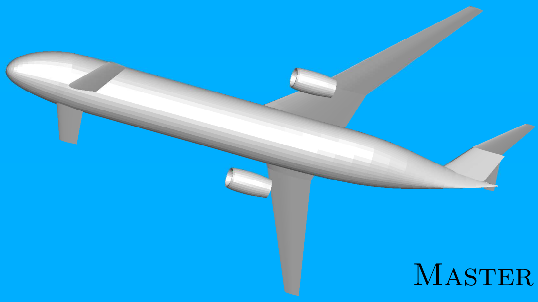
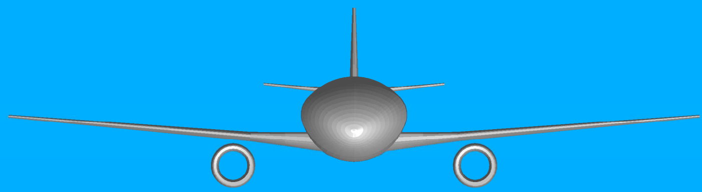


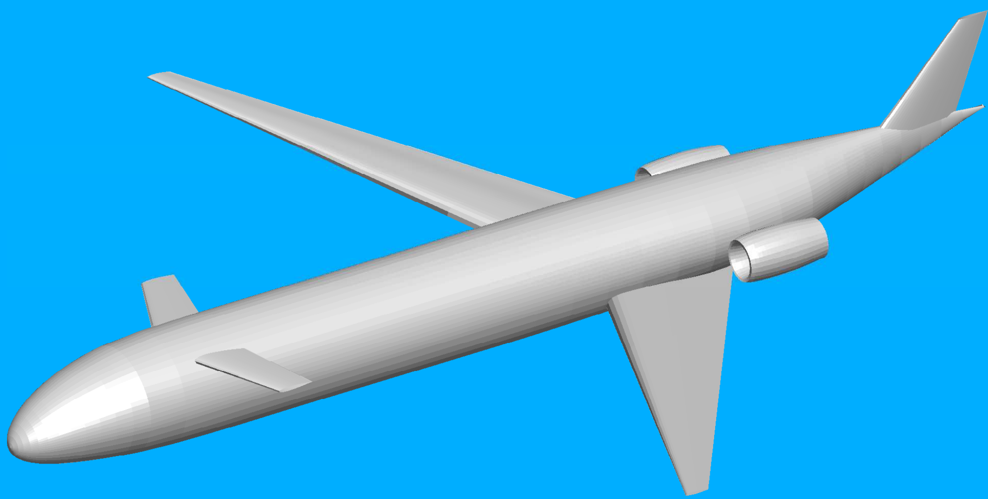
Master of Science Thesis



Optimization strategy for conceptual airplane design

P.T. Vasseur B.Sc.

May 9, 2014



Faculty of Aerospace Engineering · Delft University of Technology

TU Delft Delft University of Technology

Challenge the future

Optimization strategy for conceptual airplane design

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering at Delft University of Technology

P.T. Vasseur B.Sc.

May 9, 2014

Faculty of Aerospace Engineering · Delft University of Technology

Delft
University of
Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF
FLIGHT PERFORMANCE AND PROPULSION

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled **"Optimization strategy for conceptual airplane design"** by **P.T. Vasseur B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: May 9, 2014

Head of department: ______________________
prof.dr.ir. L.L.M. Veldhuis

Supervisor: ______________________
dr.ir. R. Vos

Reader: ______________________
dr.ir. H.G. Visser

Abstract

Due to the ever growing demand for more efficient aircraft novel aircraft concepts have to be explored. By improving design tools the potential of unconventional configurations can be further studied. This requires improvement of conceptual design tools such that more knowledge can be gathered on alternative solutions as early in the design process as possible. Multidisciplinary design optimization (MDO) can support this process by providing an environment in which the various disciplines can be designed and optimized concurrently, while a certain level of consistency is maintained.

An optimization design tool has been created to assess the potential performance gains of novel aircraft configurations. It connects with the Initiator design tool, which is a conceptual design framework. As such, it can also be used as a means to expose any analysis or design issues that may exist in the Initiator.

With the optimizer tool the following four case studies were performed: a conventional Airbus A320, a forward-swept canard aircraft, a three-surface aircraft and an oval-fuselage aircraft. For this purpose the genetic algorithm, sequential quadratic programming algorithm and a hybrid genetic algorithm were used. From the case studies followed that large improvements can be obtained with unconventional aircraft configurations when compared to the initial aircraft design proposed by the Initiator design tool. Up to 20% improvement was found with the three-surface and canard aircraft. The oval-fuselage aircraft could be improved by a solid 10%, while the lowest improvement was attained with the conventional A320.

Among all cases the most contributing factors were the wing longitudinal position, sweep angle and wing aspect ratio. There is a tendency towards lower sweep angles due to the positive effect on the weight of the wing and an underestimation of the drag rise. With the forward-swept canard relatively high sweep angles were found, which contradicts the findings of the aft-swept wings. Therefore, the aerodynamics routine needs further investigation. From the highly swept, high aspect ratio wings of the forward-swept canard aircraft followed that the weight penalty of forward swept wings is underestimated.
In three cases the fuselage fineness ratio was involved in the optimization. The results showed that changing the fineness ratio offered some reduction in fuselage weight due to a more favourable structural loading.

The sizing routine of the control surfaces is found to be inadequate, since the Initiator derives most parameters directly from the wing and does not properly take into account control and stability requirements. Results have shown that this mainly regards the sweep and dihedral angle. Especially, the sweep angle is of concern, since it changes the lift-curve slope and therefore also stall characteristics. These sizing issues also affect the static margin. It was found that class II design information was not fed back to the control surface sizing.

Other discrepancies were found with the wing dihedral. Due to a lack of lateral stability analysis of the Initiator the dihedral was driven by the lift-to-drag ratio rather than its stabilizing effect. As a result a lower dihedral was observed among the cases.

From the used optimization algorithms can be concluded that the gradient algorithm was the least effective. It had difficulties with the uncertainties in the computed results of the Initiator. It sometimes stopped prematurely or started oscillating. This was alleviated by increasing the step size of the algorithm, but at the expense of accuracy. The genetic algorithm was found to be the best option since it proved to be very robust. It is far less sensitivity to noise, because it does not use gradient information. Its computational cost could be significantly reduced by applying parallel optimization and using a caching mechanism. The hybrid algorithm was found to be too computational expensive. The obtained increase in objective value did not outweigh the added cost.

Acknowledgements

This thesis marks the final step in completing the Master Program in System Engineering and Aircraft Design at the faculty of Aerospace Engineering. I had a great time studying at Delft University of Technology, with the company of my fellow students and friends. The research presented in this report would not have been possible without the support of a number of people whom I hereby would like to thank.

Fist of all, I would like to express my gratitude to my supervisor dr.ir. Roelof Vos for his feedback, support and valuable insights during the course of my master thesis. I would also like to thank the members of my committee, prof.dr.ir. Leo Veldhuis and Dries Visser for their time to assess my work. I want to express my appreciation to Reno Elmendorp for providing feedback on the optimizer tool and helping me out with integration in the Initiator.

Last but not least, I would like to thank my parents for their continuous support and encouragement throughout my study.

Delft, The Netherlands
May 9, 2014

P.T. Vasseur B.Sc.

Contents

List of Figures

List of Tables

Nomenclature

Latin Symbols

AR	Wing aspect ratio	$[-]$
$\hat{s}$	Normalized sensitivity value	$[-]$
b	Wing span	[m]
C_D	Drag coefficient	$[-]$
C_L	Lift coefficient	$[-]$
c_r	Root chord length	[m]
c_T	Specific fuel consumption	$[g/(s \cdot kN)]$
c_t	Tip chord length	[m]
d_i	Elementary effect	$[-]$
$d_{\mathtt{f}}$	Fuselage diameter	[m]
e	Oswald efficiency number	$[-]$
L/D	Lift-to-drag ratio	$[-]$
$l_{\mathtt{f}}$	Fuselage length	[m]
M	Mach number	$[-]$
p	Grid level	$[-]$
R	Harmonic range	[km]
r	Number of trajectories	$[-]$
S	Wing surface area	[m^2]
s_L	Landing distance	[m]
s_{TO}	Take-off distance	[m]
t/c	Thickness-over-chord ratio	$[-]$

V_s	Stall speed	[m]
$W_{\mathtt{fb}}$	Block fuel mass	[kg]
$W_{\mathtt{p}}$	Payload mass	[kg]
X	Range parameter	[km]
x	Aircraft longitudinal axis	[m]

Greek Symbols

ϵ	Twist angle	[°]
Γ	Dihedral angle	[°]
Λ	Sweep angle	[°]
λ	Taper ratio	[−]
$\lambda_{\mathtt{f}}$	Fuselage fineness ratio, $l_{\mathtt{f}}/d_{\mathtt{f}}$	[−]
μ	Mean	[−]
μ^*	Modified mean	[−]
σ	Standard deviation	[−]

Subscripts

c	Canard
f	Fuselage
h	Horizontal tail
k	Wing kink section
r	Wing root section
t	Wing tip section
v	Vertical tail

Abbreviations

AAO	All-at-once
ALGA	Augmented Lagrangian genetic algorithm
CO	Collaborative optimization
CSSO	Concurrent subspace optimization
EE	Elementary effects method
FM	Aircraft fuel mass
GA	Genetic algorithm

HT	Horizontal tail
IDF	Individual discipline feasible
KPI	Key performance indicators
MAC	Mean aerodynamic chord
MDA	Multidisciplinary analysis
MDF	Multidisciplinary feasible
MDO	Multidisciplinary design optimization
MTOM	Aircraft maximum take-off mass
MZFM	Maximum zero-fuel mass
NLP	Non-linear optimization problem
OEM	Aircraft operational empty mass
OFAT	One factor at a time
PRE	Payload-range efficiency
SAND	Simultaneous analysis and design
SM	Static margin
SQP	Sequential quadratic programming
TSA	Three-surface aircraft
UML	Unified modeling language
VEM	Value efficiency parameter w.r.t. MTOM
VEO	Value efficiency parameter w.r.t. OEM
VT	Vertical tail
XML	Extensible markup language

Part I

Thesis

Chapter 1

Introduction

Due to the ever growing demand for more efficient aircraft novel aircraft concepts have to be explored. By improving design tools the potential of unconventional configurations can be further studied. The quest for more efficient aircraft requires improvement of conceptual design tools such that more knowledge can be gathered on alternative solutions as early in the design process as possible.

Since in aircraft design many disciplines are involved, obtaining the optimal design that satisfies the requirements is not an easy task. At the conceptual design level multidisciplinary design optimization (MDO) can support this process by providing an efficient methodology in which the various disciplines can be designed and optimized concurrently, while a certain level of consistency is maintained. As such, MDO plays an important role in the coordination and optimization of the various disciplines.

As the fidelity of the disciplines increases and the coupling between the disciplines grows, the more difficult and costly it becomes to develop and maintain an efficient MDO framework. Progress has been made in the field of MDO by the development of more advanced architectures, which use system decomposition, approximation models and concurrent optimization to reduce the computational expenses, cost and required interdisciplinary communication in large-scale systems. For this reason implementing an efficient MDO strategy for the advanced preliminary and detailed design phases remains a complex matter.

Currently, a conceptual design framework is being developed and maintained by the Flight Performance and Propulsion (FPP) group at Delft University of Technology. This framework uses a multidisciplinary design approach to support the conceptual design, analysis and evaluation of conventional and novel aircraft configurations. Based on a set of top level requirements a first aircraft design can be generated, which can serve as input for higher fidelity analysis tools.

1.1 Research question and thesis goal

As has been previously discussed, the use of a multidisciplinary optimization framework in the conceptual design phase can give valuable insights in the potential of novel aircraft configurations. Therefore, the research presented in this thesis is aimed at assessing the potential performance gain of unconventional aircraft configurations through the use of an optimization tool. This leads to the following research question:

> *What effect has the developed optimization strategy on the key performance indicators of unconventional aircraft configurations?*

In order to answer the research question subquestions have to be established. They are formulated as follows:

- How can the sensitivity of the design variables be determined efficiently in order to reduce the computational cost of the optimization?
- Which optimization strategies and algorithms are most suitable for implementation in the Initiator design tool?
- What is the impact of the optimization on the design of the aircraft configurations?

In order to answer the subquestions and subsequently the main research question an optimization tool has to be developed, which will be connected to the Initiator design framework. Therefore the thesis goal can be formulated as follows:

> *The development of an optimization tool for the conceptual design of conventional and unconventional aircraft that connects with the Initiator design tool.*

1.2 Report outline

This report is dived into two parts. In the first part of the report the content with respect to the thesis is presented. In the second part the implementation details of the optimizer tool are described.

In Chapter 2 background information is given regarding the role of multidisciplinary optimization in aircraft design and various optimization strategies. Also, a brief overview of the Initiator design tool is given. The used sensitivity analysis methodology is explained in Chapter 3. It provides a screening technique, which is used to find the most important design variables. Chapter 4 elaborates on the implemented algorithms in the optimizer tool. The weak and strong points of the algorithms are discussed. The optimizer tool is described in Chapter 5. It involves the optimizer workflow, implemented optimization strategy and parallel optimization. In Chapter 6 the results of the optimizer are evaluated by means of four case studies. The chapter also present some key performance indicators, which are used to compare the obtained designs. The conclusions and recommendations are given in Chapter 7. This chapter concludes the thesis part of this report.

The second part of the report starts with the program structure of the optimizer in Chapter 8. It provides descriptions for the tool methods and properties. The user manual is described in Chapter 9. It explains how the optimizer tool should be operated.

Chapter 2

Background information

This chapter contains background information regarding the thesis, which has been collected as part of the preliminary research. In the first section the multidisciplinary design optimization process is explained with respect to aircraft design. In Section 2.2 an overview of MDO strategies is given. In Section 2.3 a brief description of the Initiator design tool is presented.

2.1 MDO in aircraft design

Aircraft design involves many disciplines such as aerodynamics, propulsion, structure and cost. The disciplines often dependent on each other, which results in a complex and iterative design process. For instance, the required strength and thus weight of the wing depends on the aerodynamic loads and total weight of the aircraft, while the latter depend on the weight of the wing. Therefore the coordination between the various disciplines plays an important role. Multidisciplinary design optimization supports this design process by providing an efficient methodology in which the various disciplines can be designed concurrently while a certain level of consistency is maintained. Decomposition of large coupled problems into smaller subproblems may positively benefit the design time by reducing the computational complexity and design groups no longer have to wait for the results of other groups [29].

The aircraft design process can be divided into three phases: conceptual design, preliminary design and detailed design. In the conceptual design the requirements are established and an initial aircraft design is created. In the preliminary design phase the concept is further developed. At this stage calculations are done using high-fidelity models and tests are performed. Based on this information the design is refined. If the decision is made to manufacture the aircraft the detailed design phase is entered. In this last phase the fabrication details of the aircraft are determined like the placement of rivets, spars and other structural elements.

These design processes are depicted in Figure 2.1. As can be seen the goal is to gather as much knowledge on the design as possible in the early design stages, while keeping a high level of freedom in the design. This can be realized by the application of MDO techniques.

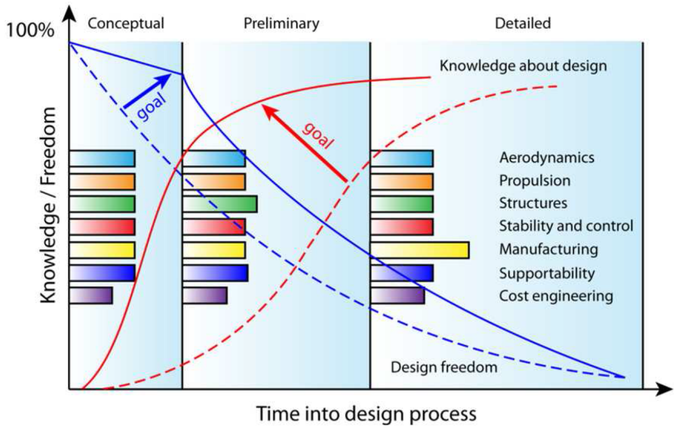


Figure 2.1: Aircraft design process [2]

2.2 Optimization strategies

The design of aircraft requires collaboration between many disciplines as described in the previous section. Because of the coupling between the various disciplines an optimization strategy has to be developed to ensure that all constraints are satisfied and that the interdisciplinary coupling variables have converged.

The MDO strategies can be categorized in two groups: the monolithic and distributed architectures [17]. The monolithic architecture uses a single optimization problem to solve the system. In the distributed approach the optimization problem is divided into smaller subproblems. Though an optimization problem can be solved by many optimization strategies, a suitable choice has to be made such that the most efficient strategy is used for the problem at hand.

2.2.1 Multidisciplinary feasible (MDF)

The multidisciplinary feasible strategy is a monolithic architecture. In this strategy the optimizer only controls the design variables an global design constraints. At the system level a multidisciplinary analysis (MDA) is performed to solve the coupling between the various disciplines. A representation of the strategy is shown in Figure 2.2. The optimization problem at the system level is as follows:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & g(x) \leq 0 \end{aligned} \tag{2.1}$$

The main advantage of MDF is that it always results in a consistent system at every design point. A second benefit is that complexity at the optimizer is reduced. A key disadvantage is that at each every evaluation of a design point a full MDA has to be performed. When using a gradient-based algorithm, the computation of the gradient requires a complete MDA run, which can be rather expensive in large problems [20]. Another downside is the high degree of coupling between the disciplines. Disciplines are likely to vary in computational difficulty, but they are run the same number of times.

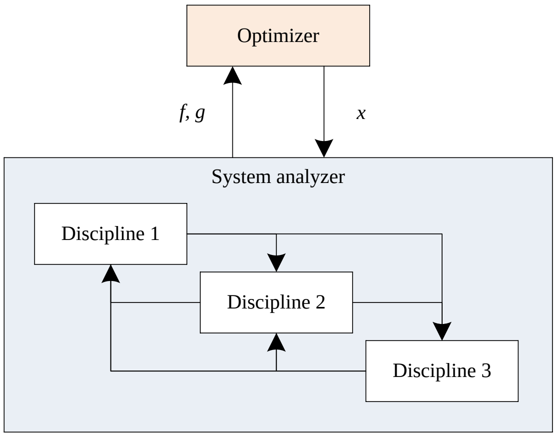


Figure 2.2: Multidisciplinary feasible strategy

2.2.2 Individual discipline feasible (IDF)

The individual discipline feasible strategy uses the optimizer to enforce compatibility between the disciplines. Like MDF, IDF is also a monolithic architecture. The optimizer provides a guess for the coupling variables to each discipline. Based on the guess the disciplines are solved individually. Convergence of the system is obtained by putting an equality constraint on the actual and guessed values of the coupling variables. A schematic representation of the IDF strategy is given in Figure 2.3. The optimization problem can be formulated as follows:

$$\begin{aligned} \min_{x,y,y'} \quad & f(x, y(x, y')) \\ \text{subject to} \quad & g(x, y(x, y')) \leq 0 \\ & y' - y = 0 \end{aligned} \tag{2.2}$$

In Equation 2.2 the coupling variables are denoted by y and the optimizer guess by y'. The IDF strategy allows the individual disciplines to be solved in parallel, since each discipline is supplied with a guess for the coupling variables. This can speed up the analysis. Generally, the strategy works well for relatively small problems. When the size of the problem grows, the number of coupling variables can become large which adversely affects its performance.

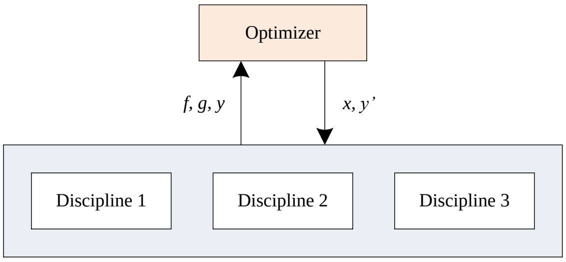


Figure 2.3: Individual discipline feasible strategy

2.2.3 All-at-once (AAO)

The all-at-once strategy is also known by the name simultaneous analysis and design (SAND). It belongs to the same category as MDF and IDF. The AAO approach further decomposes the system by simultaneously solving the state equations and the optimization problem. The state equations are formulated as equality constraints in the optimization problem. The optimization problem for AAO can be defined as follows [17]:

$$\begin{aligned} \min_{x,y} \quad & f(x,y) \\ \text{subject to} \quad & g(x,y) \leq 0 \\ & \mathcal{R}_i(x_0, x_i, y, \hat{y}_i) = 0 \quad \text{for } i = 1,..,N \end{aligned} \tag{2.3}$$

In Equation 2.3 $\mathcal{R}$ refers to the residuals of the state equations and N denotes the number of disciplines. A major disadvantage of the AAO strategy is that it quickly becomes impractical, because it requires all state equations and variables to be combined in the problem statement.

2.2.4 Collaborative optimization (CO)

Collaborative optimization is a distributed architecture. The optimization problem is solved at the system level and at the discipline level. For each discipline a optimization subproblem is formulated in which the discipline governs its own design variables and local constraints. This reduces the communication requirements in the system [1]. The role of the system-level optimizer is to minimize the design objective and the discipline-level optimizers are responsible for minimizing interdisciplinary inconsistency. The CO system-level problem can be formulated as follows:

$$\begin{aligned} \min_{x_0,\hat{x},\hat{y}} \quad & f(x_0, \hat{x}_i, .., \hat{x}_N, \hat{y}) \\ \text{subject to} \quad & g(x_0, \hat{x}_i, .., \hat{x}_N, \hat{y}) \leq 0 \\ & J_i^*(x_0, \hat{x}_i, .., \hat{x}_N, \hat{y}) = 0 \quad \text{for } i = 1,..,N \end{aligned} \tag{2.4}$$

In Equation 2.4 J^* symbolizes the interdisciplinary inconsistency of the system. The local design variables are denoted by $\hat{x}$ and the target values for the coupling variables by $\hat{y}$. The subproblem for each discipline is defined in Equation 2.5.

$$\begin{aligned} \min_{x_0,\hat{x_i},\hat{y_i}} \quad & J_i(x_0, x_i, y_i(x_0, x_i, \hat{y})) \\ \text{subject to} \quad & c(x_0, x_i, y_i(x_0, x_i, \hat{y})) \leq 0 \end{aligned} \tag{2.5}$$

The advantage of the CO approach follows from its fully separated disciplines. This strategy is useful for problems which have a low degree of coupling, since a high number of coupling variables leads to an increase in complexity and computational effort at the system level.

2.2.5 Concurrent subspace optimization (CSSO)

Concurrent subspace optimization belongs to the category of distributed architectures and decomposes the system into several independent subproblems, typically one for each discipline. Each subproblem tries to minimize the global objective with respect to its local design variables, while keeping the coupling variables constant.

The strategy starts with a full MDA of the system to obtain a consistent design. Using this design point the subspace optimizations are carried out concurrently. Each subsystem optimization results in a different design. These designs are used to generate an approximation model of the objective function, which is used by the system-level optimizer to solve the coordination problem and obtain convergence among the disciplines. After each iteration the approximation model is updated. The system-level problem can be defined as follows:

$$\begin{aligned} \min_{x,\tilde{y}} \quad & f(x, \tilde{y}) \\ \text{subject to} \quad & g(x, \tilde{y}) \leq 0 \end{aligned} \tag{2.6}$$

In Equation 2.6 $\tilde{y}$ denotes the state of the coupling variables of the other subspaces. Each subspace optimization can be formulated using Equation 2.7.

$$\begin{aligned} \min_{x,y_i,\tilde{y}} \quad & f(x, y_i, \tilde{y}_{j\neq i}) \\ \text{subject to} \quad & g(x, y_i, \tilde{y}_{j\neq i}) \leq 0 \end{aligned} \tag{2.7}$$

The main advantage of CSSO is the separation of the disciplines into subspace optimization problems, which can be evaluated in parallel. A downside is that the accuracy of the approximation models needs to be checked and validated [7]. Also, extensive tuning may be required in order to run CSSO efficiently, especially on large non-linear problems [17].

2.3 Initiator

The optimizer tool that is designed for the purpose of this thesis uses the Initiator design tool. It is a conceptual aircraft design tool and is mainly written in Matlab . It uses a modular structure to represent the components and disciplines of the system. The advantage this approach is that the components and analysis routines can be easily added or changed. The tool is driven by top level requirements, which are specified through a configuration file.

A simplified workflow of the Initiator is shown in Figure 2.4. As can be seen in this diagram the Initiator starts with the top level requirements. Using these requirements the sizing modules are called, which execute the class I design methods. At this point the initial geometry of the aircraft is generated and rough estimates for the weight and performance are obtained.

Next, the Initiator advances to the analysis modules which calculate the properties and characteristics of the aircraft in more detail by using class II and class II.V design methods. Amongst these modules is EMWET, which estimates the weight of the wing. The method has been developed by Elham as part of his PhD thesis [13]. The fuselage weight estimation module is designed by Schmidt for his Master's thesis [25], which can handle both conventional and novel fuselage shapes. The aerodynamics module is based on AVL, which is a vortex-lattice method developed by Drela [11].

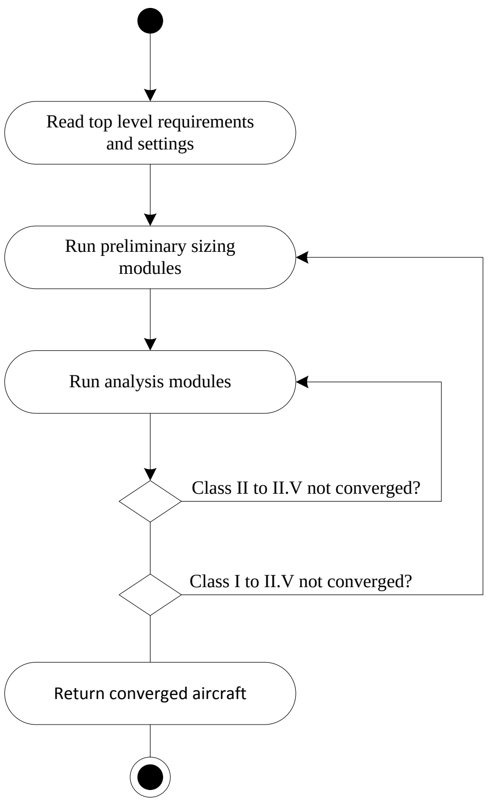


Figure 2.4: High-level activity diagram of the Initiator

When all analysis modules have been run, the results of the class II.V weight estimation are checked against the class II method. If the error is too large an iteration is performed until the results converge. Next, the results of the class II.V methods are compared to the class I estimates. An iteration of the design is performed when the results are too far off. At the end a fully converged aircraft is obtained for the specified requirements.

