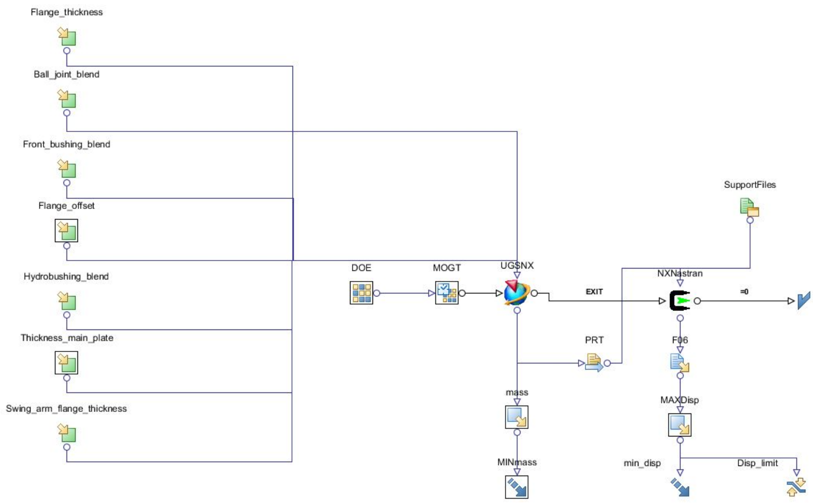


Figure 3.29: MOGT workflow

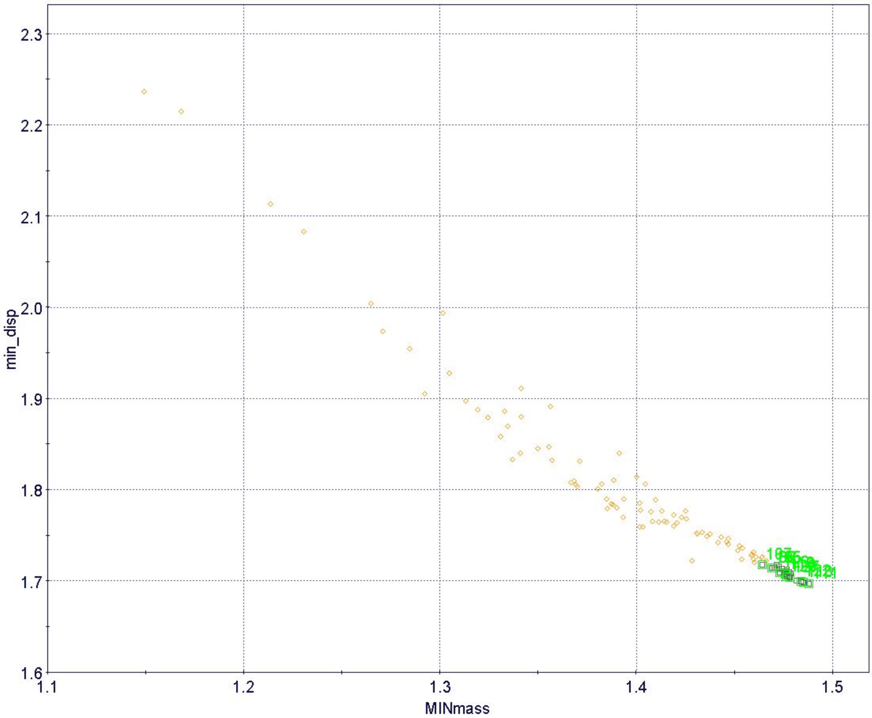


Figure 3.30: MOGT pareto curve with limiter

Figure 3.30 shows the entire MOGT run with all the generated designs.

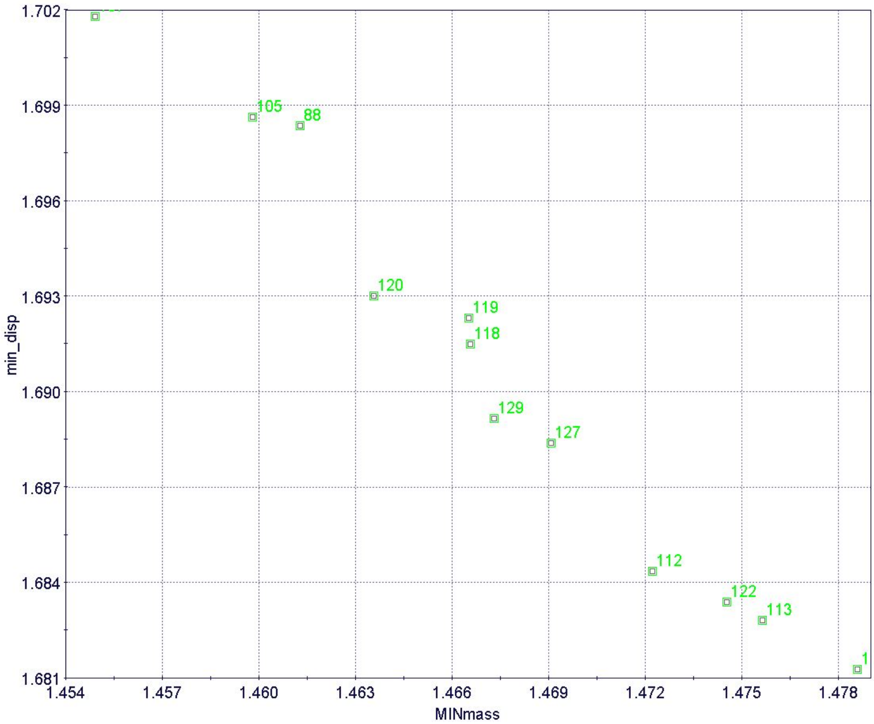


Figure 3.31: Marked designs MOGT

how

Figure 3.31 shows the best designs within the limiter that modeFRONTIER marks.

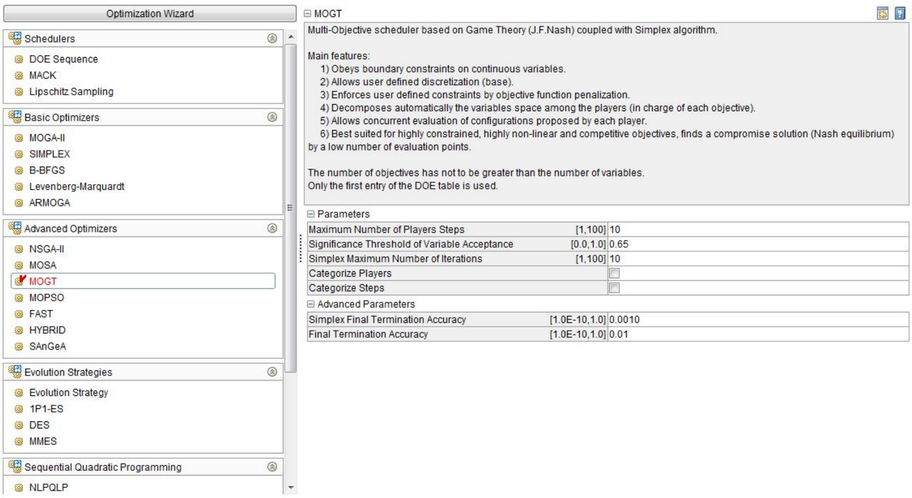


Figure 3.32: MOGT preferences

In this scheduler standard values has been used. This is for the same reasons as mentioned in chapter 3.3.5.

	Design values (ID 107)
Ball_joint_blend [mm]	18,1
Flange_offset [mm]	0,95
Flange_thickness [mm]	6,5
Front_bushing_blend [mm]	19,8
Hydrobushing_blend [mm]	17,4
Swing_arm_flange_thickness [mm]	16
Thickness_main_plate [mm]	0,09
MINmass [kg]	1,4549
Min_disp [mm]	1,7018

Table 3.4: Selected MOGT designs

	ID	Values	Improvement	NX Improvement
MINmass	107	1,4549 kg	-0,45 %	-3,21 %
Min_disp		1,7018 mm	0,18 %	0,16 %

Table 3.5: MOGT goal achievement

The best result from the MOGT run is presented in table 3.5. The weight reduction achieved with this algorithm is not very good, final weight is even worse than the initial weight. This algorithm has made the model 0,45 % heavier than initial weight. This result is bad even when considering that a run takes only 60 minutes, which is the fastest of the tested algorithms in modeFRONTIER.

3.3.7 Simplex

Simplex is a single objective optimization algorithm which resembles the NX algorithm work method. The fact that it is single objective is clearly shown by the workflow depicted in figure 3.33 that shows only one goal node (min_weight). This algorithm is included although it is single objective to compare directly with an algorithm similar to NX.

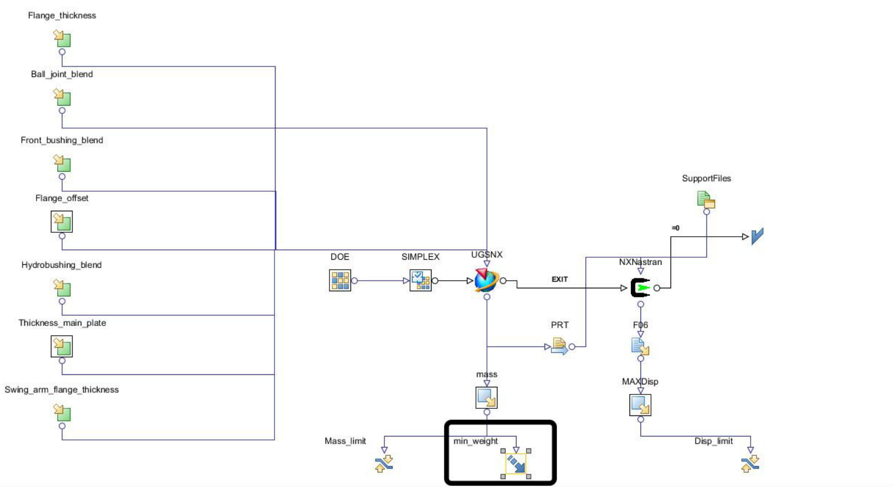


Figure 3.33: Simplex workflow

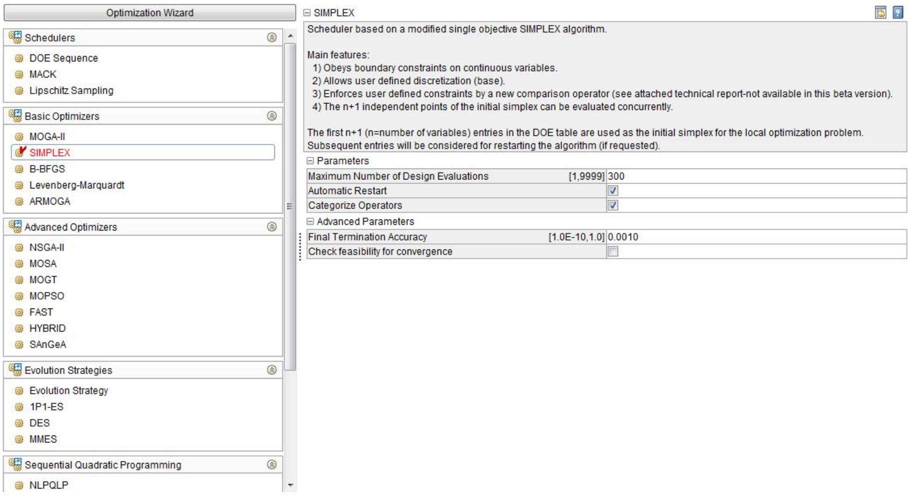


Figure 3.34: Simplex preferences

These preferences are also default, except for number of design evaluations that has been reduced to save computational time. Automatic restart has been activated to ensure that the algorithm will continue to run even though it might have found a local optimum. This means that the algorithm will start a new simplex run with a new set of DOEs when it otherwise would consider the run to be completed.

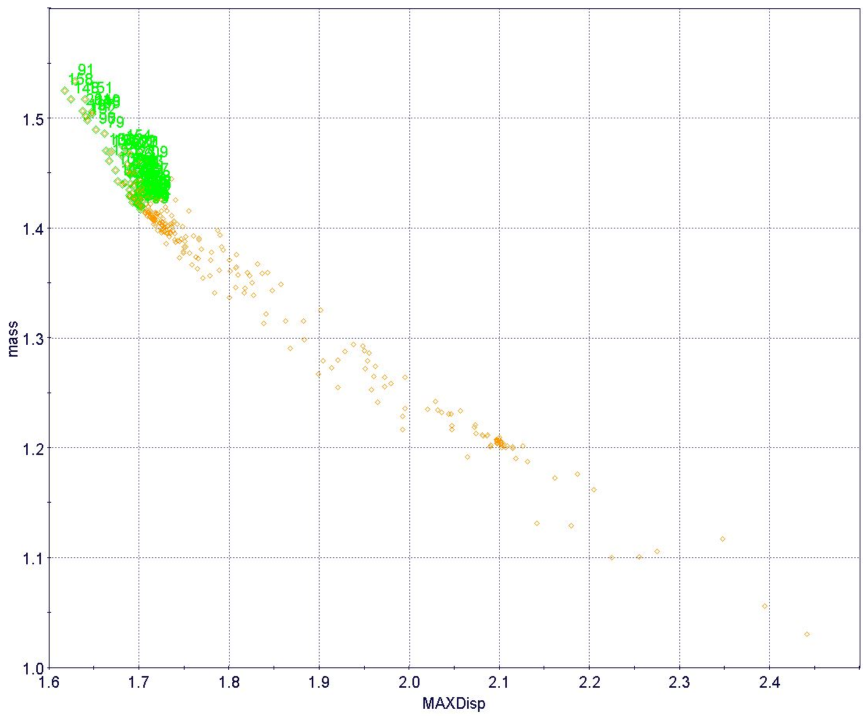


Figure 3.35: Pareto curve simplex

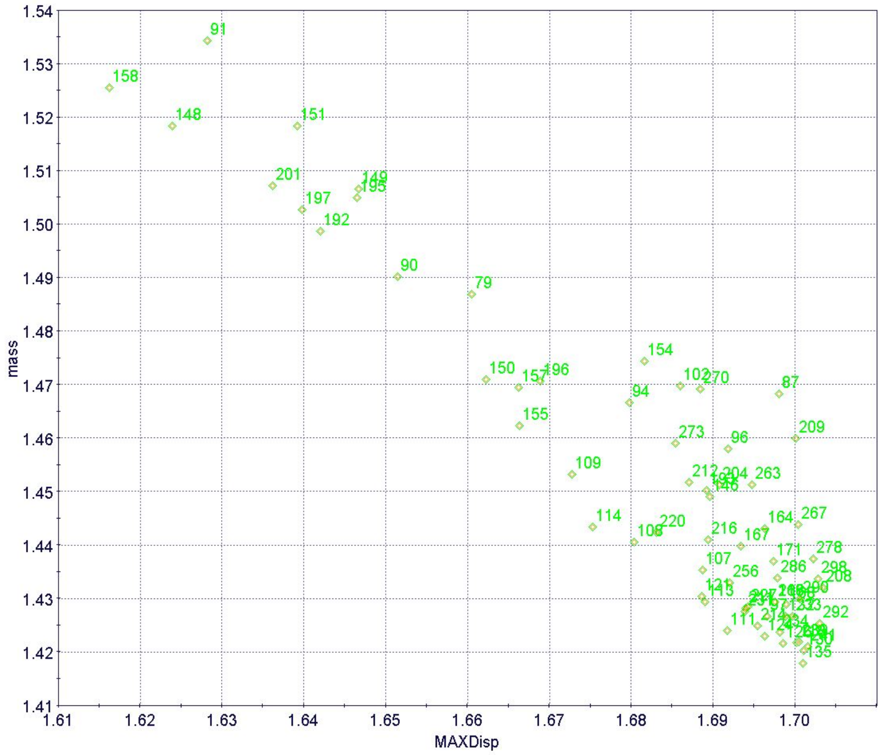


Figure 3.36: Marked design simplex

This particular run was not able to mark pareto designs automatic. A selection of designs had to be marked manually within a given range. These are depicted in figure 3.36 and also includes dominated design solutions (solutions located outside the pareto front). The best results from the chosen designs are presented in table 3.6 and 3.7.

	Design values (ID 135)	Design values (ID 130)
Ball_joint_blend [mm]	17,5	17,4
Flange_offset [mm]	0,56	0,5
Flange_thickness [mm]	7	7
Front_bushing_blend [mm]	18,6	19
Hydrobushing_blend [mm]	17,5	17,8
Swing_arm_flange_thickness [mm]	16	16
Thickness_main_plate [mm]	0,88	0,93
MINmass [kg]	1,418	1,4204
Min_disp [mm]	1,7008	1,7009

Table 3.6: Selected simplex designs

	ID	Values	Improvement	NX Improvement
MINmass	135	1,418 kg	2,09 %	-0,6 %
Min_disp		1,7008 mm	0,24 %	0,22 %
	ID	Values	Improvement	NX Improvement
MINmass	130	1,4204 kg	1,95 %	-0,77 %
Min_disp		1,7009 mm	0,24 %	0,21 %

Table 3.7: Simplex goal achievement

The best design generated with this algorithm did only improve the weight by 2,09 %, which is 0,6 % worse than NX best design. This is considered quite bad compared with the NX run given the fact that it required 300 iterations and 2,5 computational hours.

3.3.8 MOSA

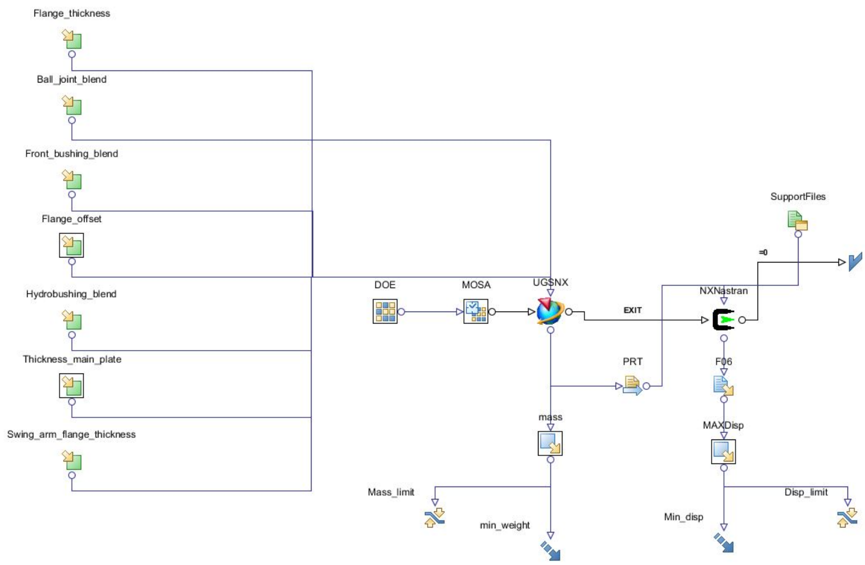


Figure 3.37: MOSA workflow

This workflow also contained an additional limiter for the mass in case it generated a lot of pareto designs.

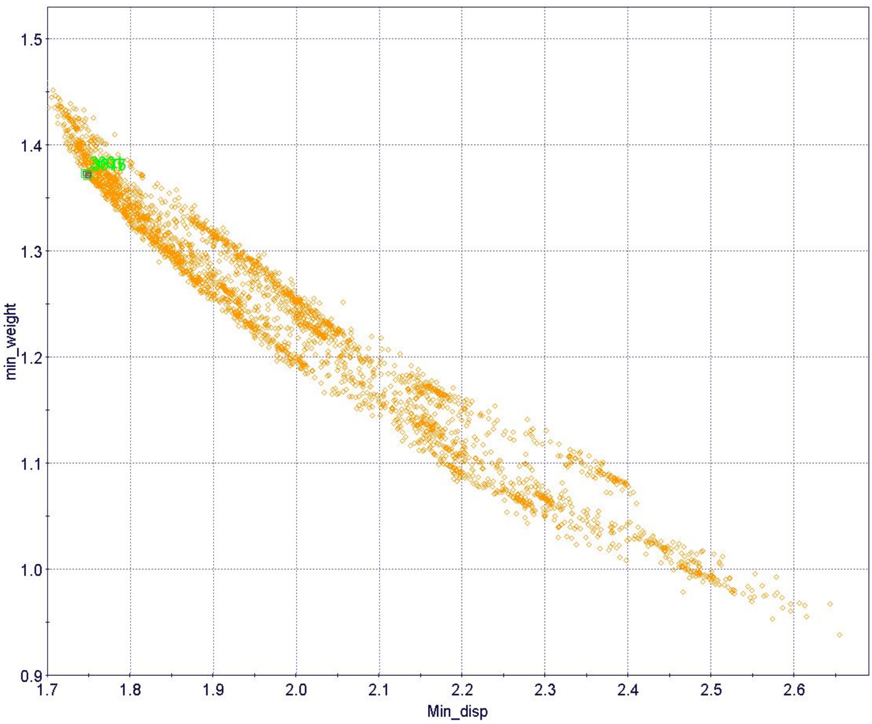


Figure 3.38: MOSA pareto curve

The MOSA run consists of almost 3000 iterations. As one can see not very many pareto designs were generated. The reason for that is the weight limiter value being too strict so only some of the designs were highlighted as satisfactory results. As a result of this, an additional limiter will not be used in the following optimizations.