Besides the sizing and analysis modules, there are also design and workflow modules. The design modules involve the more detailed design of some part of the aircraft. They are placed outside the analysis workflow. Examples are the design of the cabin, the design of control surfaces like ailerons and elevators or the design of the landing gear. The workflow modules are used for tools or routines that control the Initiator workflow or to process module results. The optimizer tool will be part of this category. For an in-depth description of the Initiator design tool the reader is referred to Elmendorp [14].

Chapter 3

Sensitivity analysis

Aircraft design is a multidisciplinary design process which involves many design parameters. Due to the complex nature of the analysis routines and the couplings between the various disciplines it is often difficult to predict what impact each design variable has on the aircraft characteristics. This is where sensitivity analysis comes into play. In Section 3.1 variable screening is explained and in Section 3.2 a description of the elementary effects method is given.

3.1 Variable screening

Variable screening is a subcategory in the area of sensitivity analysis and is used to identify the contribution of input variables to the outputs of a model. This way the most influential parameters can be selected, such that optimization complexity and computational cost can be reduced.

In screening the aim is to qualify the measure of importance of the input factors rather than quantifying the exact sensitivity values. As such, screening is a useful addition to a design optimization strategy.

Once the sensitivity data has been obtained, the input factors can be ranked based on their importance. By selecting only the most important design variables, the dimensionality of the optimization problem can be reduced leading to faster optimization.

3.2 Elementary effects method

One of the most commonly used screening approaches is the elementary effects (EE) method. It employs the one-factor-at-a-time (OFAT) principle and provides a global sensitivity analysis. The computational cost of this approach is relatively low compared to other screening methods [8], which makes it a prime candidate when computationally expensive models are involved, like in the Initiator.

The elementary effects method is based on the work of Morris [18]. His method provides two sensitivity measures to determine the importance of input variables based on a series of experiments: the mean μ, which signifies the overall importance of an input factor, and the standard deviation σ, which indicates non-linear effects and interactions. These sensitivity measures are obtained by conducting a series of experiments in which the inputs are changed one at a time.

The sensitivity measures are obtained by changing the k-dimensional input vector x one component at a time in random order. This creates a so-called trajectory through the input space. The more trajectories are used the more reliable the sensitivity measures become as more input space is sampled. An example of a trajectory in 3-dimensional space is shown in Figure 3.1.

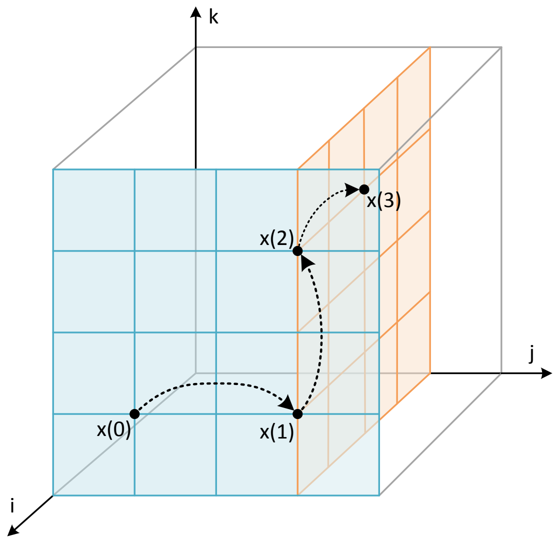


Figure 3.1: Example trajectory in 3-dimensional space using a five-level grid

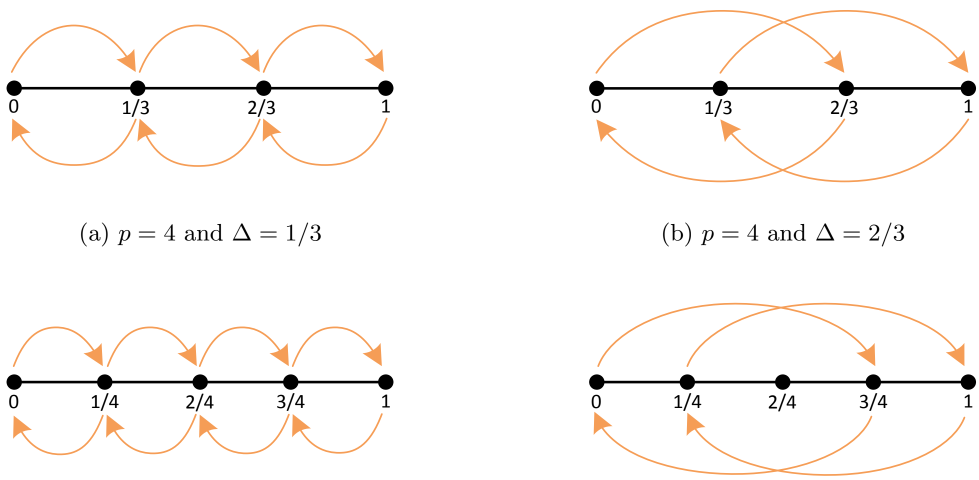
In the example it can be seen that in each subsequent step only one input is changed. The start of the trajectory $x^{(0)}$ is obtained by selecting a random point in the input space $[0,1]^k$, which is discretized into a p-level grid Ω. The next point $x^{(1)}$ is acquired by increasing or decreasing one component from $x^{(0)}$ with Δ such that $x^{(1)}$ is still in Ω. This is done until all components of x have been displaced with Δ. It follows that $k+1$ model runs are required to compute a single trajectory. So, for each input x_i the elementary effect d_i can be defined as follows:

$$d_i(x) = \frac{y(x_1,..,x_{i-1},x_i+\Delta,x_{i+1},..,x_k) - y(x)}{\Delta} \tag{3.1}$$

The step size Δ must be a predefined multiple of $1/(p-1)$. Though different combinations of p and Δ can be chosen, there exists some values for p and Δ for which the grid points have equal probability of being sampled. This occurs when p is an even number and Δ is calculated using Equation 3.2.

$$\Delta = \frac{p}{2(p-1)} \tag{3.2}$$

The effect of choosing different combinations of p and Δ is illustrated using the examples in Figure 3.2. Figure 3.2a shows that for $p = 4$ and $\Delta = 1/3$ the two outer points are less likely to be sampled. When Δ is changed to $2/3$ in Figure 3.2b, it can be seen that the sampling probability is equal for all grid points. Using $p = 5$ and $\Delta = 1/4$ as shown in Figure 3.2c the same problem arises as with example 3.2a. The two outer points have a lower sampling probability. Example 3.2d uses $p = 5$ and $\Delta = 3/4$. Here the center grid point is never sampled.



(a) $p = 4$ and $\Delta = 1/3$

(b) $p = 4$ and $\Delta = 2/3$

(c) $p = 5$ and $\Delta = 1/4$

(d) $p = 5$ and $\Delta = 3/4$

Figure 3.2: Sampling probability using different values for p and Δ

The choice of p is also related to the number of trajectories r. When a high-level grid is used more trajectories are required to make sure that all possible levels are explored. In this thesis a four-level grid ($p = 4$) is used with a Δ of $2/3$ and a total of 4 trajectories. According to Morris [18] a sample size of at least 4 is needed to obtain a reasonably reliable result.

With the calculated elementary effects the sensitivity measures can be determined. The mean μ_i of each input parameter follows from Equation 3.3.

$$\mu_i = \frac{1}{r}\sum_{j=1}^{r} d_i(x) \tag{3.3}$$

In this equation the parameter r refers to the number of trajectories. The corresponding standard deviation σ_i is given by Equation 3.4.

$$\sigma_i = \sqrt{\frac{1}{r-1}\sum_{j=1}^{r}(d_i(x) - \mu_i)^2} \tag{3.4}$$

An improved version was developed by Campolongo et al.[9], who added a modified mean μ^*. It uses the absolute values of the elementary effects to avoid cancellation effects when the function is non-monotonic. The formula is displayed in Equation 3.5.

$$\mu_i^* = \frac{1}{r}\sum_{j=1}^{r}|d_i(x)| \tag{3.5}$$

In order to rank the input parameters by importance the Euclidean distance with respect to modified mean μ^* and standard deviation σ is used. Though the value of μ^* alone would suffice to rank the parameters, results show that inputs with a high value for μ^* generally have a high value for σ as well [24].

$$s_i = \sqrt{\sigma_i^2 + (\mu_i^*)^2} \tag{3.6}$$

The elementary effects method is demonstrated in Chapter 6. In this chapter four case studies are presented for which the screening method is used to reduce the number of design variables.

Chapter 4

Optimization algorithms

In this chapter the algorithms used in the optimizer tool are presented. Multidisciplinary optimization problems are generally very costly in terms of computation time, so it is important to find a suitable algorithm that offers the most gain while keeping the computational effort as low as possible. Since each optimization problem has different requirements and characteristics, there is no algorithm that fits every case. Aspects like the available resources, required accuracy, model noise and chosen optimization strategy may affect the selection of an algorithm.

For this thesis a gradient-based algorithm, a genetic algorithm and a genetic hybrid algorithm is used. In the first section a formulation of the general optimization problem is given. In Section 4.2 gradient-based algorithms are discussed. This is followed by a description of the genetic algorithm in Section 4.3. The hybrid algorithm is explained in Section 4.4.

4.1 Problem description

The general non-linear optimization problem (NLP) can be defined as follows:

$$
\begin{aligned}
\min_x f(x) & \\
x \in \mathbb{R}^n & \\
\text{subject to } g_i(x) \leq 0, & \quad i = 1, .., j \\
h_i(x) = 0, & \quad i = 1, .., k \\
x_l \leq x \leq x_u &
\end{aligned}
\tag{4.1}
$$

In Equation 4.1 $f(x)$ is the objective function, which is a measure for the optimality of the design. An example of an objective function could be the payload-range efficiency or the lift-to-drag ratio. The problem is subject to inequality constraints $g_i(x)$ and equality constraints $h_i(x)$. Examples of constraints could be noise restrictions, emission regulations or coupling variables. Furthermore, the design vector x is restricted by an upper and lower bound.

4.2 Gradient-based algorithms

Gradient-based methods rely on first and second-order derivatives of the objective function to compute the search directions. One of the primary advantages of gradient-based algorithms is that tthey tend to convergence rather rapidly, especially near the optimum. In general the computational cost scales linearly with the number of design variables [30]. Another advantage is that they have a straightforward termination criterion. When the step size has been reduced by a certain order of magnitude it can be said with certainty that at least a local minimum has been found.

A disadvantage of gradient methods is its intolerance towards noise in the objective function. The algorithm might get stuck and stop prematurely or start to oscillate around a certain point. Also, there is no guarantee that a global minimum is found. Furthermore, the starting point may influence the outcome, because a different starting location might direct the algorithm towards another basin of attraction yielding a different optimum.

For this thesis the sequential quadratic programming (SQP) method is used. It is one of the more popular gradient methods and it is quite robust [15]. The idea behind SQP is that an approximation is made for the Hessian using a quasi-Newton updating method. Therefore this method can be seen as an extension to the Newton methods to the field of constraint optimization. SQP solves the non-linear problem by creating a quadratic programming (QP) subproblem at each iteration. The results of each QP subproblem are used to approximate the next search step. The QP subproblem can be set up using a Taylor expansion [6]:

$$\begin{aligned} \min_{d} \quad & f(x_k) + \nabla f(x_k)^T d + \frac{1}{2} d^T \nabla^2 \mathcal{L}(x_k, \lambda_k, \mu_k) d \\ \text{subject to} \quad & g(x_k) + \nabla g(x_k)^T d = 0 \\ & h(x_k) + \nabla h(x_k)^T d \leq 0 \\ \text{where} \quad & d = x - x_k \end{aligned} \tag{4.2}$$

In Equation 4.2 x_k is the approximation at the current iteration and $\mathcal{L}$ denotes the Lagrangian function of the problem. This function is given in Equation 4.3. Here λ and μ are the Lagrange multipliers.

$$\mathcal{L}(x_k, \lambda_k, \mu_k) = f(x) + \lambda^T g(x) + \mu^T h(x) \tag{4.3}$$

The optimizer tool that is designed for the purpose of this thesis uses the built-in SQP algorithm from MATLAB by means of the `fmincon` function. This function uses the popular Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm to approximate the Hessian $H = \nabla^2 \mathcal{L}$. This approximation is shown in Equation 4.4.

$$\begin{aligned} & H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k s_k s_k^T H_k^T}{s_k^T H_k s_k} \\ \text{where} \quad & s_k = x_{k+1} - x_k \\ \text{and} \quad & q_k = \nabla \mathcal{L}(x_{k+1}, \lambda_{k+1}) - \nabla \mathcal{L}(x_k, \lambda_k) \end{aligned} \tag{4.4}$$

When the QP subproblem is solved the new iterate x_{k+1} can be computed:

$$x_{k+1} = x_k + a_k x_k \tag{4.5}$$

The step length parameter a_k in Equation 4.5 follows from a line search to determine an appropriate step size. Finally, the algorithm advances to the new iterate x_{k+1} and a new QP subproblem is generated. This procedure is repeated until a termination criterion halts the algorithm. For more information regarding the algorithm implementation the reader is referred to MATLAB manual [4].

4.3 Genetic algorithm

The genetic algorithm (GA) is an evolutionary algorithm. Instead of relying on derivative information like gradient-based methods, it uses the principle of natural selection. The algorithm uses a population of individual solutions. Usually, the algorithm starts by initializing a randomly generated population. At each iteration all individuals of the current generation are ranked according to their fitness value, which follows from the objective function. Then the following selection rules are applied to create the next generation:

- *Crossover*: children are created by combining the design vectors of a pair of parents
- *Mutation*: children are created by making random changes to a single parent
- *Elite*: individuals with the best fitness values survive to the next generation

In Figure 4.1 these operations are visualized. Fitter solutions are more likely to be selected to create children. As the algorithm progresses the average fitness value of the population will increase, because only the best solutions survive to the next generation. The algorithm terminates when the best fitness value is not increasing anymore for a number of generations.

Compared to gradient-based methods, the genetic algorithm is more robust [28]. It can operate in noisy environments and is able to solve non-smooth optimization problems. The algorithm is less likely to be trapped in a local optimum, since multiple solutions are used to explore the design space. However, there is no guarantee that the global optimum is found. The algorithm may suffer from early convergence leading to a suboptimal solution

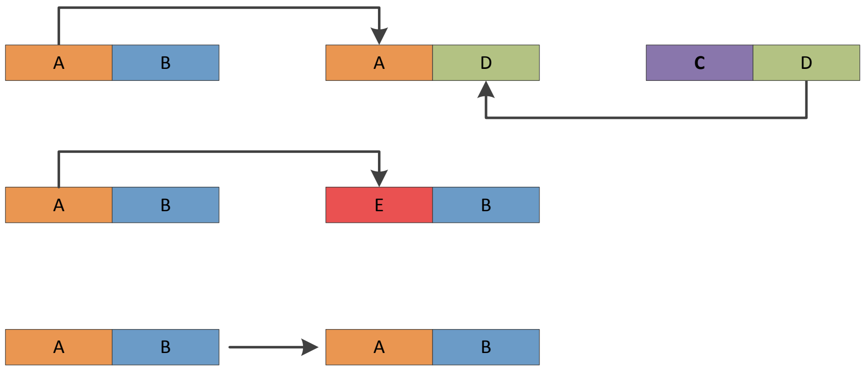


Figure 4.1: The crossover, mutation and elite selection procedures of the genetic algorithm

[26]. Increasing the mutation rate may alleviate the problem. Another disadvantage of the genetic algorithm is that convergence tends to be slow near the optimum [30]. Furthermore, the algorithm requires many function evaluations, because each individual within the population has to be computed every generation. This can be overcome by caching all individual solutions, which is explained in Section 5.3. The process can be further speed up by computing individual solutions in parallel as is described in Section 5.4.

In the optimizer tool the built-in function `ga` from MATLAB is employed. It uses the augmented Lagrangian genetic algorithm (ALGA) and is based on the work of Conn et al.[10]. The ALGA algorithm treats the bounds and linear constraints separately from the non-linear constraints. The non-linear constraints and the fitness function $f(x)$ are combined into a subproblem, which is shown in Equation 4.6

$$\begin{aligned}\Phi(x,\lambda,s,\rho) = f(x) - \sum_{i=1}^{m}[\lambda_i s_i \log\left(s_i - c_i(x)\right)] - \sum_{i=m+1}^{mt}[\lambda_i \log ceq_i(x)] \\ + \frac{\rho}{2}\sum_{i=m+1}^{mt}[\lambda_i ceq_i(x)^2]\end{aligned} \tag{4.6}$$

In this equation λ are Lagrange multiplier estimates, s is a vector containing non-negative shifts and ρ is a positive penalty parameter. Each subproblem uses fixed values for λ, s and ρ. The Lagrangian estimates are updated when the subproblem converges to feasible conditions. When the constraints cannot be satisfied the penalty parameter is increased. In both cases this leads to a new subproblem formulation. This procedure is repeated until the termination criteria are reached. Examples of termination criteria for the genetic algorithm are: no improvement in fitness value over a specified number of generations, exceeding the maximum number of generations or an imposed time limit.

4.4 Hybrid algorithm

A hybrid algorithm combines two or more distinct optimization routines to solve the same optimization problem. The idea behind this approach is to use the strong points of each method such that the combined algorithm is better than the individual algorithms.

A well-known example is the hybrid genetic algorithm in which the genetic algorithm is paired with a more fine-grained solver, which is often gradient-based. This type is also used for the optimizer tool. It uses the previously discussed genetic and SQP algorithm to form a hybrid algorithm.

The primary role of the genetic algorithm in this setup is to explore the design space in order to find the region with the most promising optimum, which is the global optimum in the ideal case. The region discovered by the genetic algorithm can then be used by the gradient method to hit the exact optimum in that region. The difficulty of the hybrid algorithm lies in determining the decision criterion that dictates when the algorithm must switch to the local solver. Studies showed that there are multiple criteria that can be used for this purpose, but the effectiveness of these criteria depends on the type of problem and requires tuning of the termination parameters [12].

Chapter 5

Optimizer tool description

Since many disciplines are involved in aircraft design a multidisciplinary design optimization (MDO) tool is required. The interactions between the disciplines greatly increase the complexity of the system. Since analysis routines can be computationally expensive, finding a feasible design within a reasonable amount of time can be more valuable than finding the optimal design from a mathematical point of view.

The optimizer tool has been written in MATLAB . It is implemented as a workflow module in the Initiator framework, which has previously been described in Section 2.3. This way the optimizer can be operated through the established routes in the Initiator and it also facilitates easier implementation and linking with the design tool.

In this chapter first the workflow of the optimizer is explained. This is followed by a description of the implemented optimization strategy. In Section 5.3 the caching technique is discussed. In the final section the application of parallel optimization is presented.

5.1 Optimizer workflow

The high-level workflow of the optimizer tool is illustrated in Figure 5.1 and can be described as follows. When the optimizer the tool starts with reading the optimization problem and settings. The optimization problem contains information regarding the objective function, design variables, constraints and selected optimization algorithm. Next, the optimizer uses the Initiator to compute the initial design point. The analysis results of this baseline aircraft and its geometry data are saved for later use. Then the number of design variables is checked. If this number exceeds the specified maximum number of design variables, a sensitivity analysis is performed first. The sensitivity analysis routine uses the elementary effects method described in Chapter 3. When the analysis has completed, the design variables are ranked according to their computed sensitivity value and the most important variables are selected. By default the top 5 variables are taken. Using the selected variables the optimization problem is set up. After the provided objective, constraints and algorithm options have been set, the specified algorithm is called. Depending on the options a parallel MATLAB session may be started. Once the algorithm has completed, the resulting optimum design vector is used to compute the final aircraft. Its properties and analysis results are saved to disk, along with the optimization and baseline aircraft data.

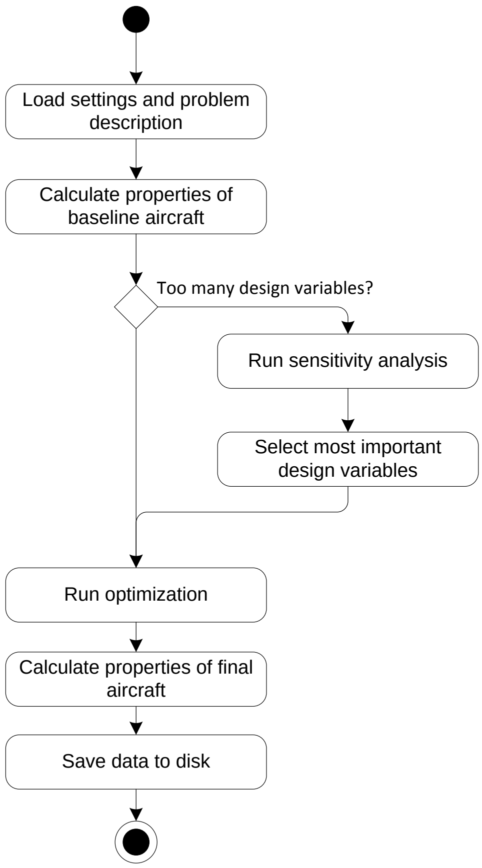


Figure 5.1: High-level activity diagram of the optimizer

A more detailed description of the routines and functions can be found in Chapter 8. In this chapter flowcharts are given of the sensitivity routine and the optimization functions.

5.2 Optimization strategy

For the optimizer tool the multidisciplinary feasible optimization strategy was chosen. Due to the existing structure of the Initiator, which already featured a design convergence module that performs the multidisciplinary analysis, the MDF architecture was adopted. It is a traditional MDO strategy in which the optimizer is in charge of the design variables and global design constraints.

Figure 5.2 shows the MDF implementation of the optimizer tool. Here it can be seen that the optimizer passes the design vector x to the cache layer (see Section 5.3). If there are already results for the passed design vector the cache immediately returns the corresponding objective and constraint values. Otherwise the design vector is forwarded to the Initiator and a full MDA is done. The MDA is performed by the Initiator's design convergence module. This module takes care of obtaining consistency across all disciplines and compliance with system constraints.

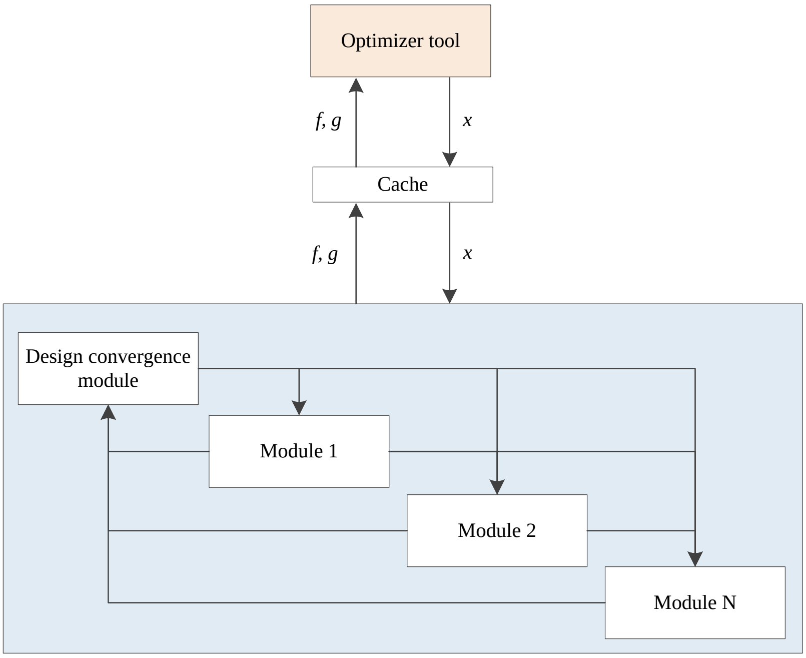


Figure 5.2: Multidisciplinary feasible implementation of the optimizer tool

As already has been discussed in Chapter 2, the major advantage of MDF is that it always returns a consistent system for a given design vector. Another benefit is that when changes are made in the Initiator, which require adjustments in the MDA routine, the optimizer does not have to be modified. The loose coupling promotes maximum flexibility. When a strategy like IDF, CO or CSSO was chosen the optimizer would have to be updated every time someone decides to make a change to a particular module. This requires knowledge of the whole optimization structure, which is an undesirable situation. For the same reason approximation models or response surface techniques are not incorporated in the implementation. Since the Initiator is subject to changes maintaining and updating such models would be too costly. As such MDF is a robust and intuitive strategy, which is most suitable for this environment.

5.3 Caching results

In order to speed up the optimization the effect of adding a caching mechanism at the system level has been investigated. The caching mechanism allows to quickly retrieve the results of design points that have already been processed. This way an expensive recalculation is avoided.

This technique is only beneficial for algorithms that may re-examine prior results. A perfect example is the genetic algorithm. In each generation part of the population survives to the next generation. Therefore the results of the surviving design vectors will be needed again. The effect of caching using the genetic algorithm is shown in Figure 5.3.

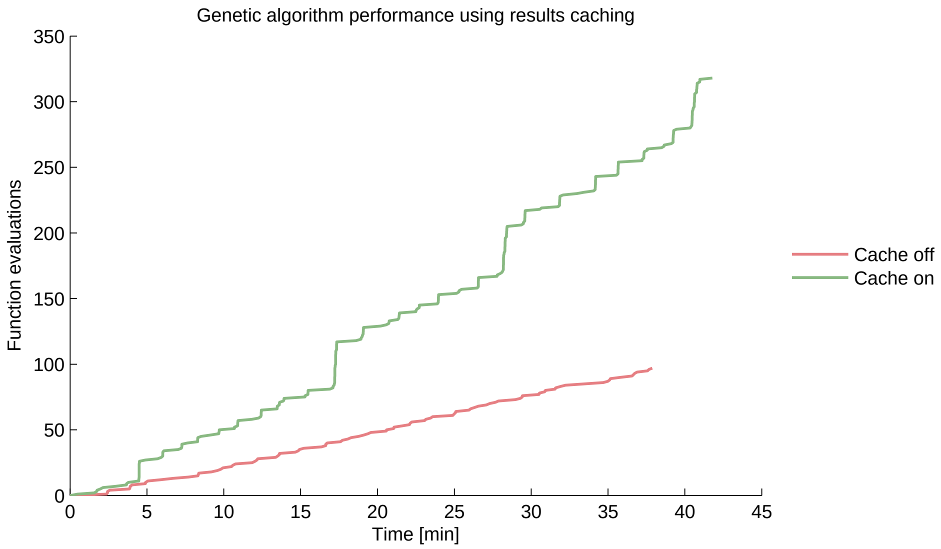


Figure 5.3: Effect of caching on the genetic algorithm

In this figure it can be seen that with caching the algorithm is able to evaluate much more solutions in the same period of time. The steps in the line indicate that a match in the cache has been found. The big steps can be explained by the adjustment of the Lagrangian estimates of the genetic algorithm. The Lagrangian multipliers are used to solve the fitness function with respect to the nonlinear constraints and are updated after every couple of optimization steps, as was described in Section 4.3. When this happens previous solutions are checked against the new multiplier estimates. The results can immediately be pulled from the cache, which saves a lot of time.

Generally, gradient methods do not benefit from caching, because along its gradient-based search path it is unlikely that the same point will be requested again. Nevertheless, the caching layer is always on by default, because the overhead of this mechanism is very small compared to a full MDA computation.

5.4 Parallel optimization

Since multidisciplinary analysis can be very expensive, the use of parallel computing has been researched. In order for parallelization to be effective, there should be multiple tasks that can be performed simultaneously and have to be of sufficient workload. If the tasks are too small, the communication overhead will become too large resulting in parallel slowdown. Also, attention must be paid to race conditions, because two processes may request the same resource at the same time.

Not all algorithms are suitable for this technique. Algorithms at which the next solution depends on previous iterate, the solutions can only be computed one at a time. Sometimes only particular subroutines of an algorithm may be suitable for parallelization. An example is a gradient-based algorithm which estimates the gradients simultaneously, but can only evaluate one design point at a time.

The multi-solution approach of the genetic algorithm makes it an excellent candidate for parallel optimization. Each individual solution of a particular generation can be computed independently. A benchmark has been performed to determine how well the genetic algorithm scales with the number of workers. The test has been performed on a modern quadcore processor, which allows up to four parallel workers. The population size is set to 12 to ensure that the workload is equally distributed across the available workers in each test case. The caching layer has been disabled for benchmark to test the raw performance. The results of this benchmark are shown in Figure 5.4. A trend line has been fitted through each set of data points.

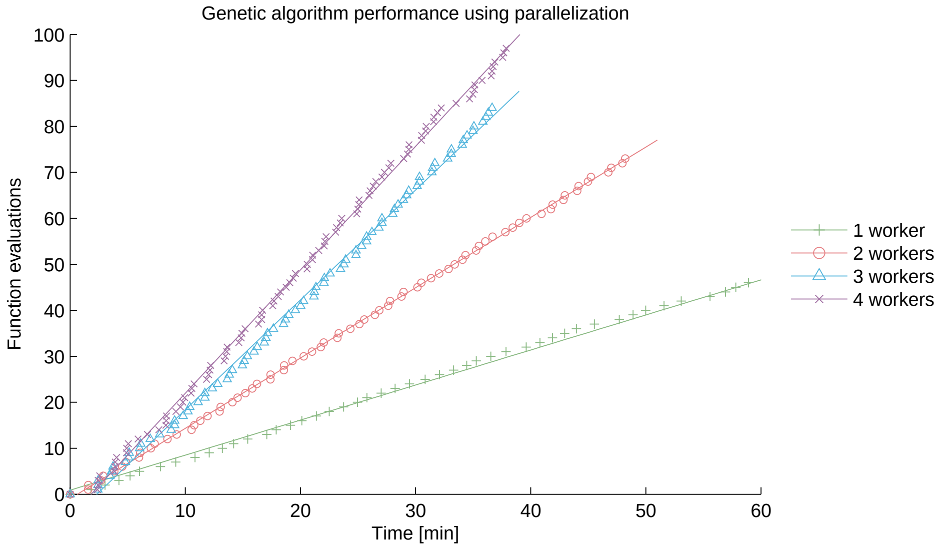


Figure 5.4: Scaling the genetic algorithm using parallel computing

From the results it can be seen that the algorithm scales linearly up to three parallel workers. By using a single worker about 0.76 function evaluations per minute are performed. By adding a second and third worker this number increases to 1.53 and 2.39 respectively. The performance gain diminishes when the system's maximum of four workers is used. Because the system processes also require processor time, the fourth available thread cannot be fully utilized. A maximum of 2.68 function evaluations per minute is reached, which is 2.5 times more than with a single worker. So parallelization of the genetic algorithm proves to be very beneficial.

Chapter 6

Optimization case studies

In this chapter four aircraft configurations are evaluated to assess the effect of the optimization on their performance and characteristics. For this purpose the following aircraft configurations are selected: a conventional Airbus A320, a canard aircraft with forward swept wings, a three-surface aircraft and an oval-fuselage aircraft.

In Section 6.1 the key performance indicators are established. Next, the optimization procedure is explained in Section 6.2. Using this procedure the case studies are performed. Their results can be found in Section 6.3 through 6.6. In the final section the cases are compared and an evaluation of the algorithms is done.

6.1 Key performance indicators

The performance of an aircraft can be expressed in many ways. The key performance indicators (KPI) are likely to vary depending on the perspective. Aircraft manufacturers, consumers, airline companies and legislators all have different opinions with respect to performance and efficiency parameters. For commercial airliners fuel efficiency and operating cost are very important factors. Also, environmental issues like noise and emissions are becoming more and more important, which put constraints on the design space.

The selection of a key performance indicator is very important as it defines which objective has to be considered for optimization. This choice will have consequences for the entire design of the aircraft, so a sensible parameter must be used. A suitable measure of aircraft performance is the payload-range efficiency (PRE) [19]. It is defined as follows:

$$\mathtt{PRE} = \frac{W_{\mathtt{p}} \cdot R}{W_{\mathtt{fb}}} \tag{6.1}$$

In Equation 6.1 $W_{\mathtt{p}}$ is the payload mass, R is the harmonic range and $W_{\mathtt{fb}}$ is the block fuel load. The block fuel is the total fuel minus any reserve fuel. The harmonic range is the furthest distance the aircraft can fly with maximum payload. Other ranges are the maximum fuel range and ferry range and they are typically indicated in a payload-range diagram. An example of such a diagram is given in Figure 6.1, which is based on the Airbus A320 from the first test case.

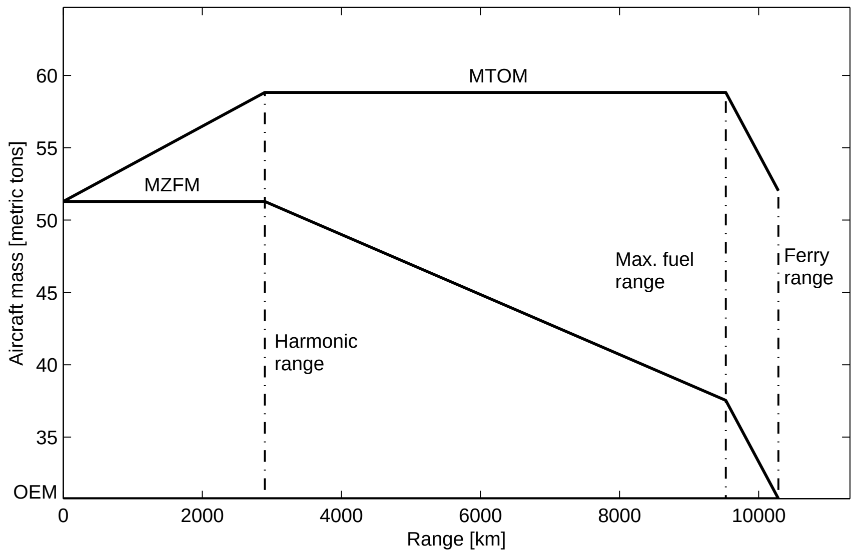


Figure 6.1: An example of a payload-range diagram

Another important performance indicator is the range parameter X shown in Equation 6.2. The range parameter follows from the first part of the Breguet range equation. It denotes the aerodynamic and propulsion efficiency of an aircraft.

$$X = \frac{V \cdot L/D}{c_T} \tag{6.2}$$

In this equation L/D is the lift-to-drag ratio and c_T is the specific fuel consumption.

6.2 Optimization procedure

In this section the optimization procedure is laid out. It consists of two main steps. The first step is running the sensitivity analysis. This is explained in Section 6.2.1 and is conducted once for each aircraft case. Next, the actual optimization is performed. This is done three times, once for each algorithm. It is described in Section 6.2.2. In the last section the used Initiator and aircraft settings are given.

6.2.1 Sensitivity analysis

For each case a sensitivity analysis is performed to identify the most important design variables. This is done using the elementary effects method described in Chapter 3.

According to Morris [18] a sample size of at least 4 is needed to obtain a reliable result. Therefore the number of trajectories is set to 4, which is a fair trade-off with respect to the computational cost. Each design variable is varied across four levels on the grid.

After the screening has been performed the most influential parameters are selected based on their sensitivity value. Therefore a maximum of 5 design variables is established. The sensitivity value s that is used in this chapter is the normalized version of Equation 3.6:

$$\hat{s}_i = \frac{s_i}{s_{\mathtt{max}}} \cdot 100 \tag{6.3}$$

6.2.2 Optimization

The top 5 design variables that follow from the sensitivity analysis are used in the optimization routine. Three runs will be done per aircraft, each time using a different optimization algorithm. For this purpose the genetic algorithm, gradient-based SQP algorithm and the genetic–SQP hybrid algorithm are used. The results of the optimization are evaluated afterwards.

As already explained earlier in the key performance indicators section, the payload-range efficiency is used as the optimization objective. The following objective function has been set up:

$$\min f(x) = -\frac{\mathtt{PRE}}{1000} \tag{6.4}$$

As can be seen in Equation 6.4 a minus sign is added and a scaling factor is introduced to normalize its value. Though most constraints are handled by the Initiator itself, the nose loading constraint is not forced. To solve this problem the following two inequality constraints are defined:

$$g_1(x) = -\frac{\hat{x}_{cg_{\mathtt{min}}} - \hat{x}_{\mathtt{fwd}}}{\mathtt{MAC}} \leq 0 \tag{6.5}$$

$$g_2(x) = -\frac{\hat{x}_{\mathtt{aft}} - \hat{x}_{cg_{\mathtt{max}}}}{\mathtt{MAC}} \leq 0 \tag{6.6}$$

Equation 6.5 denotes the maximum nose loading constraint and Equation 6.6 the minimum nose loading constraint. Note that $\hat{x}$, which refers to the longitudinal position, is not the same as the design vector x. Both constraints are expressed in terms of the MAC.

A maximum runtime of two hours is maintained for the genetic and gradient algorithm. From running the design tool multiple times it followed that after two hours there was no significant gain in objective value. In some cases the algorithms even stopped before the two hour limit. In Figure 6.2 the improvement in objective value against the computation time is shown.

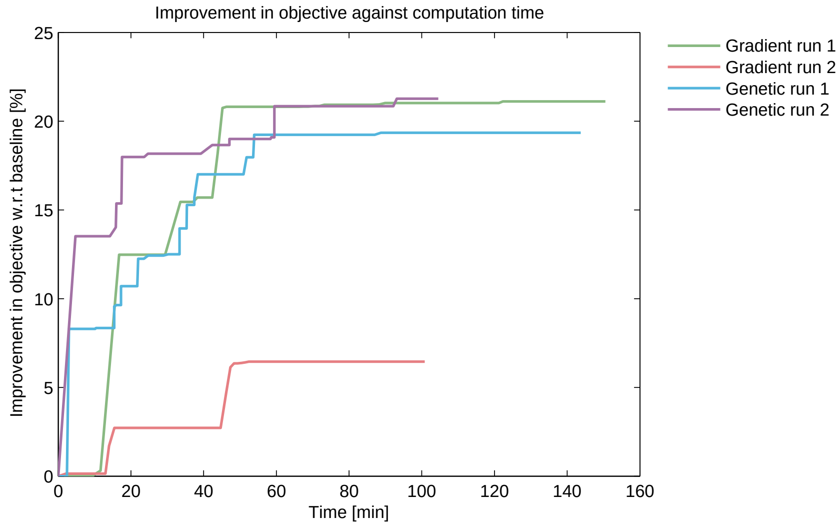


Figure 6.2: Improvement in objective value against computation time

The hybrid method is given a maximum of three hours. The run time of the first stage, the genetic algorithm, is limited to two hours. Then the local solver is run, which is limited to 1 hour.

For the genetic and gradient-based SQP algorithm most options were left at their default values. The population size of the genetic algorithm was adjusted to 12 with an elite count of 2. Though a higher population count yields more diverse individuals, it takes considerable more time to compute a single generation and therefore less generations can be evaluated in the same amount of time. It was found that 12 individuals is a fair trade-off. This way it takes about 5 to 10 minutes per generation depending on the computational difficulty of the aircraft configuration. The algorithm stops when there is no improvement in best fitness value after 4 generations.
The minimal step length of the SQP algorithm has been set to 10^{-3} to overcome the output noise of the design tool. The default objective tolerance of 10^{-6} is maintained.

Design variables

The next step is to establish a set of design variables. The selected variables are gathered in Table 6.1. Most parameters are related to the wing geometry, since it is expected that they have the most influence on the objective value. Also, the longitudinal position of the wing and the diameter of the fuselage are added.

The wing reference area must be kept constant when setting its geometry after a design point has been chosen in the preliminary sizing. Therefore the root chord of the wing is treated as a dependent variable and is calculated based on the given aspect ratio, taper ratio and sweep angle. Changes in wing area due to twist, dihedral and thickness ratio are assumed to be small. The wing position is selected because of its important role in the longitudinal stability of the aircraft. It influences the size of the control surfaces and the position of the landing gear. It is expressed in terms of the fuselage length. For the same reasons the canard position is added, but it is only used for the three-surface aircraft case.

Variable	Symbol	Min.	Max.
Aspect ratio	$\mathcal{R}$	8.0	15.0
Sweep angle[1]	Λ	0.0°	35.0°
Taper ratio	λ	0.0	1.0
Root thickness ratio	$(\frac{t}{c})_{\mathtt{r}}$	−25%	+25%
Kink thickness ratio[2]	$(\frac{t}{c})_{\mathtt{k}}$	−25%	+25%
Tip thickness ratio	$(\frac{t}{c})_{\mathtt{t}}$	−25%	+25%
Dihedral angle	Γ	−5°	+5°
Root twist angle	$\epsilon_{\mathtt{r}}$	−5°	+5°
Kink twist angle[2]	$\epsilon_{\mathtt{k}}$	−5°	+5°
Tip twist angle	$\epsilon_{\mathtt{t}}$	−5°	+5°
Wing longitudinal position	$f_{x_{\mathtt{w}}}$	−0.15	+0.15
Canard longitudinal position[3]	$f_{x_{\mathtt{c}}}$	0.05	0.20
Fuselage diameter	$d_{\mathtt{f}}$	−25%	+25%

Table 6.1: List of design variables

[1]For a forward swept wing the bounds are inversed
[2]Not applicable to the canard aircraft due to its forward swept wing
[3]Three-surface aircraft only

The fuselage diameter is added due to its effect on its structural weight and moment arm with respect to the control surfaces. Based on the given diameter the design tool calculates the required length to make sure that enough seats can be placed to carry the required number of passengers. In case of the oval-fuselage aircraft the width is controlled instead of the diameter.

6.2.3 Initiator and aircraft settings

The Initiator settings are mostly kept at their default values. The most important settings are mentioned here. The allowed convergence error between the class II and II.V weight estimation is kept at 1%. The weight error between the class I and II.V estimation is 0.5%. The minimum nose gear loading is set to 5% and the maximum nose loading is set to 20%. The passenger mass is 80 kg and the luggage mass per passenger is 25 kg.

For all cases the Boeing 737 airfoils are used for the main wing. The airfoil shown in Figure 6.3a is used for the root section, while the airfoil from 6.3b is placed at the kink and tip section. The kink location is fixed at 30% of the wing semi-span.

(a) Boeing 737-based root section airfoil

(b) Boeing 737-based kink and tip section airfoil

(c) NACA0012 airfoil

Figure 6.3: Airfoils used in the case studies

The horizontal tails have an aspect ratio of 5.0 and a taper ratio of 0.35. The other properties are automatically sized by the Initiator.

The vertical tail has an aspect ratio of 1.0 and a taper ratio of 0.35. In case of a T-tail configuration the values are 1.6 and 0.7 respectively. For the canard an aspect ratio of 5 and taper ratio of 0.60 is used. The sweep and dihedral are derived from the main wing as well. For all control surfaces the NACA0012 airfoil is used, which is shown in Figure 6.3c.

6.3 Case 1: Airbus A320

The first case that is considered is an aircraft which has similar requirements as the Airbus A320-family. It is a conventional aircraft for short to medium range. It must be able to carry 150 passengers over a range of 2870 km at a cruise speed of Mach 0.78. All top level requirements are gathered in Table 6.2.

Pax.	Payload mass	$M_{\mathtt{C}}$	Altitude	Range	$s_{\mathtt{TO}}$	$s_{\mathtt{L}}$
150	20.5 tons	0.78	11.3 km	2870 km	2180 m	1440 m

Table 6.2: Airbus A320 top level requirements

The Initiator uses this information to create an aircraft that fulfils these requirements. The resulting aircraft properties are shown in Table 6.3. A 3-dimensional model of the aircraft is illustrated in Figure 6.4.

AR	9.4	–	Γ	6.0	°	$AR_{\mathtt{h}}$	4.9	–	MTOM	58.9	tons
S	126	m^2	$f_{x_{\mathtt{w}}}$	0.45	–	$S_{\mathtt{h}}$	25	m^2	OEM	30.8	tons
b	34.5	m	$(\frac{t}{c})_{\mathtt{r}}$	0.151	–	$b_{\mathtt{h}}$	11.1	m	FM	7.6	tons
Λ	26.2	°	$(\frac{t}{c})_{\mathtt{k}}$	0.104	–	$AR_{\mathtt{v}}$	1.6	–	$R_{\mathtt{h}}$	2900	km
c_r	7.6	m	$(\frac{t}{c})_{\mathtt{t}}$	0.104	–	$S_{\mathtt{v}}$	18	m^2	PRE	7880	km
λ	0.16	–	$l_{\mathtt{f}}$	40.6	m	$b_{\mathtt{v}}$	5.2	m	L/D	17.6	–
ϵ	0	°	$d_{\mathtt{f}}$	4.2	m				$C_{L_{\mathtt{max,clean}}}$	1.24	–

Table 6.3: Airbus A320 properties using the Initiator

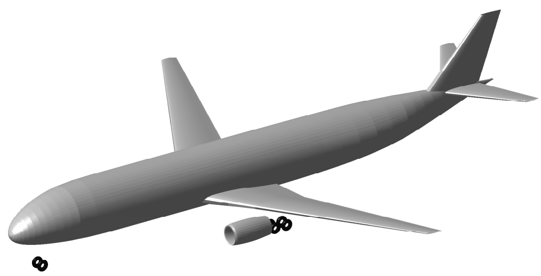


Figure 6.4: Airbus A320 model

First a sensitivity analysis is performed using the design variables mentioned in Table 6.1. In total 52 runs had to be performed, which took 62 minutes to complete. The resulting mean and standard deviation of each parameter are shown in Figure 6.5a and the sensitivity index is given in Figure 6.5b. The variables are numbered according to their importance.

(a) Mean vs. standard deviation (b) Sensitivity index

Figure 6.5: Airbus A320 sensitivity analysis results

In the graphs it can be seen that two parameters stand out, which are the wing position and the sweep angle. The other variables are grouped in the left bottom corner. The section twist angles have the least impact on the payload-range efficiency. The top 5 design variables are selected for optimization.

The optimizer is run for the genetic, gradient-based and hybrid algorithm. The resulting optimum design vector of each algorithm is shown in Table 6.4. A detailed overview of the changes in geometry and performance with respect to the initial design are given in Table 6.5.

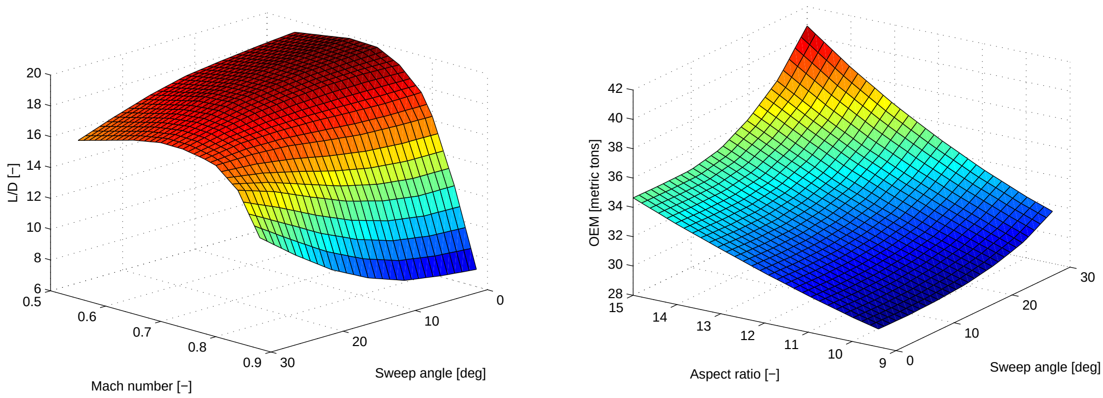
	Æ	f_{x_w}	Λ	$(\frac{t}{c})_t$	Γ	PRE	Δ_{PRE}	t
Initiator	9.4	0.45	26.2°	0.104	6.0°	7880 km	–	–
Genetic	11.6	0.53	0.1°	0.081	1.9°	8240 km	+4.6%	108 min
Gradient	14.4	0.48	6.2°	0.078	6.0°	8250 km	+4.6%	101 min
Hybrid	13.1	0.45	14.5°	0.079	1.5°	8400 km	+6.6%	169 min

Table 6.4: Optimum design vectors for the Airbus A320

In Table 6.4 it can be seen that the hybrid algorithm obtained the best payload-range efficiency. It improved by 6.6% with respect to the reference aircraft, but it also took the most time. The payload-range found by the genetic and gradient algorithm is about 2% less, but both completed within 2 hours. They all agree on a slightly thinner wing tip section, but there are significant differences when comparing the other design parameters.

Looking at the genetically optimized A320 in Figure 6.9 the very low sweep angle of the wing immediately stands out. At a cruise speed of Mach 0.78 one expects a higher sweep angle to reduce the drag rise due to compressibility effects. Therefore it seems that the drag is underestimated. In Figure 6.6a it can be seen that up to Mach 0.80 a low sweep angle is beneficial for the lift-to-drag ratio. The optimizer takes advantage of this by trading sweep angle for a higher aspect ratio. From Figure 6.6b follows that at lower sweep angles the aspect ratio has less impact on the operational empty mass.

Though the structural weight increases at higher aspect ratios, it is compensated by a better lift-to-drag ratio due to lower induced drag. This in turn benefits the payload-range efficiency.

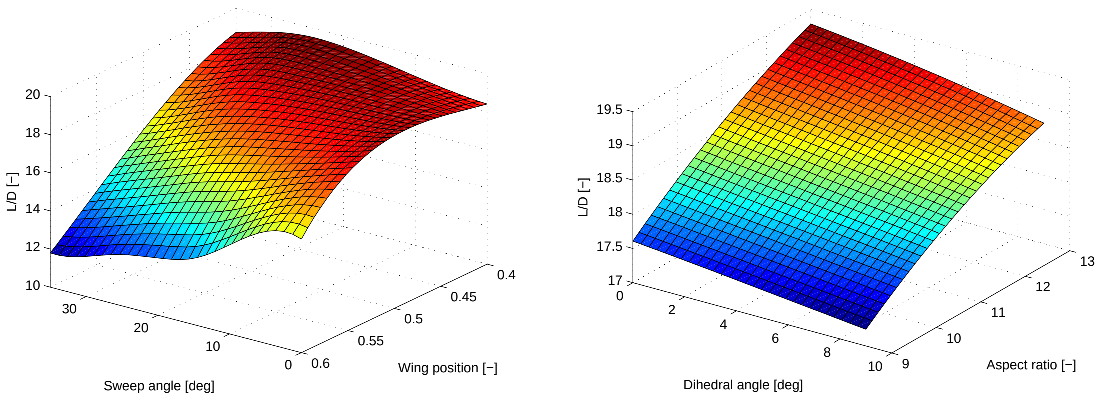


(a) Lift-over-drag ratio for varying sweep angle and Mach number ($A\!\!R = 9.4$)

(b) Operational empty mass for varying aspect ratio and sweep angle

Figure 6.6: Airbus A320 lift-to-drag ratio and operational empty mass

The wing position of the genetic solution is a bit more aft than the other designs. The very low sweep angle causes the wing to shift a bit aft. This effect is shown in Figure 6.7a. There is a strong correlation between the sweep angle, wing position and lift-to-drag ratio. With increasing sweep angle the wing has to shift forward to attain a better L/D, but nose loading constraints limit this movement.



(a) Lift-over-drag ratio for varying sweep angle and wing position ($A\!\!R = 9.4$)

(b) Lift-over-drag ratio for varying dihedral angle and aspect ratio

Figure 6.7: Airbus A320 lift-to-drag ratio for varying parameters

The hybrid algorithm improved on the genetic solution by a few percent. Though it started at the optimum design vector of the genetic algorithm, it came up with a rather different combination of aspect ratio, sweep angle and wing position. The resulting aircraft is shown in Figure 6.11. The aspect ratio increased from 11.6 to 13.1 and the sweep angle increased to 14.5°. Up to about 15° sweep the aspect ratio weight penalty remains roughly the same when observing Figure 6.6b. In combination with the more forward wing position a better optimum was found with these parameters.