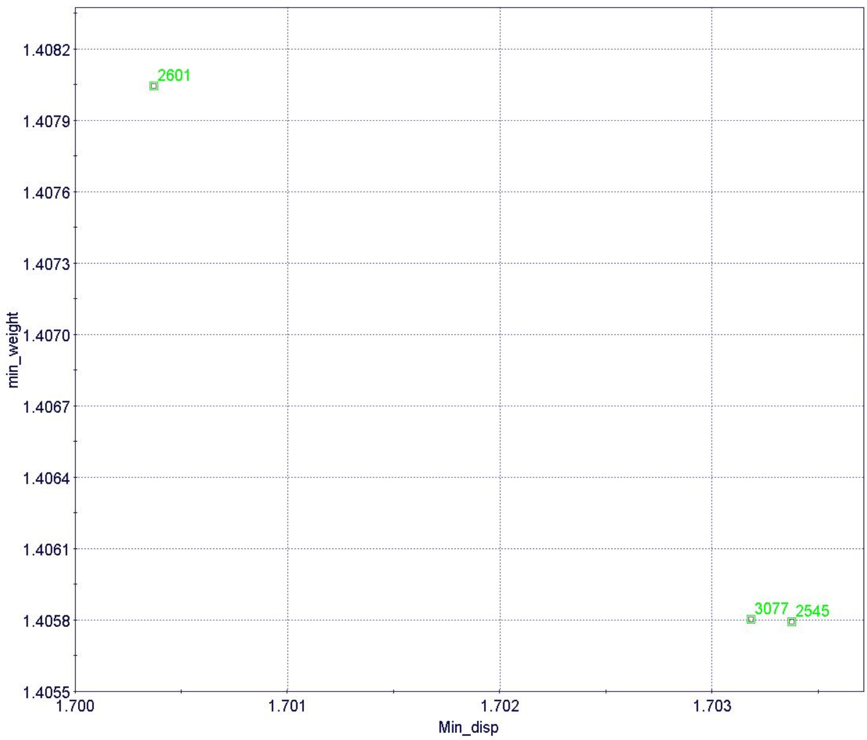


Figure 3.39: Marked designs MOSA

Even though this run did three times more iteration than MOGA - II, it only generated three designs that satisfy the displacement limit of 1,705 mm.

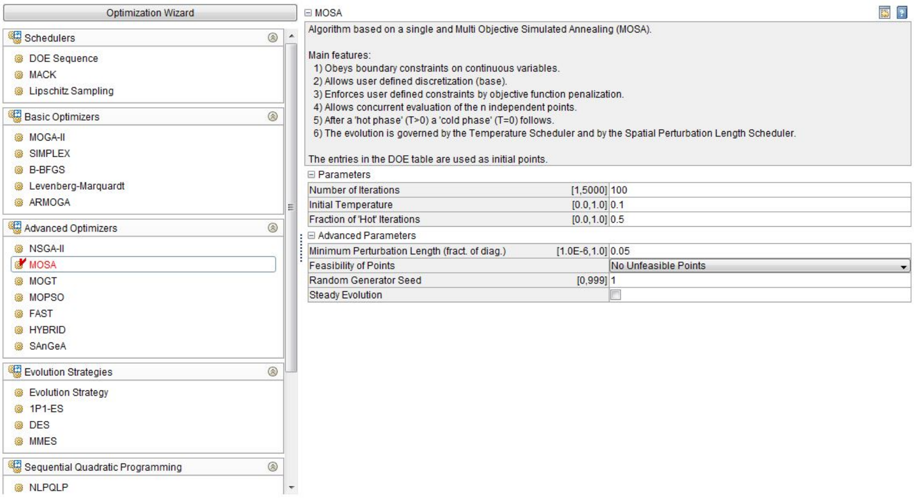


Figure 3.40: MOSA preferences

These preferences are also set to default. The initial temperature parameter controls whether the iteration process is going to be robust or not. A higher value makes it robust and a lower one will ensure faster convergence. Fraction of hot iterations will control the importance of hot and cold phases. Higher values will give the algorithm more robustness and lower gives the most converging results. The values chosen gives a good compromise.

	Design values (ID 2545)	Design values (ID 3077)	Design values (ID 2601)
Ball_joint_blend [mm]	17,3	17,3	17,1
Flange_offset [mm]	0,08	0,06	0,01
Flange_thickness [mm]	6,8	6,8	6,8
Front_bushing_blend [mm]	19,2	19,2	19,2
Hydrobushing_blend [mm]	16	16,1	16,1
Swing_arm_flange_thickness [mm]	15,9	15,9	15,9
Thickness_main_plate [mm]	0,88	0,9	0,91
MINmass [kg]	1,4058	1,4058	1,4080
Min_disp [mm]	1,7037	1,7035	1,7007

Table 3.8: Selected MOSA designs

	ID	Values	Improvement	NX Improvement
MINmass	2545	1,4058 kg	2,98 %	0,26 %
Min_disp		1,7037 mm	0,07 %	0,05 %
	ID	Values	Improvement	NX Improvement
MINmass	3077	1,4058 kg	2,93 %	0,26 %
Min_disp		1,7035 mm	0,08 %	0,06 %
	ID	Values	Improvement	NX Improvement
MINmass	2601	1,4080 kg	2,78 %	0,11 %
Min_disp		1,7007 mm	0,25 %	0,22 %

Table 3.9: MOSA goal achievement

These are the result acquired by a MOSA run. The entire run is made up of nearly 3000 iterations that takes almost 100 hours to finish. This is by far the most computational expensive algorithm. It is also the most exploratory which one would think yielded the best results because of an extensive design space. Despite its large search, it ends up almost as good as the MOGA - II run.

3.3.9 Hybrid

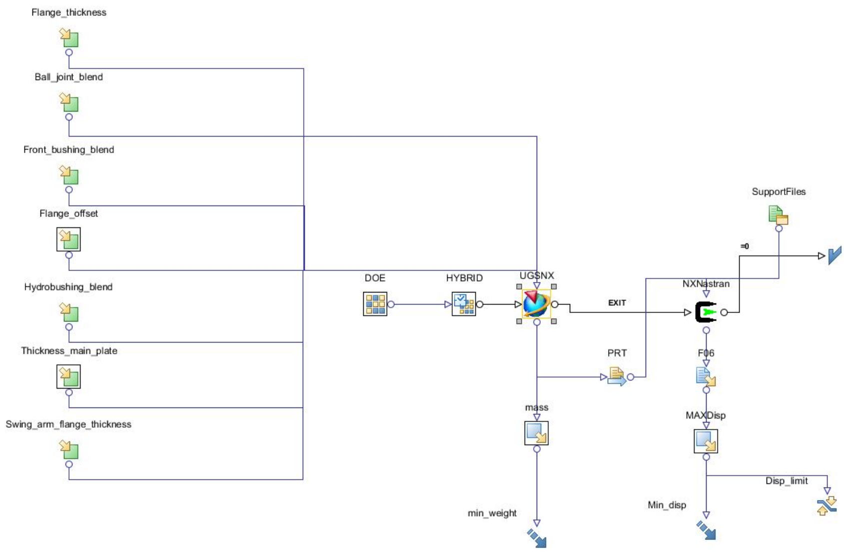


Figure 3.41: Hybrid workflow

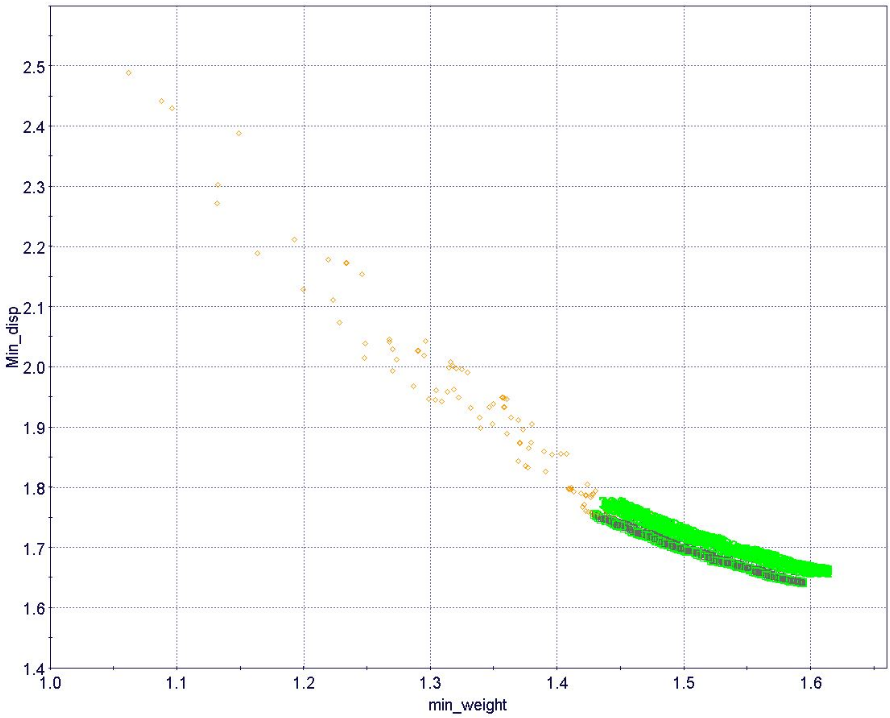


Figure 3.42: Hybrid pareto curve

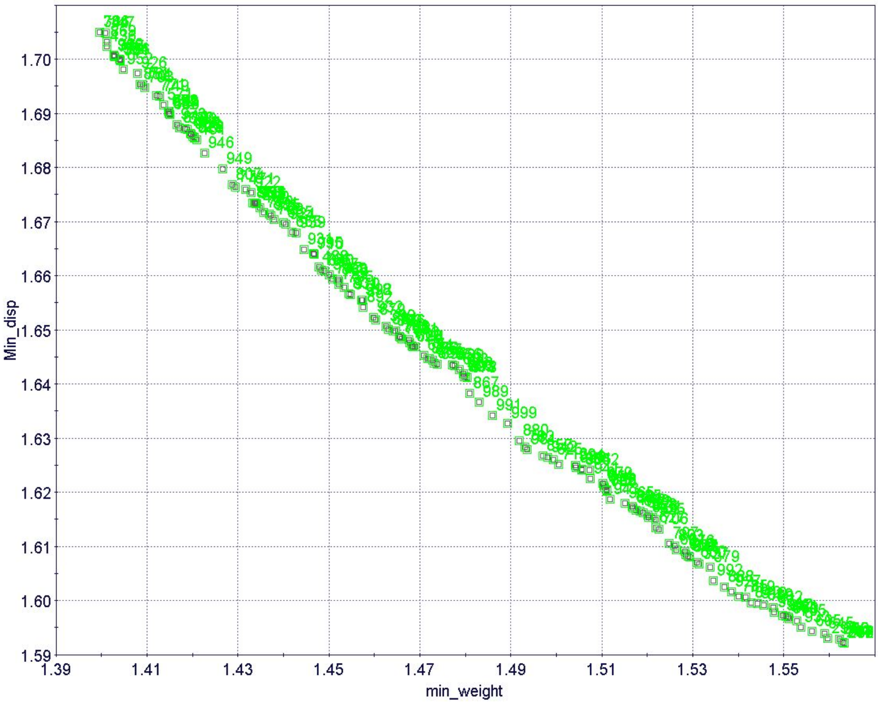


Figure 3.43: Marked designs Hybrid

This run was overall better than many of the previous optimization runs. This run presents a lot of acceptable designs that fall within the limiters requirement and pareto front. Both figure 3.42 and 3.43 shows this fact.

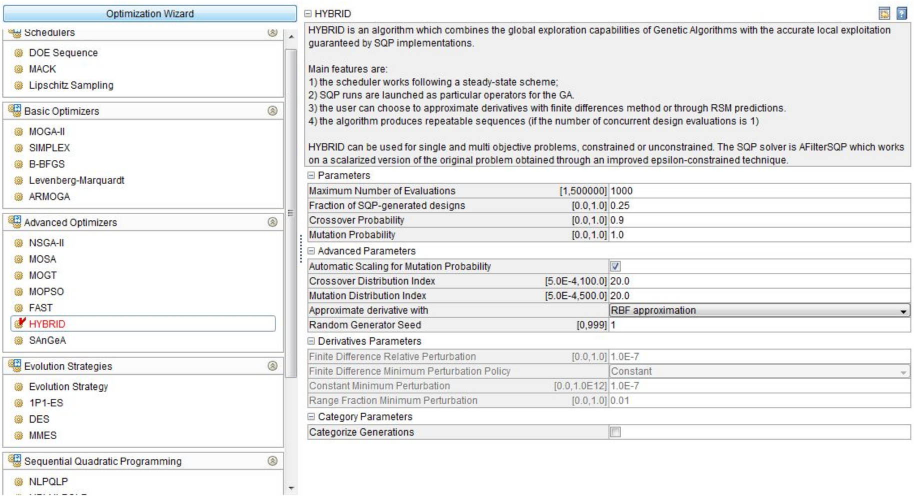


Figure 3.44: Hybrid preferences

As with all of the other scheduler properties, this one is also using default configuration. The only thing altered is number of maximum iterations which is set to 1000 to save time. This run still took 11,5 hours to complete.

	Design values (ID 344)	Design values (ID 786)	Design values (ID 147)
Ball_joint_blend [mm]	17,112	17,11	17,1
Flange_offset [mm]	0,01	0,01	0,01
Flange_thickness [mm]	6,8393	6,839	6,839
Front_bushing_blend [mm]	18,95	18,95	1,92
Hydrobushing_blend [mm]	16	16,00	16
Swing_arm_flange_thickness [mm]	15,96	15,96	15,96
Thickness_main_plate [mm]	0,9934	0,9934	0,993
min_weight [kg]	1,3995	1,3996	1,4008
Min_disp [mm]	1,7050	1,7050	1,7048

Table 3.10: Selected Hybrid designs

	ID	Values	Improvement	NX Improvement
min_weight		1,3995 kg	3,368 %	0,7123 %
Min_disp		1,7050 mm	0 %	-0,023 %
	ID	Values	Improvement	NX Improvement
min_weight		1,3996 kg	3,362 %	0,7 %
Min_disp		1,7050 mm	0 %	-0,02 %
	ID	Values	Improvement	NX Improvement
min_weight		1,4008 kg	3,27 %	0,62 %
Min_disp		1,7048 mm	0,01 %	-0,01 %0

Table 3.11: Hybrid goal achievement

This Hybrid run yielded the best results of all five initial runs.

3.3.10 Additional Searches

As the initial runs with the different algorithms has been completed, it is established which of the two algorithms that works best on this problem. This is why two additional runs with these schedulers is to be run. The sensitivity analysis in chapter 3.3.2 indicates which parameters is the most important for the overall goal. The resolution of each of the important design variables are therefore highlighted. The increments (steps) of these variables has been increased, resulting in a bigger base (number of possible designs for the specified range) that in turn creates additional design possibilities for the DOE sampler and scheduler. These two runs will use the same DOE sampling (ULH with 50 samplings) with the same incremental steps and the same design base as shown in 3.45.

Input Variable

	Name	Lower Bound	Central Value	Upper Bound	Base	Step
0	Flange_offset	0.01	1.505	3.0	300	0.01
1	Thickness_main_plate	0.01	0.505	1.0	100	0.01
2	Flange_thickness	6.0	6.5	7.0	11	0.1
3	Ball_joint_blend	17.1	17.6	18.1	11	0.1
4	Front_bushing_blend	18.0	20.0	22.0	81	0.05
5	Hydrobushing_blend	16.0	17.0	18.0	201	0.01
6	Swing_arm_flange_thickness	12.0	14.0	16.0	5	1.0

Figure 3.45: Modified design increments

The method for extracting results is the same as for the previous runs, this is why only the results are presented for these last runs.

3.3.11 Additional MOGA - II Search

From the previous runs one has established that MOGA - II seemed to be one of the most effective algorithm for this kind of problem. This is why it is interesting to run an additional optimization based on this information.

A final run with MOGA - II is to be executed with the use of an increased DOE sampling of 50 design and 100 generations in the MOGA - II scheduler (100 generations was specified in the MOGA - II preferences). The increments from figure 3.45 is used. This result is from a quite comprehensive optimization consisting of over 4000 iterations.

	Design values (ID 701)	Design values (ID 816)	Design values (ID 1482)
Ball_joint_blend [mm]	17,1	17,1	17,1
Flange_offset [mm]	0,06	0,03	0,03
Flange_thickness [mm]	6,9	6,9	6,9
Front_bushing_blend [mm]	18,05	18	18
Hydrobushing_blend [mm]	16,01	16	16,06
Swing_arm_flange_thickness [mm]	16	16	16
Thickness_main_plate [mm]	1	1	1
MINmass [kg]	1,3989	1,4006	1,4012
Min_disp [mm]	1,702	1,7000	1,6996

Table 3.12: The best MOGA - II designs

<table>
<tr><th></th><th>ID</th><th>Values</th><th>Improvement</th><th>NX Improvement</th><th>Improvement last MOGA - II</th></tr>
<tr><td>min_weight</td><td rowspan="2">701</td><td>1,3989 kg</td><td>3,41 %</td><td>0,89 %</td><td>0,36 %</td></tr>
<tr><td>Min_disp</td><td>1,702 mm</td><td>0,17 %</td><td>0,03 %</td><td>-0,17 %</td></tr>
<tr><td></td><td>ID</td><td>Values</td><td>Improvement</td><td>NX Improvement</td><td></td></tr>
<tr><td>min_weight</td><td rowspan="2">816</td><td>1,4006 kg</td><td>3,29 %</td><td>0,63 %</td><td></td></tr>
<tr><td>Min_disp</td><td>1,700 mm</td><td>0,29 %</td><td>0,26 %</td><td></td></tr>
<tr><td></td><td>ID</td><td>Values</td><td>Improvement</td><td>NX Improvement</td><td></td></tr>
<tr><td>min_weight</td><td rowspan="2">1482</td><td>1,4012 kg</td><td>3,25 %</td><td>0,59 %</td><td></td></tr>
<tr><td>Min_disp</td><td>1,6996 mm</td><td>0,31 %</td><td>0,29 %</td><td></td></tr>
</table>

Table 3.13: Additional MOGA - II goal achievement

This final run has proven to yield better results than the initial MOGA - II run. The best results generated was 0,36 % better than the last one. The NX improvement was 0,89 % better and compared to the base line weight it was improved by 3,41 %.

The iteration time was 115 hours total which was almost 15 times more than the first run. The computation time required for this last run was 107 hours more for an improvement of 0,36 %. These are better results, but this is a small improvement since the last run, considering the additional 107 hours of computational time. Because the algorithm evolves, one would expect the latest design ids in a run consisting of over 4000 iterations to yield the best result. In spite of this, some of the first iterations (701, 816 and 1482) saves the most weight in this optimization.

3.3.12 Additional Hybrid Search

A second run with Hybrid algorithm has also been done to see if it is possible to generate even better results than the previous Hybrid and MOGA - II run. The new search will use the same increments for use with the same DOE sampler as shown in figure 3.45. In the new run a maximum of 3000 iterations has been specified. This is three times the amount of iterations as the first Hybrid run. The computational time was 75 hours for this optimization.

	Design values (ID 1318)	Design values (ID 2494)	Design values (ID 2174)
Ball_joint_blend [mm]	17,2	17,108	17,1
Flange_offset [mm]	0,019	0,0144	0,0396
Flange_thickness [mm]	6,79	6,72	6,87
Front_bushing_blend [mm]	18	18	18,07
Hydrobushing_blend [mm]	16,1	16	16
Swing_arm_flange_thickness [mm]	16	15,99	16
Thickness_main_plate [mm]	0,995	0,855	0,997
MINmass [kg]	1,3966	1,3976	1,3977
Min_disp [mm]	1,7045	1,7031	1,7030

Table 3.14: The best Hybrid designs

	ID	Values	Improvement	NX Improvement	Improvement last Hybrid
min_weight	1318	1,3966 kg	3,569 %	0,9181 %	0,2 %
Min_disp		1,7045 mm	0,029 %	0,005 %	0,02 %
	ID	Values	Improvement	NX Improvement	
min_weight	2494	1,3976 kg	3,5 %	0,84 %	
Min_disp		1,7031 mm	0,11 %	0,08 %	
	ID	Values	Improvement	NX Improvement	
min_weight	2174	1,3977 kg	3,49 %	0,84 %	
Min_disp		1,7030 mm	0,11 %	0,09 %	

Figure 3.46: Additional Hybrid goal achievement

This additional Hybrid run did not yield much better results than the initial Hybrid even though it took almost seven times longer. There was some improvement in result compared to the MOGA - II run. The two algorithms both produced good results although MOGA - II consisted of more iterations. Even in this analysis one of the earlier iterations stands out among the best, although one might think that the latest iterations should be the most optimized.

3.3.13 Local search

The last step in an optimization after an extensive search would be to fine tune the best designs. This can be done by using the best designs from previous runs for a new local search based on these designs. Marked designs is imported into the DOE scheduler and then an optimization run is performed as usual. This local search used the marked designs as DOE scheduler and make use of the Hybrid scheduler with default preferences.

	Design values (ID 6562)
Ball_joint_blend [mm]	17,27
Flange_offset [mm]	0,0161
Flange_thickness [mm]	6,79
Front_bushing_blend [mm]	18
Hydrobushing_blend [mm]	16
Swing_arm_flange_thickness [mm]	16
Thickness_main_plate [mm]	0,989
min_weight [kg]	1,3958
Min_disp [mm]	1,7048

Table 3.15: Best designs from the local search

	ID	Values	Improvement	NX Improvement	Improvement from additional Hybrid
min_weight	6562	1,3958 kg	3,6243 %	0,9748 %	0,05 %
Min_disp		1,7048 mm	0,0117 %	-0,011 %	-0,017 %

Table 3.16: Local search goal achievement

This last run was not expected to give much better results than the ones before since the design space is already considered explored. The main idea is to fine tune the best designs to get the last potential for improvement. An improvement of 0,05 % from the last additional run is an infinitesimal small improvement, but the small improvements indicates that the earlier searches have been thorough.

3.3.14 Discussion

In chapter three, all of the modeFRONTIER related topics regarding Multi-objective Design Optimization has been elaborated. It starts with DOE and scheduler theory, then continue with configuration of modeFRONTIER and how to extract results. The standardized DOE sequence chosen for the initial benchmark runs is presented. Sensitivity analysis is included to illustrate how one can use this information to improve optimizations. Then results from the initial benchmark runs is presented. This set of benchmark runs has been conducted to identify the algorithms that produces the best results. The best ones is then tested further to find out if they can perform better, which they prove is possible. Although they perform better it is not as much as the objective implies. A local search has been run in addition to the two extensive searches to explore more thoroughly amongst the best designs. This last search has been run to complement the overall search method.

There is reason to believe that the control arm part has been previously optimized manually by car manufacturers since it has been in production for many years. The potential for further optimization at this stage may be reduced because of the fact that it has been optimized in advance.