Both the genetic and hybrid solution have a lower wing dihedral angle. A lower dihedral angle yields a higher effective planform area, which leads to a slight increase in lift [23]. This is also in accordance with Figure 6.7b. The dihedral angle largely depends on the trade-off between lateral stability and roll control. Especially low-wing aircraft like the A320 require some dihedral due to the wing-fuselage interaction, which is usually in the range of 5° to 7° [27]. The dihedral angle is also constraint by the engine ground clearance requirement and tip clearance during take-off rotation and landing. The Initiator does not yet cover lateral stability and therefore the dihedral is entirely driven by the lift-to-drag ratio.

The gradient-based solution resulted in the heaviest aircraft. It is depicted in Figure 6.10. The operational empty weight increased by almost 12%. This can be mainly attributed to wing as can be observed in Figure 6.8. The aspect ratio of 14.4 allowed it to reach a lift-to-drag ratio of 20.3 at the cost of a significant increase in wing weight. Therefore it does not outperform the genetic solution with an L/D of 19.1.

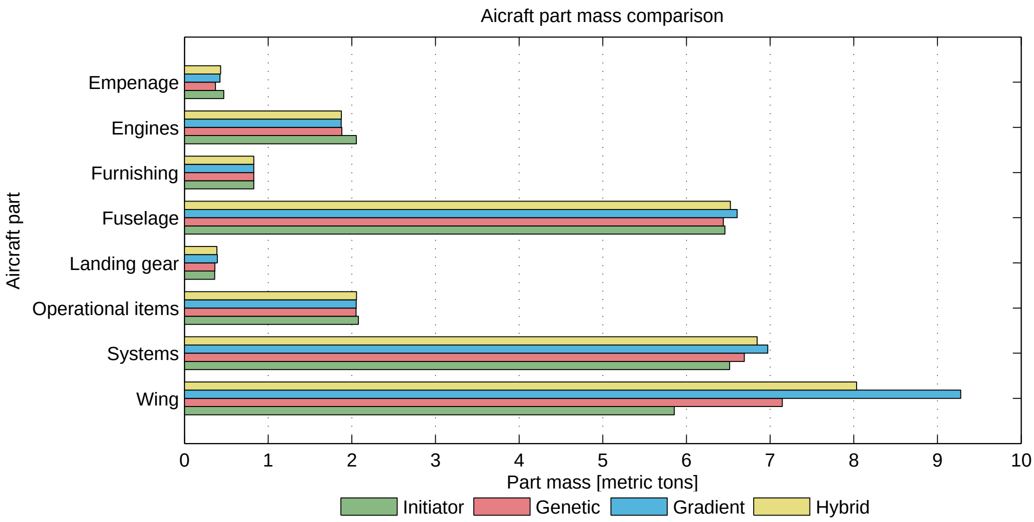


Figure 6.8: Airbus A320 change in part mass after optimization

When looking at the tail surfaces of each design it can be observed that their sweep and dihedral angles are coupled to the main wing. The sweep angle alters the lift curve slope, which has consequences for the stall angle of attack and maximum lift coefficient. Therefore the shape of the tail surfaces are likely to be far from optimal with respect to the control and stability of the aircraft. Also, the high trailing edge sweep angle of the genetic and hybrid solution create an unfavourable condition for the placement of elevator and rudder control surfaces.

All solutions show a rather high static margin. This means that the center of gravity of the aircraft is relatively far ahead of the neutral point. A lower static margin allows a reduction in tail size and requires a lower download from the tail. A smaller tail could reduce the static margin. The design tool does not use the class II design information to update the tail size, so this may be a point of improvement. Another option is to move the wing more aft. However, the constraint on minimum nose loading limits its position.

	Parameter	Description	Unit	Initiator	Genetic	Gradient	Hybrid
Key performance indicators	PRE	Payload-range efficiency	km	7880	8240	8240	8400
	X	Range parameter	km	7060	7630	8140	8050
	$R_{\mathtt{h}}$	Harmonic range	km	2900	2890	2890	2890
	$R_{\mathtt{max.fuel}}$	Max. fuel range	km	9560	6440	6710	7420
	MTOM	Maximum take-off mass	tons	58.9	59.3	62.2	60.5
	OEM	Operational empty mass	tons	30.8	31.6	34.5	32.9
	FM	Fuel mass	tons	7.6	7.2	7.2	7.0
	L/D	Lift-to-drag ratio	–	17.6	19.1	20.3	20.1
	$C_{L_{\mathtt{max,clean}}}$	Max. lift coefficient	–	1.24	1.39	1.37	1.30
	$V_{s_{\mathtt{clean}}}$	Stall speed clean	m/s	77.7	73.2	73.7	75.5
	W/S	Wing loading	kg/m^2	466	465	464	464
	T/W	Thrust loading	–	0.27	0.24	0.23	0.23
	SM	Static margin	%MAC	68	69	59	56
	c.g. travel	Center of gravity travel	%MAC	34	37	37	38
Wing	AR	Aspect ratio	–	9.4	11.6	14.4	13.1
	S	Planform area	m^2	126	127	134	130
	b	Span	m	34.5	38.5	43.9	41.4
	c_r	Root chord	m	7.6	4.1	4.6	5.7
	MAC	Mean aerodynamic chord	m	4.5	3.5	3.3	3.6
	λ	Taper ratio	–	0.16	0.45	0.36	0.26
	Λ	Sweep angle	°	26.2	0.1	6.2	14.5
	Γ	Dihedral angle	°	6.0	1.9	6.0	1.5
	$f_{x_{\mathtt{w}}}$	Wing position fraction	–	0.45	0.53	0.48	0.45
Fuselage	$l_{\mathtt{f}}$	Fuselage length	m	40.6	40.6	40.6	40.6
	$d_{\mathtt{f}}$	Fuselage width	m	4.2	4.2	4.2	4.2
	$\lambda_{\mathtt{f}}$	Fineness ratio	–	9.6	9.6	9.6	9.6
HT	$AR_{\mathtt{h}}$	Aspect ratio	–	4.9	4.9	4.8	4.9
	$S_{\mathtt{h}}$	Planform area	m^2	25	20	20	20
	$b_{\mathtt{h}}$	Span	m	11.1	9.9	9.8	10.0
	$c_{r_{\mathtt{h}}}$	Root chord	m	3.3	2.9	2.9	3.0
	$\Lambda_{\mathtt{h}}$	Sweep angle	°	29.3	0.1	6.9	16.2
	$\Gamma_{\mathtt{h}}$	Dihedral angle	°	6.0	1.9	6.0	1.5
VT	$AR_{\mathtt{v}}$	Aspect ratio	–	1.6	1.6	1.6	1.6
	$S_{\mathtt{v}}$	Planform area	m^2	18	20	24	21
	$b_{\mathtt{v}}$	Span	m	5.2	5.6	6.1	5.8
	$c_{r_{\mathtt{v}}}$	Root chord	m	4.8	5.2	5.6	5.3
	$\Lambda_{\mathtt{v}}$	Sweep angle	°	39.2	0.1	9.3	21.7

Table 6.5: Airbus A320 optimization results

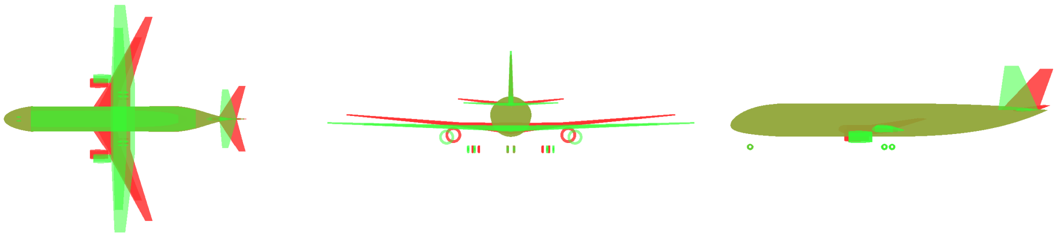
The results of this optimization case can also be viewed with respect to the actual Airbus A320-200. Its specifications are given in Table 6.6. The aircraft has a harmonic range of 2870 km and a maximum payload of 20.5 tons, which is the same as the top-level requirements of this case. At maximum payload the fuel mass is 12.5 tons, which yields a payload-range efficiency of approximately 4700 km. This value is significantly lower than the Initiator reference aircraft and the three optimizations. This stems from the lower estimated aircraft mass and the underestimated drag penalty due to the onset of compressibility effects. Both affect the lift-to-drag ratio, which plays a major role in this key performance indicator.

AR	9.5	–	l_f	37.6	m	AR_v	1.8	–	MTOM	73.5	tons
S	122	m^2	d_f	4.1	m	S_v	22	m^2	OEM	39.7	tons
b	34.1	m	λ_f	9.1	–	b_v	6.3	m	PM	20.5	tons
Λ	25.0	°	AR_h	5	–	Λ_v	35	°	R_h	2870	km
c_r	6.1	m	S_h	31	m^2	W/S	600	kg/m^2	PRE	4710	km
λ	0.24	–	b_h	12.5	m	T/W	0.31	–			
Γ	5.0	°	Λ_h	28	°						

Table 6.6: Airbus A320-200 specifications [21, 22]

The geometry of the A320-200 has most resemblance with the Initiator reference design. For all obtained designs the horizontal tail planform area is considerably smaller than the actual A320. This also indicates that the sizing routine of the tail surfaces needs further investigation.

It can be concluded that with the conventional A320 only relatively small improvements can be found with respect to the reference aircraft. It seems that the drag rise is underestimated, which followed from analyzing the optimizations and comparing the results with the actual A320-200. The beneficial weight effect of lower sweep angles outweighs the drag penalty. Also, the sizing method of the tail surfaces could use some improvement. The sizing should be based on stability and control requirements, rather than only using the main wing as reference. Purely looking at the objective value the hybrid algorithm found the best aircraft. Its computational time is a bit higher, but in this case it can be justified.

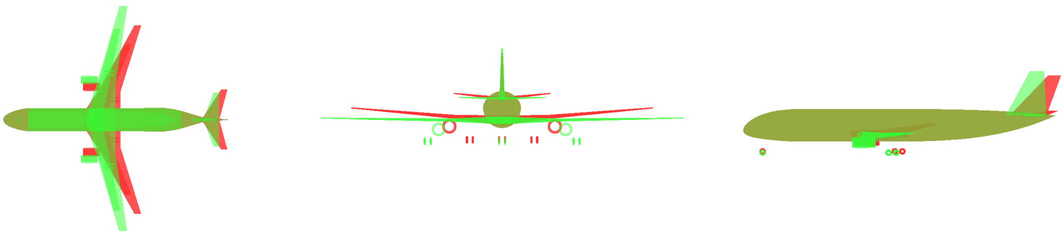


(a) Top view (b) Front view (c) Side view

Figure 6.9: Airbus A320 geometry after genetic optimization

(a) Top view (b) Front view (c) Side view

Figure 6.10: Airbus A320 geometry after gradient-based optimization



(a) Top view (b) Front view (c) Side view

Figure 6.11: Airbus A320 geometry after hybrid optimization

6.4 Case 2: Canard aircraft

The canard aircraft shares its top level requirements with the Airbus A320, which are repeated in Table 6.7. It has a forward swept wing and a canard instead of a horizontal tail. The engines are mounted to the rear of fuselage.

Pax.	Payload mass	M_C	Altitude	Range	s_{TO}	s_L
150	20.5 tons	0.78	11.3 km	2870 km	2180 m	1440 m

Table 6.7: Canard aircraft top level requirements

For this aircraft the control-canard is used. The primary role of a control-canard is to provide longitudinal control for the aircraft. The other variant is the lifting-canard, which also carries part of the lift during normal flight. This type of canard usually has a higher aspect ratio to reduce its lift-induced drag.
The canard generates an upward force to control the aircraft, while a horizontal tail produces negative lift that must be compensated by the wing. This seems to make the canard configuration the better choice due to the improved lift capability. However, the downwash of the canard affects the airflow over the main wing which may worsen its aerodynamic performance. Also, the canard must always stall first to ensure that the aircraft pitches down during such event. Therefore the main wing can never reach its maximum lift coefficient.

The forward swept wing has some advantages over an aft swept wing. It generally requires a lower leading edge sweep angle to cope with the compressibility effects at high Mach numbers. The aerodynamics model is not capable of fully computing these effects [11], so this will not be reflected in the results. The downside is that the structure must be rigid enough to withstand bending and torsion, especially at high sweep angles. This may lead to a serious weight penalty. In addition, aeroelasticity effects can be problematic as the tip may have flutter tendencies. The design tool does not evaluate the aeroelasticity, so these effects are not taken into account.

Using the aforementioned requirements the canard aircraft is generated using the Initiator. The resulting aircraft properties and performance figures are listed in Table 6.8. A 3-dimensional representation of the model is shown in Figure 6.12.

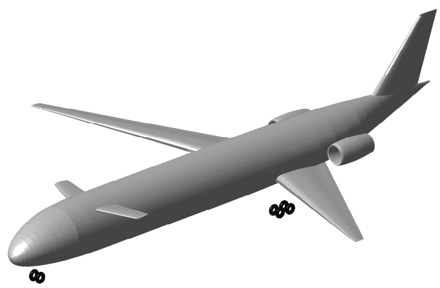


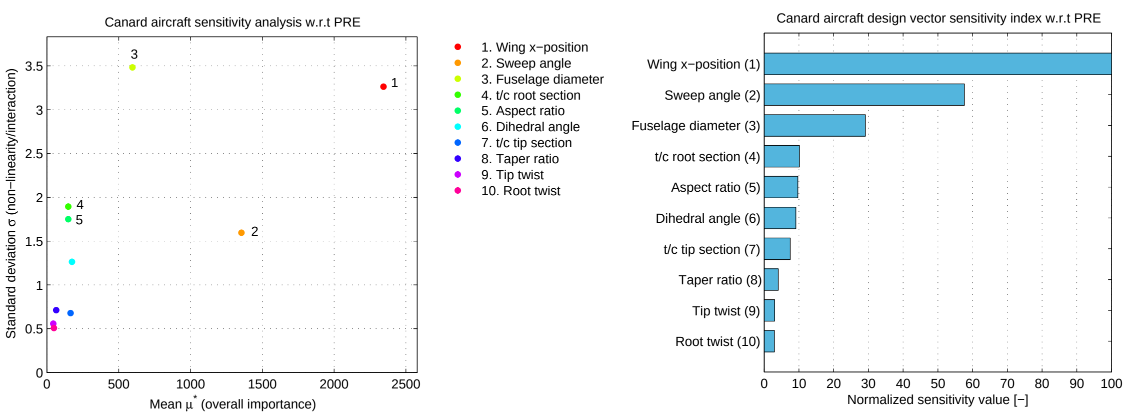
Figure 6.12: Canard aircraft model

$A\!\!R$	9.4	–	Γ	6.0	°	$A\!\!R_{\mathtt{v}}$	1.6	–	MTOM	57.1	tons
S	121	m^2	$f_{x_{\mathtt{w}}}$	0.60	–	$S_{\mathtt{v}}$	17	m^2	OEM	28.5	tons
b	33.7	m	$(\frac{t}{c})_{\mathtt{r}}$	0.151	–	$b_{\mathtt{v}}$	5.3	m	FM	8.0	tons
Λ	-26.2	°	$(\frac{t}{c})_{\mathtt{t}}$	0.104	–	$A\!\!R_{\mathtt{c}}$	5.1	–	$R_{\mathtt{h}}$	2900	km
c_r	5.4	m	$l_{\mathtt{f}}$	40.6	m	$S_{\mathtt{c}}$	14	m^2	PRE	7420	km
λ	0.16	–	$d_{\mathtt{f}}$	4.2	m	$b_{\mathtt{c}}$	8.5	m	L/D	15.6	–
ϵ	0	°							$C_{L_{\mathtt{max,clean}}}$	1.84	–

Table 6.8: Canard aircraft properties using the Initiator

Next, a sensitivity analysis is conducted to reduce the number of design variables. Since the wing is swept forward there is no kink section. So from the design variables listed Table 6.1 the thickness ratio and twist angle at the kink are left out. Furthermore the sweep angle boundaries are inversed, giving it a lower bound of $-35°$ and an upper bound of $0°$. In total 10 design variables are sampled, requiring 44 analysis runs. The sensitivity analysis took 54 minutes to complete.

The results of the sensitivity analysis are shown in Figure 6.13. The top 5 variables are indicated in the graph. It can be seen that the wing position, sweep angle and fuselage diameter have the most influence on the objective when observing their μ^* and σ values. This is also reflected in the sensitivity index in Figure 6.13b.



(a) Mean vs. standard deviation (b) Sensitivity index

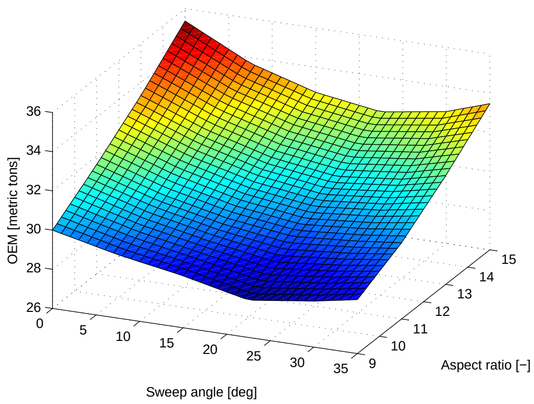
Figure 6.13: Canard aircraft sensitivity analysis results

Using all three optimization algorithms the results shown in Table 6.9 and 6.10 are obtained. The latter gives a more extensive overview the aircraft properties. It can be seen that the payload-range efficiency has been greatly improved, especially for the genetic and hybrid algorithm. The gradient algorithm performed the worst as it attained the lowest payload-range efficiency using the same amount of time as the genetic algorithm.

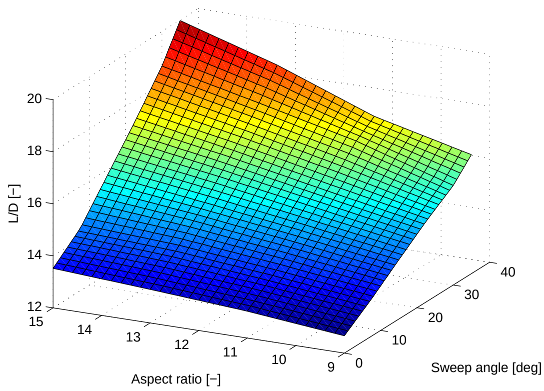
	$A\!\!\!R$	f_{x_w}	Λ	$(\frac{t}{c})_r$	d_f	PRE	Δ_{PRE}	t
Initiator	9.4	0.60	$-26.2°$	0.151	4.2 m	7420 km	–	–
Genetic	10.0	0.56	$-19.2°$	0.118	4.3 m	8590 km	+15.8%	124 min
Gradient	10.1	0.59	$-31.9°$	0.119	4.8 m	8260 km	+11.3%	129 min
Hybrid	12.3	0.59	$-23.4°$	0.116	4.7 m	8850 km	+19.3%	170 min

Table 6.9: Optimum design vectors for the canard aircraft

All algorithms remain close to the initial wing position of 0.60. In contrast to the aft-swept A320 where the sweep was drastically reduced among all optimizations, the algorithms maintained a higher sweep angle with a forward swept wing in canard configuration, which is curious. The relation between the sweep angle, aspect ratio and operational empty mass becomes clear when viewing Figure 6.14a. From this graph follows that a moderate sweep angle of 20° to 25° results in the lowest aircraft weight. The wing mass seems to be underestimated at high aspect ratios and sweep angles as their is no severe increase in structural mass required to resists the large bending stresses of a forward swept wing.



(a) Operational empty mass for varying aspect ratio and sweep angle



(b) Lift-over-drag ratio for varying aspect ratio and sweep angle

Figure 6.14: Effect of aspect ratio and sweep angle on the canard aircraft

The gain in lift-to-drag ratio with respect to aspect ratio and sweep angle is depicted in Figure 6.14b. From this graph can be observed that their relation with lift-to-drag ratio is much stronger than witnessed in the A320 case. Also, with increasing sweep angle the effect on the lift-to-drag ratio becomes more pronounced. Results showed that this effect mainly stems from a decrease in drag from the vortex-lattice based AVLVLM module. This contradicts the findings with the aft-swept wing of the A320. Therefore further investigation is required in the aerodynamics routines.
As such, the algorithms tried to find a compromise between a higher lift-to-drag by increasing the sweep angle and aspect ratio, while keeping the weight increase to a minimum such that the fuel consumption is kept as low as possible.

Looking at Figure 6.15 there are significant differences in the aircraft part masses between the four designs. The heaviest design follows from the hybrid solution, which has an operational empty mass of 31.1 metric tons. The largest contributor is the main wing as it gained over 3 tons in mass. Due to its relatively large aspect ratio a heavier structure is required.

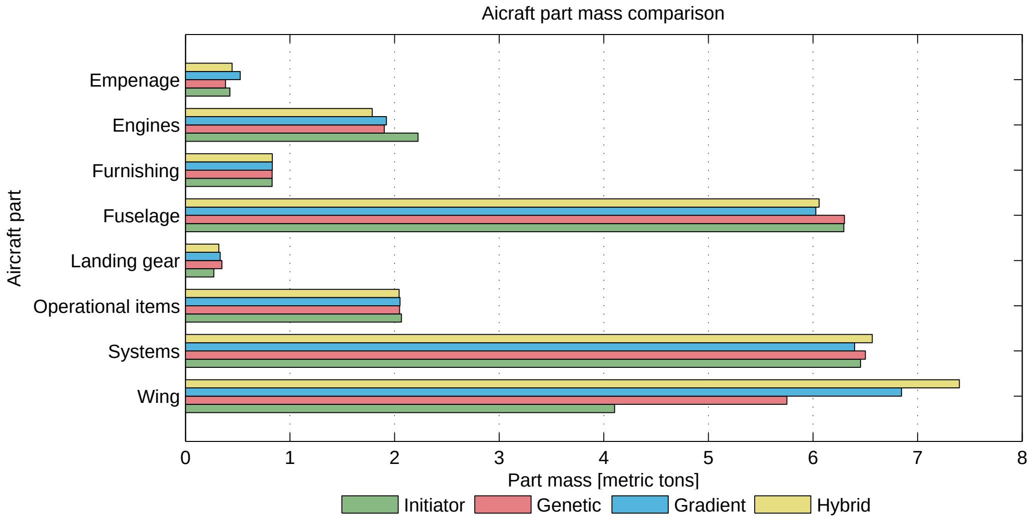


Figure 6.15: Canard aircraft change in part mass after optimization

The gradient and hybrid algorithm tried to save weight on the fuselage by decreasing its fineness ratio. This is visualized in Figure 6.18 and 6.19. The genetic algorithm maintained roughly the same ratio (Figure 6.17). Due to the shorter moment arm a slight increase in empenage weight is observed. The improved aerodynamic efficiency allowed for a better thrust-to-weight ratio, thereby saving on engine weight.

The three optimization solutions all show a negative static margin. For a canard aircraft to be longitudinally stable the static margin must be positive. In order to get a positive static margin the wing or canard can be moved aft, or the canard size can be decreased for example. The static margin as a function of sweep angle and wing position is given in Figure 6.16. Here it can be seen that the margin becomes less with increasing sweep and a more forward wing position. So for the moderate sweep angles of the optimized designs a wing position fraction of around 0.65 is required to reach the feasible static margin region. However, at such aft position the minimum nose loading constraint is violated.
The location of the canard is fixed, so the option that remains is reducing its size. After investigating the sizing method of the canard it followed that it is linearly scaled with the main wing's planform area, MAC and position from the class I design methods. To obtain a better static margin the information from the class II methods should be used to adjust the canard size.

Like in the A320 case, the same tail–wing sizing relation is found. The vertical tail sweep is heavily affected by the main wing sweep. Also, the sweep angle of the canard is sized according to the main wing. As mentioned earlier, the control surface sizing should be improvement in order to meet stability and control requirements instead of depending on empirical geometric functions.

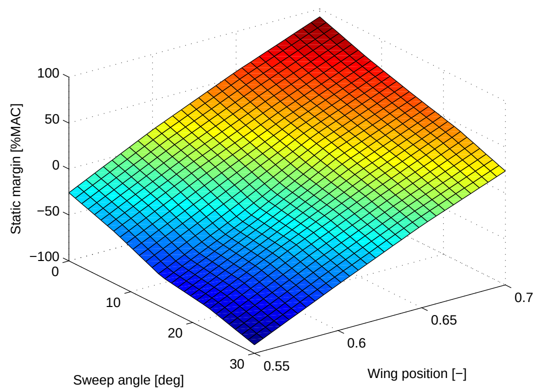


Figure 6.16: Canard aircraft static margin vs. sweep angle and wing position ($AR = 9.4$)

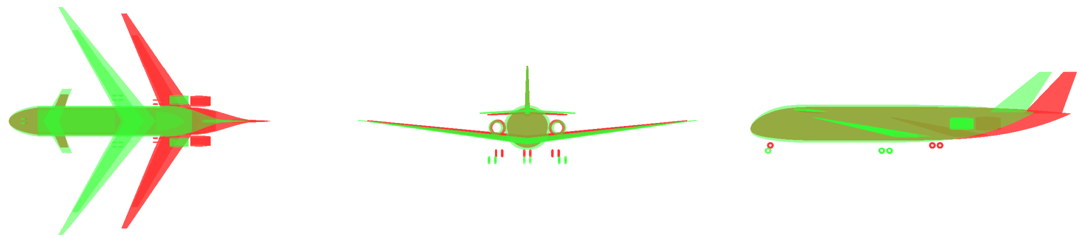
From the optimization of the canard aircraft it can be concluded that a large improvement in objective value can be obtained. A slightly higher wing aspect ratio and lower sweep angle resulted in a nearly 20% higher payload-range efficiency, at the cost of a 5% to 10% heavier aircraft. Some weight savings are achieved with the engine and fuselage. Improvements with respect to the static margin could be done by feeding class II design information back into the canard sizing. Compared to the A320, the baseline canard aircraft performed worse than the conventional aircraft, but after optimization the canard configuration obtained a superior payload-range efficiency. Lastly, it must be noted that some characteristics of the forward swept wing, as mentioned earlier in the case description, could not been taken into account as it is not covered in the design tool.

	Parameter	Description	Unit	Initiator	Genetic	Gradient	Hybrid
Key performance indicators	PRE	Payload-range efficiency	km	7420	8590	8260	8850
	X	Range parameter	km	6250	7700	7520	8240
	$R_{\mathtt{h}}$	Harmonic range	km	2900	2900	2890	2890
	$R_{\mathtt{max.fuel}}$	Max. fuel range	km	7310	6470	6670	6230
	MTOM	Maximum take-off mass	tons	57.1	57.4	58.7	58.4
	OEM	Operational empty mass	tons	28.5	29.9	31.0	31.1
	FM	Fuel mass	tons	8.0	6.9	7.2	6.7
	L/D	Lift-to-drag ratio	–	15.6	19.2	18.8	20.6
	$C_{L_{\mathtt{max,clean}}}$	Max. lift coefficient	–	1.84	1.22	0.77	1.25
	$V_{s_{\mathtt{clean}}}$	Stall speed clean	m/s	64.1	78.1	98.4	77.0
	W/S	Wing loading	kg/m^2	471	464	465	462
	T/W	Thrust loading	–	0.30	0.25	0.25	0.23
	SM	Static margin	%MAC	33	-57	-76	-47
	c.g. travel	Center of gravity travel	%MAC	67	53	55	43
Wing	$Æ$	Aspect ratio	–	9.4	10.0	10.1	12.3
	S	Planform area	m^2	121	124	126	126
	b	Span	m	33.7	35.2	35.7	39.4
	c_r	Root chord	m	5.4	5.1	5.4	4.8
	MAC	Mean aerodynamic chord	m	3.7	3.6	3.7	3.3
	λ	Taper ratio	–	0.16	0.22	0.13	0.18
	Λ	Sweep angle	°	-26.2	-19.2	-31.9	-23.4
	Γ	Dihedral angle	°	6.0	6.0	6.0	6.0
	$f_{x_{\mathtt{w}}}$	Wing position fraction	–	0.60	0.56	0.59	0.59
Fuselage	$l_{\mathtt{f}}$	Fuselage length	m	40.6	40.2	35.9	36.3
	$d_{\mathtt{f}}$	Fuselage width	m	4.2	4.3	4.8	4.7
	$\lambda_{\mathtt{f}}$	Fineness ratio	–	9.6	9.5	7.5	7.7
Canard	$Æ_{\mathtt{c}}$	Aspect ratio	–	5.1	5.1	5.1	5.1
	$S_{\mathtt{c}}$	Planform area	m^2	14	18	20	18
	$b_{\mathtt{c}}$	Span	m	8.5	9.7	10.1	9.6
	$c_{r_{\mathtt{c}}}$	Root chord	m	2.1	2.4	2.5	2.4
	$\Lambda_{\mathtt{c}}$	Sweep angle	°	23.5	17.3	28.7	21.0
	$\Gamma_{\mathtt{c}}$	Dihedral angle	°	-3.0	-3.0	-3.0	-3.0
VT	$Æ_{\mathtt{v}}$	Aspect ratio	–	1.6	1.6	1.6	1.6
	$S_{\mathtt{v}}$	Planform area	m^2	17	15	16	19
	$b_{\mathtt{v}}$	Span	m	5.3	4.9	5.1	5.5
	$c_{r_{\mathtt{v}}}$	Root chord	m	4.9	4.5	4.7	5.1
	$\Lambda_{\mathtt{v}}$	Sweep angle	°	39.2	28.8	47.8	35.1

Table 6.10: Canard aircraft optimization results

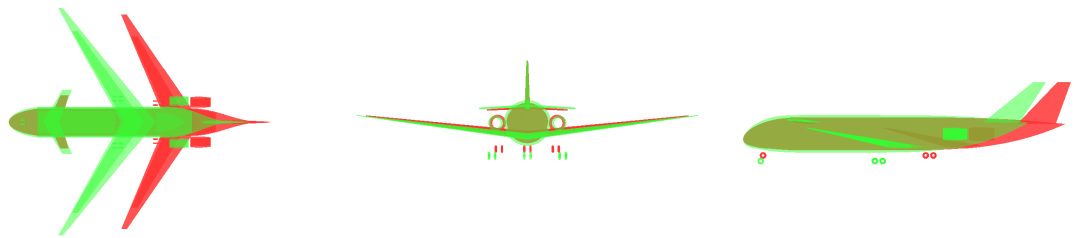
(a) Top view (b) Front view (c) Side view

Figure 6.17: Canard aircraft geometry after genetic optimization



(a) Top view (b) Front view (c) Side view

Figure 6.18: Canard aircraft geometry after gradient optimization



(a) Top view (b) Front view (c) Side view

Figure 6.19: Canard aircraft geometry after hybrid optimization

6.5 Case 3: Three-surface aircraft

The three-surface aircraft features three horizontal surfaces: a canard, main wing and horizontal tail. A well-known example is the Piaggio P.180 Avanti, which achieved lower weight and drag thanks to its three-surface configuration [3].
Traditional aircraft with only a horizontal tail rely on the tailplane to balance and control the aircraft. The tailplane provides a negative lift to counteract the moment due to the lift of the wing which. This in turn must be compensated by additional lift of the wing. By adding a canard the required counteracting moment can be shared with the horizontal tail. Because the canard provides an upward force, the wing loading becomes lower and therefore the wing size can be reduced. A schematic overview of the equilibrium of a three-surface aircraft is shown in Figure 6.20.

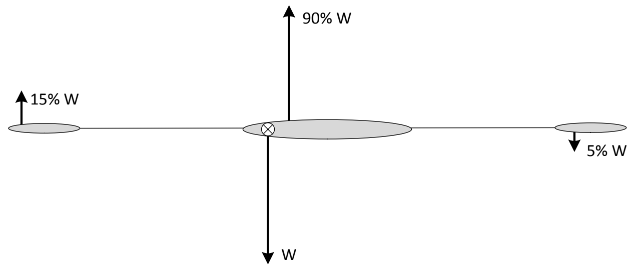


Figure 6.20: Three-surface equilibrium

The three-surface aircraft has similar requirements as the A320 and the canard aircraft. It has a high-wing configuration, low canard and a T-tail, such that the surfaces are not in each other's wake. The requirements are shown in Table 6.11.

Pax.	Payload mass	M_{C}	Altitude	Range	s_{TO}	s_{L}
150	20.5 tons	0.78	11.3 km	2870 km	2180 m	1440 m

Table 6.11: Three-surface aircraft top level requirements

From these requirements an aircraft is generated using the Initiator. The resulting design is depicted in Figure 6.21. The aircraft properties are listed in Table 6.12.

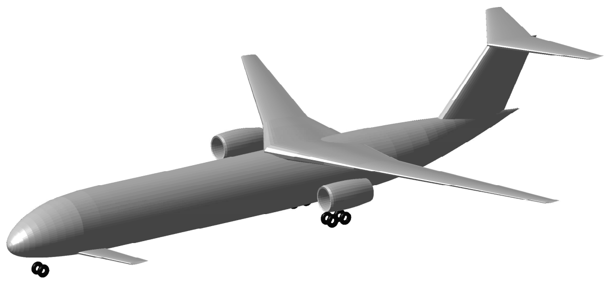
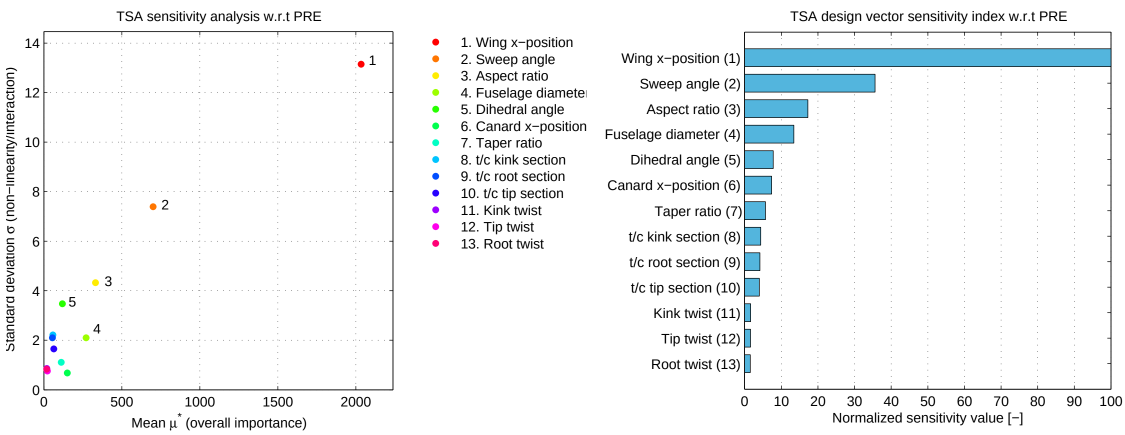


Figure 6.21: Three-surface aircraft model

$A\!\!R$	9.4	–	$f_{x_\mathtt{w}}$	0.60	–	$S_\mathtt{c}$	19	m^2	MTOM	64.5	tons
S	136	m^2	$(\frac{t}{c})_\mathtt{r}$	0.151	–	$b_\mathtt{c}$	9.7	m	OEM	34.5	tons
b	35.8	m	$(\frac{t}{c})_\mathtt{k}$	0.104	–	$A\!\!R_\mathtt{h}$	4.9	–	FM	9.5	tons
Λ	26.2	°	$(\frac{t}{c})_\mathtt{t}$	0.104	–	$S_\mathtt{h}$	30	m^2	$R_\mathtt{h}$	2900	km
c_r	7.1	m	$l_\mathtt{f}$	40.6	m	$b_\mathtt{h}$	12.1	m	PRE	6270	km
λ	0.16	–	$d_\mathtt{f}$	4.2	m	$A\!\!R_\mathtt{v}$	1.0	–	L/D	14.7	–
ϵ	0	°	$f_{x_\mathtt{c}}$	0.10	–	$S_\mathtt{v}$	24	m^2	$C_{L_{\mathtt{max,clean}}}$	1.02	–
Γ	0.0	°	$A\!\!R_\mathtt{c}$	4.9	–	$b_\mathtt{v}$	4.8	m			

Table 6.12: Three-surface aircraft properties using the Initiator

The sensitivity analysis was done using all 13 design variables from Table 6.1. It took 89 minutes to perform all 56 runs. The results from the sensitivity analysis are shown in Figure 6.22a in which the top 5 variables are indicated. Using the mean and standard deviation of the parameters the sensitivity index is composed. This index is given in Figure 6.22b.



(a) Mean vs. standard deviation (b) Sensitivity index

Figure 6.22: Three-surface aircraft sensitivity analysis results

It can be observed that the wing position is by far the most influential parameter. The objective is also very sensitive to the sweep angle. The top 5 is concluded by the aspect ratio, fuselage diameter and dihedral angle. The longitudinal position of the canard did not make it to the selection. It was a near tie with the dihedral angle. Apparently, canard sizing benefits due to better positioning with respect to the wing and tail surfaces does not change the payload-range efficiency very much.

The selected design variables are used for optimization of which the results are displayed in Table 6.13. A more detailed overview of the aircraft properties is gathered in Table 6.14. For the gradient optimized three-surface aircraft also an example report is generated by the Initiator. This report is shown in Appendix A.

	AR	f_{x_w}	Λ	Γ	d_f	PRE	Δ_{PRE}	t
Initiator	9.4	0.60	26.2°	0.3°	4.2 m	6270 km	–	–
Genetic	11.4	0.50	11.1°	−0.7°	4.1 m	7550 km	+20.4%	114 min
Gradient	11.7	0.47	12.9°	−3.6°	4.7 m	7570 km	+20.7%	152 min
Hybrid	11.5	0.50	11.5°	−0.8°	4.1 m	7550 km	+20.4%	172 min

Table 6.13: Optimum design parameters for the three-surface aircraft

A large gain in payload-range efficiency is obtained through optimization. The best objective value is achieved by the gradient-based algorithm, but the other two algorithms are not far behind. The initial value of 6270 is increased by roughly 20% for all algorithms. When the computation time is taken into account, it can be said that the genetic algorithm performed best. Looking at the design vectors there is a trend towards a slightly higher aspect ratio and a more forward wing position. These notable differences in geometry are clearly visible in the top views of Figure 6.26, 6.27 and 6.28. There are also some notable differences with respect to the fuselage diameter and sweep angle.

The weight decreased for all solutions with respect to the baseline version. Weight savings were mainly achieved by smaller engines and lighter wing structures as can be seen in Figure 6.23.

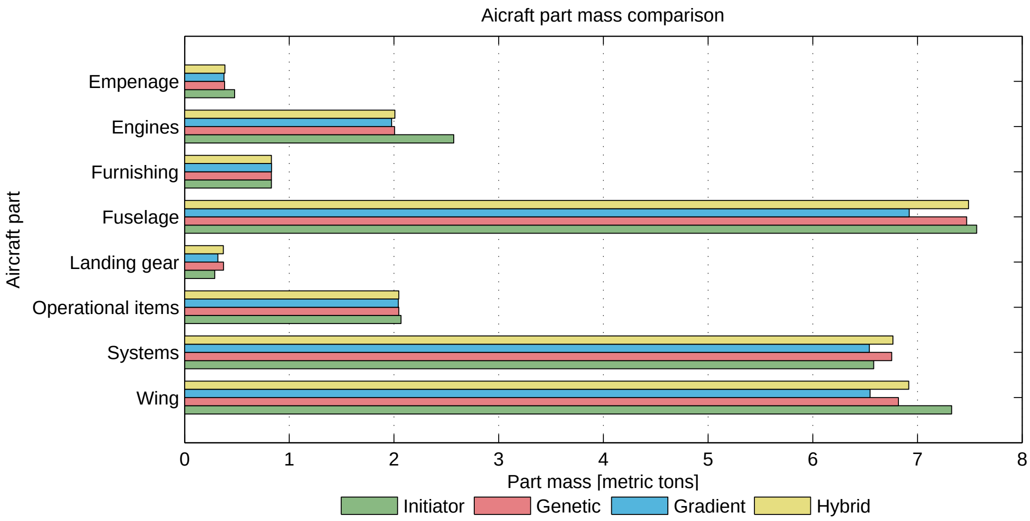
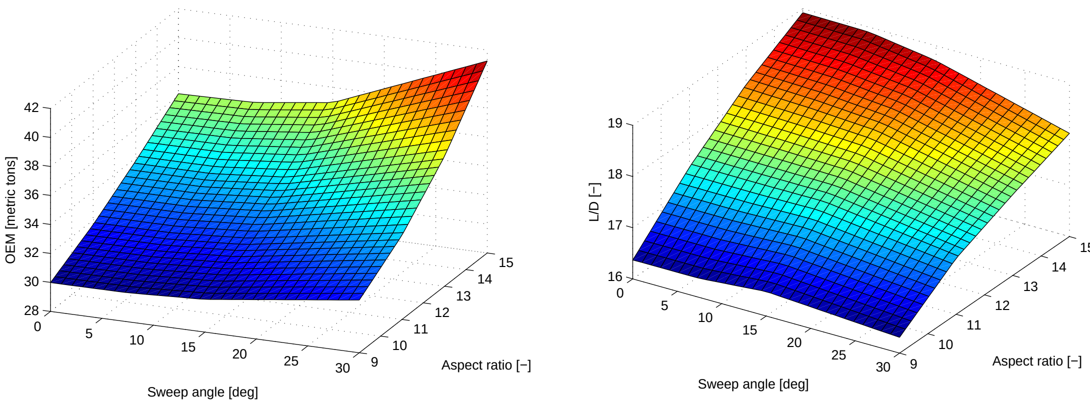


Figure 6.23: Three-surface aircraft change in part mass after optimization

Most weight was saved by the gradient-based algorithm. This follows from its low fuselage fineness ratio. The difference with respect to the baseline geometry can be clearly noticed in Figure 6.27. A shorter fuselage has less bending stresses and therefore the structure can be lighter. This is also reflected in the system components mass. The high-wing configuration resulted in a high fuselage mass when compared to the low-wing aircraft from the first two cases.

The weight of the wing is largely influenced by the sweep angle and aspect ratio. At large sweep angles this effect becomes more pronounced. This is shown in Figure 6.24a. The lift-to-drag ratio, which plays an important role in the payload-range efficiency, benefits from a larger aspect ratio. This is depicted in Figure 6.24b. At larger sweep angles the L/D decreases a bit, which is likely caused by an increase in lift-dependent drag. So a trade-off arises between weight and aerodynamic efficiency.

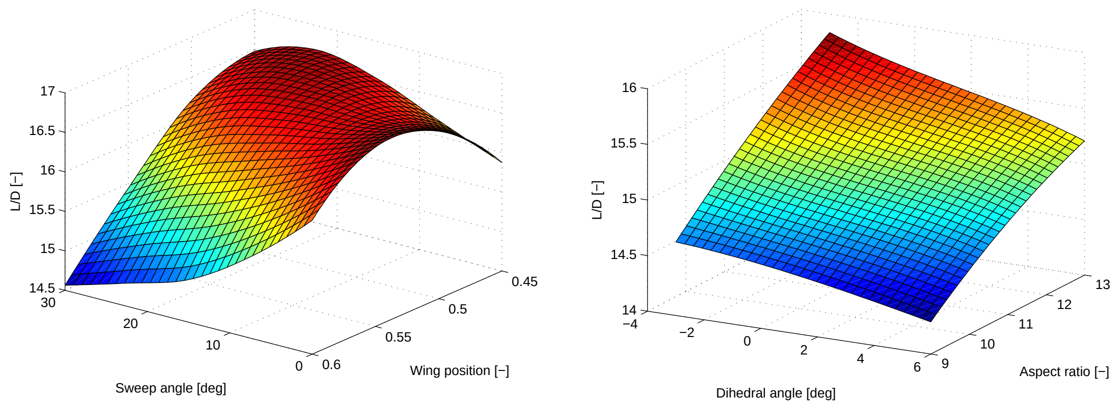


(a) Operational empty mass vs. aspect ratio and sweep angle

(b) Lift-over-drag ratio vs. aspect ratio and sweep angle

Figure 6.24: Three-surface aircraft with varying aspect ratio and sweep angle ($f_{x_w} = 0.5$)

Clearly, when observing these graphs an unswept wing would be the best choice. This conclusion does not match with the sweep angle obtained from the optimizations. As can be seen in Figure 6.25a the optimum sweep angle is also dictated by the position of the wing. The optimum sweep angle becomes higher as the wing is located further aft.



(a) Lift-over-drag ratio vs. sweep angle and wing position ($\mathcal{R} = 10$)

(b) Lift-over-drag ratio vs. aspect ratio and dihedral angle

Figure 6.25: Three-surface aircraft lift-to-drag for varying parameters

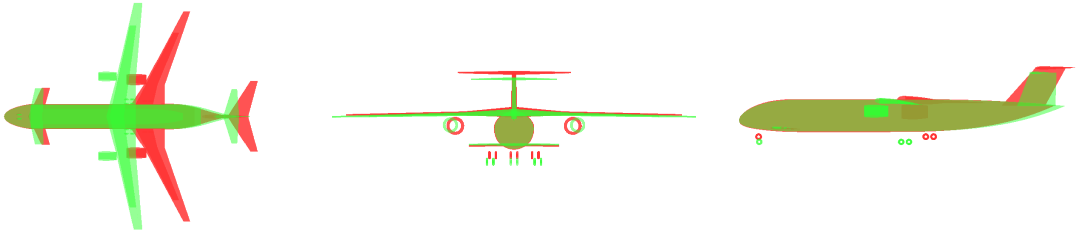
The dihedral angle became lower for all three solutions. As can be seen in Figure 6.25b the L/D improves with decreasing dihedral. A higher aspect ratio slightly enhances this effect. A similar effect has been witnessed in the A320 case. The high-wing nature of the three-surface aircraft makes it more laterally stable with respect to the dihedral effect. Therefore the obtained dihedral angles are not very unrealistic. However, since the Initiator does not compute the lateral stability yet, the lower dihedral is purely driven by the beneficial L/D instead of taking into account the dihedral effect.

Again, the same control surface sizing discrepancies are witnessed. The sweep of the horizontal tail, vertical tail and canard are based on the main wing. The same can be said of the dihedral angle.

From this case it can be concluded that a large improvement in payload-range efficiency can be attained with the three-surface aircraft. The initial wing position of 0.60 is too aft. A better initial guess would be a value 0.50. The optimization resulted in two distinct fuselage designs, but with similar payload-range efficiency. The gradient algorithm found the highest optimum, but the genetic algorithm resulted in the most gain in the shortest amount of time. Comparing to the Airbus A320 and canard aircraft, which have the same top level requirements, the three-surface configuration has the worst payload-range efficiency. Even after the optimization the aircraft is no match for the conventional A320.

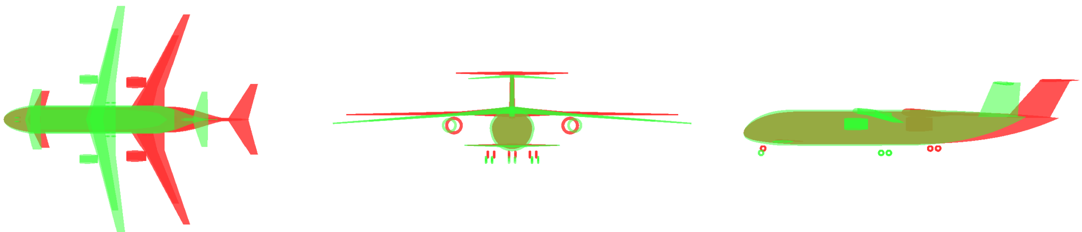
	Parameter	Description	Unit	Initiator	Genetic	Gradient	Hybrid
Key performance indicators	PRE	Payload-range efficiency	km	6270	7550	7570	7550
	X	Range parameter	km	5890	7100	6920	7120
	$R_{\mathtt{h}}$	Harmonic range	km	2900	2900	2890	2900
	$R_{\mathtt{max.fuel}}$	Max. fuel range	km	6170	5450	5050	5470
	MTOM	Maximum take-off mass	tons	64.5	61.1	59.7	61.3
	OEM	Operational empty mass	tons	34.5	32.7	31.3	32.8
	FM	Fuel mass	tons	9.5	7.9	7.9	7.9
	L/D	Lift-to-drag ratio	–	14.7	17.7	17.3	17.8
	$C_{L_{\mathtt{max,clean}}}$	Max. lift coefficient	–	1.02	1.41	1.44	1.40
	$V_{s_{\mathtt{clean}}}$	Stall speed clean	m/s	86.1	72.8	72.2	73.1
	W/S	Wing loading	kg/m^2	473	466	468	466
	T/W	Thrust loading	–	0.32	0.25	0.25	0.25
	SM	Static margin	%MAC	119	34	15	38
	c.g. travel	Center of gravity travel	%MAC	69	28	23	30
Wing	Æ	Aspect ratio	–	9.4	11.4	11.7	11.5
	S	Planform area	m^2	136	131	128	132
	b	Span	m	35.8	38.8	38.6	38.9
	c_r	Root chord	m	7.1	5.0	5.1	5.0
	MAC	Mean aerodynamic chord	m	4.1	3.3	3.2	3.3
	λ	Taper ratio	–	0.16	0.29	0.27	0.29
	Λ	Sweep angle	°	26.2	11.1	12.9	11.5
	Γ	Dihedral angle	°	0.3	-0.7	-3.6	-0.8
	$f_{x_{\mathtt{w}}}$	Wing position fraction	–	0.60	0.50	0.47	0.50
Fuselage	$l_{\mathtt{f}}$	Fuselage length	m	40.6	42.0	36.4	42.1
	$d_{\mathtt{f}}$	Fuselage width	m	4.2	4.1	4.7	4.1
	$\lambda_{\mathtt{f}}$	Fineness ratio	–	9.6	10.3	7.7	10.4
Canard	Æ$_{\mathtt{c}}$	Aspect ratio	–	4.9	5.0	5.0	5.0
	$S_{\mathtt{c}}$	Planform area	m^2	19	18	21	18
	$b_{\mathtt{c}}$	Span	m	9.7	9.6	10.3	9.6
	$c_{r_{\mathtt{c}}}$	Root chord	m	2.4	2.4	2.6	2.4
	$\Lambda_{\mathtt{c}}$	Sweep angle	°	23.5	10.0	11.6	10.3
	$\Gamma_{\mathtt{c}}$	Dihedral angle	°	-0.2	0.4	1.8	0.4
HT	Æ$_{\mathtt{h}}$	Aspect ratio	–	4.9	5.0	5.1	5.1
	$S_{\mathtt{h}}$	Planform area	m^2	30	17	18	17
	$b_{\mathtt{h}}$	Span	m	12.1	9.2	9.5	9.3
	$c_{r_{\mathtt{h}}}$	Root chord	m	3.6	2.7	2.8	2.8
	$\Lambda_{\mathtt{h}}$	Sweep angle	°	29.3	12.5	14.4	12.8
	$\Gamma_{\mathtt{h}}$	Dihedral angle	°	0.3	-0.7	-3.6	-0.8
VT	Æ$_{\mathtt{v}}$	Aspect ratio	–	1.0	1.0	1.0	1.0
	$S_{\mathtt{v}}$	Planform area	m^2	24	18	19	18
	$b_{\mathtt{v}}$	Span	m	4.8	4.2	4.4	4.3
	$c_{r_{\mathtt{v}}}$	Root chord	m	5.7	5.0	5.2	5.0
	$\Lambda_{\mathtt{v}}$	Sweep angle	°	39.2	16.7	19.4	17.2

Table 6.14: Thee-surface aircraft optimization results



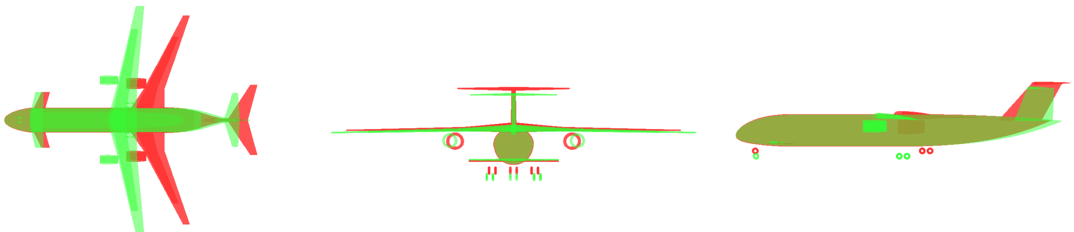
(a) Top view (b) Front view (c) Side view

Figure 6.26: Thee-surface aircraft geometry after genetic optimization



(a) Top view (b) Front view (c) Side view

Figure 6.27: Thee-surface aircraft geometry after gradient optimization



(a) Top view (b) Front view (c) Side view

Figure 6.28: Thee-surface aircraft geometry after hybrid optimization

6.6 Case 4: Oval-fuselage aircraft

The oval-fuselage aircraft has an ellipsoidal shaped fuselage cross-section. The wider fuselage allows more passengers seats abreast. So for the same fuselage length more passengers can be carried. Conventional aircraft have circular shaped fuselages, which are structurally more efficient with respect to pressurization loads. For more information regarding oval fuselages in conventional and novel aircraft configurations the reader is referred to Schmidt [25].