It is possible that the part might have been optimized in regard to other criteria such as buckling. In the previous optimizations the part should ideally have had more requirements regarding buckling and other types of requirements related to its use.

Chapter 4

Conclusion

An optimization guide has been developed for NX and modeFRONTIER. Different types of algorithms and software has been evaluated. The guide also provides a detailed step by step explanation of configuration and extraction of results in NX and modeFRONTIER, both in the thesis and more in-depth in the provided appendices. The thesis starts by presenting an optimal strategy for identifying which expressions that are most suited for the largest impact on product requirements, which creates a good parameterization. Based on this parameterization an automatic optimization was carried out in NX with its built-in geometry optimization module. The NX algorithm excelled on performance compared to time spent when the results from this optimization was compared with modeFRONTIERs different algorithms later on. A brief explanation of some of the available scheduler and DOE algorithms has also been provided in the multiobjective design optimization chapter. The modeFRONTIER optimization section aims to present an optimal approach for optimizations with no information to predict which algorithm performs best.

Five different algorithms were benchmarked with a standardized DOE setup to compare results with the most uniform basis. The goal of the benchmark was to evaluate the result with respect to ease of use and final product requirements. The criteria for ease of use is based on how much time each optimization took. After this initial benchmarking, the two algorithms that performed best were chosen for an extended optimization run, as well as a local search based on previous designs.

Algorithm	Weight reduction	Time spent [h]	Number of iterations
NX single objective	2,67 %	0,5	16
MOGA - II	3,05 %	8	731
MOGT	-0,46 %	1	104
MOSA	2,93 %	100	2935
Simplex	2,09 %	2,5	297
Hybrid	3,36 %	11,5	973
Additional extended runs			
New MOGA - II	3,41 %	115	4056
New Hybrid	3,56 %	75	2917
Local search			
Hybrid	3,62 %	22	1936

Table 4.1: Goal achievement in percentage

From the modeFRONTIER runs there is a pattern predicting that the amount of time spent is related to goal achievement. MOSA stand out in the low end of the scale and performs exceptionally bad when computational time is compared to goal achievement. The results generated by NX is the best result relative to time spent solving. From table 4.1 one can see that both MOGA - II and Hybrid performs well. Hybrid generates better results at the expense of more time used computing (in the initial run). From the additional optimizations, Hybrid gives a better result than MOGA - II in almost half the time.

From table 3.12 and 3.14 the best designs from large searches are shown. It is not always one of the latest iterations performed in the run that produces the best result, as one would expect from evolutionary algorithms. This does however not mean that it is a waste of time to run large searches as these searches usually return better overall results.

The local Hybrid search produced very small relative improvement from the extended search, but this was expected in advance because of an already explored search space.

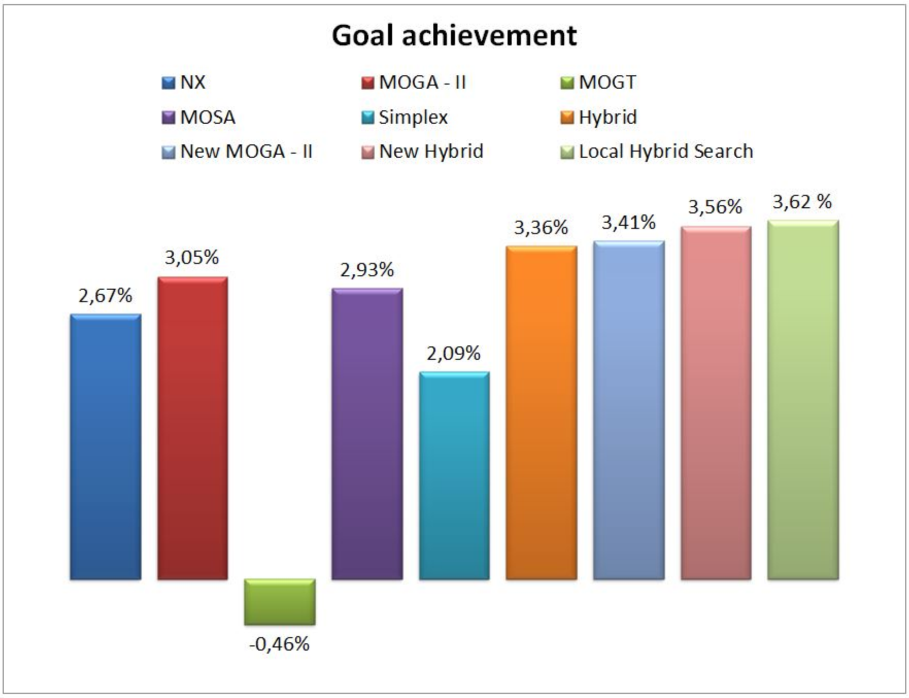


Figure 4.1: Goal achievement chart

A particularly interesting thing was the initial Hybrid run that produced very good results compared to its time spent solving. The first Hybrid run results was close to the additional extensive MOGA - II run in just one tenth of the time.

Based on modeFRONTIERs design optimization results considering goal achievement and time spent solving, the best algorithm was Hybrid. This is because it generates the best results although it is somewhat time consuming compared to NX, yet faster than the additional MOGA - II run which it is directly comparable with.

NX optimization is considered the best option if saving time is the most important aspect. This optimization is in its own class regarding time because it is a built in module. This saves a lot of time compared to modeFRONTIER because it does not need to call up external applications the way modeFRONTIER does.

Most of the modeFRONTIER runs involves very extensive searches. It is possible to scale down these searches to save time by reducing the number of total iterations necessary to complete a run. By use of the sensitivity analysis, one could remove one or more input variables to decrease the number of iterations necessary to perform a good exploration run.

The goal of 10 % weight reduction was not reached, but a goal achievement of 3,56 % is considered very good since the part has been optimized by the automobile industry for several years.

Problems stated in this thesis with its subtasks has been solved in addition to provide a practical guide for optimization both in this thesis and appendices.

Chapter 5

Further Work

One potential topic for further work is the specifications options (change the settings from default) and determine if that could generate better designs. It is conceivable that there exists more undiscovered functionality and additional algorithms within modeFRONTIER that could help in getting better results faster. More local searches with different algorithms based on good designs might have the potential to improve the optimization further. This has not been covered in depth due to insufficient time available. It might have been interesting to run optimizations within both NX and modeFRONTIER that also include multiaxial load cases. Buckling included in the optimization would give a more comprehensive picture of the overall design requirements and goal achievement. In addition to wider searches with more iteration. Most of these tasks would require enormous computational time which is unfortunate in a product improvement process where time is a great concern. It might be possible to configure modeFRONTIER to run NX in batch mode (a mode that does not require the actual program to start visually). This would save a lot of time since NX is currently restarted for each iteration.

If a more automated sorting of results were desirable, one could create a macro that exports modeFRONTIER design results. From this excel sheet it would be possible to rank and sort the most desired designs based on weighting of goals. This best design could in turn be imported into NX and export a CAD model with updated optimized designs.

Bibliography

[1] Raufoss AS. Fwd: Sv: Lastscenarioer for flca suplight; masteroppgave ved ntnu. E-mail, January 2013.

[2] cenaero.be. Pareto front. http://www.cenaero.be/Page.asp?docid=27103&langue=EN.

[3] UGS Corp. *NX Nastran 3 Quick reference guide*, volume 2. UGS Corp., 2004.

[4] Ph.D Dr. Marc Ratzel, Fatma Kocer. Cfd optimization with altair hyperworks. http://blog.altair.co.kr/wp-content/uploads/2011/03/hyperstudy_10-0.pdf, May 2007.

[5] Altair Engineering. Introduction to hyperworks. http://www.altairhyperworks.com/(S(3fu2zyrlbyi03xcofiue25jd))/hwhelp/Altair/hw11.0/help/hst/hst.htm?least_squares_regression.htm, 1999-2012.

[6] ESTECO. Doe techniques. http://www.esteco.com/home/mode_frontier/Optimization/DOE.html, February.

[7] Esteco. Schedulers. http://www.esteco.com/home/mode_frontier/Optimization/Schedulers.html, 2012.

[8] Esteco. Esteco, 2013. modeFRONTIER documentation.

[9] Peter Aitchison G. Gary Wang, Zuomin Dong. Adaptive response surfacemethod - a global optimization scheme for approximation-based design problems. http://www.tandfonline.com/doi/pdf/10.1080/03052150108940940, April 2007.

[10] Steffen Johnsen. Structural topology optimization. Master's thesis, NTNU, 2013.

[11] Marijke Keet. Genetic algorithms - an overview. http://www.meteck.org/gaover.html, May 2002.

[12] Peter W. Christensen; Anders Klarbring. *An Introduction to Structural Optimization*. Springer, 2009.

[13] Granta Design Limited. Ces edupack. Software, 2012. v. 11.9.9.0.

[14] J. Haslinger; R.A.E. Mäkinen. *Introduction to Shape Optimization. Theory, Approximation, and Computation*. SIAM - Society for Industrial and Applied Mathematics, 2003.

[15] S. Rechak J.M. Roelandt N. Amoura, H. Kebir. Axisymmetric and two-dimensional crack identification using boundary elements and coupled quasi-random downhill simplex algorithms. http://www.sciencedirect.com/science/article/pii/S0955799710000317, June 201.

[16] Opel. Insignia 5-dørs. http://www.opel.no/kjoretoy/opel-serie/biler/insignia-5-dors/indeks.html, February 2013.

[17] S. Saha S. Bandyopadhyay. Some single- and multiobjective optimization techniques. http://link.springer.com/chapter/10.1007%2F978-3-642-32451-2_2, 2013.

[18] Siemens. *Siemens NX 8.5 Documentation*. Siemens Product Lifecycle Management Software Inc., 8.5 edition, October 2012.

[19] Tea Robic Silvia Poles, Enrico Rigoni. Moga 2 performance on noisy optimization problems. http://dis.ijs.si/tea/Publications/Poles04MOGA.pdf. downloaded as pdf from google.

[20] SuPLight. Welcome to suplight homepage! http://www.suplight-eu.org/, January 2013.

[21] Wikipedia. Cygwin. http://en.wikipedia.org/wiki/Cygwin, May 2013.

[22] Wikipedia. Makro (datateknik). http://sv.wikipedia.org/wiki/Makro_(datateknik), march 2013.

[23] Wikipedia. No free lunch in search and optimization. http://en.wikipedia.org/wiki/No_free_lunch_in_search_and_optimization, February 2013.

[24] Wikipedia. Simulated annealing. http://en.wikipedia.org/wiki/Simulated_annealing, April 2013.

[25] David L. Pepyne Yu-Chi Ho. Simple explanation of the no free lunch theorem of optimization. http://www.cc.gatech.edu/~jlee716/ml/NoFreeLunch.pdf, December 2001. Downloaded as pdf via google.

Appendix A

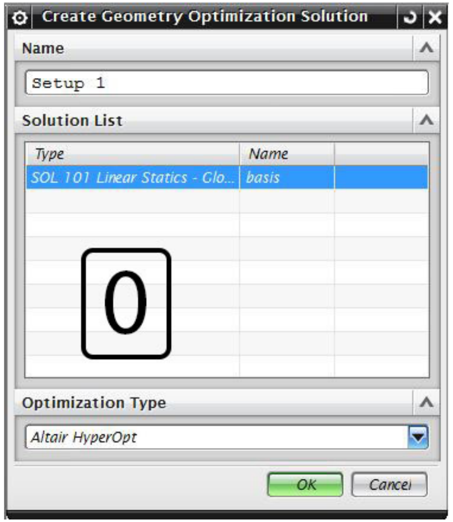
A3 Briefing NX

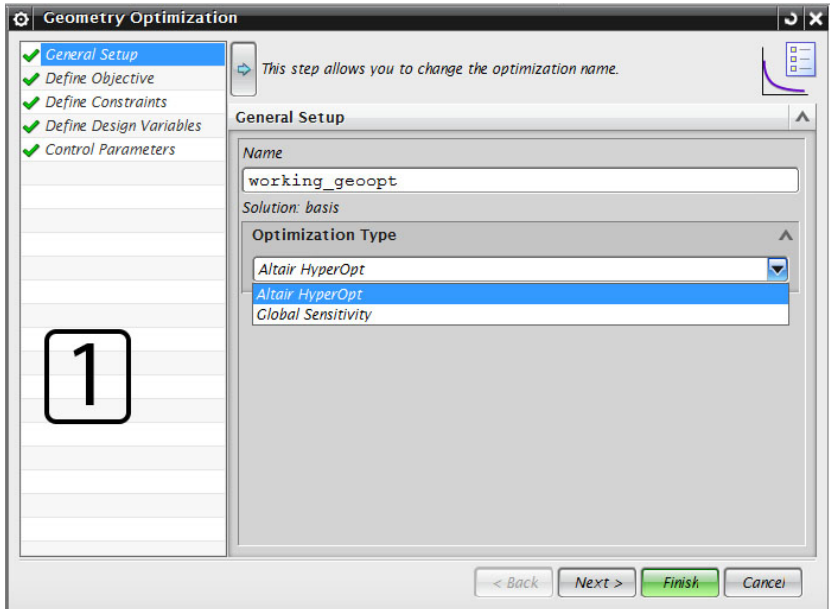
Geometry Optimization

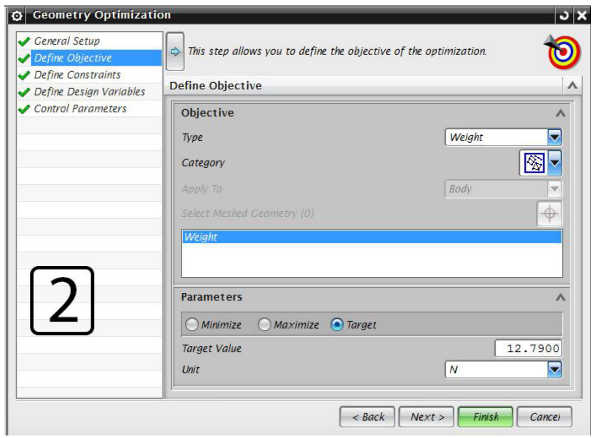
Topic: NX8.5 Optimization Brief | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: March 2013

To perform a geometry optimization in NX it is nescessary to do a initial standard linear simulation with the correct load cases. This is because the geometry analysis uses this as a base line. This is shown in step 0.

1. First off you get to choose what type of optimization you want - choose Altair Hyperopt
2. In this next step it is possible to define your objective. Based on experience it is best to define a target rather than just minimize.







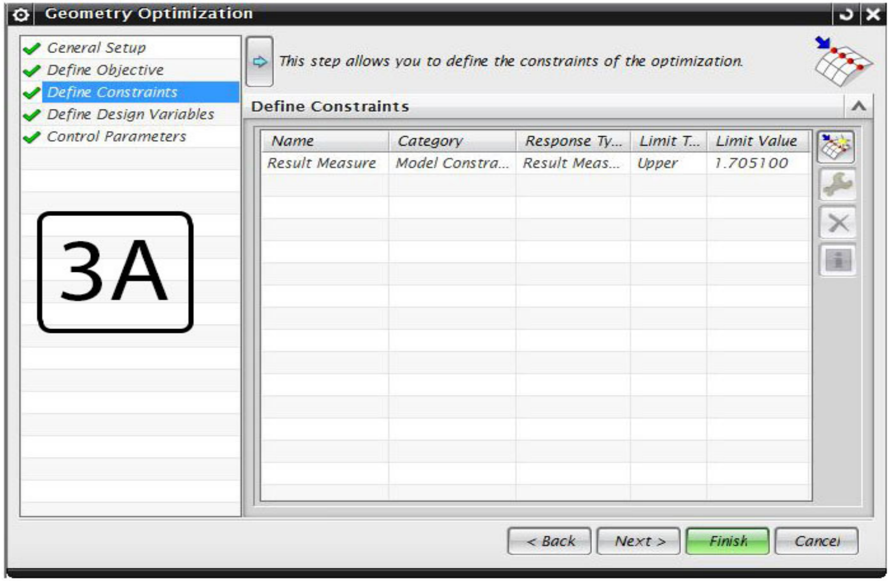
suplight

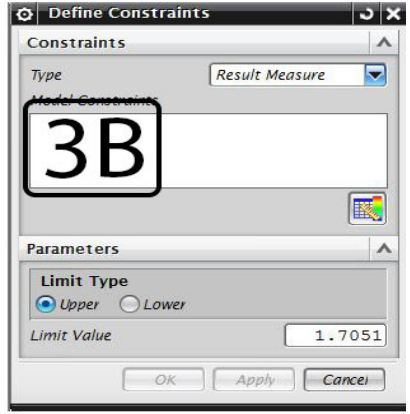
UGS The PLM Company | modeFRONTIER | NTNU

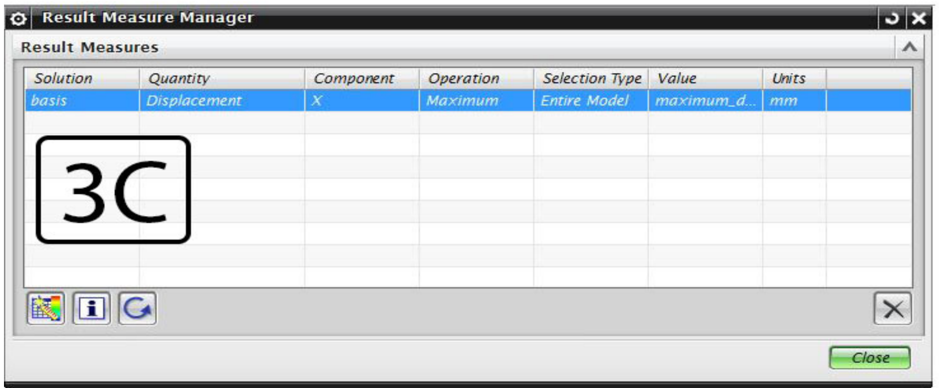
Geometry Optimization

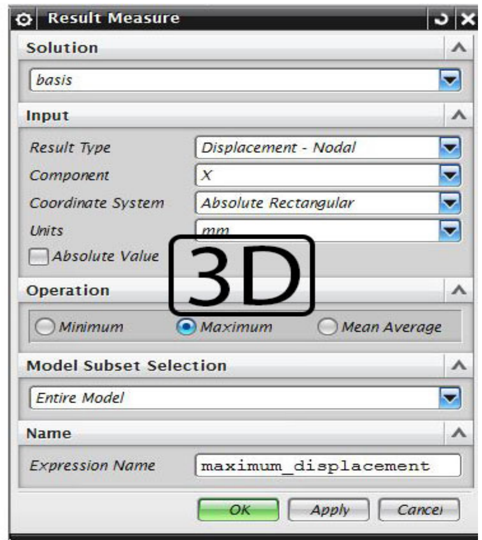
Topic: NX8.5 Optimization Brief | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: March 2013

3. This step is where you link your requirement to the base line that were run before the optimization creation. It is also possible to define which coordinate system the solution should use. In the menu shown in picture 3B it is possible to define some leeway in the constraint for the solution.









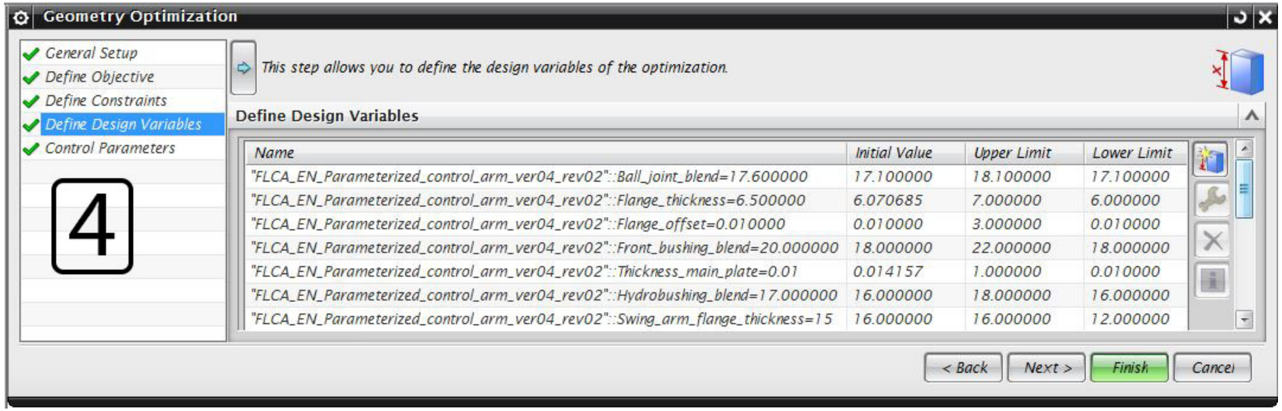
Geometry Optimization

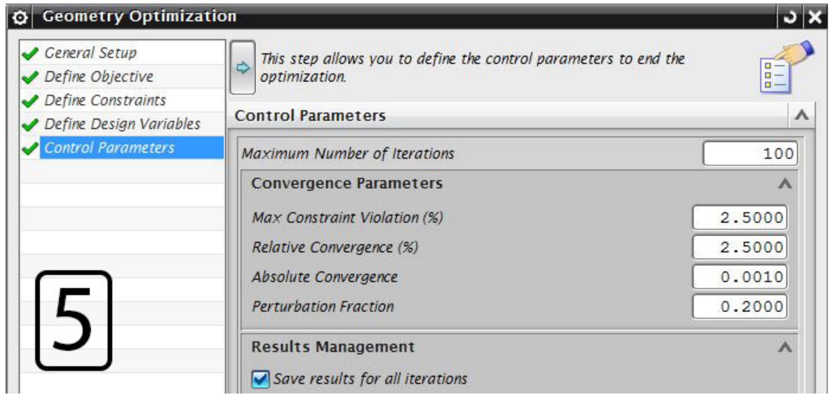
Topic: NX8.5 Optimization Brief | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: March 2013

4. Here is the tab that lets you define which variables to define as your design space. The easiest thing is to use expressions that is associated with the model.

5. This is the last step containing some more options. None of these seem to have an enormous impact on the solution, but might help you to some extent. Number of iterations can be kept high since it seldom seem to exceed 30 iterations. Max constraint violations tells NX how much more deviation from target it can tolerate. Relative and absolute convergence decides when NX is satisfied with the results and terminates the iterations. Perturbation fraction is how big percentage of the predefined design space(proportion between upper and lower limit of the design variables) it is allowed to alter between each iteration.