The design requirements are somewhat different from the previous cases. The first three cases concerned short-range aircraft. In this fourth case an aircraft for medium to long range is considered. It must have a harmonic range of 5900 km at a cruise speed of Mach 0.78. The maximum payload is established at 42 metric tons and it must be able to carry 400 passengers. All top level requirements are gathered in Table 6.15.

Pax.	Payload mass	$M_{\mathtt{C}}$	Altitude	Range	$s_{\mathtt{TO}}$	$s_{\mathtt{L}}$
400	42 tons	0.78	11.3 km	5900 km	1960 m	1490 m

Table 6.15: Oval fuselage aircraft top level requirements

The Initiator uses this information to create an aircraft that fulfils these requirements. A model of the oval-fuselage aircraft is displayed in Figure 6.29. The corresponding aircraft properties are shown in Table 6.16. From these properties it can be observed that the fuselage width is 8 meters, which is about 23% larger than its height due to its oval shape.

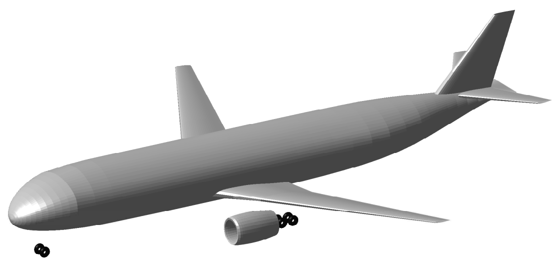


Figure 6.29: Oval-fuselage aircraft model

AR	9.5	–	Γ	6.0	°	$h_{\mathtt{f}}$	6.8	m	MTOM	159	tons
S	291	m^2	$f_{x_{\mathtt{w}}}$	0.45	–	$AR_{\mathtt{h}}$	5.1	–	OEM	78.2	tons
b	52.5	m	$(\frac{t}{c})_{\mathtt{r}}$	0.151	–	$S_{\mathtt{h}}$	57.0	m^2	FM	38.9	tons
Λ	26.2	°	$(\frac{t}{c})_{\mathtt{k}}$	0.104	–	$b_{\mathtt{h}}$	17.0	m	$R_{\mathtt{h}}$	5960	km
c_r	11.5	m	$(\frac{t}{c})_{\mathtt{t}}$	0.104	–	$AR_{\mathtt{v}}$	1.6	–	PRE	6430	km
λ	0.16	–	$l_{\mathtt{f}}$	60.1	m	$S_{\mathtt{v}}$	40.1	m^2	L/D	16.5	–
ϵ	0	°	$d_{\mathtt{f}}$	8.0	m	$b_{\mathtt{v}}$	8.0	m	$C_{L_{\mathtt{max,clean}}}$	1.25	–

Table 6.16: Oval-fuselage properties using the Initiator

The design variables from Table 6.1 are used to perform the sensitivity analysis with. The required 52 runs were completed in 96 minutes. The results are shown in Figure 6.30. The variables are numbered according to their sensitivity value. It can be seen that two parameters stand out, which are the wing position and the sweep angle. The other variables have much less impact. They are grouped on the left side of the plot due to their low overall importance.

(a) Mean vs. standard deviation (b) Sensitivity index

Figure 6.30: Oval-fuselage aircraft sensitivity analysis results

The 5 most sensitive parameters from Figure 6.30b were selected for optimization. The optimization results for the genetic, gradient-based and hybrid algorithm are given in Table 6.17. A more extensive overview of the changes in geometry and performance with respect to the initial design are given in Table 6.18.

	$Æ$	f_{x_w}	Λ	$(\frac{t}{c})_k$	d_f	PRE	Δ_{PRE}	t
Initiator	9.5	0.45	26.2°	0.104	8.0 m	6430 km	–	–
Genetic	12.3	0.48	5.2°	0.106	7.4 m	7040 km	+9.5%	121 min
Gradient	9.4	0.43	24.1°	0.108	8.4 m	6720 km	+4.5%	129 min
Hybrid	12.6	0.47	5.1°	0.108	7.5 m	7070 km	+10.0%	183 min

Table 6.17: Optimum design parameters for the oval-fuselage aircraft

The hybrid algorithm achieved the best payload-range efficiency, but it remains very close to the genetic solution. An improvement of about 10% is obtained. The gradient-based solution did not go beyond a meager 4.5% increase in objective value. Looking at the computation time the genetic algorithm clearly wins. The hybrid algorithm managed to find a sightly more optimal solution, but at higher computational expense.

The kink thickness ratio increased slightly in all optimizations. A thicker section increases the stiffness of the structure, but negatively affects the lift-to-drag ratio. The differences over the baseline are rather small. Another trend that follows from the optimized design vectors is the low sweep angle. This was also seen at the Airbus A320 and three-surface aircraft cases.

Significant differences in weight can be found when comparing the designs. An overview of the weight components is given in Figure 6.31. The largest differences can be found in the wing. The gradient-based solution obtained the lightest wing structure. The genetic and hybrid optimized designs have the heaviest wing, but saved on engine weight by having better thrust-to-weight ratios due to an increase in L/D.

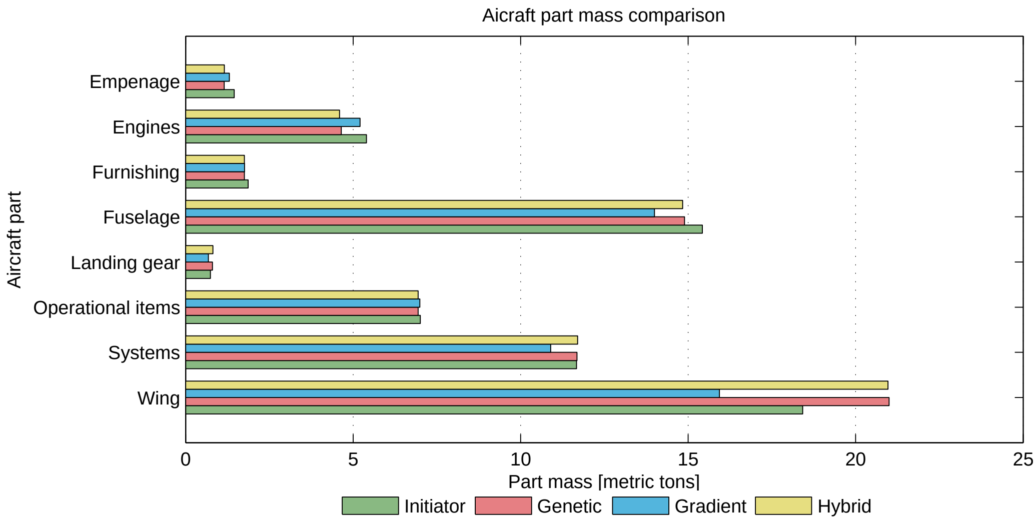
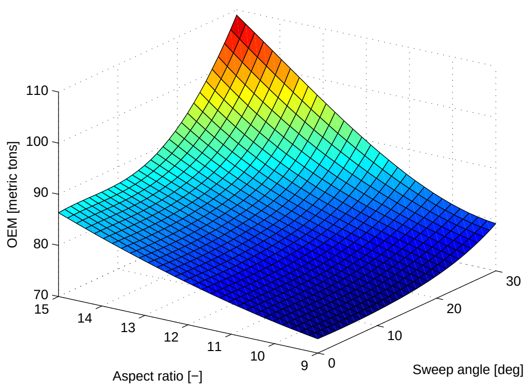
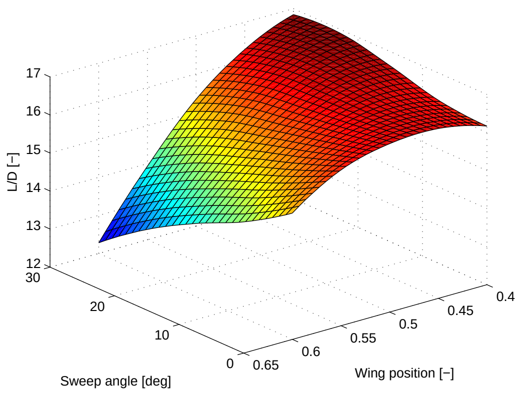


Figure 6.31: Oval-fuselage aircraft change in part mass after optimization

The weight of the wing and thus the aircraft increases with aspect ratio. The weight penalty becomes even higher at sweep angles beyond 20° as can be seen in Figure 6.32a. The choice for a low sweep angle of the genetic and hybrid algorithm allowed for a larger aspect ratio to further improve the lift-to-drag ratio at a reduced weight penalty.



(a) Operational empty mass vs. aspect ratio and sweep angle ($f_{x_w} = 0.45$)



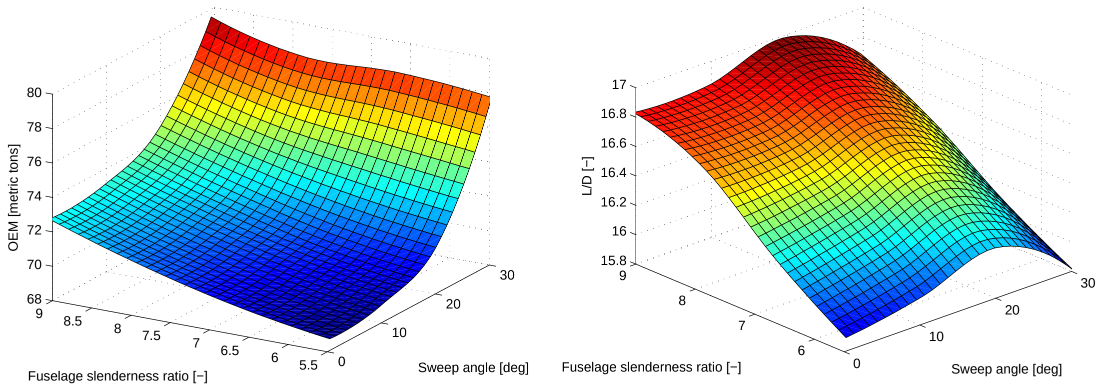
(b) Lift-over-drag ratio vs. wing position and sweep angle ($Æ = 9.5$)

Figure 6.32: Oval-fuselage aircraft operational empty mass and lift-to-drag ratio

The low sweep angle is also driven by the wing position as is shown in Figure 6.32b.

This relation has already been explained in the A320 case. The L/D deteriorates with higher sweep angle at more aft wing positions due to an increasing tail size. This effect is stronger for the oval-fuselage aircraft than for the A320.

The gradient-based solution saved on fuselage and systems weight thanks to its shorter fuselage. Its lower fuselage slenderness ratio is clearly distinguishable in the top view of Figure 6.35. The effect of the fuselage slenderness on the operational empty weight is depicted in Figure 6.33a. Here it can be observed that the weight decreases with lower slenderness ratios. This follows from the lower forces in the fuselage structure, leading to a less heavy design. The benefit decreases a bit as the sweep angle becomes larger.



(a) Operational empty mass vs. slenderness ratio and sweep angle

(b) Lift-over-drag ratio vs. slenderness ratio and sweep angle

Figure 6.33: Oval-fuselage aircraft fuselage slenderness ratio effects ($\mathcal{R} = 9.5$)

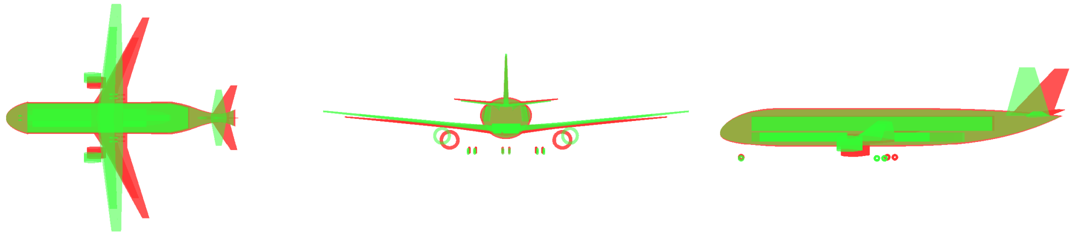
The slenderness ratio has some effect on the lift-to-drag ratio. Less slender fuselages result in lower L/D values, which is illustrated in Figure 6.33b. Due to the lower moment arm of the horizontal tail it must increase in size to provide a sufficient counterbalancing moment leading to an increase in drag and weight. When comparing the gradient-based design to the genetic and hybrid designs, it has a 30% to 35% higher horizontal tail planform area due to its shorter fuselage. Also, the drag of the fuselage changes with slenderness ratio. A lower ratio resulted in a higher drag coefficient.

It can be concluded that quite some improvement in payload-range efficiency can be achieved over the baseline design. Most benefit can be gained by lowering the sweep angle and increasing the aspect ratio. The parameters are closely related to the wing position. By adjusting the width of the fuselage the weight and lift-to-drag ratio can be further tuned. Improvements in L/D also yield in better thrust-to-weight ratios leading to smaller and lighter engines.

When comparing the algorithms, the genetic showed most gain in the shortest amount of time. The additional gradient-based step of the hybrid algorithm only improvement marginally on the genetic solution. The gradient algorithm got stuck on a local optimum as its objective is only increased by 4.5%, which is only half as much as the other algorithms.

	Parameter	Description	Unit	Initiator	Genetic	Gradient	Hybrid
Key performance indicators	PRE	Payload-range efficiency	km	6430	7040	6720	7070
	X	Range parameter	km	6620	7390	6610	7420
	R_{h}	Harmonic range	km	5960	5950	5960	5940
	$R_{\mathtt{max.fuel}}$	Max. fuel range	km	12080	8720	11620	8750
	MTOM	Maximum take-off mass	tons	159	155	151	155
	OEM	Operational empty mass	tons	78.2	77.7	72.2	77.6
	FM	Fuel mass	tons	38.9	35.5	37.3	35.3
	L/D	Lift-to-drag ratio	–	16.5	18.5	16.5	18.5
	$C_{L_{\mathtt{max,clean}}}$	Max. lift coefficient	–	1.25	1.51	1.32	1.53
	$V_{s_{\mathtt{clean}}}$	Stall speed clean	m/s	83.7	75.7	81.5	75.0
	W/S	Wing loading	kg/m^2	547	539	547	538
	T/W	Thrust loading	–	0.30	0.26	0.30	0.26
	SM	Static margin	%MAC	67	50	50	50
	c.g. travel	Center of gravity travel	%MAC	22	26	21	26
Wing	AR	Aspect ratio	–	9.5	12.3	9.4	12.6
	S	Planform area	m^2	291	288	277	288
	b	Span	m	52.5	59.6	50.9	60.2
	c_r	Root chord	m	11.5	6.9	10.8	6.8
	MAC	Mean aerodynamic chord	m	6.7	5.2	6.5	5.1
	λ	Taper ratio	–	0.16	0.37	0.18	0.37
	Λ	Sweep angle	°	26.2	5.2	24.1	5.1
	Γ	Dihedral angle	°	6.0	6.0	6.0	6.0
	$f_{x_{\mathtt{wing}}}$	Wing position fraction	–	0.45	0.48	0.43	0.47
Fus.	$l_{\mathtt{f}}$	Fuselage length	m	60.1	59.7	52.9	59.3
	$d_{\mathtt{f}}$	Fuselage width	m	8.0	7.4	8.4	7.5
	$h_{\mathtt{f}}$	Fuselage height	m	6.8	6.3	7.1	6.4
	$\lambda_{\mathtt{f}}$	Fineness ratio	–	7.5	8.0	6.3	7.9
HT	AR_{h}	Aspect ratio	–	5.1	5.1	5.1	5.1
	S_{h}	Planform area	m^2	57	43	56	42
	b_{h}	Span	m	17.0	14.7	16.9	14.5
	$c_{r_{\mathrm{h}}}$	Root chord	m	5.0	4.4	5.0	4.3
	Λ_{h}	Sweep angle	°	29.3	5.8	26.9	5.7
	Γ_{h}	Dihedral angle	°	6.0	6.0	6.0	6.0
VT	AR_{v}	Aspect ratio	–	1.6	1.6	1.6	1.6
	S_{v}	Planform area	m^2	40	44	40	44
	b_{v}	Span	m	8.0	8.5	8.0	8.4
	$c_{r_{\mathrm{v}}}$	Root chord	m	7.5	7.8	7.4	7.8
	Λ_{v}	Sweep angle	°	39.2	7.8	36.1	7.7

Table 6.18: Oval-fuselage aircraft optimization results



(a) Top view (b) Front view (c) Side view

Figure 6.34: Oval-fuselage aircraft geometry after genetic optimization

(a) Top view (b) Front view (c) Side view

Figure 6.35: Oval-fuselage aircraft geometry after gradient-based optimization

(a) Top view (b) Front view (c) Side view

Figure 6.36: Oval-fuselage aircraft geometry after hybrid optimization

6.7 A comparison of the obtained aircraft designs

In order to be able to compare the designs that have been obtained from the four case studies, first some parameters have to be defined on which the comparison can be based. For this purpose the value efficiency parameters defined by Nangia [19] are used. These parameters are based on the payload-range efficiency, but are normalized with respect to the weight of the aircraft. The first parameter is the value efficiency parameter with respect to the maximum take-off mass, which is abbreviated to VEM. It is defined as follows:

$$\mathtt{VEM} = \frac{\mathtt{PRE}}{W_{\mathtt{TO}}} \tag{6.7}$$

The second parameter is the value efficiency with respect to the operational empty mass and is denoted as VEO. It is shown in Equation 6.8.

$$\mathtt{VEO} = \frac{\mathtt{PRE}}{W_{\mathtt{OEM}}} \tag{6.8}$$

The above efficiency parameters values are determined for each design and these results are shown in Table 6.19 for the VEM parameter and in Table 6.20 for the VEO parameter.

	Initiator [km/kN]	Genetic [km/kN]	Gradient [km/kN]	Hybrid [km/kN]
1. Airbus A320	13.6	14.2	13.5	14.2
2. Canard aircraft	13.2	15.2	14.3	15.4
3. Three-surface aircraft	9.9	12.6	12.9	12.5
4. Oval-fuselage aircraft	4.1	4.6	4.5	4.6

Table 6.19: Comparison of the case studies using the VEM parameter

	Initiator [km/kN]	Genetic [km/kN]	Gradient [km/kN]	Hybrid [km/kN]
1. Airbus A320	26.1	26.6	24.4	26.0
2. Canard aircraft	26.5	29.2	27.2	29.0
3. Three-surface aircraft	18.5	23.5	24.6	23.4
4. Oval-fuselage aircraft	8.4	9.2	9.5	9.3

Table 6.20: Comparison of the case studies using the VEO parameter

When comparing the efficiency values of the first three cases, which all have the same top level requirements, if follows that the canard aircraft is the most efficient design. The canard aircraft designs obtained from the genetic and hybrid algorithm show the best results. The three-surface aircraft designs perform significantly less compared to the

A320 and canard aircraft. Although the baseline TSA can be improved a lot through optimization, it is no match against the other configurations. The conventional Airbus A320 showed the least improvement after optimization. When looking at the VEM and VEO values of the gradient optimized A320, it has an even worse efficiency value compared to the initial design although it has a better payload-range efficiency. This is caused by the relatively large increase in weight. The OEM and MTOM increased by 5.6% and 11.9% respectively, while its payload-range efficiency only increased by 4.6%.

All oval-fuselage aircraft solutions have much lower efficiency values with respect to the first three cases. This follows from its higher range and payload requirements, which causes the weight of the aircraft to increase more rapidly than the payload-range efficiency. This trend is in accordance with the results obtained by Nangia [19].

6.8 Evaluation of the algorithms

For the case studies three optimization algorithms have been used: the genetic algorithm, the SQP gradient algorithm and the hybrid genetic–SQP algorithm. Based on the obtained results and experience with the optimizer tool it was found that the genetic algorithm worked best. The main arguments for choosing the genetic algorithm over the other methods are its robustness and tolerance towards noise in the model outputs. The computation time can be reduced significantly by using parallel optimization, which eliminates one of the weak points of the algorithm.

The output noise of the Initiator proved to be troublesome for the gradient algorithm. It may cause the gradient algorithm to start oscillating around a certain point in the design space due improper gradient information. It often required tuning of the algorithm settings like the minimum step size to overcome the noise.

In Figure 6.37 the presence of noise is demonstrated by plotting the aspect ratio against the obtained payload-range efficiency. The aspect ratio was increased from 9 to 14 with increments of 0.1 for the Airbus A320 aircraft. In this figure it can be seen that a small increase in aspect ratio may result in lower payload-range efficiency, while a higher value is expected or vice versa. In other words, the change in results due to a small change in the design may contradict the global trend, which causes the gradient algorithm to take a search step in the wrong direction.

The hybrid algorithm produced slightly better results in some cases, but at higher computational cost. The added benefit of the local gradient-based solver does not seem to outweigh the required computation time. Also, the noise adversely affects the capability of gradient-based algorithm to find the exact optimum in the region provided by the genetic algorithm.

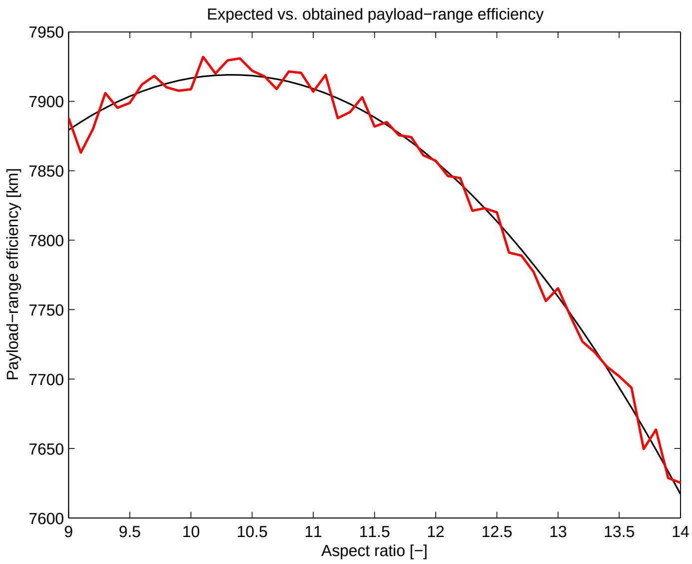


Figure 6.37: Payload-range efficiency for increasing aspect ratio using the Initiator

Chapter 7

Conclusions and recommendations

7.1 Conclusions

The goal of the thesis was to develop an optimization tool for the conceptual design of conventional and unconventional aircraft. This optimization tool is used to answer the research question: *What effect has the developed optimization strategy on the key performance indicators of unconventional aircraft configurations?*.

Through the years lots of data has been gathered on conventional aircraft and therefore design rules and estimates for such aircraft became fairly accurate. However, this does not apply to unconventional and novel configurations, for which far less data is available and design approaches are sometimes rather crude. If certain edge cases are not covered well, the optimizer might exploit this loophole in an attempt to find even better solutions. This may result in strange or unrealistic designs. Therefore, the outcome of the optimization strongly depends on the behaviour and flexibility of the analysis routines. Similarly, limitations of analysis modules put constraints on the design space.

From the case studies it followed that large improvements can be obtained with unconventional aircraft configurations with respect to the reference aircraft proposed by the Initiator design tool. The highest payload-range efficiency was obtained with the hybrid optimized canard aircraft. Most improvement was found with the three-surface aircraft. All three optimizations showed an increase of over 20% compared to the initial design. The oval-fuselage aircraft could be improved by a solid 10%, while the lowest improvement was obtained with the conventional A320.

When comparing the results of the first three cases, which share the same top level requirements, it is clear that the canard aircraft is the best concept with respect to the objective. It obtained the highest payload-range efficiency. It yielded a 5% higher payload-range efficiency compared to the best solution from the A320 case. The three-surface aircraft showed the least promising results. However, these statements are only valid with respect to the output provided by the Initiator. Due to several discrepancies in the sizing and analysis routines the actual performance of the considered aircraft configurations might be very different. Therefore the results should be interpreted with caution and should be mainly used as an indication of the maturity and validity of the Initiator design tool.

Among all cases the most contributing factors were the wing longitudinal position, sweep angle and wing aspect ratio. There is a tendency towards lower sweep angles due to the positive effect on the weight of the wing. The drag rise penalty due to the lower sweep seems to be underestimated, which is exploited by the optimizer by trading sweep for a higher aspect ratio to minimize the weight penalty of the latter. In the canard case relatively high sweep angles were found. From this result followed that the weight penalty of forward swept wings due to sweep is underestimated. It also contradicts the findings of the aft-swept wing cases in which a lower sweep was actually more beneficial. This can be traced back to an error in the drag estimation, especially with respect to compressibility effects.

In three cases the fuselage fineness ratio was involved in the optimization. The results showed that changing the ratio offered some reduction in fuselage weight due to a more favourable structural loading at the expensive of more drag.

The uncertainties in the computed results of the Initiator were not handled well by the gradient-based algorithm. The gradient algorithm either stopped prematurely or started oscillating around a certain design point when too much noise was present. This was alleviated by increasing the step size of the algorithm, but at the expense of accuracy. Also, determining the starting point of the gradient algorithm remains difficult. Not every starting point yields a feasible design and a change in start location might lead to a different basin of attraction.

The genetic algorithm was found to be very robust. It is far less sensitivity to noise, because it does not use gradient information. Its multi-solution approach allows the algorithm to explore multiple sites at the same time, which allows it to continue searching in other sites when an infeasible region is encountered. Its computational cost was significantly reduced by applying parallel optimization and using a caching mechanism. The hybrid algorithm was found to be too computational expensive. The obtained increase in objective value did not outweigh the added cost.

7.2 Recommendations

The following recommendations and considerations can be made for further improvement on the developed optimizer tool.

In order to further investigate unconventional aircraft configurations improvements in the analysis tools are required. For instance, the current aerodynamics implementation underestimates the compressibility effects which has consequences for the drag estimates and therefore the overall aircraft design. This heavily affects the sweep angle. Through the use of a better aerodynamic solver the potential of novel aircraft configurations can be studied more accurately and different optimization solutions might be obtained.

The sizing routine of the control surfaces is found to be inadequate, since the Initiator derives most parameters directly from the wing and does not properly take into account control and stability requirements. Results have shown that this mainly regards the sweep and dihedral angle. Especially, the sweep angle is of concern, since it changes the lift-curve slope and therefore also stall characteristics. These sizing issues also affect the static margin. It was found that class II design information was not fed back to the control surface sizing.

A related concern is the static margin. Mainly due to changes in sweep angle, the optimizer moved the wing to make sure that the nose loading constraints were satisfied. This effect outweighs the weight savings due to smaller control surfaces that would have been obtained with a lower static margin. This could be solved by imposing a constraint on the allowed static margin. In order to do this reliably, the Initiator's control surface sizing routine should be improved first.

The dihedral angle is driven by the lift-to-drag ratio, while it should also take into account lateral effects. Currently, the Initiator does not compute the lateral stability yet, which affected the outcome of the dihedral angles of the optimizations.

Another issue that currently affects the design space is the EMWET weight estimation module. Wings with a high aspect ratio or unconventional shape are found to be problematic. It also seems to underestimate the weight penalty of forward swept wings. For a better evaluation of the aircraft designs this module should be improved.

The design variables could be expanded by including, for instance, the engine location. In this thesis their positions were fixed with respect to the wing span or fuselage. The current set of design variables exposed some large discrepancies, which should be solved first.

The Initiator has rather limited support for the blending-wing body concept and the Prandtlplane. When the analysis with respect to these concepts have matured, optimizations of these concepts could be performed with the developed tool to discover any further issues. At the time of this thesis the maturity level was found to be too inadequate and therefore they were not included in the case study.

Part II

Code documentation

Chapter 8

Program structure

In this chapter the program structure of the optimizer is described. The optimizer tool is written in MATLAB . It is part of the workflow modules. In the first section the optimizer class, its properties and its methods are explained. In Section 8.2 the sensitivity routine is described. Section 8.3 elaborates the optimization routine.

8.1 Optimizer class

The optimization routines and properties are housed in a single module class. As has been explained in Section 2.3, this class inherits from the `WorkflowModule` class. Workflow modules are placed outside the analysis chain and are used to control the workflow of the Initiator. The relationship is shown in the UML diagram of figure 8.1.

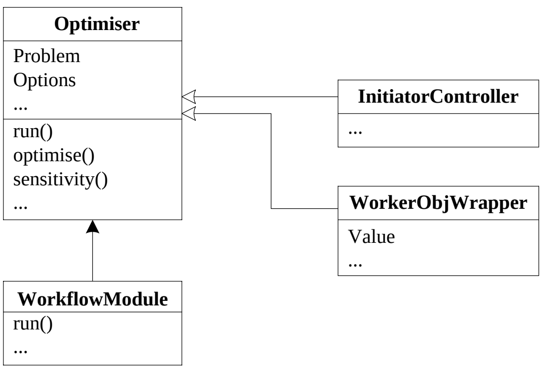


Figure 8.1: UML class diagram of the optimizer

The optimizer class depends on the `InitiatorController` and the `WorkerObjWrapper` classes. The Initiator controller is the main class of the Initiator. The optimizer uses this controller to control the workflow and to communicate with the modules. The

`WorkerObjWrapper` class has been developed by MathWorks [5] and is used during parallel optimization. Normally, data is destroyed and recreated when a parallel worker advances to the next iteration. This class allows to retain the data of the parallel worker, such that expensive recreation of the Initiator instance is not required.

The optimizer class has several public properties that can be accessed. These properties are listed in Table 8.1. The `Debug` property triggers debug mode when set to true. In this mode the optimizer will output debug information to the command window. The `Problem` property holds the problem structure in which the optimization problem is described. The `Options` property contains the sensitivity analysis and optimization options.

Property	Description
Debug	Debug mode
Problem	Problem description
Options	Structure containing all options
Results	Structure with sensitivity and optimization results
ResultsDirectory	Directory in which the results are saved

Table 8.1: Public properties of the optimizer class

The `Problem` property that holds the optimization problem, which is required to perform a sensitivity analysis or an optimization. The available fields are shown in Table 8.2.

Field	Description
ObjFcn	Cell array with one or more objective functions
ObjScaling	Cell array containing objective scaling parameters
AssignFcn	Cell array with an assign function per parameter
DesignVarScaling	Cell array with scaling parameters for design variables
ConFcn	Cell array with constrain functions
ModuleList	List of modules to run
Algorithm	Optimization algorithm
LowerBound	Lower bound of the design variables
UpperBound	Upper bound of the design variables
Start	Starting point
Selected	Selected design variables
Labels	Contains labels used for plotting

Table 8.2: Problem structure fields

The module list contains the modules that are executed during the sensitivity analysis and optimization. By default this is the `DesignConvergence` module, but any module can be used. The `Algorithm` field currently accepts the following three algorithms: *gradient*, *genetic* and *hybrid*. Through the `Selected` field the design variables can be easily activated or deactivated. This is especially useful when there are many design variables. The `Label` field can be used to provide names for the objective, assign and constraint functions. This is for plotting purposes only.

The exposed optimizer methods are listed in Table 8.3. Normally, the optimizer is run through the Initiator controller, but by obtaining its module handle these methods can be called. This may offer some more fine-grained control over the optimizer. The usage of these methods is explained in Chapter 9.

Method	Description
addConstraint	Adds a constraint to the optimization problem
addDesignVar	Adds a design variable to the optimization problem
addObjective	Adds an objective to the optimization problem
elemEffects	Elementary effects routine
listFiles	Lists all available results files
loadData	Loads the problem, option and result data from disk
optimise	Starts the optimization
resetOptions	Resets all options to default
resetProblem	Resets the problem description
resetResults	Clears the results
resume	Resumes optimization from a previous run
run	Performs sensitivity analysis and optimization
saveData	Saves the problem, option en result data to disk
showOptimPlots	Shows the optimization plots
showProblem	Shows the problem description in command window
showSensPlot	Shows the sensitivity analysis plots

Table 8.3: Public methods of the optimizer class

8.2 Sensitivity Analysis

The optimizer module contains a sensitivity analysis routine to screen the design variables. It can be called by using the method `sensitivity`. The screening procedure is able to identify the design variables which have the most impact on the objective function. This way the most influential parameters can be selected for the optimization phase, which reduces its complexity and decreases computation time.

The screening is performed by using the elementary effects method. This method consists of individually randomised one-at-a-time experiments. Each time a factor is changed its impact is measured.

A flowchart of the sensitivity analysis process is shown in Figure 8.2. After initialization a copy of the Initiator controller is made. This is done such that the state of the current session is not altered. Next, the routine calculates the elementary effect of each variable. This is repeated for the specified number of trajectories. During this process an estimate for the remaining time is given based on the average computation time of previous iterations. Then, the sensitivity values are calculated. Based on these values the optimizer automatically selects the most important design variables. The other variables are disabled. The results are stored in the property `Results.Sensitivity`. A description of the sensitivity results structure can be found in Table 8.4. The available sensitivity analysis options are listed in Chapter 9.

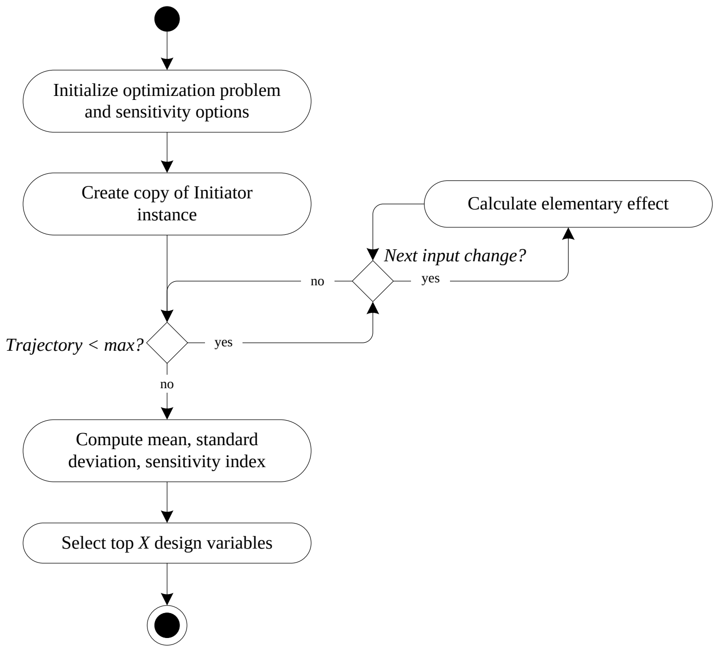


Figure 8.2: Flowchart of the sensitivity analysis

Field	Description
ObjValues	Objective values of each iteration
DesignVars	Design vectors of each iteration
Method	Sensitivity method name
Mu	Mean μ
Mu_s	Improved mean μ^*
Sigma	Corrected standard deviation σ
Sigma_n	Uncorrected standard deviation σ_n
Euclidean	Euclidean distance for ranking the variables
Ranks	Design vector ranks based on the Euclidean
Labels	Labels for plotting

Table 8.4: Sensitivity analysis results structure

8.3 Optimization

The actual optimization is governed by the `optimise` method. It currently supports the genetic algorithm, gradient algorithms and a genetic–gradient hybrid algorithm. All algorithms rely on MATLAB implementations. The genetic algorithm is based on `ga`, the gradient on `fmincon` and the hybrid on both `ga` and `fmincon`. Because the MATLAB optimization functions have different input and output formats, each algorithm is wrapped inside a separate class method. This allows a uniform approach.

The optimizer offers the possibility to perform parallel optimization. This only applies to the genetic and hybrid algorithm, since the gradient method cannot compute the objective function in parallel. Parallel mode is turned on by setting `Options.<Algorithm>.UseParallel` to *always*, or off by setting it to *never*. All other available optimization options can be found in Section 9.3.1 of Chapter 9.

The top-level flowchart of the optimization process is shown in Figure 8.3. It starts with running the Initiator with default values. After the initial design has been computed, its results are saved. Then, a parallel MATLAB session may be opened depending on the aforementioned setting. Next, multiple copies of the current Initiator instance are created. This prevents polluting the state of the current Initiator instance and avoids race conditions during parallel optimization. In single-threaded mode a single copy will be created. For parallel optimization this depends on the configured number of parallel workers.

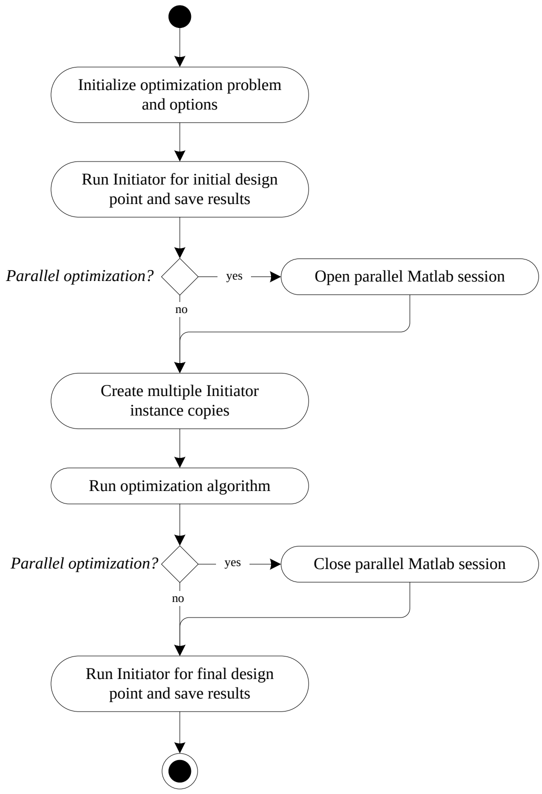


Figure 8.3: Top-level flowchart of the optimization process

At this point the optimization algorithm is started. For the objective function and non-linear constraint function internal class methods are assigned. These methods use the `initiatorRunner` routine to obtain the objective and constraint values. The flowchart of this routine is shown in Figure 8.4.

The `initiatorRunner` function starts with obtaining a copy of the Initiator instance using the task number assigned by MATLAB . Next, a cache lookup is performed for the requested design vector. When it is a cache hit, the results are gathered and returned immediately. A cache miss leads to a full multidisciplinary analysis. It begins with resetting all modules and rescaling the design vector to the actual values. These design values are assigned to the modules and then the design convergence module is run.

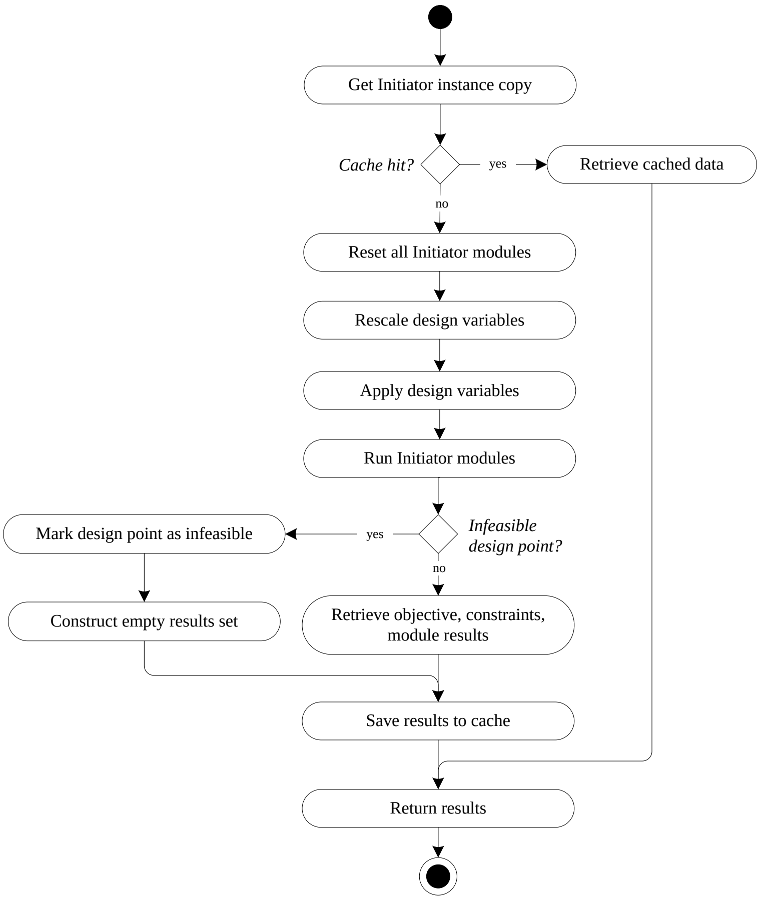


Figure 8.4: Flowchart of the `initiatorRunner` method

When the design point is feasible, the results are collected. In case of an infeasible design the results an empty results set is created. After storing the data corresponding to the requested design vector, the results are returned to the calling function. Once the optimization algorithm has finished, the parallel session is closed and the Initiator is run for the final design point.

The results of the optimization are stored in the `Results.Optimisation` property. The data is also automatically stored to a `mat`-file. Besides the results it also includes the optimization problem and options structs, which allows the user to restore the current optimizer state at a later point in time. A description of the results structure can be found in Table 8.5.

Field	Description
Algorithm	Algorithm specific output data
ConEvals	Array containing constraint function evaluations
Final	Final aircraft data
Labels	Labels used for plotting
ObjEvals	Array containing objective function evaluations
Original	Baseline aircraft data
StateData	Array containing algorithm state data

Table 8.5: Optimization results structure

Chapter 9

User manual

This chapter serves as the user manual of the optimizer. This guide assumes that the reader is already familiar with the Initiator. For more information on using the Initiator the reader is referred to Elmendorp [14].

9.1 Requirements

The requirements for the optimizer are as follows:

- The Initiator design tool
- MATLAB 2012a; version 2013b or higher is recommended
- Windows 7 or higher, Mac OS X 10.7[1] or higher, Linux[1]
- At least 8 GB memory is recommended for parallel optimization

In addition to these requirements Subversion may be useful to retrieve the latest version from the repository.

9.2 Setting up a problem description

In order to use the optimizer module a problem description must be set up first. This description contains the information that is required to perform a sensitivity analysis and an optimization. A default problem statement is loaded automatically when no user input is specified. The standard objective is the payload-range efficiency and the default algorithm is the genetic algorithm. The default design variables are as follows:

[1] Compatibility with these operating systems depends on the installed libraries [14].

1. Aspect ratio
2. Wing x-position
3. Fuselage diameter
4. Sweep angle
5. Taper ratio
6. Dihedral angle
7. Twist angle per wing section
8. Thickness-over-chord ratio per wing section

The problem description can be changed by means of module inputs. These module inputs can be specified in the aircraft input file. A description of the available inputs is given in Section 9.3.1

9.3 Operating the optimizer

Once the problem description has been set up the optimizer can be run. When the Initiator calls its `run` method, first the number of design variables will be checked. If this number exceeds the configured maximum, a sensitivity analysis will be done first.

Depending on the number of design variables the sensitivity analysis may take some time. Using the default design variables the analysis takes one to two hours. The optimizer will show a remaining time estimate. When the sensitivity values have been obtained the most important design variables will be selected to perform the optimization with.

The optimization starts with storing the state of the initial aircraft. Then the selected algorithm is called. Depending on the settings and chosen algorithm a parallel MATLAB session may be opened. When the algorithm has found an optimum, the corresponding design vector is used to compute the final aircraft. Finally, the optimization data is saved to a `mat`-file and the results are shown.

9.3.1 Module input

The default problem setup can be changed by providing module inputs in the aircraft configuration file. The user is not required to specify all elements in order for the optimizer to work. The supplied input will simply overwrite the default values. When for instance design variables are supplied, they will only replace the standard design variable list. An example is given in Listing 9.1.

```
<moduleInputs>
  <input module="Optimiser">
    <problem>
      <objectives>
        ..
      </objectives>
      <designVars>
        ..
      </designVars>
      <constraints>
        ..
      </constraints>
      <algorithm>genetic</algorithm>
      <moduleList>GeometryEstimation,DesignConvergence</moduleList>
    </problem>
  </input>
</moduleInputs>
```

Listing 9.1: Module inputs example for optimizer module

Inside the input section of the optimizer module there must be a main element called `problem`. This element holds the entire problem description.

The objectives, design variables and constraints can be provided with the `objectives`, `designVars` and `constraints` elements respectively. They are explained in the following subsections.

The `algorithm` field can be used to provide the algorithm. Currently, there are three algorithms available: *genetic*, *gradient* and *hybrid*. By default the genetic algorithm is loaded.

The modules that need to be run in the sensitivity analysis and optimization can be changed with the `moduleList` field. The module names must be separated by a comma. By default the geometry estimation and design convergence modules are called.

Objective functions

The objective can be specified with an `objectives` element. It requires a `label`, `module` and `value` element. The label is used for plotting and can be any string. An example is given in Listing 9.2.

```
<problem>
  <objectives>
    <label>PRE</label>
    <module>PerformanceEstimation</module>
    <value> - KPI.PRE</value>
    <scaling>0.001</scaling>
  </objectives>
</problem>
```

Listing 9.2: Module input example for objective functions

The module field refers to a module from the Initiator, which is the performance estimation module in this case. The value field specifies which result value must be used from the module. In the example the payload-range efficiency result from the key performance indicators is used. A minus sign can be added in front of the value field if necessary. A scale factor can be added to change the order of magnitude of the objective value.

There can be multiple `objectives` elements. Note that only the first objective is considered during optimization. Multiple entries may be useful when a sensitivity analysis must be performed for several objectives.

Design variables

The design variables can be specified with the `designVars` element. There are two variants, which are given in Listing 9.3.

```xml
<problem>
  <designVars>
    <label>Wing x-position</label>
    <module>GeometryEstimation</module>
    <value>MainWingXPosition</value>
    <lowerBound>0.30</lowerBound>
    <upperBound>0.60</upperBound>
    <start>0.45</start>
  </designVars>
  <designVars>
    <label>Aspect ratio</label>
    <type>ConfigurationParameter</type>
    <value>WingAspectRatio</value>
    <lowerBound>8</lowerBound>
    <upperBound>15</upperBound>
    <start>10</start>
    <scaling>0.1</scaling>
  </designVars>
</problem>
```

Listing 9.3: Module input example for design variables

The first design variable in the code example sets the longitudinal position of the wing. Here the geometry estimation module is used to set the value of *MainWingXPosition*. The lower bound and upper bound are set to 0.30 and 0.60 respectively. Optionally a starting position can be provided. By default the mean between the bounds is used.

The second design variable controls an aircraft configuration parameter by means of the type field. In the example the wing aspect ratio is controlled through the *WingAspectRatio* value. Again, a lower bound and upper bound must be provided. A scale factor can be added to change the order of the design variable. It also accepts the value *auto*, which transforms the parameter such that it has a range of $[-1, 1]$ [16]. The governing equation is as follows:

$$\bar{x} = \frac{2x}{x_u - x_l} - \frac{x_u + x_l}{x_u - x_l} \tag{9.1}$$

In Equation 9.1 $\bar{x}$ is the scaled variable, x_l represents the lower bound and x_u denotes the upper bound.

Constraint functions

The constrain functions can be specified with the `constraints` element. It requires a label and function. An example is given in Listing 9.4.

```
<problem>
  <constraints>
    <label>Nose loading</label>
    <function>myNoseLoadingConstraint</function>
  </constraints>
</problem>
```

Listing 9.4: Module input example for constraints

As can be seen in the code example a MATLAB function name must be supplied to the function field. Since constraints can involve extensive code, it has been chosen to keep the actual constraint logic in MATLAB . When a custom function is built, one must sure that it takes the main Initiator controller, worker controller and design vector as input arguments. The output must be two vectors containing the inequality and equality constraints.

Optimizer settings

The settings of the optimizer can be changed through the settings file of the Initiator. Settings that are provided by the user will overwrite the default values of the optimizer. An example is given in Listing 9.5.

```
<settings>
  <setting>
    <name>Optimiser-General-MaxTime</name>
    <value>8000</value>
  </setting>
  <setting>
    <name>Optimiser-ElemEffects-Trajectories</name>
    <value>4</value>
  </setting>
  <setting>
    <name>Optimiser-Genetic-PopulationSize</name>
    <value>10</value>
  </setting>
</settings>
```

Listing 9.5: Optimizer settings example

As can be seen in this example each setting has a `name` and `value` element. The name of each setting consists of three parts, which are separated by hyphens. First the name of the optimizer module is provided, followed by the category and the name of the setting. Currently there are five categories: *General*, *Genetic*, *Gradient*, *Hybrid* and *ElemEffects*.

The settings available in the general category are listed in Table 9.1 and control the global parameters of the optimizer.

Setting	Description	Default
Debug	Enable or disable debug mode	true
MaxDesignVars[1]	Maximum number of design variables	5
MaxTime[2]	Optimization time limit in seconds	7200
PoolSize[1]	Number of parallel workers	-[3]
ResultsDir	Directory to write the results data to	/Data/Optimiser
SensScaleFactor[1]	Factor reducing the range of design vectors	0.5
ShowPlots	Enable or disable plots	true
TolCache[2]	Cache tolerance for matching design vectors	`eps()`
UseCache[2]	Enable or disable results cache	true

Table 9.1: General optimizer settings
[1]Sensitivity analysis only [2]Optimization only [3]System dependent

The options of the elementary effects method are given in Table 9.2. The genetic, gradient and hybrid algorithms use the options specified in the MATLAB manual [4].

Setting	Description	Default
Grid	Grid sizing parameter	4
Retries	Number of trajectory retries after an error	5
Trajectories	Number of trajectories	4

Table 9.2: Elementary effects method settings

9.3.2 Using the module handle

To get more fine-grained control one can obtain the optimizer module handle from the Initiator. This way the sensitivity analysis and optimization can be run individually and additional functions are available. The sensitivity analysis can be started separately by calling the `sensitivity` method. To only start the optimization the `optimise` method can be used.

There is also the possibility to resume a previous optimization run. This is dony by running the `resume` method, which expects a results file name as parameter. The optimizer will continue at the previously found optimum.

At any point in time the state of the optimizer can be saved to disk. This can be done by calling the `saveData` method. It will automatically generate a file name if none is provided. The save data includes the problem statement, options and results. By default the data is stored in `/Data/Optimiser`.

The data can be loaded through the `loadData` method. It restores the problem statement, options and results to its previous state. A list of available files can be retrieved by calling `listFiles`.

An overview of the problem statement can be printed in the command window by using the `showProblem` method. This may give some extra insight in the problem setup. Plots can be shown using the `showSensPlots` and `showOptimPlots` methods for the sensitivity and optimization results respectively.

References

[1] Allison, J. Complex system optimization: A review of analytical target cascading, collaborative optimization, and other formulations. Master's thesis, University of Michigan, 2004.

[2] Anon. White paper on industrial experience with mdo, 1999. AIAA Technical Committee on Multidisiplinary Design Optimization.

[3] Anon. *Piaggio P180 Avanti II Specification and Description.* Piaggio Aero Industries SpA, Januari 2005.