Right click on the geometry solution icon to solve.





suplight

UGS The PLM Company | modeFRONTIER | NTNU

Appendix B

Optimization Results

Optimization History

Based on Altair HyperOpt

Design Objective Function Results	0	1	2	3	4
Target Weight (12.790000) [N]	14,20773	14,39885	14,25795	13,86296	14,23893

Design Variable Results					
Name	0	1	2	3	4
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_thickness=6.5	6,5	6,7	6,5	6,5	6,5
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Ball_joint_blend=17.6	17,6	17,6	17,8	17,6	17,6
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_offset=0.01	0,01	0,01	0,01	0,608	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Front_bushing_blend=20	20	20	20	20	20,8
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Thickness_main_plate=0.01	0,01	0,01	0,01	0,01	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Hydrobushing_blend=17	17	17	17	17	17
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Swing_arm_flange_thickness=15	15	15	15	15	15

Design Constraint Results					
	0	1	2	3	4
Result Measure					
Upper Limit = 1.705100 [mm]	1,7046	1,6867	1,7012	1,7496	1,7041

Small change in design, run converged.

5	6	7	8	9	10	11	12	13	14	15	16
13,97466	14,25187	14,68163	13,57378	13,85415	13,8513	13,76318	13,82987	13,81446	13,82781	13,82635	13,82759

5	6	7	8	9	10	11	12	13	14	15	16
6,5	6,5	6,5	6	6,075832	6,301152	6,111755	6,061706	6,084425	6,06065	6,098739	6,060029
17,6	17,6	17,6	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1
0,01	0,01	0,01	0,0122	0,01	0,012278	0,010004	0,01	0,01	0,01	0,01	0,01
20	20	20	18	18	18	18	18	18	18	18	18
0,208	0,01	0,01	0,0117	0,013806	0,013923	0,01	0,014157	0,012389	0,014391	0,012656	0,013806
17	17,4	17	16	16	16	16	16	16	16	16	16
15	15	15,8	15,75904	16	15,46538	15,76215	16	15,91463	16	15,89794	16

5	6	7	8	9	10	11	12	13	14	15	16
1,7205	1,7026	1,6421	1,743	1,7012	1,7106	1,7145	1,7043	1,7069	1,7047	1,7053	1,7046

Appendix C

Sensitivity Results

Sensitivity History

Design Objective Results

Weight (Minimum) [N]

Design Variable Results	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_thickness=6.5	6,00	6,04	6,08	6,13	6,17	6,21	6,25
Objective Result	13,67	13,72	13,77	13,82	13,87	13,91	13,96
Weight							
Upper Limit = 1,7051 [mm]	1,7568	1,7518	1,7471	1,7424	1,7378	1,7335	1,729
	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Ball_joint_blend=17.6	17,10	17,14	17,18	17,23	17,27	17,31	17,35
Objective Result	14,04	14,05	14,07	14,09	14,11	14,12	14,13
Weight							
Upper Limit = 1,7051 [mm]	1,7167	1,7153	1,7142	1,713	1,7119	1,7106	1,7096
	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_offset=0.01	0,01	0,13	0,26	0,38	0,51	0,63	0,76
Objective Result	14,21	14,14	14,06	13,99	13,92	13,85	13,78
Weight							
Upper Limit = 1,7051 [mm]	1,70	1,7135	1,7226	1,732	1,7416	1,7515	1,7615
	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Front_bushing_blend=20	18,00	18,17	18,33	18,50	18,67	18,83	19,00
Objective Result	14,12	14,13	14,13	14,14	14,15	14,16	14,16
Weight							
Upper Limit = 1,7051 [mm]	1,7067	1,7066	1,7065	1,7061	1,706	1,7058	1,7057
	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Thickness_main_plate=0.01	0,01	0,05	0,09	0,13	0,18	0,22	0,26
Objective Result	14,21	14,16	14,11	14,06	14,01	13,96	13,92
Weight							
Upper Limit = 1,7051 [mm]	1,70	1,7076	1,7111	1,7143	1,7176	1,721	1,7244
	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Hydrobushing_blend=17	16,00	16,08	16,17	16,25	16,33	16,42	16,50
Objective Result	14,09	14,10	14,11	14,12	14,13	14,14	14,15
Weight							
Upper Limit = 1,7051 [mm]	1,7104	1,7098	1,7094	1,7088	1,7084	1,7079	1,7073
	Step #1	Step #2	Step #3	Step #4	Step #5	Step #6	Step #7
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Swing_arm_flange_thickness=15	12,00	12,17	12,33	12,50	12,67	12,83	13,00
Objective Result	12,50	12,60	12,69	12,78	12,87	12,97	13,06
Weight							
Upper Limit = 1,7051 [mm]	1,9981	1,9788	1,9598	1,9415	1,9234	1,9056	1,8881

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
6,29	6,33	6,38	6,42	6,46	6,50	6,54	6,58	6,63	6,67	6,71	6,75	6,79	6,83
14,00	14,04	14,08	14,13	14,17	14,21	14,25	14,29	14,33	14,37	14,41	14,45	14,48	14,52
1,7247	1,7207	1,7167	1,7126	1,7084	1,70	1,70	1,70	1,69	1,69	1,69	1,68	1,68	1,68

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
17,39	17,43	17,48	17,52	17,56	17,60	17,64	17,68	17,73	17,77	17,81	17,85	17,89	17,93
14,15	14,16	14,17	14,19	14,20	14,21	14,22	14,23	14,24	14,25	14,26	14,27	14,28	14,29
1,7088	1,7078	1,707	1,7062	1,7052	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
0,88	1,01	1,13	1,26	1,38	1,51	1,63	1,75	1,88	2,00	2,13	2,25	2,38	2,50
13,70	13,63	13,56	13,49	13,42	13,34	13,27	13,20	13,13	13,05	12,98	12,91	12,84	12,76
1,7719	1,7824	1,7932	1,8043	1,8155	1,8272	1,8391	1,8513	1,8638	1,8766	1,8899	1,9033	1,9174	1,9315

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
19,17	19,33	19,50	19,67	19,83	20,00	20,17	20,33	20,50	20,67	20,83	21,00	21,17	21,33
14,17	14,18	14,19	14,19	14,20	14,21	14,21	14,22	14,23	14,23	14,24	14,25	14,25	14,26
1,7055	1,7053	1,7052	1,71	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
0,30	0,34	0,38	0,42	0,46	0,51	0,55	0,59	0,63	0,67	0,71	0,75	0,79	0,84
13,87	13,82	13,77	13,72	13,67	13,63	13,58	13,53	13,48	13,43	13,38	13,34	13,29	13,24
1,7278	1,7315	1,7348	1,7386	1,7422	1,7458	1,7497	1,7535	1,7573	1,7614	1,7654	1,7696	1,7739	1,778

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
16,58	16,67	16,75	16,83	16,92	17,00	17,08	17,17	17,25	17,33	17,42	17,50	17,58	17,67
14,16	14,17	14,18	14,19	14,20	14,21	14,22	14,23	14,24	14,24	14,25	14,26	14,27	14,28
1,7069	1,7064	1,706	1,7055	1,71	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70	1,70

Step #8	Step #9	Step #10	Step #11	Step #12	Step #13	Step #14	Step #15	Step #16	Step #17	Step #18	Step #19	Step #20	Step #21
13,17	13,33	13,50	13,67	13,83	14,00	14,17	14,33	14,50	14,67	14,83	15,00	15,17	15,33
13,15	13,25	13,34	13,44	13,53	13,61	13,72	13,81	13,91	14,01	14,10	14,20	14,30	14,40
1,8712	1,8546	1,8381	1,8221	1,8063	1,7911	1,7757	1,7611	1,7465	1,7322	1,7182	1,70	1,69	1,68

Step #22	Step #23	Step #24	Step #25
6,88	6,92	6,96	7,00
14,56	14,60	14,64	14,67
1,67	1,67	1,66	1,66

Step #22	Step #23	Step #24	Step #25
17,98	18,02	18,06	18,10
14,30	14,31	14,32	14,32
1,70	1,70	1,70	1,70

Step #22	Step #23	Step #24	Step #25
2,63	2,75	2,88	3,00
12,69	12,62	12,55	12,48
1,9462	1,9612	1,9766	1,9925

Step #22	Step #23	Step #24	Step #25
21,50	21,67	21,83	22,00
14,27	14,27	14,28	14,28
1,70	1,70	1,70	1,70

Step #22	Step #23	Step #24	Step #25
0,88	0,92	0,96	1,00
13,19	13,14	13,10	13,05
1,7824	1,7869	1,7914	1,7961

Step #22	Step #23	Step #24	Step #25
17,75	17,83	17,92	18,00
14,29	14,30	14,31	14,32
1,70	1,70	1,70	1,70

Step #22	Step #23	Step #24	Step #25
15,50	15,67	15,83	16,00
14,50	14,60	14,70	14,80
1,66	1,65	1,64	1,63

Appendix D

Stress vs Displacements Results

Optimization History

Based on Altair HyperOpt

Design Objective Function Results

	0	1	2	3	4
Target Weight (12.630000) [N]	14,02617	14,11322	14,25287	13,6761	14,06009

Design Variable Results

Name	0	1	2	3	4
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Ball_joint_blend=17.600000	17,1	17,3	17,1	17,1	17,1
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_thickness=6.500000	6,5	6,5	6,7	6,5	6,5
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_offset=0.010000	0,01	0,01	0,01	0,608	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Front_bushing_blend=20.000000	20	20	20	20	20,8
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Thickness_main_plate=0.01	0,01	0,01	0,01	0,01	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Hydrobushing_blend=17.000000	17	17	17	17	17
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Swing_arm_flange_thickness=15	15	15	15	15	15

Design Constraint Results

	0	1	2	3	4
Result Measure					
Upper Limit = 1.705000 [mm]	1,7168	1,7108	1,6961	1,7623	1,7161

Small change in design, run converged.

weight		
0,987071	1,292928	%
Constraint		
0,991554	0,844595	%

5	6	7	8	9	10	11	12	13	14	15	16	17	18
13,79494	14,07023	14,52854	13,66092	13,9729	13,85915	13,79803	13,8359	13,84053	13,79977	13,83783	13,81659	13,83769	13,84666

5	6	7	8	9	10	11	12	13	14	15	16	17	18
17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1	17,1
6,5	6,5	6,5	6	6,150957	6,079332	6,042366	6,066222	6,068292	6,043717	6,067157	6,053409	6,06616	6,071852
0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,0119	0,01	0,01	0,01	0,01	0,01	0,01
20	20	20	18	18	18	18	18	18	18	18	18	18	18
0,208	0,01	0,01	0,0117	0,013395	0,013923	0,01404	0,014157	0,014274	0,014391	0,013689	0,013806	0,013923	0,01404
17	17,4	17	16	16	16	16	16	16	16	16	16	16	16
15	15	15,8	15,8929	16	16	16	16	16	16	16	16	16	16

5	6	7	8	9	10	11	12	13	14	15	16	17	18
1,7331	1,7144	1,6528	1,7301	1,6877	1,7003	1,7085	1,7035	1,7028	1,7083	1,7031	1,7061	1,7032	1,702

19
13,84482
19
17,1
6,070685
0,01
18
0,014157
16
16
19
1,7023

Optimization History

Based on Altair HyperOpt

Design Objective Function Results

Minimum Weight [N]	0	1	2	3	4
	14,20059	14,25089	14,39484	13,85575	14,2332

Design Variable Results

Name	0	1	2	3	4
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Ball_joint_blend=17.600000	17,6	17,8	17,6	17,6	17,6
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_thickness=6.500000	6,5	6,5	6,7	6,5	6,5
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_offset=0.010000	0,01	0,01	0,01	0,608	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Front_bushing_blend=20.000000	20	20	20	20	20,8
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Thickness_main_plate=0.01	0,01	0,01	0,01	0,01	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Hydrobushing_blend=17.000000	17	17	17	17	17
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Swing_arm_flange_thickness=15	15	15	15	15	15

Design Constraint Results

	0	1	2	3	4
Result Measure					
Upper Limit = 1.705000 [mm]	1,7046	1,7011	1,6864	1,7494	1,7038

Small change in design, run converged.

Weight		
0,973117	2,688345	%
Constraint		
1,00176	-0,17599	%

5	6	7	8	9	10
13,9705	14,24472	14,68159	13,69784	13,94918	13,81883

5	6	7	8	9	10
17,6	17,6	17,6	17,1	17,1	17,1
6,5	6,5	6,5	6	6,134102	6,141145
0,01	0,01	0,01	0,01	0,01	0,010006
20	20	20	18	18	18
0,208	0,01	0,01	0,0117	0,011713	0,01177
17	17,4	17	16	16	16
15	15	15,8	15,95024	16	15,77989

5	6	7	8	9	10
1,7202	1,7024	1,642	1,7246	1,6902	1,7076

Optimization History

Based on Altair HyperOpt

Design Objective Function Results

Target Weight (12.690000) [N]	0	1	2	3	4
	14,10541	14,18727	14,32798	13,75555	14,07568

Design Variable Results

Name	0	1	2	3	4
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Ball_joint_blend=17.600000	17,1	17,3	17,1	17,1	17,1
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_thickness=6.500000	6,5	6,5	6,7	6,5	6,5
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_offset=0.010000	0,01	0,01	0,01	0,608	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Front_bushing_blend=20.000000	22	22	22	22	21,2
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Thickness_main_plate=0.01	0,01	0,01	0,01	0,01	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Hydrobushing_blend=17.000000	17	17	17	17	17
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Swing_arm_flange_thickness=15	15	15	15	15	15

Design Constraint Results

	0	1	2	3	4
Result Measure					
Upper Limit = 66.000000 [N/mm^2(MPa)]	65,691	65,859	67,207	67,588	65,498

Maximum number of iterations reached, no convergence.

Weight		
0,978671	2,132938	%
Constraint		
1,052808	-5,28078	%

5	6	7	8	9	10	11	12	13	14	15	16	17	18
13,86965	14,15233	14,60863	12,62395	12,77739	12,70033	12,68363	12,67068	12,63615	12,53647	12,46546	12,54044	12,4113	12,68329

5	6	7	8	9	10	11	12	13	14	15	16	17	18
17,1	17,1	17,1	17,1	17,50932	17,40834	18,1	18,1	18,07151	17,10001	17,10001	17,11855	17,10085	17,92609
6,5	6,5	6,5	6	6,185563	6,328841	6,127284	6,555893	6,247688	6,12825	6,13053	6,191427	6,059029	6,118945
0,01	0,01	0,01	0,01	0,612015	0,613568	0,616131	0,627509	0,623825	0,603455	0,609044	0,560652	0,606884	0,603464
22	22	22	21,66168	20,81555	20,05437	20,34823	20,58558	18,63569	21,84901	21,83067	21,87655	21,81729	21,73863
0,208	0,01	0,01	0,01	0,010002	0,01002	0,010027	0,010011	0,010045	0,01	0,01	0,01	0,010258	0,01
17	17,4	17	16,17448	17,89261	18	18	18	18	16,88738	16,74464	16,63444	16,75994	16,92932
15	15	15,8	13,80836	13,4809	13,16253	12,96672	12,21779	12,82024	13,83616	13,73995	13,70815	13,76421	13,23014

5	6	7	8	9	10	11	12	13	14	15	16	17	18
69,383	66,205	65,146	131,96	72,425	73,308	75,34	77,352	75,54	103,78	107,63	95,079	129,61	74,986

19	20	21	22	23	24	25	26	27	28	29	30	31	32
12,67937	12,70583	12,68465	12,71088	12,70711	12,70258	12,69773	12,69047	12,73309	12,69547	12,85283	13,02328	13,30776	13,23185

19	20	21	22	23	24	25	26	27	28	29	30	31	32
17,73904	17,97328	17,81338	17,92461	17,95456	18,06391	17,75224	17,37817	17,66355	18,02037	17,89911	17,97232	17,49484	18,1
6,423113	6,287258	6,260754	6,157506	6	6,137161	6,302657	6,51539	6,250658	6,455907	6,331954	6,269482	6,503702	6,074801
0,597916	0,601458	0,605179	0,574187	0,585688	0,585759	0,589541	0,600745	0,517518	0,657872	0,681231	0,532418	0,643228	0,614493
21,00602	21,70537	21,03985	20,28279	21,04945	21,54189	19,46244	19,39716	22	18,00002	20,20982	22	22	19,6026
0,01	0,01	0,010001	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01
17,52026	16,21378	16,26932	17,48798	18	18	17,69995	17,98022	18	17,72653	17,98662	17,84739	17,8782	17,82502
12,6987	13,10059	13,24134	13,15667	13,27731	12,89966	13,00398	12,82356	12,94008	12,7549	13,12162	13,28389	13,77484	14,22689

19	20	21	22	23	24	25	26	27	28	29	30	31	32
75,213	91,957	91,203	73,979	74,194	75,621	75,178	74,096	74,303	75,031	73,853	72,634	69,691	70,294

33	34	35	36	37	38	39	40	41	42	43	44	45	46
13,4208	13,46497	13,54309	13,52769	13,53779	13,53752	13,66549	13,67992	13,72074	13,41034	13,59151	13,74857	13,7564	13,77174

33	34	35	36	37	38	39	40	41	42	43	44	45	46
17,79372	17,1	17,60681	17,1	17,58916	18,1	17,26218	17,36078	17,26716	18,1	18,1	17,28539	17,29778	17,30588
6,210654	6,809523	6,304796	6,74149	6,385194	6,05139	6,530509	6,539539	6,544348	6	6,08518	6,521282	6,557492	6,5282
0,626151	0,616029	0,618671	0,623622	0,590485	0,645375	0,618478	0,599353	0,619222	0,67009	0,64981	0,623014	0,606945	0,628476
20,94629	20,42756	22	22	20,35862	21,92813	22	21,8231	22	18,21656	20,50678	22	21,69428	21,99472
0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01	0,01
17,82947	17,82999	17,73155	17,71871	17,693	17,72128	17,7541	17,6784	17,7599	17,66906	17,71607	17,75111	17,69565	17,73835
14,417	13,84236	14,50544	14,00773	14,4511	14,72279	14,52925	14,48181	14,60119	14,92826	14,87447	14,68059	14,62693	14,70243

33	34	35	36	37	38	39	40	41	42	43	44	45	46
69,711	69,024	68,812	68,792	69,186	68,568	67,581	68,452	67,435	68,686	69,055	66,843	68,059	66,681

47	48	49
13,77763	13,79221	13,80455

47	48	49
17,29305	17,22567	17,32214
6,562111	6,567304	6,536552
0,608306	0,621681	0,609094
21,62292	22	21,6768
0,01	0,01	0,01
17,69381	17,74178	17,69258
14,67146	14,71538	14,7376

47	48	49
67,628	67,425	69,16

Optimization History

Based on Altair HyperOpt

Design Objective Function Results

	0	1	2	3	4
Minimum Weight [N]	14,20067	14,25127	14,39183	13,85574	14,2332

Design Variable Results

Name	0	1	2	3	4
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Ball_joint_blend=17.600000	17,6	17,8	17,6	17,6	17,6
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_thickness=6.500000	6,5	6,5	6,7	6,5	6,5
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Flange_offset=0.010000	0,01	0,01	0,01	0,608	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Front_bushing_blend=20.000000	20	20	20	20	20,8
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Thickness_main_plate=0.01	0,01	0,01	0,01	0,01	0,01
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Hydrobushing_blend=17.000000	17	17	17	17	17
"FLCA_EN_Parameterized_control_arm_ver04_rev02"::Swing_arm_flange_thickness=15	15	15	15	15	15

Design Constraint Results

Result Measure	0	1	2	3	4
Upper Limit = 66.000000 [N/mm^2(MPa)]	67,208	67,28	67,193	67,546	66,93

Maximum number of iterations reached, no convergence.