[4] Anon. Matlab *r2013b (8.2.0.701).* The MathWorks Inc., Natick, Massachusetts, August 2013.

[5] Anon. *Worker Object Wrapper.* The MathWorks Inc., Natick, Massachusetts, November 2013. http://www.mathworks.com/matlabcentral/fileexchange/31972.

[6] Boggs, P., and Tolle, J. Sequential quadratic programming. *Acta Numerica* (1996).

[7] Breitkopf, P., and Coelho, R., Eds. *Design Optimization in Computational Mechanics.* Wiley, 2010.

[8] Campolongo, F., and Cariboni, J. Sensitivity analysis: how to detect important factors in large models. Tech. Rep. JRC37120, European Comission, Directorate-General Join Research Centre, 2007.

[9] Campolongo, F., Cariboni, J., Saltelli, A., and Schoutens, W. Enhancing the morris method. In *Sensitivity Analysis of Model Output* (2005).

[10] Conn, A., Gould, N., and Toint, P. A globally convergent augmented lagrangian algorithm for optimization with general constraints and simple boundsle. *SIAM Journal on Numerical Analysis 28* (1991), 545–572.

[11] Drela, M. *Athena Vortex Lattice (AVL).* Massachusetts Institute of Technology, http://web.mit.edu/drela/Public/web/avl/avl_doc.txt. Last accessed: 16/4/2014.

[12] El-Mihoub, T., Hopgood, A., Nolle, L., and Battersby, A. Hybrid genetic algorithms: A review. Engineering Letters, 2006.

[13] Elham, A. *Weight Indexing for Multidisciplinary Design Optimization of Liing Surfaces.* PhD thesis, Delft University of Technology, 2013.

[14] Elmendorp, R. Synthesis of novel aircraft concepts for future air travel. Master's thesis, Delft University of Technology, January 2014.

[15] Gill, P., Murray, W., and Saunders, M. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM Journal on Optimization 12* (2002), 979–1006.

[16] Gill, P., Murray, W., and Wright, M. *Practical Optimization.* Emerald Group, January 1982.

[17] Martins, J., and Lambe, A. Multidisciplinary design optimization: A survey of architectures. *AIAA Journal 51* (2013), 2049–2075.

[18] Morris, M. Factorial sampling plans for preliminary computational experiments. In *Technometrics* (May 1991), vol. 33 No. 2, pp. 161–174.

[19] Nangia, R. Efficiency parameters for modern commercial aircraft. *The Aeronautical Journal* (August 2006).

[20] Parte, Y., Auroux, D., Clément, J., Masmoudi, M., and Hermetz, J. *Collaborative optimization.* Multidisciplinary design optimization in computational mechanics. Wiley-ISTE, April 2010, pp. 321–368.

[21] Rhodes, D., Jenkinson, L., and Simpkin, P. *Civil Jet Aircraft Design.* Elsevier, 1999.

[22] Roux, É. *Avions civils à réaction : plan 3 vues et données caractéristiques.* É. Roux, 2007.

[23] Sadraey, M. *Aircraft Design: A Systems Engineering Approach.* Wiley, 2012.

[24] Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. *Sensitivity analysis in practice: a guide to assessing scientific models.* Wiley, 2004.

[25] Schmidt, R. A semi-analytical weight estimation method for oval fuselages in novel aircraft configurations. Master's thesis, Delft University of Technology, October 2013.

[26] Taherdangkoo, M., Paziresh, M., Yazdi, M., and Bagheri, M. An efficient algorithm for function optimization: modified stem cells algorithm. *Central European Journal of Engineering* (2013), 36–50.

[27] Torenbeek, E. *Synthesis of Subsonic Airplane Design.* Springer, 1982.

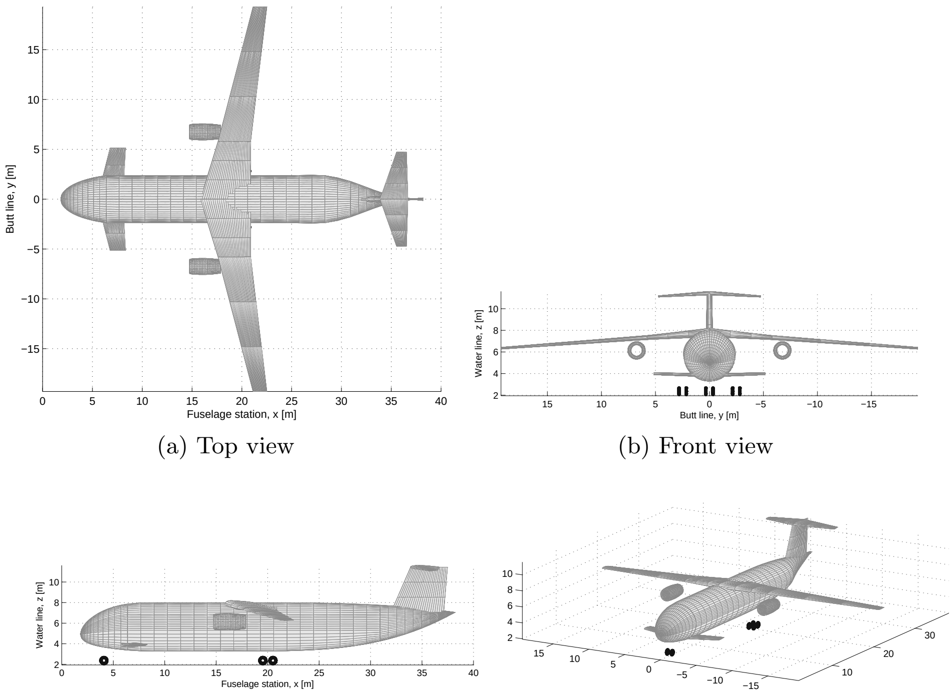
[28] Wright, J., and Alajmi, A. The robustness of genetic algorithms in solving unconstrained building optimization problems. In *Building Simulation 2005* (2005), Ninth International IBPSA Conference.

[29] Zhang, K. *Concurrent subspace optimization for aircraft system design, Aeronautics and Astronautics.* InTech, 2011.

[30] Zingg, D., Nemec, M., and Pulliam, T. A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization. *European Journal of Computational Mechanics* (2008), 103–126.

Appendix A

Example three-surface aircraft report



(a) Top view

(b) Front view

(c) Side view

(d) 3D view

Figure A.1: Aircraft geometry (all dimensions in meters)

A.1 General Characteristics

Aircraft "TSA" generated by the Initiator version . The aircraft is a three-surface aircraft with a high wing and an aspect ratio of 11.7. The aircraft is designed to transport 150 passengers with a total payload mass of 20536kg over 2870km.

A.2 Specification

Table A.1: Max payload

Pax	150	-
Payload Mass	20536	kg
Cruise Mach	0.78	-
Altitude	11278	m
Range	2870	km
Take Off Distance	2180	m
Landing Distance	1440	m

A.3 Optimiser

Table A.2: Optimiser results

Algorithm	Gradient		
Objective value	PRE	7570	km
Design variable 1	Aspect ratio	11.7	-
Design variable 2	Wing x-position	0.47	-
Design variable 3	Sweep angle	12.9	°
Design variable 4	Dihedral angle	-3.6	°
Design variable 5	Fuselage diameter	4.7	m

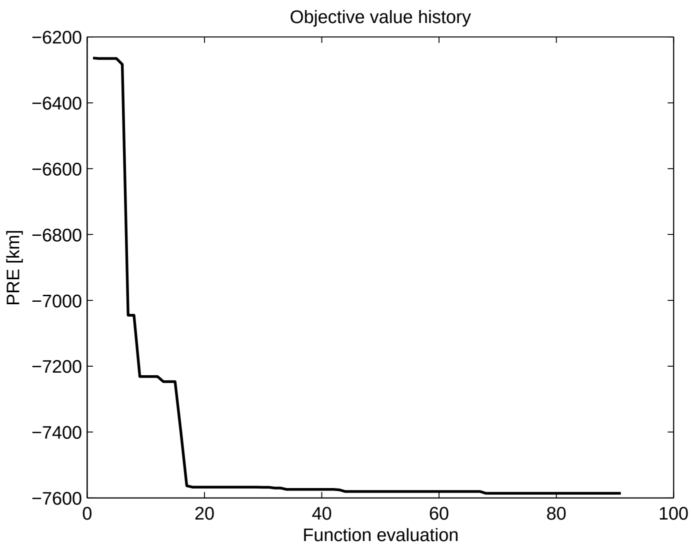
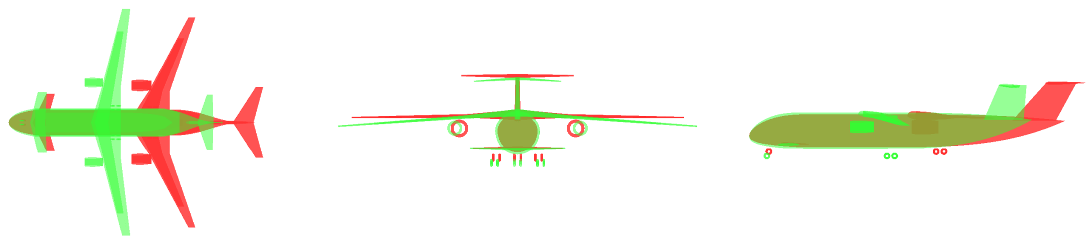


Figure A.2: Objective value history



(a) Top view (b) Front view (c) Side view

Figure A.3: Aircraft geometry changes

A.4 Operational Performance

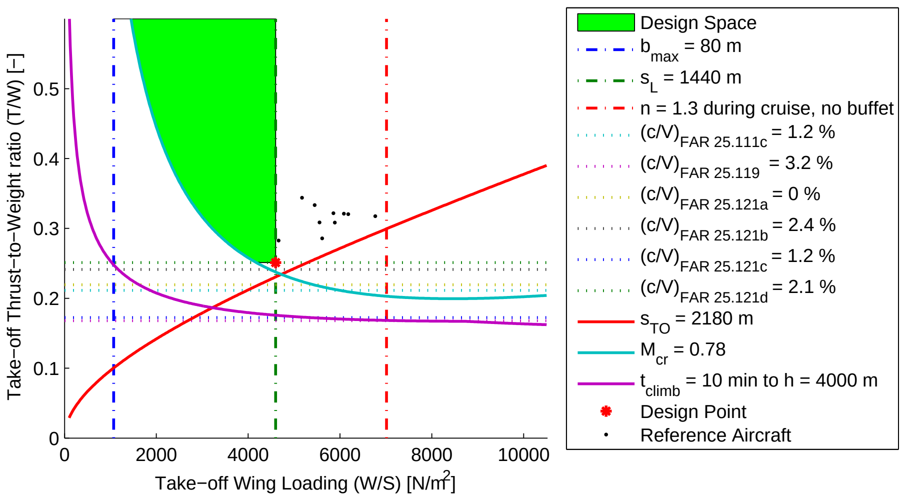


Figure A.4: Loading Diagram

Result: Wing loading at MTOM: 4597 N/m^2
Thrust-to-weight ratio: 0.251 -

Table A.3: Performance results

L/D_{cruise}	17.3	-
Cruise altitude	11278	m
Maximum take-off mass	59640	kg
Operational empty mass	31280	kg
Payload mass	20540	kg
Fuel mass	7830	kg
Harmonic range	2890	km
Ferry range	5730	km
Maximum fuel range	5050	km

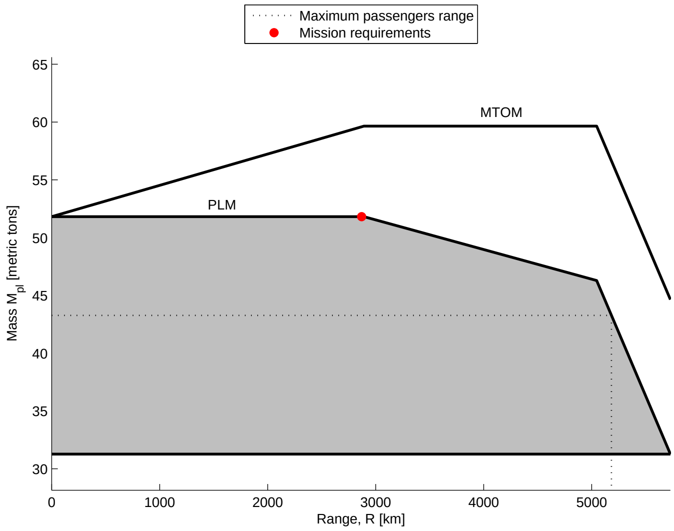


Figure A.5: Payload-Range

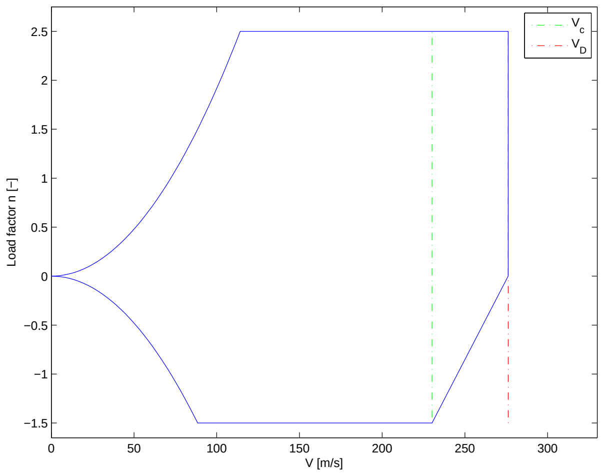


Figure A.6: V-n diagram

A.5 Weight estimation

Table A.4: Mass summary

Pax	12000	kg
Cargo	8540	kg
DLM	53040	kg
Diversion FM	0	kg
End Cruise Mass	53570	kg
Extension FM	0	kg
FM	7830	kg
Initial Cruise Mass	58160	kg
Loiter FM	0	kg
MLM	54880	kg
MRM	60860	kg
MTOM	59640	kg
Max FM	11280	kg
Mission FM	6610	kg
OEM	31280	kg
PLM	20540	kg
Reserve FM	0	kg
ZFM	51810	kg

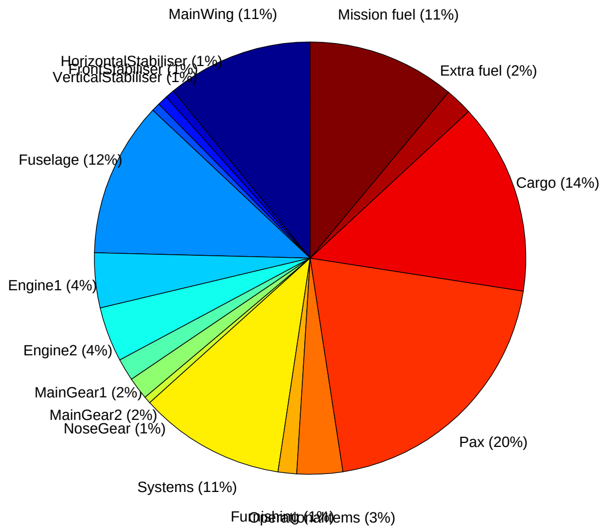


Figure A.7: Mass distribution

Table A.5: Component masses

Engine1	1974	kg
Engine2	1974	kg
Front Stabiliser	457	kg
Furnishing	830	kg
Fuselage	6914	kg
Horizontal Stabiliser	403	kg
Main Gear1	990	kg
Main Gear2	990	kg
Main Wing	6523	kg
Nose Gear	317	kg
Vertical Stabiliser	375	kg
APU	1837	kg
Air Conditioning	1166	kg
Anti Ice	119	kg
Avionics	766	kg
Electrical	395	kg
Flight Controls	242	kg
Fuel System	79	kg
Handling Gear	18	kg
Hydraulics	1804	kg
Instruments	111	kg

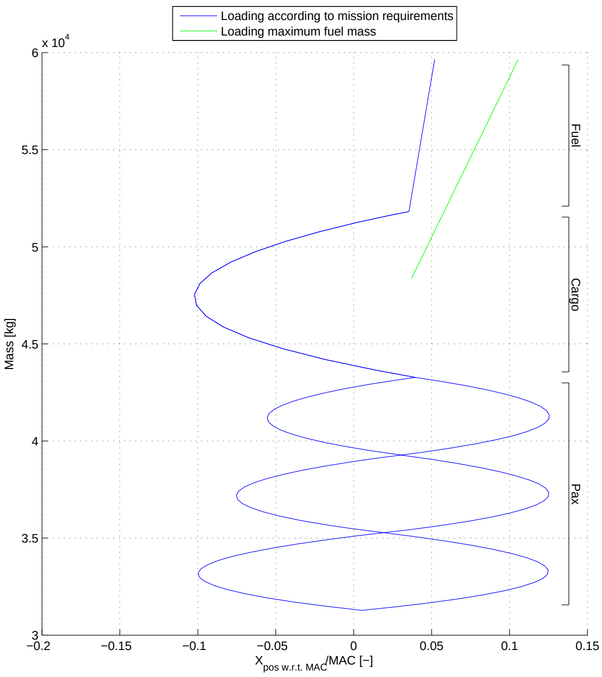


Figure A.8: Loading diagram

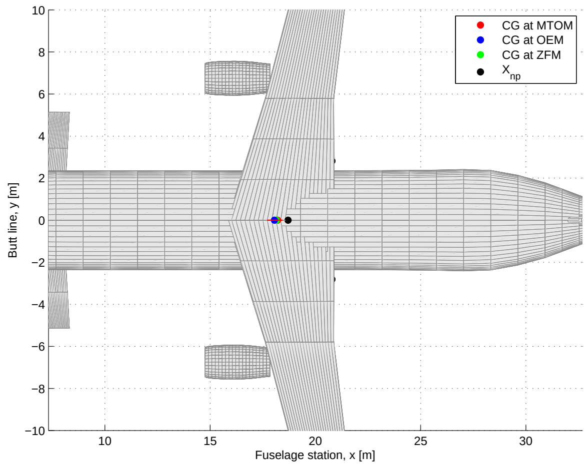


Figure A.9: CG location

Table A.6: Centre-of-gravity locations

X_{cg} (MTOM)	16.4	m
X_{cg} (OEM)	16.2	m
X_{cg} (ZFM)	16.3	m
X_{np}	16.9	m
SM	15	%

A.6 Aerodynamics

Table A.7: Aerodynamic properties at cruise

$C_{L,\text{cruise}}$	0.5	-
$C_{D,\text{cruise}}$	288	cts
L/D_{cruise}	17.3	-
C_{D_0} (Clean)	204	cts
C_{D_0} (Take-Off)	549	cts
C_{D_0} (Landing)	1049	cts
Oswald factor (e) (Clean)	0.799	-
Oswald factor (e) (Take-Off)	0.849	-
Oswald factor (e) (Landing)	0.899	-
C_{L_α}	5.42	rad^{-1}
C_{m_α}	-0.828	rad^{-1}
$C_{L_{\text{max,clean}}}$	1.44	-
$C_{L_{\text{max,take-off}}}$	2.2	-
$C_{L_{\text{max,landing}}}$	2.8	-

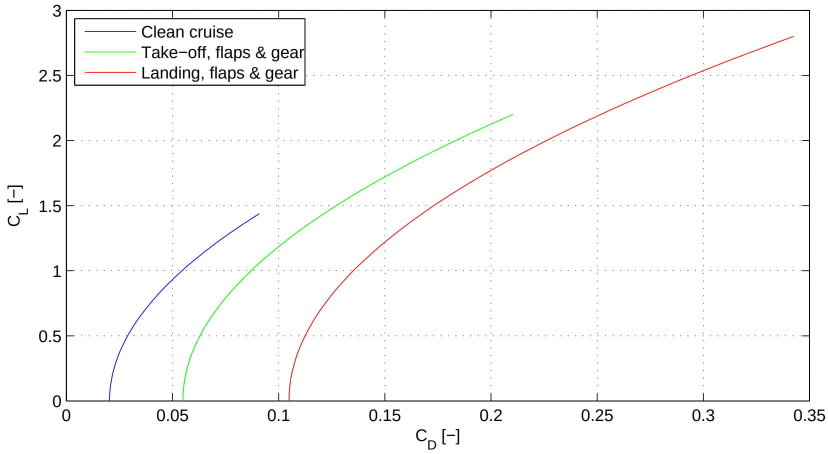


Figure A.10: Drag Polars

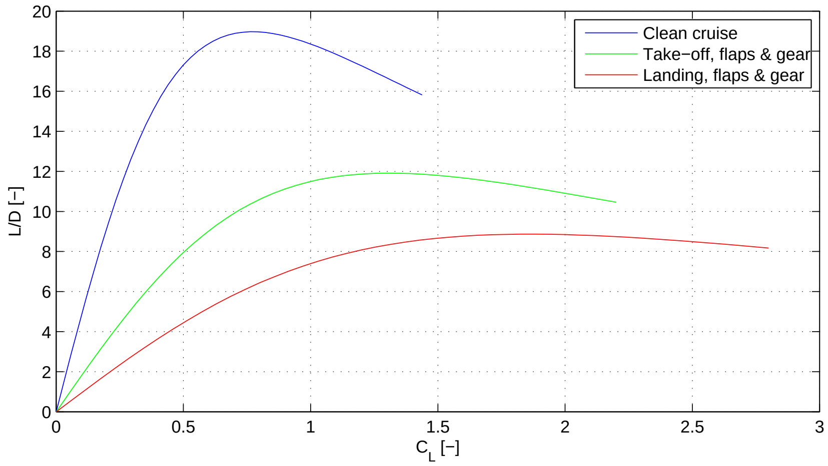


Figure A.11: Aerodynamic efficiency of the aircraft

A.7 Propulsion

Table A.8: Propulsion

Number of engines	2	-
SFC_{cruise}	0.575	h^{-1}
Bypass Ratio	6	-
Diameter	1.6	m
Length	3.13	m

A.8 Aircraft Geometry

Table A.9: Main Wing dimensions

Span	38.6	m
Planform area	111	m^2
MAC	3.24	m
Root Chord	4.37	m
Root t/c	0.151	-
Tip Chord	1.39	m
Tip t/c	0.103	-
Sections (root to tip)	boeing-a, boeing-b, boeing-c	
Sweep 0.25c	12.9	$^\circ$
Taper ratio	0.318	-
Twist	3.9e-15	$^\circ$
Dihedral	-3.6	$^\circ$

Table A.10: Horizontal Stabiliser dimensions

Span	9.44	m
Planform area	17.65	m^2
MAC	2.03	m
Root Chord	2.76	m
Root t/c	0.118	-
Tip Chord	0.979	m
Tip t/c	0.118	-
Sections (root to tip)	N0012, N0012	
Sweep 0.25c	14.4	$^\circ$
Taper ratio	0.355	-
Twist	0	$^\circ$
Dihedral	-3.6	$^\circ$

Table A.11: Front Stabiliser dimensions

Span	10.3	m
Planform area	20.99	m^2
MAC	2.1	m
Root Chord	2.54	m
Root t/c	0.118	-
Tip Chord	1.54	m
Tip t/c	0.118	-
Sections (root to tip)	N0012, N0012	
Sweep 0.25c	11.6	$^\circ$
Taper ratio	0.606	-
Twist	6.1e-15	$^\circ$
Dihedral	1.8	$^\circ$

Table A.12: Vertical Stabiliser dimensions

Span	4.37	m
Planform area	18.93	m^2
MAC	4.42	m
Root Chord	5.06	m
Root t/c	0.118	-
Tip Chord	3.6	m
Tip t/c	0.118	-
Sections (root to tip)	N0012, N0012	
Sweep 0.25c	19.4	°
Taper ratio	0.713	-
Twist	0	°
Dihedral	0	°

Table A.13: Fuselage dimensions

Length	36.4	m
Floor Position	-57	% of fuselage height
Diameter	4.7	m
Nose Fineness Ratio	0.18	-
Aft Fineness Ratio	0.55	-
Cabin Height	1.54	m
Nose Length	4.56	m
Aft Cutoff	0.8	-
Aft Ratio	0.05	-

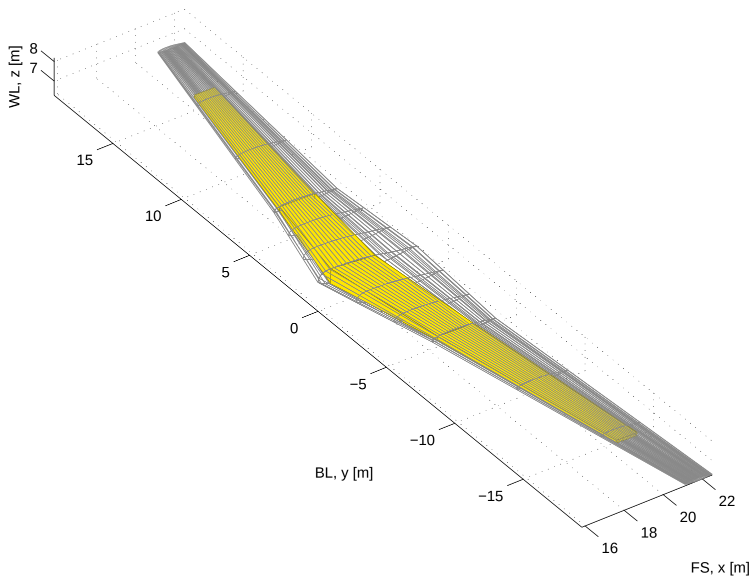


Figure A.12: Fuel tank layout

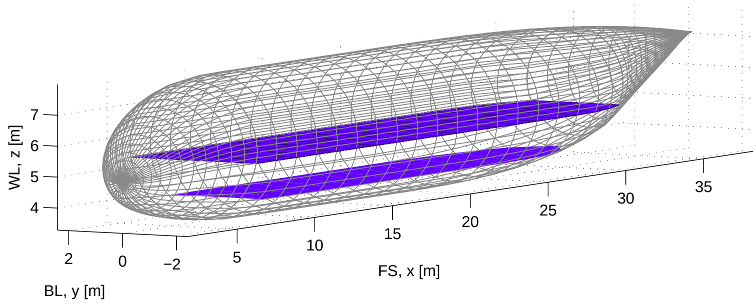


Figure A.13: Fuselage geometry; (blue = cargo ULDs, purple = floors)