Weight		
0,863016	13,6984	%
Constraint		
1,17245	-17,245	%

5	6	7	8	9	10	11	12	13	14	15	16	17	18
13,9676	14,24177	14,68158	13,92081	13,88116	11,37895	11,1845	11,35709	11,62653	11,43804	12,1131	11,53447	11,73429	11,36904

5	6	7	8	9	10	11	12	13	14	15	16	17	18
17,6	17,6	17,6	17,1	17,70658	17,1	17,21682	17,1	17,1	17,1	18,1	17,1	17,74063	17,28632
6,5	6,5	6,5	6	6,499302	6,203051	6	6	6,130571	6	6	6	6	6,164748
0,01	0,01	0,01	0,010003	0,607995	0,611548	0,71744	0,49248	0,601083	0,509404	0,47424	0,46816	0,50464	0,681211
20	20	20	22	20,00305	18	18	19,673	19,91881	18	22	18	18	18
0,208	0,01	0,01	0,01	0,01	0,0119	0,012	0,0121	0,01	0,01	0,01	0,0118	0,0119	0,01
17	17,4	17	16	17,00017	16	16	16	16	16	16,82959	16,77636	16	16
15	15	15,8	16	15,00108	12	12	12	12,45	12,3	12,15	12,27649	12,27777	12

5	6	7	8	9	10	11	12	13	14	15	16	17	18
71,434	67,166	66,66	91,775	68,336	146,85	182,2	137,71	145,87	128,44	108,96	137,57	122,11	175,82

19	20	21	22	23	24	25	26	27	28	29	30	31	32
11,95602	11,98015	11,52109	11,3765	11,30465	11,31303	11,27213	11,41311	11,5964	11,75113	11,67845	11,66567	11,3452	12,48979

19	20	21	22	23	24	25	26	27	28	29	30	31	32
17,1	17,11533	17,1	17,1	17,32083	17,45295	17,1	17,1	17,1	17,70141	17,32905	17,10948	17,1	18,1
6,58667	6	6	6	6	6	6	6,090076	6	6	6,085254	6,285326	6	6,577554
0,49248	0,494051	0,617429	0,74176	0,678298	0,586329	0,71744	0,602883	0,6255	0,73568	0,604461	0,591646	0,590301	0,613377
18	18	21,6487	21,91627	20,14575	18,14645	20,46326	21,06157	18	22	18,48633	18	19,5644	18,09487
0,0121	0,01187	0,01	0,0117	0,0118	0,010234	0,010286	0,010532	0,010356	0,01	0,01	0,01	0,0119	0,012
16	16	16	16	16	16	16	16	17,19542	16	17,99982	17,66243	16,37325	18
12,09159	13,39277	12,3	12,15	12	12	12	12	12,45	12,3	12,15	12	12	12

19	20	21	22	23	24	25	26	27	28	29	30	31	32
96,688	137,61	138,65	178,86	183,22	175,19	179,02	173,32	133,04	175,02	105,25	119,04	139,79	78,439

33	34	35	36	37	38	39	40	41	42	43	44	45	46
13,48204	12,07694	11,58566	12,34912	14,49585	11,29631	11,84813	11,66886	12,20759	12,09071	12,28401	12,473	11,97368	12,24946

33	34	35	36	37	38	39	40	41	42	43	44	45	46
18,1	17,20619	17,1	17,1	17,74316	17,1	17,44901	17,1	17,1	18,03475	17,1	17,1	17,93642	18,1
7	6	6,275636	7	6,576869	6	6,252331	6,403699	6,525979	6	6,501055	6,591076	6	6
0,611959	0,609671	0,636281	0,74176	0,623411	0,71136	0,604244	0,662981	0,596303	0,618954	0,583026	0,603739	0,50464	0,53634
21,91828	20,96284	18	18	22	18	18	18	18	18	18	18	18,7404	19,55974
0,01	0,010248	0,010839	0,010585	0,0118	0,01	0,010251	0,0121	0,01049	0,01	0,01	0,01	0,01	0,010646
18	17,24005	16	16	16	16	16	16,01014	16,31171	18	16	16,72528	18	18
12,97865	13,12878	12,3	12,15	16	12,23904	12,3498	12,21042	12,6801	12,3	12,93002	12,91049	12	12,34206

33	34	35	36	37	38	39	40	41	42	43	44	45	46
71,241	127,32	117,35	103,5	68,875	175,23	138,44	131,49	84,821	79,612	91,451	74,428	93,484	78,931

47	48	49
12,59189	12,70655	12,2554

47	48	49
17,1	18,09747	18,09623
6,650606	6,349688	6,134271
0,723357	0,606144	0,599388
18	19,81163	18
0,01	0,01	0,01
18	16,24112	17,99965
12,84049	13,05269	12,3

47	48	49
74,024	87,977	78,798

Appendix E

A3 Briefing modeFRONTIER Configuration

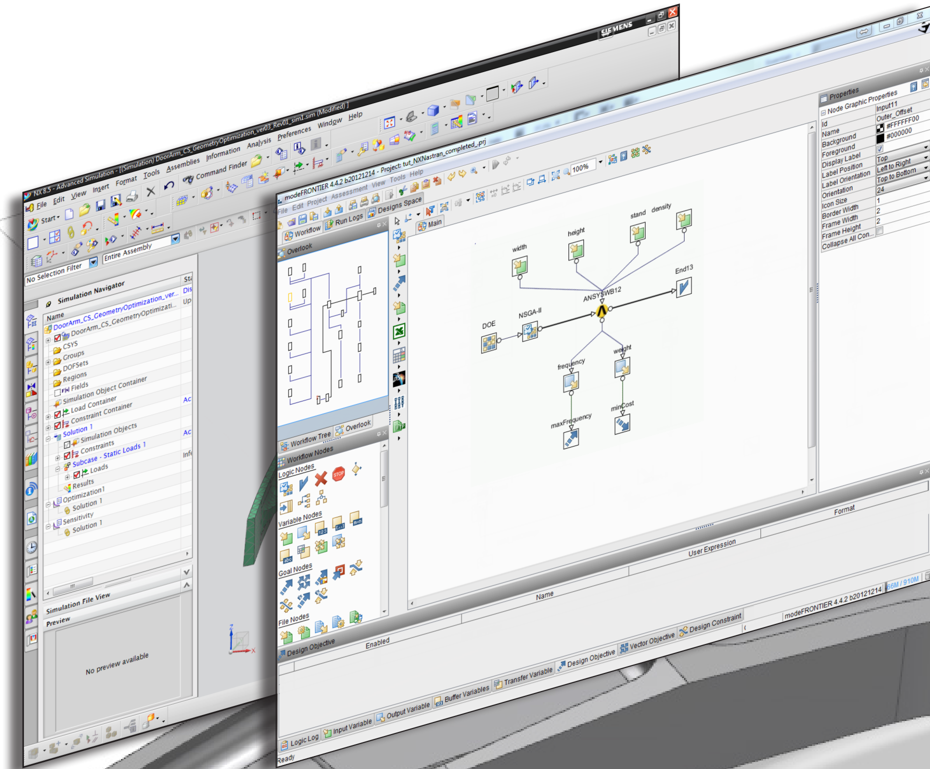
modeFRONTIER Configuration

Topic: Optimization modeFRONTIER	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 10.04.2013

modeFRONTIER is a multiobjective optimization software which allows you to connect several different CAD or FEA software together. Through the graphical interface you are able to set up a workflow consisting of nodes (the icons) and links (lines between the nodes)

We will here demonstrate how it is possible to build a workflow which can interact with simulations performed with NX Advanced Simulations.

The objectives of this simulation will be mass and displacement.

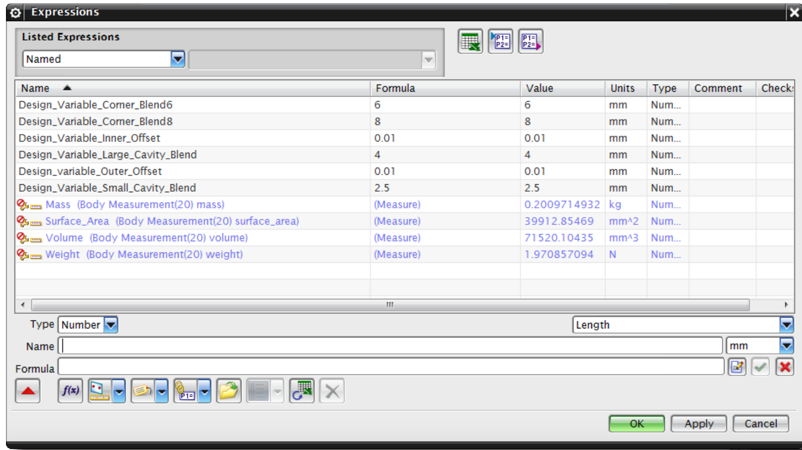


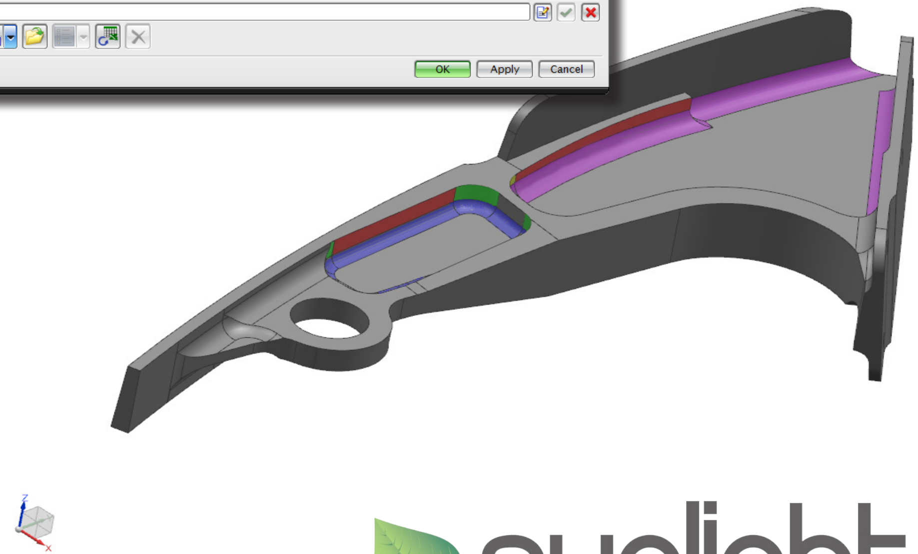
modeFRONTIER Configuration

Topic: Necessary Preperations | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

The following preparations are necessary in order to run the optimization.

1. Parameterize the model (Expressions) in NX
2. Use Measure Bodies (Analysis --> Measure Bodies) to apply measurements like volume, mass, etc to the expressions list
3. Run a simulation i NX Advanced Simulation. Make sure you have a .fem- and .sim-file which you store in the same folder as the .prt-file
4. Install modeFRONTIER. It is recommended to install it in a directory path without spaces.
5. Download Cygwin (cygwin.com). Install it in c:\cygwin\ if possible
6. In addition you need the file killer.vbs stored in the same folder as the rest of the modeling and simulation files
7. You will also need to record a macro with meshing and simulation. Store the macro in the same folder as the rest of the files. Watch the instruction video (NX_macro.mp4)





suplight

modeFRONTIER Configuration

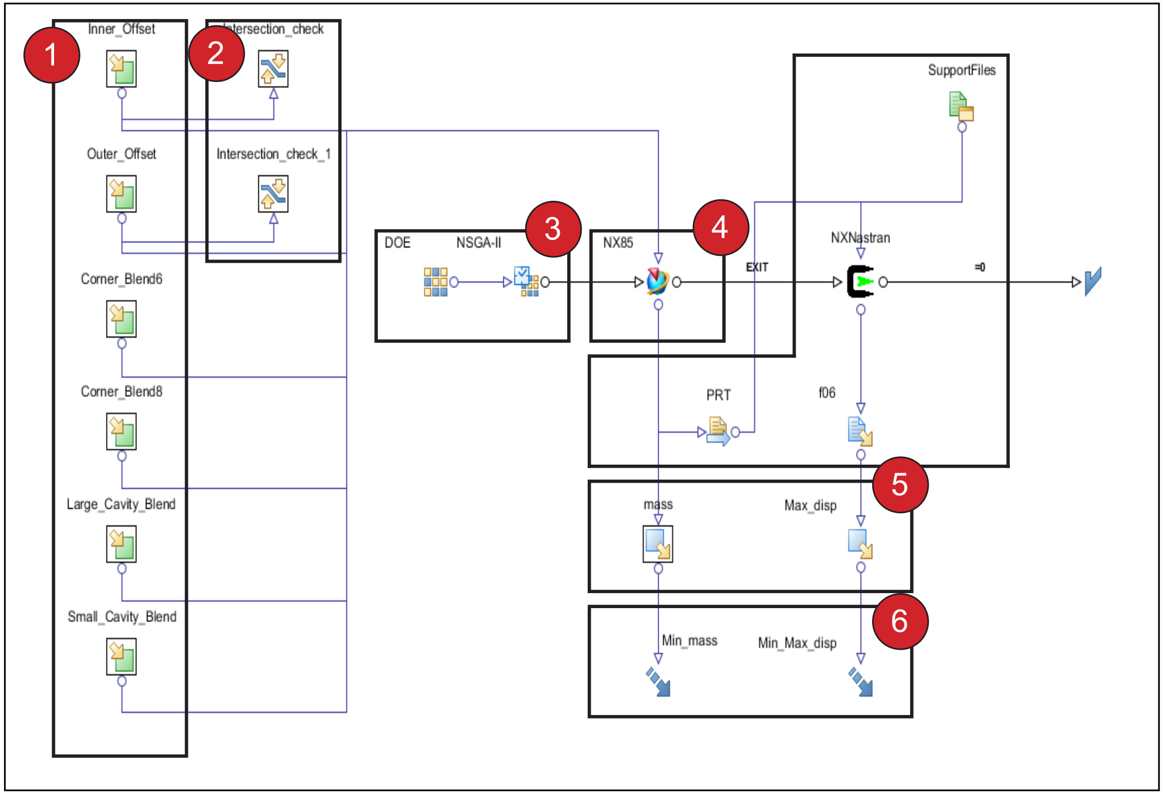
Topic: Workflow in modeFRONTIER | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

The workflow shows how modeFRONTIER can control NX and NX Advanced Simulation to optimize a prt-file

The following nodes are necessary to run a optimization run:

1. Input Variables: defines design space
2. Constraints: constraints on variables.
3. DOE and Scheduler: DOE and algorithms provides different values for the input variables
4. NX CAD Node: Interacts with NX expressions
5. Output Variables: design output variables
6. Objective: minimizing or maximizing output variables

The last bulk consists of four nodes that are neccessary to derive data outputs from NX Advanced Simulations.



UGS The PLM Company | mode FRONTIER | suplight | NTNU

modeFRONTIER Configuration

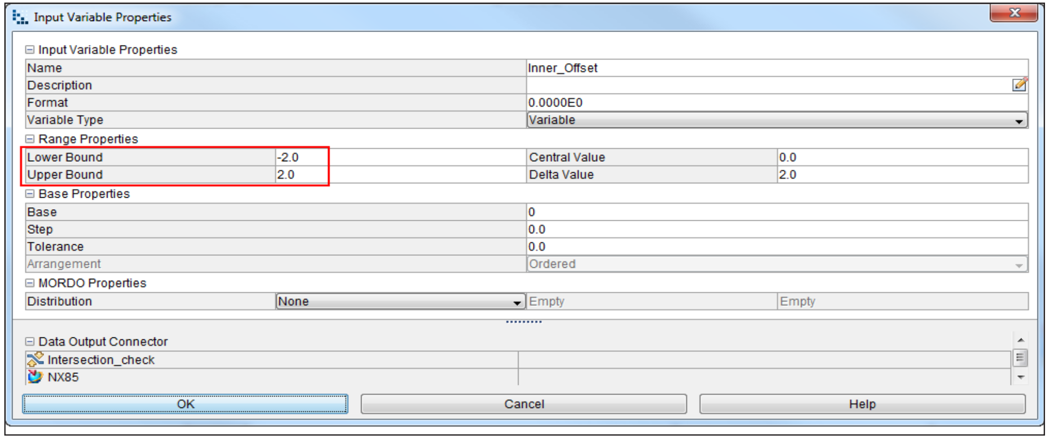
Topic: Define Input Variables | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

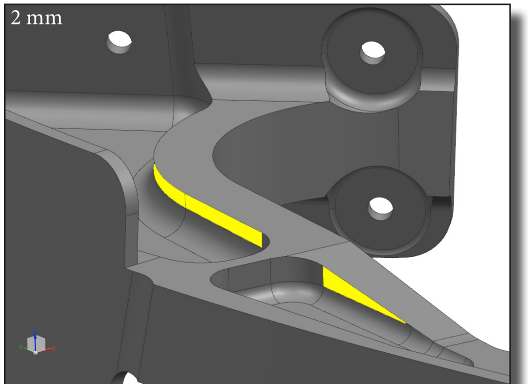
Input variables: Defines the design space by the upper and lower bound

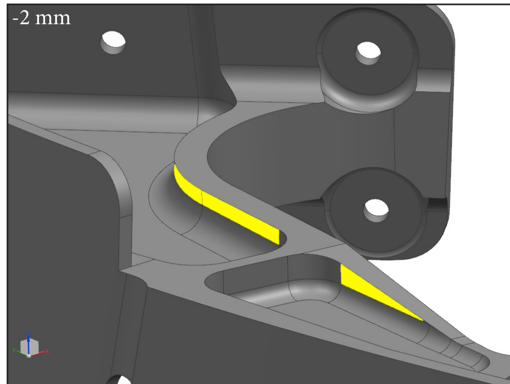
The value range (design space) needs to be defined in order to run a successful optimization run.

1. Open the Input Variable Properties by double clicking the input node icon.
2. Under Range Properties you are able to specify the lower and upper bound for the design parameter. This tells the scheduler what range it should keep within while changing the design parameter.

The screen shots underneath shows how the CAD model responds when the parameter inner_offset is changed.







modeFRONTIER Configuration

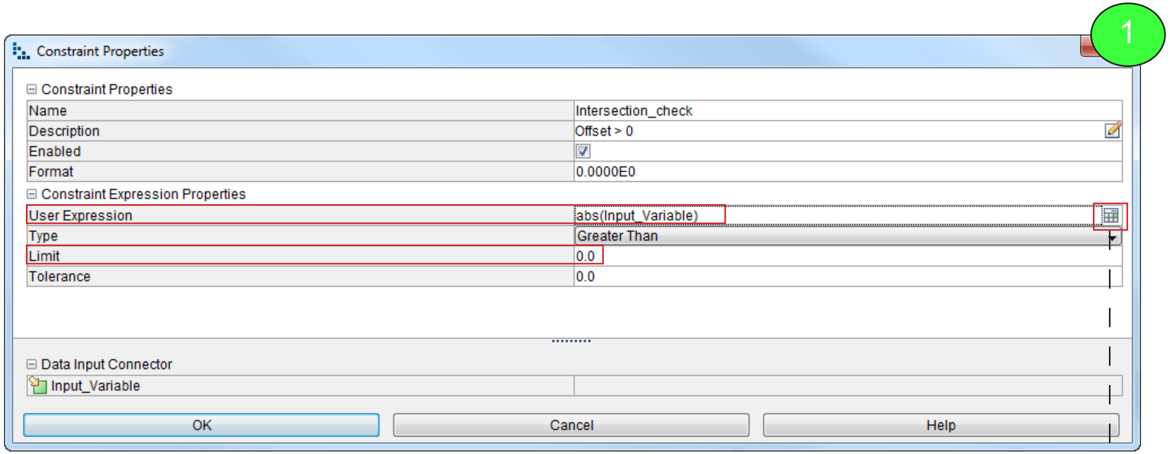
Topic: Define Input Constraint | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

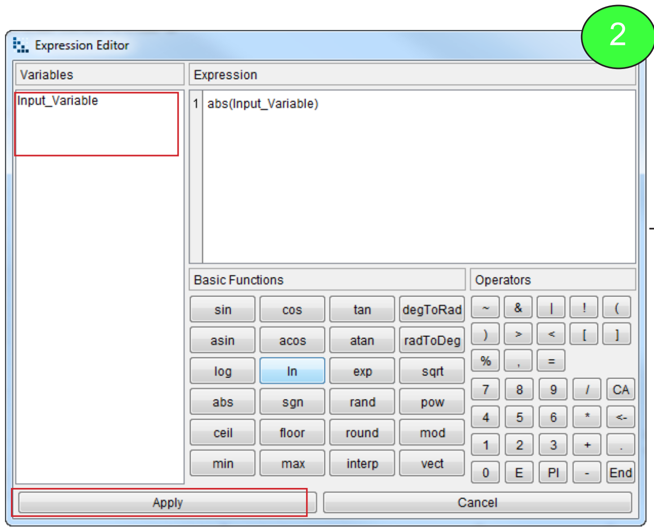
Design Constraint: Additional contraints added to the output or input variable

It is also possible to define additional constraints to the Input or output variable. If one of the design parameters for instance can´t be equal to zero, a constraint node can be linked to the input variable node.

1: What is needed to be defined here is the "User Expression" and "Limit". To edit the user expression, click the calculator icon behind the "User Expression".

2: In this case the expression is written the absolute value of the input variable. The variables connected to the constraint node will occur in a list to the left. Click "Apply" to save the changes. You will then return to the Constraint Properties. Set the limit equal to zero. Save the changes and close the window by clicking "Ok".





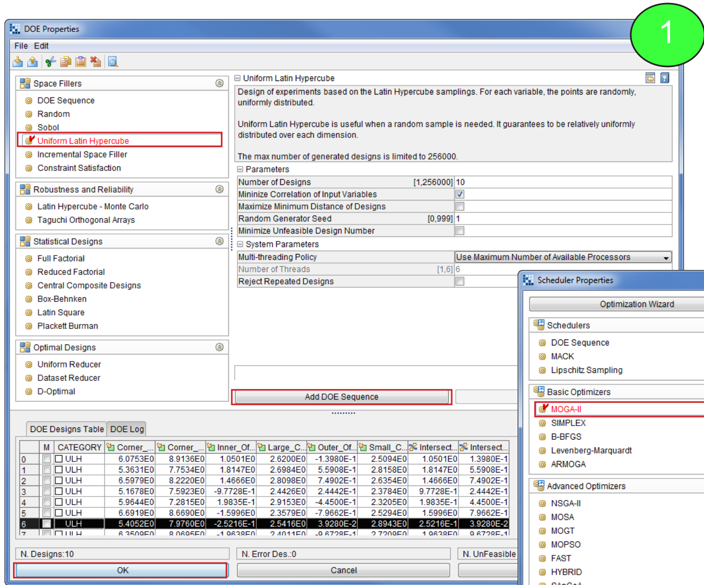
modeFRONTIER Configuration

Topic: DOE and Schedulers	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 10.04.2013

Design of experiments (DOE): Necessary sample of the design space which the scheduler algorithm can base its optimization algorithms on.

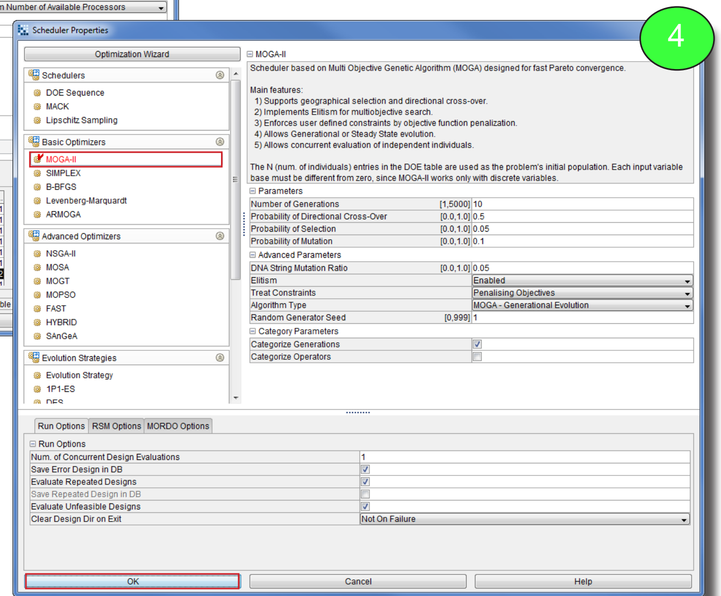
From the DOE Properites you can choose what kind of DOE setting you will like to use. In this tutorial we will use the Uniform Latin Hypercube.

1: Remove the DOE design table by holding shift and mark all the lines. Press "Delete
2: Use the same proprties as specified and click "Add DOE Sequence"
3: Click "Ok" to save and close the window.



Scheduler: Contains different types of algorithms that work in various ways to reach optimization goals based on the problem at hand.

The scheduler uses the initial DOE to build a population of new designs. The way it controls the evolution varies from algorithm type selected.
As a example we will use the scheduler called MOGA-II.



4: Select "MOGA-II" from the list to the left
5: Click "Ok" to save and close the window.

Look in the help section for more details regarding DOE and Schedulers

modeFRONTIER Configuration

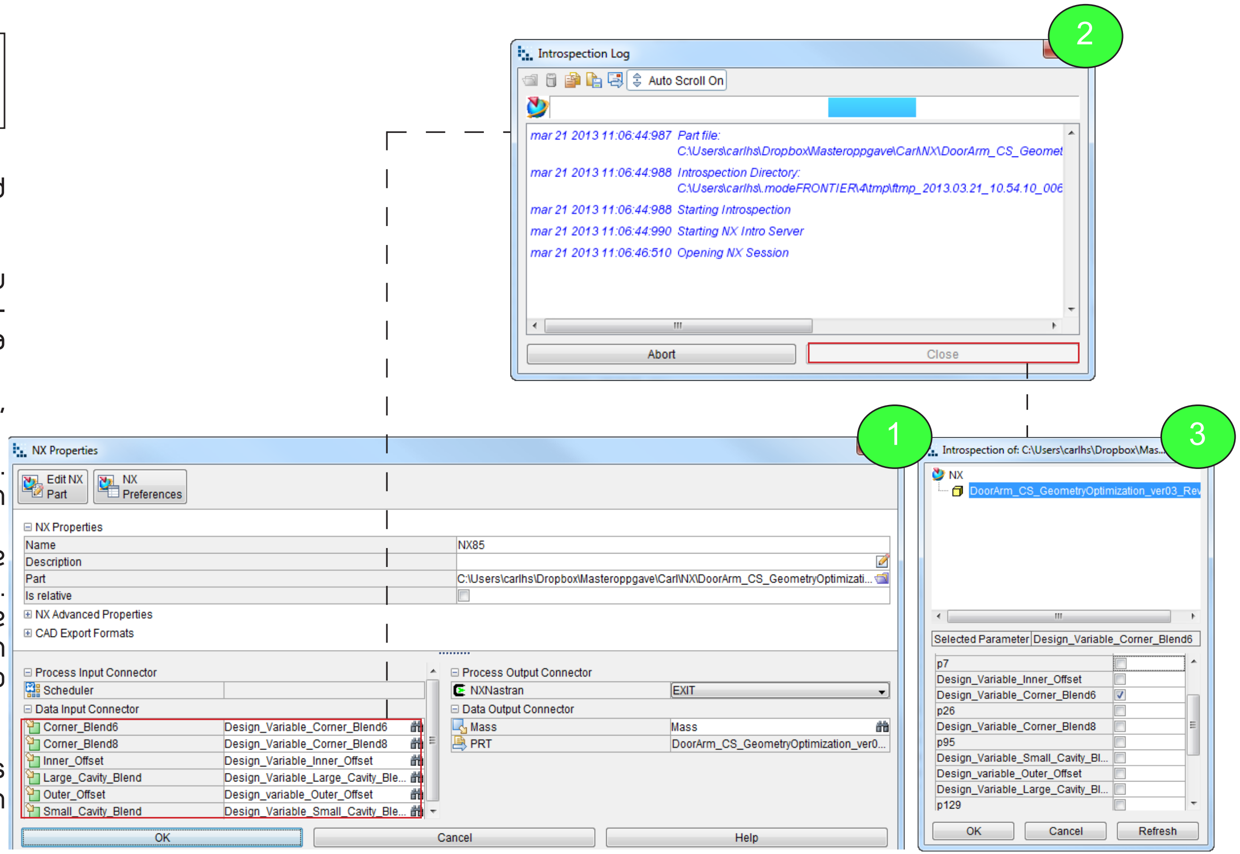
Topic: Assign Input/Output Variables from part-file. | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

NX CAD Node: interact with the user expressions in a NX .prt-file.

The NX CAD node can only interact with inputs and outputs involving the geometry.

1: Define all the input and output variables you would like to use by clicking on each of the binoculars behind the input variables listed under "Data Input Connector".
2: Wait for the introspection log to finish loading, then click close.
3: A new window will open. Click on the part name. A list will be shown underneath with all the design parameters from the expression list in NX.
4: A list will be shown underneath with all the design parameters from the expression list in NX. Tick the design parameter which corresponds to the input node you want to assign. You have to click on each binocular to assign one design parameter to each of the input nodes.
5: The same way you assign the output variables. You will need to apply all the input and output nodes you need in the workflow to be able to assign each and every one of them.

The data output connector called PRT is for the simulations results and has no binoculars.



suplight

modeFRONTIER Configuration

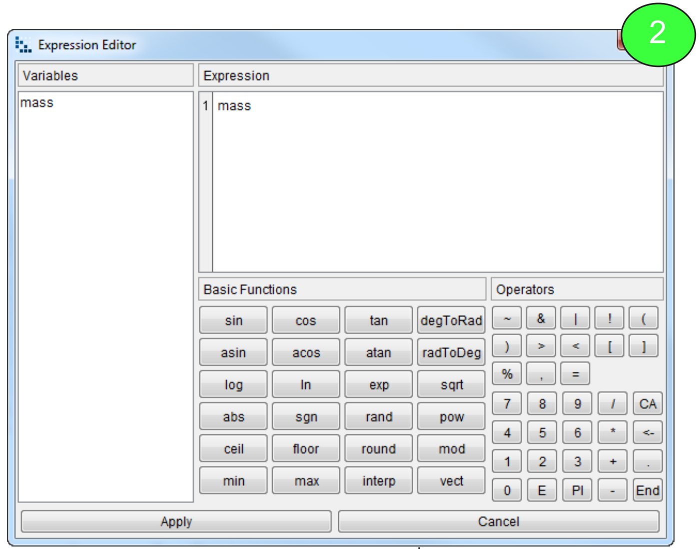
Topic: Define Objective	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 10.04.2013

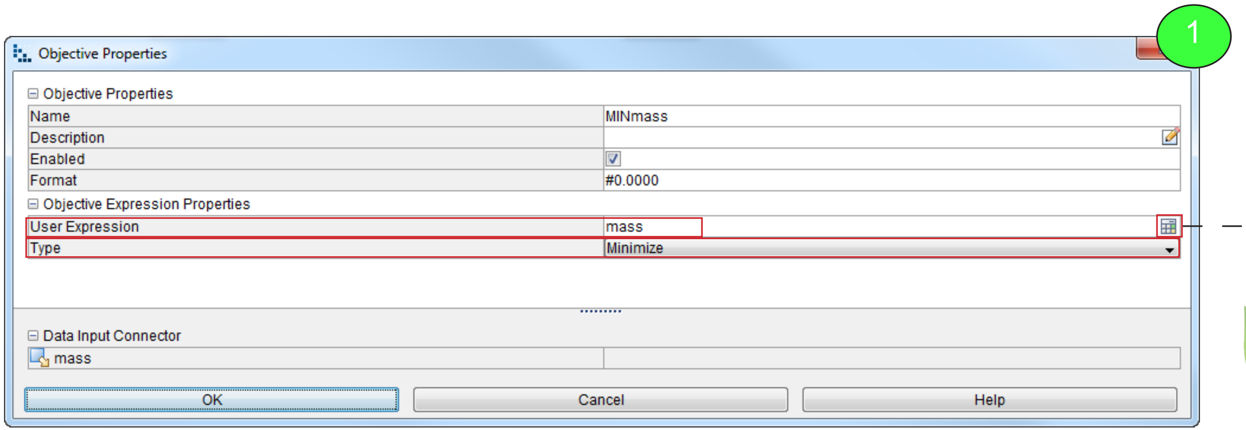
Deisgn Objective: Identifies the output node and represents the optimization objective

The objective node is connected to output node and allows you to either minimize or maximize the output.

1: Double-click the objective node. The objective node is linked to a output node called "mass". Click on the calculator.
2: In the Expression Editor define the expression output value through the calculator icon.

Under "Type" you could either choose minimize or maximize depending on the objective for the variable chosen.

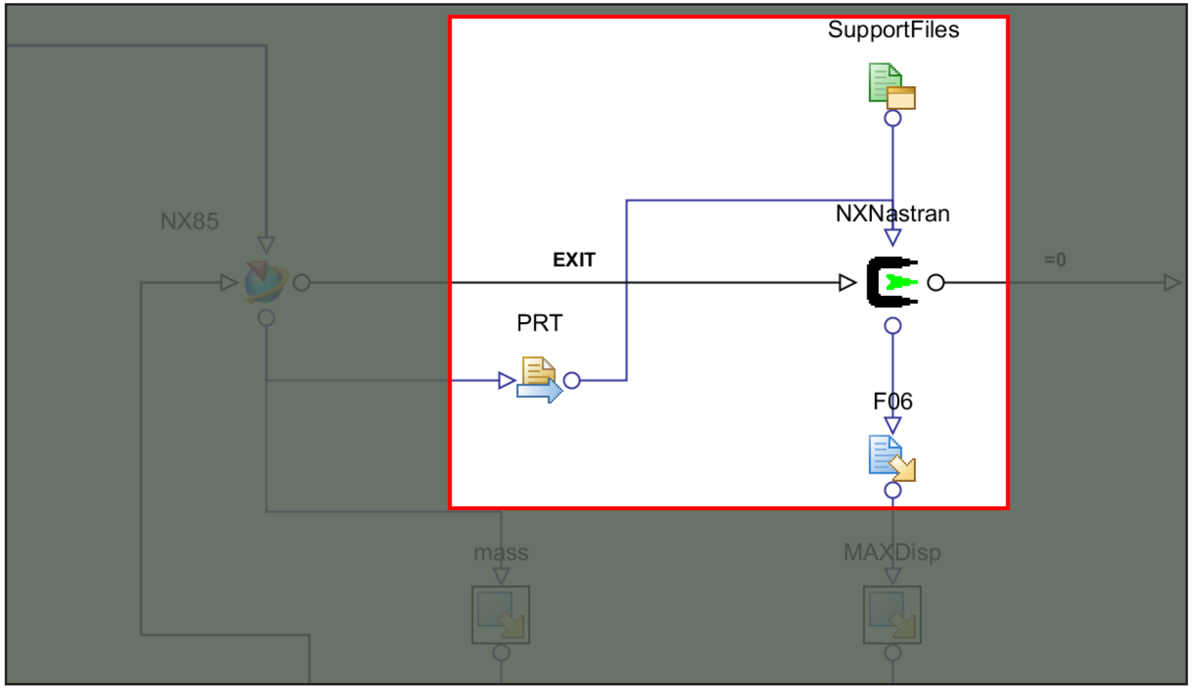




modeFRONTIER Configuration

Topic: Retrieving Simulations Outputs (displacement)	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 10.04.2013

The NX geometry node is only able to deal with the expressions defined in the .prt-file . Unfortunatly, there is no standard simulation node which can interact with NX Advanced Simulations. The way of retrieving simulation results into modeFRONTIER is to use the Cygwin node. The Cygwin node allows you to run a Script which will run the macro recorded in NX.



Transfer File: Transfers file(s) from one application node to another.

SupportFiles: Shows the absolute path for all the files included in the script

Cygwin Shell Script: Executes a script which will load NX and run a macro

Output File: Uses a mining rule which is able to read results out of the .F06 file.

In the next slides we will show you how each of the nodes needs to be defined, changes needed to be done in the script, and how to record a macro in NX 8.5.

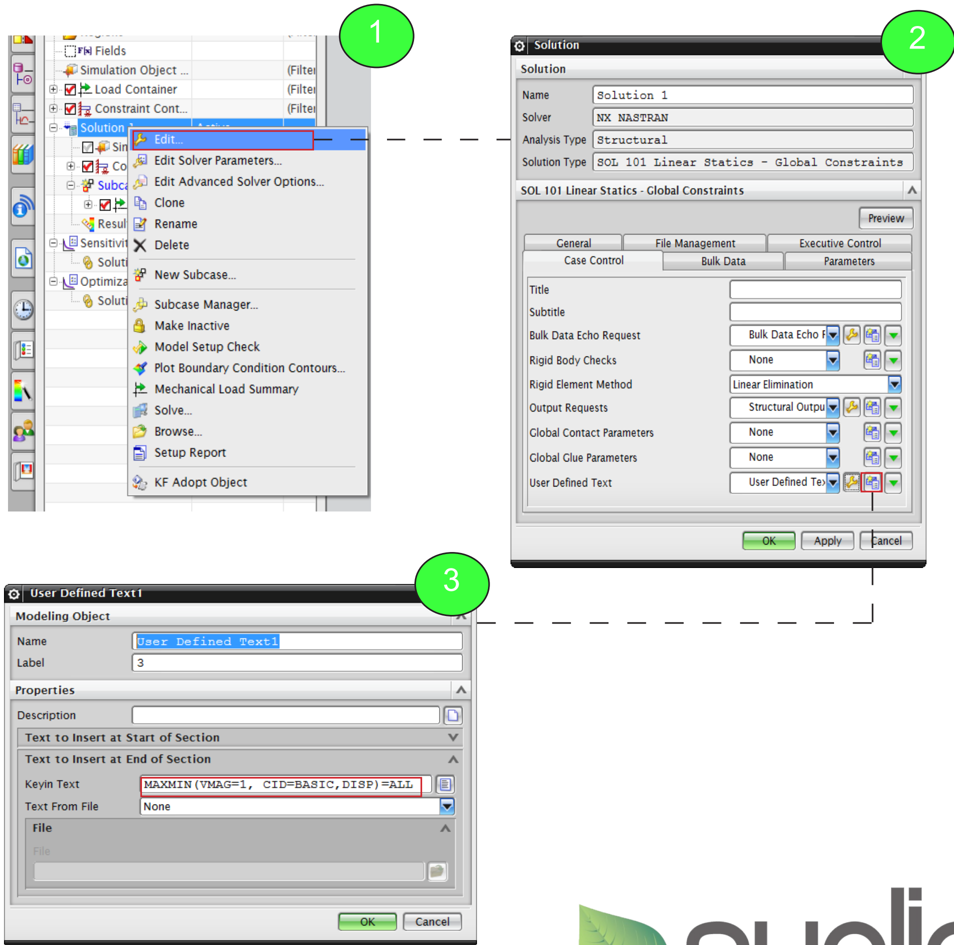
modeFRONTIER Configuration

Topic: Write Results to the .F06-File | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

In order to extract the maximum displacement magnitude, this setting needs to be configured in NX

1: Open the .sim-file and right-click on the solution.
2: Click on the icon "Creating Modeling Object"
3: In the "Keyin Text" input box, copy-paste:

MAXMIN(VMAG=1, CID=BASIC,DISP)=ALL

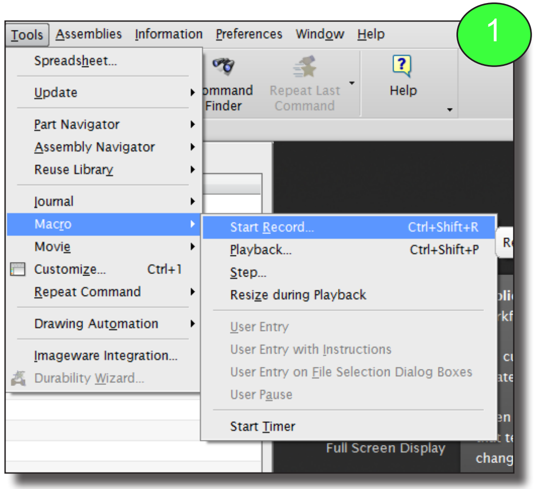


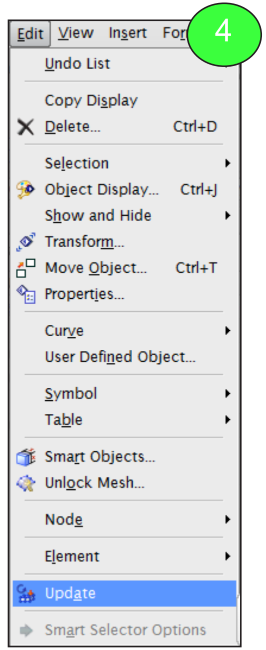
modeFRONTIER Configuration

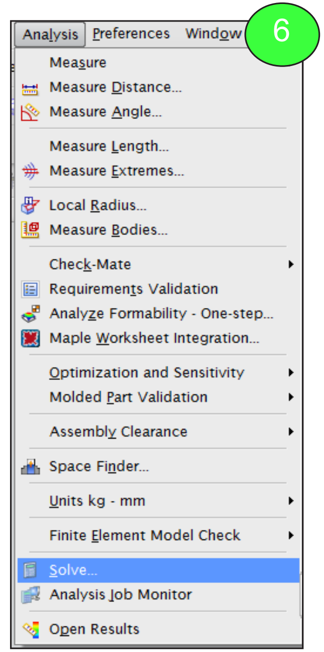
Topic: Record the macro	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 10.04.2013

A macro needs to be recorded in NX in order to run the simulations. It is important to do this exactly as described here, in order to make this work.
Before recording the macro, make sure you have a .prt, .fem and .sim file. Save the part file without updating the finite element model so the update finite element model button is clickable.

1: Start recording the macro from Tools --> Macro --> Start Recording. Name the macro "NX_Macro"
2: Open the part-file (File --> Open)
3: Open the .fem file (File --> Open)
4: Update the finite element model (Edit --> Update)
5: Open the .sim file (File --> Open)
6: Run the analysis (Analysis --> Solve)
7: Let the simulation finish and close NX without saving (File --> Exit)







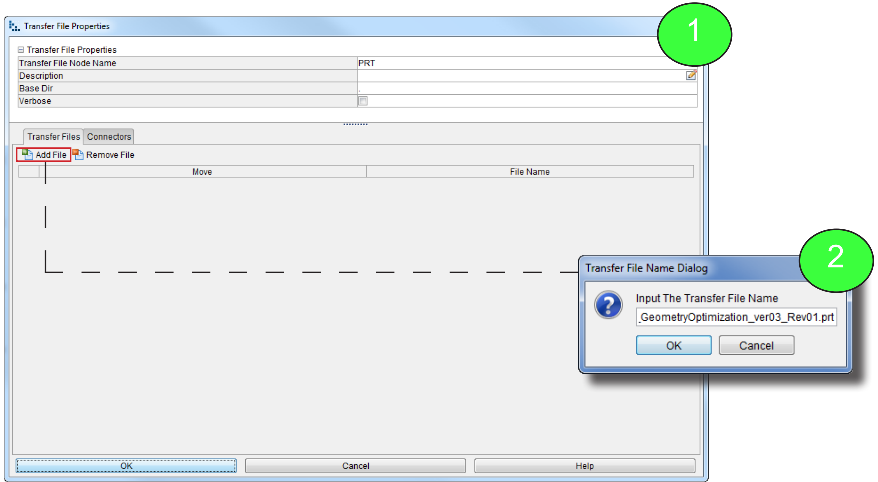
modeFRONTIER Configuration

Topic: Transfer File and Support Files | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

Transfer File: Transfers file(s) from one application node to another.

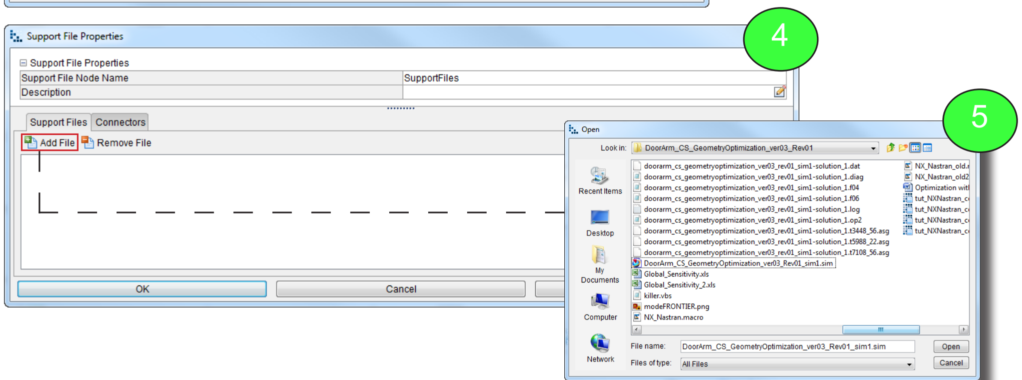
In order to get displacement as output on each of the design iterations the transfer node is needed. The node makes a temporary copy of each of the generated designs and couples it with the cygwin node.

1: Click on Add File
2: Write the .prt-file name. Since you want the temporary copy of the file, insert: filename_copy.prt.
3: If there is other files in the list, remove them by clicking Remove File.



SupportFiles: Shows the absolute path for all the files included in the script

4: Click on Add File
5: Choose the five files: .prt, .fem, .sim, killer.vbs and NX_Nastran.macro.
6: If there is other files in the list, remove them by clicking Remove File.



.

UGS The PLM Company | modeFRONTIER | NTNU

modeFRONTIER Configuration

Topic: Cygwin Shell Script | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

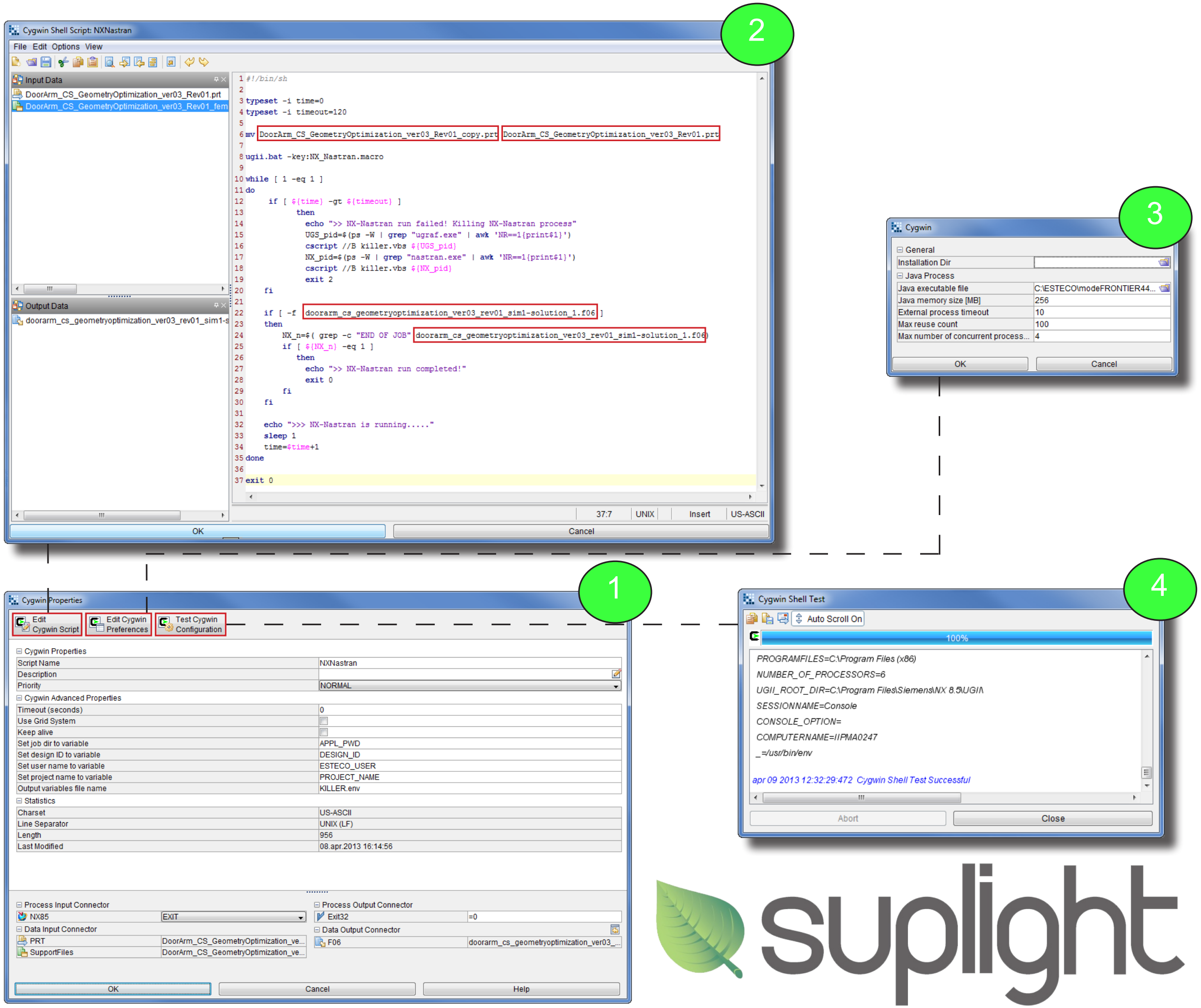
Cygwin Shell Script: Executes a script which will load NX and run a macro

1: There are no changes needed to be done in the main window. Go ahead and click on "Edit Cygwin Script"

2: All the text marked within a red box on the picture underneath needs to be updated with the correct filenames. This involves the .prt-file, the copy of the .prt file and the .F06 resultfile. These can be picked from the list to the left.

3: If the NX does not open during the run, define the java executable file. Do this by specifying the path for java.exe. Normally this is stored in modeFRONTIER installation folder: (c:\ESTECO\modeFRONTIER442\jre\bin\java.exe)

4: Check if the script is able to load.



modeFRONTIER Configuration

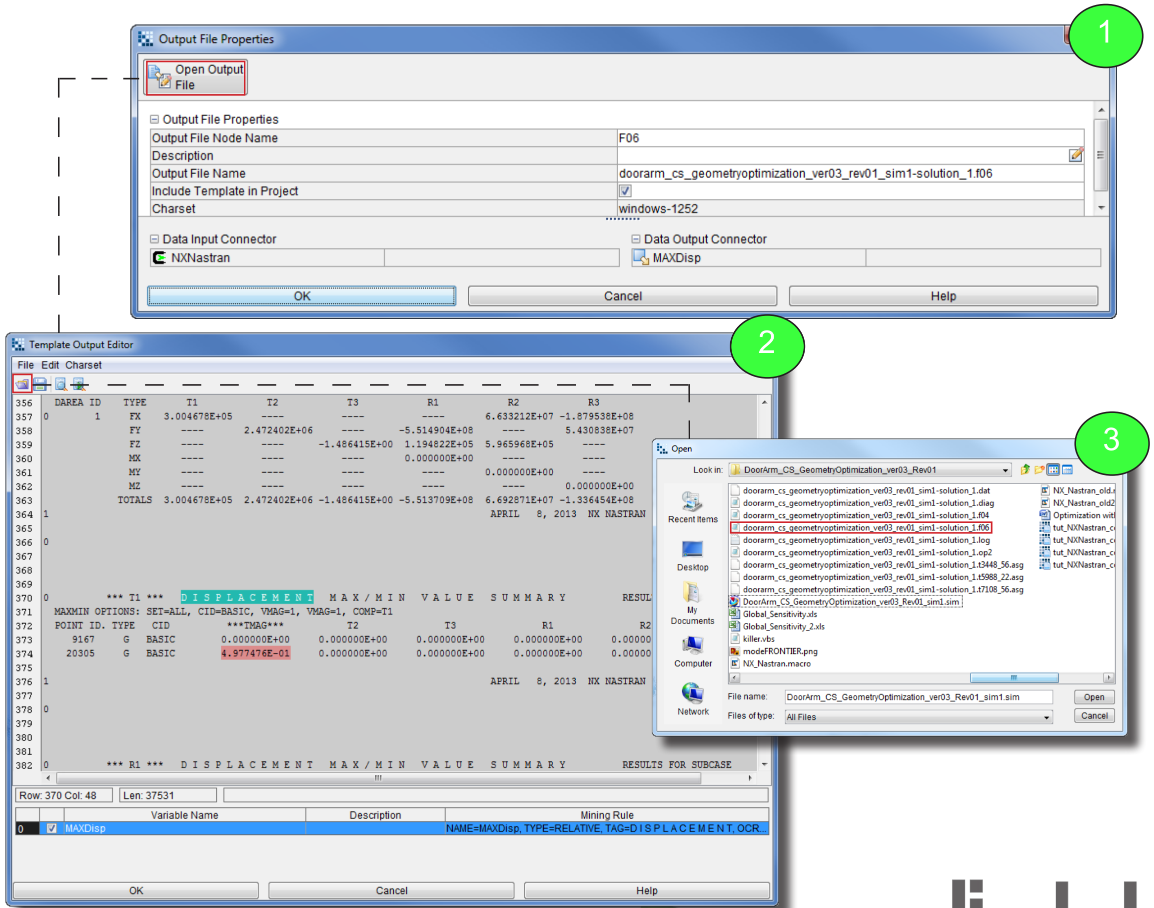
Topic: Retrieving Simulations Outputs (displacement) | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 10.04.2013

Output File: Uses a mining rule which is able to read out results from the .F06 file and assign it to the output variable

1: Click on "Open Output File"
2: Click open to update the .F06 file
3: Browse to the .F06 file and double-click it.

There is already a mining rule defined. If you need to add a new mining rule for displacement:

5: Mark the text "Displacement"
6: Right-click and choose "Relative Position"
7: Mark the value you will like obtain from the .F06-file, left
8: Right-click and choose "Select Relative".
Displacement should become green, and the value red.



Finally, run the analysis by clicking project --> run

Appendix F

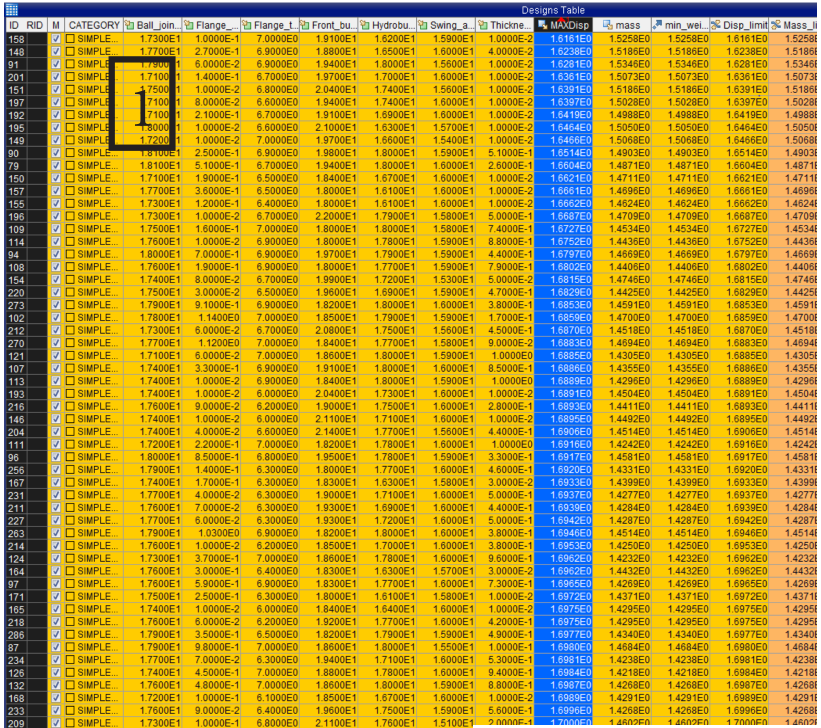
modeFRONTIER Postprocessing

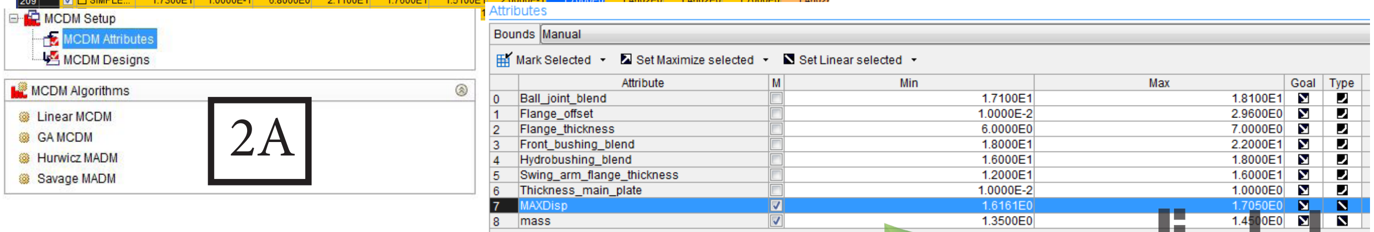
modeFRONTIER Postprocessing

Topic: Postprocessing	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 08.05.2013

This A3 intends to provide a guide to decision making once an optimization has been done. This briefing shows an approach with the use of modeFRONTIERs built in MCDM(-Multi Criteria Decision Making).

1. When an optimization has been run in modeFRONTIER, it presents all the iterations with its design parameters in a design table.
From this table one can range the smallest to largest and mark the designs that falls within an acceptable range regarding one of the two goals. This is done by highlighting the designs that might be good enough. Right click in the highlighted field and click: mark designs->mark selected. The chosen designs should now be ticked off. In this particular case displacement was chosen as the limiting factor for estimating the best design.
2. A. The next step is: Assessment->Open MCDM panel. Here the variables and goals are displayed in a list in MCDM attributes. Check the boxes regarding goal parameters and proceed. Desired range of the parameters can also be specified manually in the attributes.





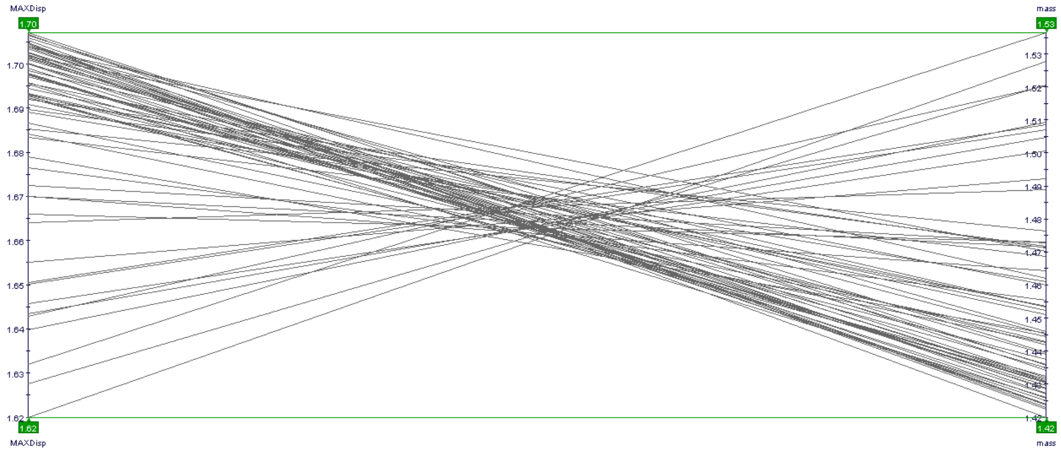
modeFRONTIER Postprocessing

Topic: Postprocessing | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 08.05.2013

2. B. Choose the tab called MCDM designs in the left top corner. Here the designs is presented in both a table and a parallel chart. It is possible to slide the green numbers on the chart to isolate the designs within a given range. This chart is mainly to see which designs that are related to each other.

3. From here one is given the choice between different algorithms and preferences regarding these. The linear MCDM seems to give the best results in this particular example. Click create MCDM.

2B



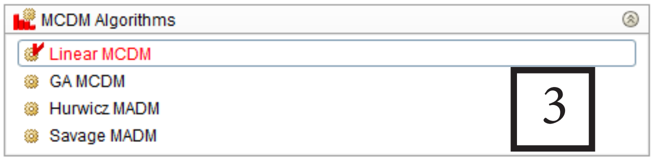
Linear MCDM

Linear Search Algorithm for MCDM. It helps the research of a reasonable solution among a set of available ones. Main features are :

1) Respects all the attributes relationships
2) Respects all the designs relationships
3) Generates a ranking list of solutions
4) Is very precise and fast with few attributes
5) Does not allow the use of more than 4 attributes

Parameters

Training Cycles	[0,28]	28
Preference Margin	[0.0,1.0]	1.0
Indifference Margin	[0.0,1.0]	0.01



3

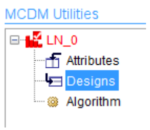
suplight

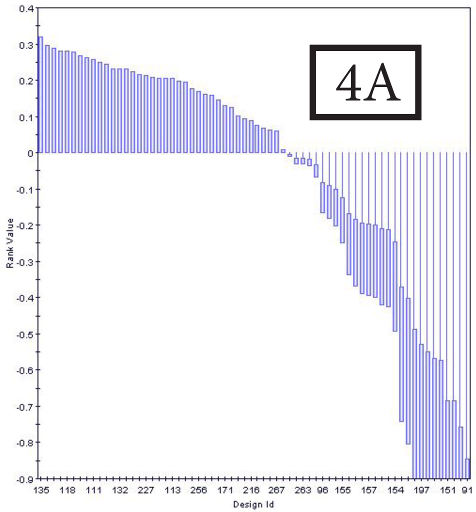
UGS The PLM Company | mode FRONTIER | NTNU

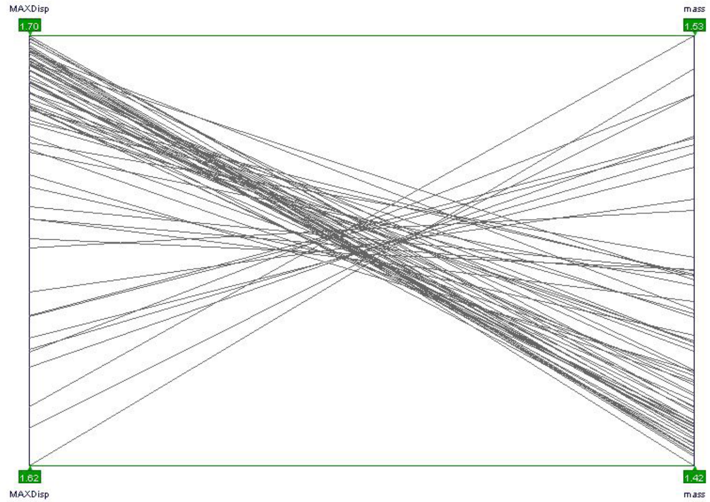
modeFRONTIER Postprocessing

Topic: Postprocessing	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 08.05.2013

4. A. ModeFRONTIER now creates results under MCDM utilities. The attributes tab shows information regarding the setup. The most interesting is the designs tab that contains designs in ranked order.
4. B. This can be exported to the mode-FRONTIER desktop by right clicking MCDM utility(LN_0 in this case) and choose mcdm ranking.
4. C. These tables is now located in design space tab->desktop tab.



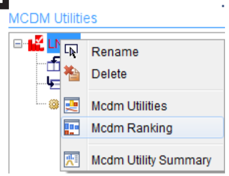




Designs

ID	MAXDisp	mass	Rank Value
135	1.7009E0	1.4180E0	0.320
130	1.7010E0	1.4204E0	0.296
241	1.7015E0	1.4211E0	0.289
126	1.6984E0	1.4218E0	0.282
118	1.7001E0	1.4219E0	0.281
239	1.7003E0	1.4221E0	0.279
124	1.6962E0	1.4232E0	0.268
234	1.6981E0	1.4238E0	0.262
111	1.6916E0	1.4242E0	0.258
214	1.6953E0	1.4250E0	0.250
292	1.7029E0	1.4255E0	0.245
233	1.6996E0	1.4268E0	0.232
132	1.6987E0	1.4268E0	0.232
97	1.6965E0	1.4269E0	0.231
231	1.6937E0	1.4277E0	0.223
211	1.6939E0	1.4284E0	0.216
227	1.6942E0	1.4287E0	0.213
168	1.6989E0	1.4291E0	0.209
218	1.6975E0	1.4295E0	0.205
165	1.6975E0	1.4295E0	0.205
113	1.6889E0	1.4296E0	0.204
290	1.7004E0	1.4301E0	0.199
121	1.6885E0	1.4305E0	0.195
208	1.7033E0	1.4323E0	0.177
256	1.6920E0	1.4331E0	0.169
298	1.7027E0	1.4339E0	0.161
286	1.6977E0	1.4340E0	0.160
107	1.6886E0	1.4355E0	0.145
171	1.6972E0	1.4371E0	0.129

4B

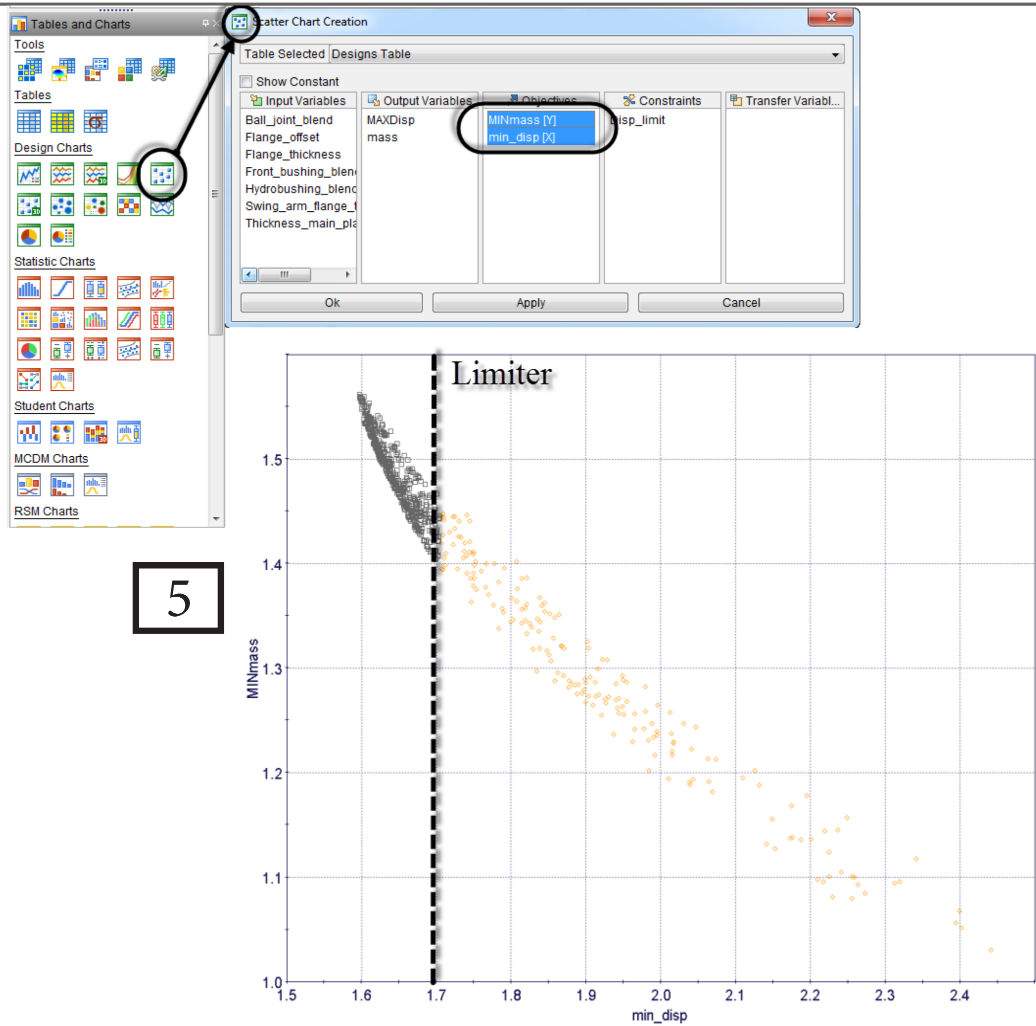


4C

modeFRONTIER Postprocessing

Topic: Postprocessing | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 08.05.2013

5. In the design desktop it is possible to generate a pareto curve by creating a scatter plot of goals. This is done by clicking scatter plot under tables and chart and then select what to display on X and Y axes, as shown to the right. Right clicking the scatter plot gives an option to mark pareto designs(if they exist). Right click: mark designs->mark pareto designs->only real. Here the best designs can be identified visually and marked manually to identify the best results.



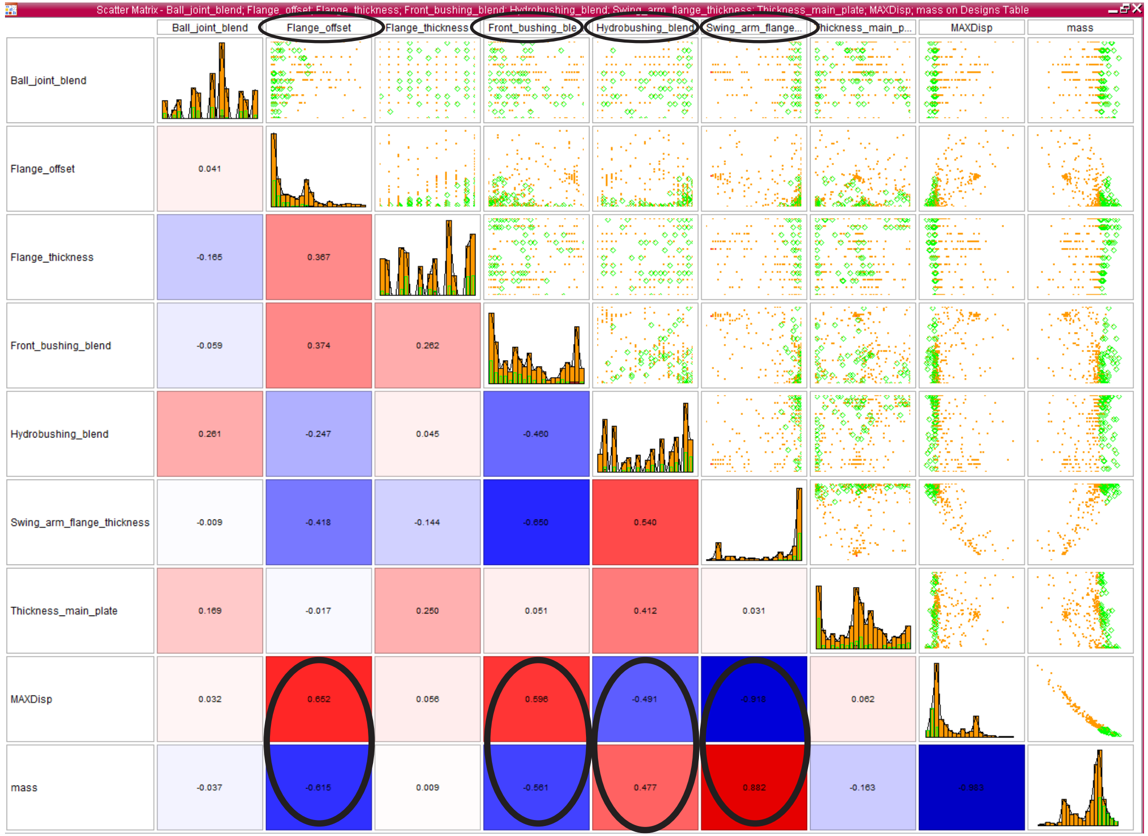
5

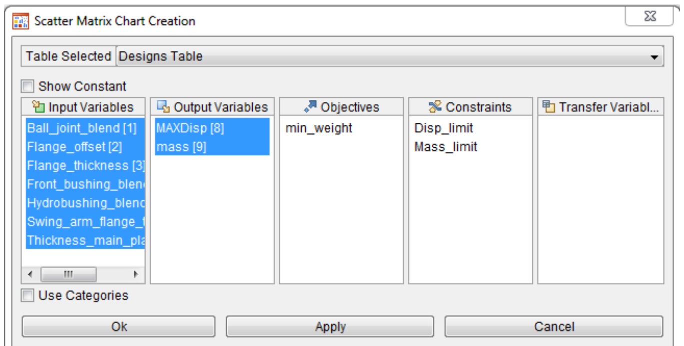
modeFRONTIER Postprocessing

Topic: Sensitivity analysis | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 08.05.2013

6. In case parameter sensitivity is of interest, this can be created by the scatter matrix chart under statistics chart. These tools is located in the left bottom corner of the modeFRONTIER desktop.(remember to be located in:design space tab->desktop tab) Choose all the input and output variables. Click ok and the matrix scatter to the right appears. This shows the correlation between parameters and goals. The four most important parameters is highlighted and shows by a strong color and correlation value, the impact each parameter has on the goal. As one can see the parameters has almost an opposite effect on the conflicting goals as one would expect. This can be used to identify parameters that can be excluded from optimizations to save time. As an example in this case one can see that Flange_offset has a great impact on MAXDisp and the same parameter has almost exactly the opposite effect on mass.

6





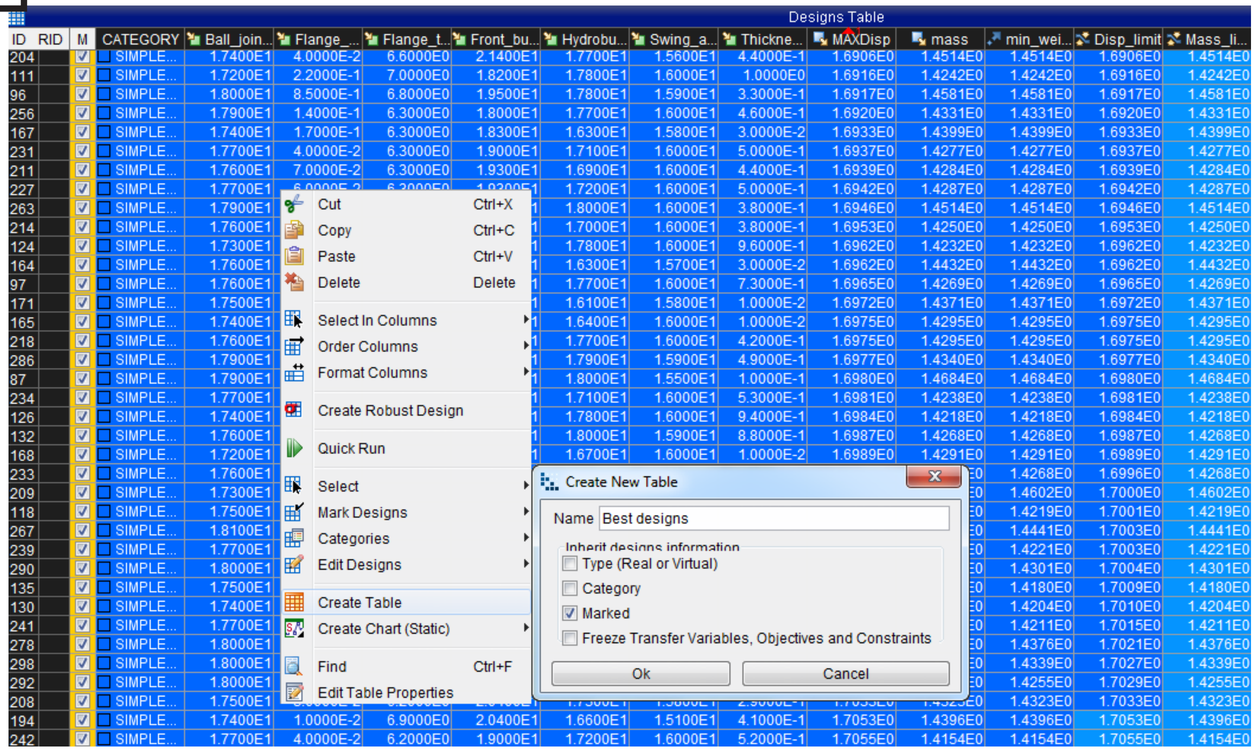
modeFRONTIER Postprocessing

Topic: Manual extraction of designs | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 08.05.2013

In case the MCDM utility does not work satisfactory or modeFRONTIER is unable to mark pareto designs automatically, it is of course possible to do the process manual. One of the ways of doing that is described in detail in the following.

7. First one would have to rank the displacement in descending order from lowest to highest. Then one would have to mark the feasible design that meets the requirements. Right click and choose create table and only keep marked ticked in the next dialog box.

7

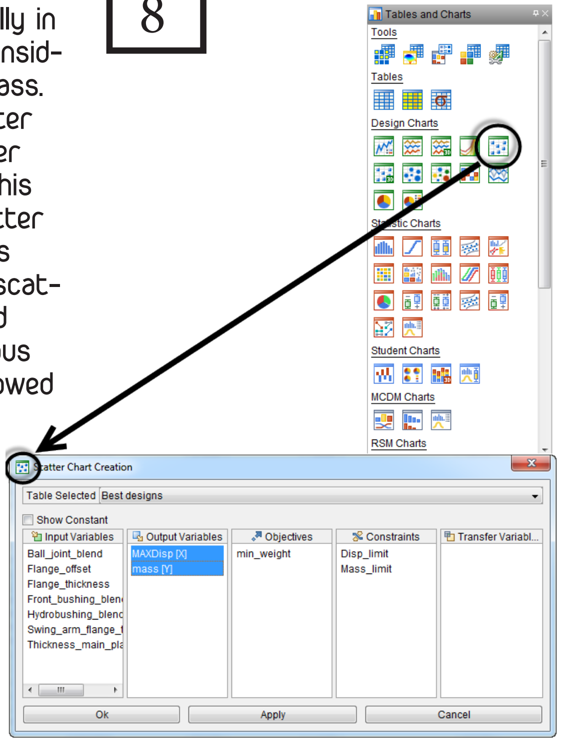


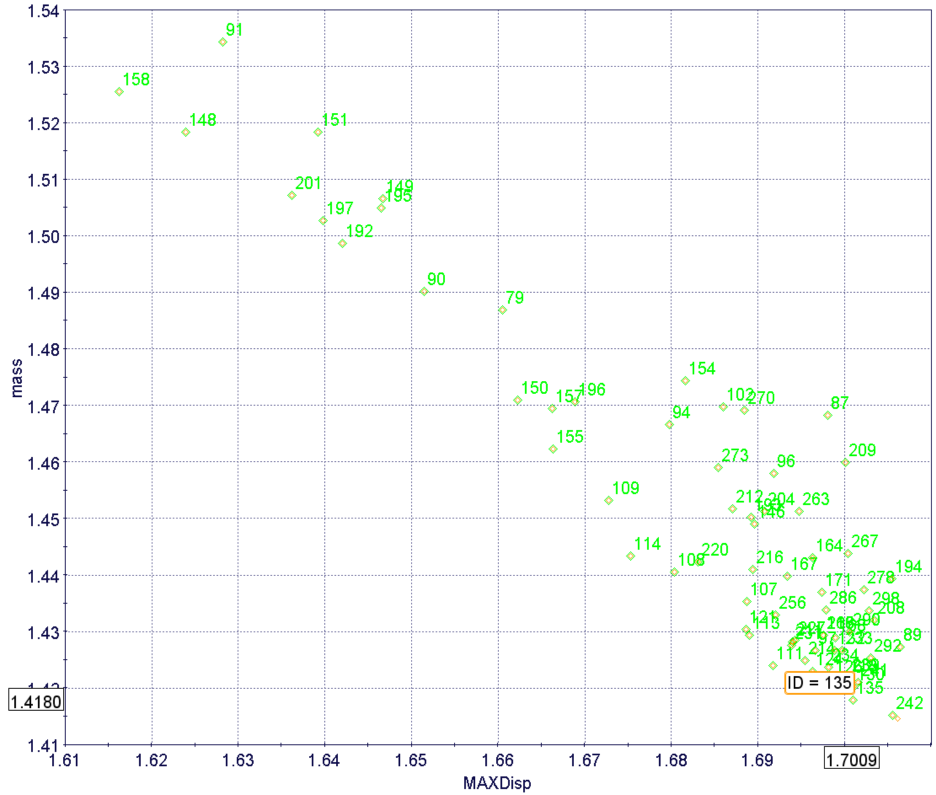
modeFRONTIER Postprocessing

Topic: Manual extraction of designs	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 08.05.2013

8

8. The next step is to identify the best design manually in the generated table by considering displacement and mass. This can be visualized better by creating a X - Y scatter of the selected designs. This is done by creating a scatter plot containing the designs from the new table. This scatter resembles the method already shown on a previous slide, but this one is narrowed down to contain only designs that is within a certain range. Clicking on the desired design shows the value and id. This makes it possible to locate the design in the table.





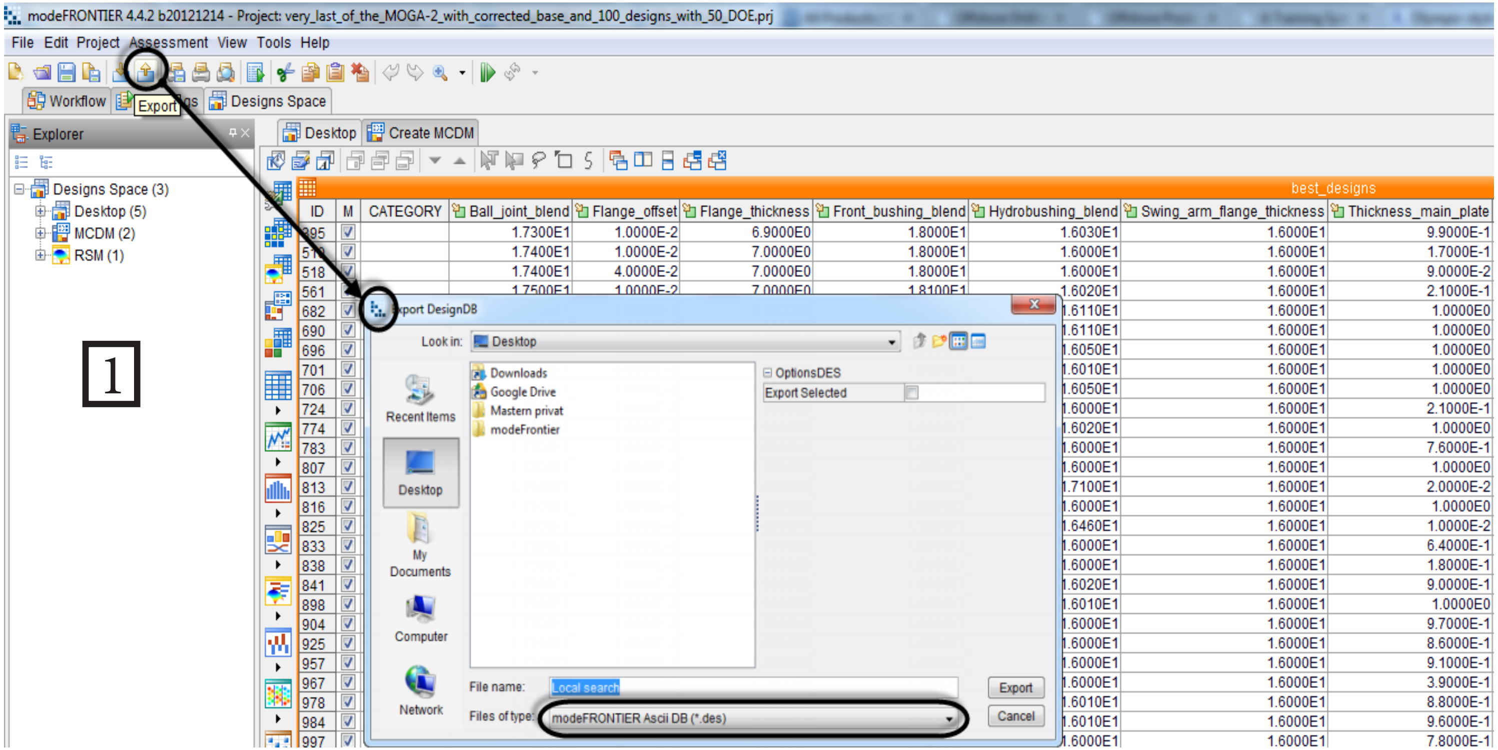
Appendix G

Local Search in modeFRONTIER

modeFRONTIER Local Optimization

Topic: Local Search using previous designs	Approved By: Terje Rølvåg	Name: Espen Nilsen, Carl Skaar	Date: 30.05.2013

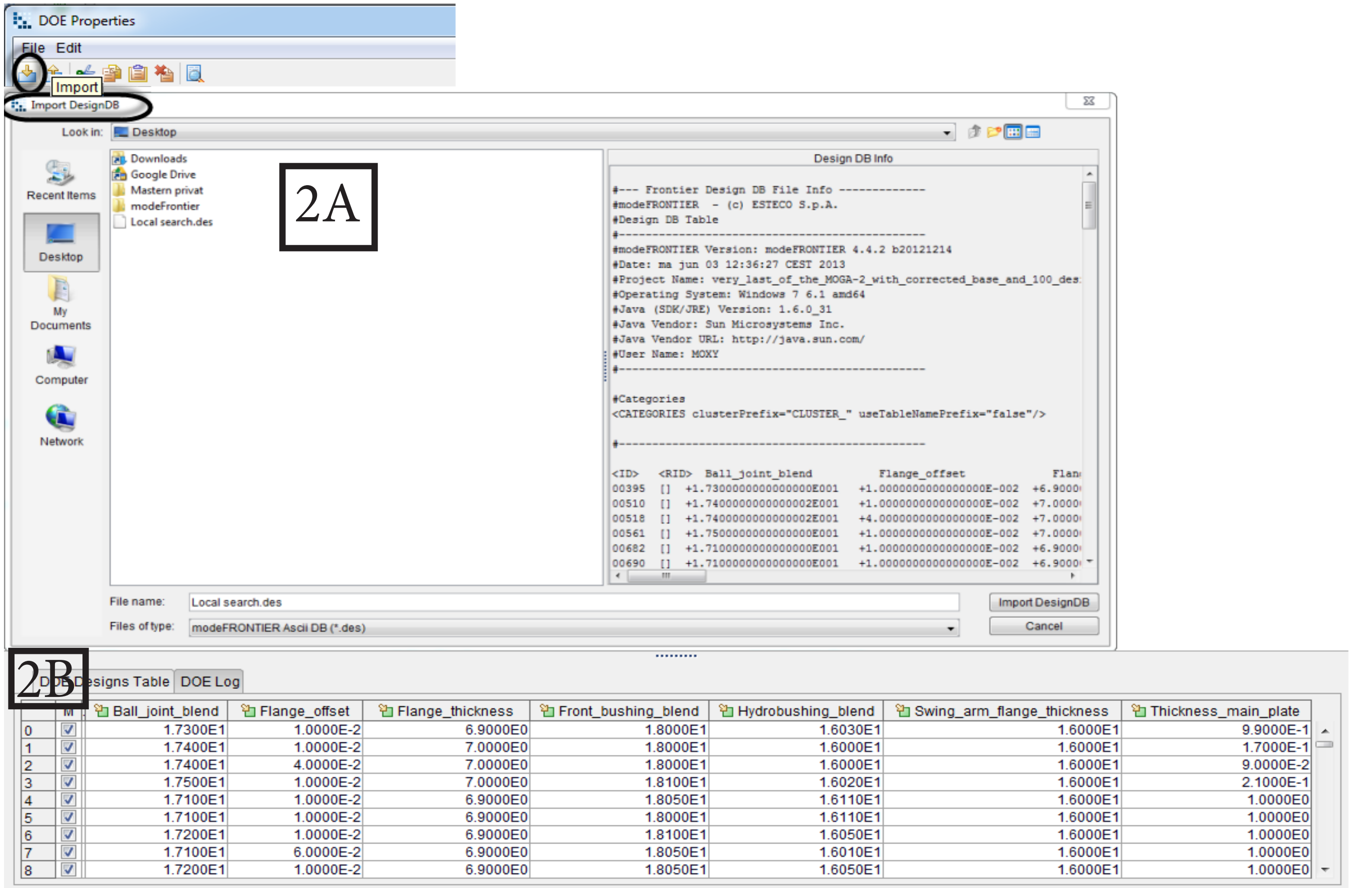
1. When the postprocessing of a run is done, one might want to explore the best designs further. A local search can then be executed based on the previous designs generated.
From the table of best designs generated one can export these designs as an ASCII file.



modeFRONTIER Local Optimization

Topic: Local Search using previous designs | Approved By: Terje Rølvåg | Name: Espen Nilsen, Carl Skaar | Date: 30.05.2013

2. A. The exported designs can now be imported to the DOE scheduler.
2. B. The imported designs are now listed in the DOE scheduler.
3. The local search can then be executed with these designs and a desired scheduler algorithm. In this particular case Hybrid was used because it had performed well on previous runs.



	M	Ball_joint_blend	Flange_offset	Flange_thickness	Front_bushing_blend	Hydrobushing_blend	Swing_arm_flange_thickness	Thickness_main_plate
0	✓	1.7300E1	1.0000E-2	6.9000E0	1.8000E1	1.6030E1	1.6000E1	9.9000E-1
1	✓	1.7400E1	1.0000E-2	7.0000E0	1.8000E1	1.6000E1	1.6000E1	1.7000E-1
2	✓	1.7400E1	4.0000E-2	7.0000E0	1.8000E1	1.6000E1	1.6000E1	9.0000E-2
3	✓	1.7500E1	1.0000E-2	7.0000E0	1.8100E1	1.6020E1	1.6000E1	2.1000E-1
4	✓	1.7100E1	1.0000E-2	6.9000E0	1.8050E1	1.6110E1	1.6000E1	1.0000E0
5	✓	1.7100E1	1.0000E-2	6.9000E0	1.8000E1	1.6110E1	1.6000E1	1.0000E0
6	✓	1.7200E1	1.0000E-2	6.9000E0	1.8100E1	1.6050E1	1.6000E1	1.0000E0
7	✓	1.7100E1	6.0000E-2	6.9000E0	1.8050E1	1.6010E1	1.6000E1	1.0000E0
8	✓	1.7200E1	1.0000E-2	6.9000E0	1.8050E1	1.6050E1	1.6000E1	1.0000E0