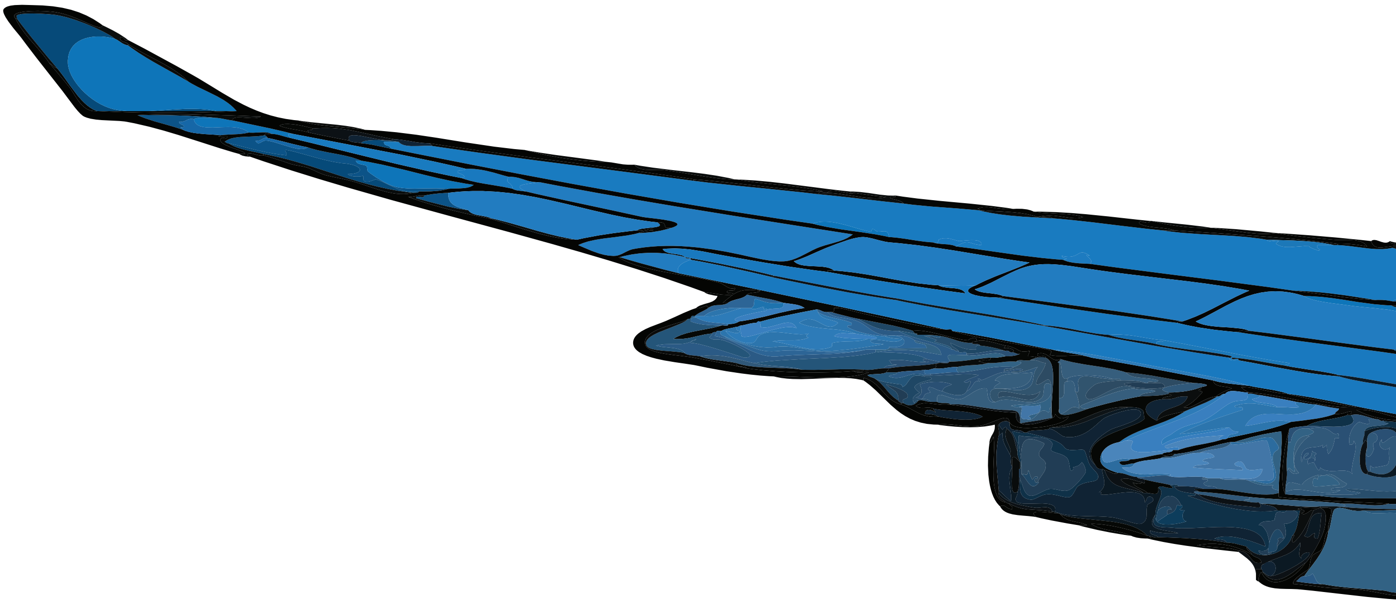
Faculty of Aerospace Engineering

Wing Shape Multidisciplinary Design Optimization

Jan Mariens

August 2, 2012



TU Delft Delft University of Technology

Challenge the future

Wing Shape Multidisciplinary Design Optimization

Master of Science Thesis

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

Jan Mariens

August 2, 2012

Faculty of Aerospace Engineering · Delft University of Technology

TU Delft

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 **"Wing Shape Multidisciplinary Design Optimization"** by **Jan Mariens** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: August 2, 2012

Head of Department: ____________________
Dr. ir. Dries Visser

Supervisor: ____________________
Ali Elham, MSc.

Reader one: ____________________
Prof. dr. ir. Egbert Torenbeek

Reader two: ____________________
Dr. ir. Roelof Vos

Summary

Multidisciplinary design optimizations have shown great benefits for aerospace applications in the past. Especially in the last decades with the advent of high speed computing. Still computational time limits the desire for models with high level of fidelity cannot be always fulfilled. As a consequence, fidelity is often sacrificed in order to keep the computing time of the optimization within limits. There is always a compromise required to select proper tools for an optimization problem.

In this final thesis work, the differences between existing weight modeling techniques are investigated. Secondly, the results of using different weight modeling techniques in multidisciplinary design optimization of aircraft wings is compared. The aircraft maximum take-off weight was selected as the objective function. The wing configuration of a generic turboprop and turbofan passenger aircraft were considered for these optimizations. This should aid future studies of wing shapes in early design stages to select a proper weight prediction technique for a given case. A quasi-three-dimensional aerodynamic solver was developed to calculate the wing aerodynamic characteristics. Various statistical prediction methods (low level of fidelity) and a quasi-analytical method (medium level of fidelity) are used to estimate the structural wing weight. Furthermore, the optimal wing shape was found using a local optimization algorithm and is compared to the results found using a novel optimization algorithm to find the global optimum.

The quasi-three-dimensional aerodynamic solver was validated using experimental data and other available aerodynamic tools. Compared to the results generated by other tools, the developed solver has a wider range of validity. Most important of all, it is up to 10 times faster and the results show good agreement with other data. Several test cases were used to prove the robustness and effectiveness of the global optimization algorithm. A comparison of the different weight estimation methods indicated that the lower level fidelity methods are insensitive for some wing parameters. The results of the optimizations showed that the optimum wing shape is affected by the used weight modeling technique. Use of different weight prediction methods strongly affects the computational times and the convergence history. The global optimization algorithm was able to find the global solution for the wing shape optimization. However, the search for the global optimum comes at a cost: the computational time is significantly larger.

Acknowledgements

This graduate thesis report is written as part of a Master Program in Aircraft Design at the Faculty of Aerospace Engineering – Delft University of Technology. The report forms the end of my time as an Aerospace Engineering student. Looking back at the last months, I realize that there a number of people who made all this possible.

First of all, I would like to thank my supervisor Ali Elham, for his invaluable advise, expertise and excellent support during the past year. Furthermore, I would also like to thank professor Egbert Torenbeek for the interesting discussions we had that gave me a lot of new ideas and second thoughts. I also would like to thank the members of my committee: Dries Visser, Ali Elham, Egbert Torenbeek and Roelof Vos.

Many thanks to all of my friends and (ex) fellow students of room NB2.40, whom made it besides a busy period also very pleasant one. A very special thanks goes to my parents, who have always supported me in everything I did and encourage my dreams. Finally, I would like to address a special person, my girlfriend Catherine for her continuing support, involvement and encouragements.

Jan Mariens
Deflt, July 2012

Contents

List of Figures

List of Tables

Nomenclature

Latin symbols

(t/c)	Airfoil thickness-to-chord ratio	[-]
AR	Aspect ratio of the wing	[-]
b	Span width	[m]
b_s	Structural span	[m]
$\bar{c}$	Mean aerodynamic chord	[m]
c	Chord length	[m]
C_d	Local two-dimensional drag coefficient	[-]
C_D	Drag coefficient	[-]
C_{D_c}	Compressibility (or wave) drag coefficient	[-]
C_{D_i}	Induced drag coefficient	[-]
C_{D_f}	Friction drag coefficient	[-]
C_{D_p}	Pressure drag coefficient	[-]
C_{D_prof}	Profile drag coefficient	[-]
C_p	Local two-dimensional pressure coefficient	[-]
C_L	Lift coefficient	[-]
d	Local two-dimensional drag	[N/m]
D	Aerodynamic drag force	[N]
g	Gravitational acceleration	[m/s^2]
h	Flight altitude	[m]
K	Number of observations	[-]
l	Local two-dimensional lift coefficient	[N/m]
L	Aerodynamic lift force	[N]
M	Mach number	[-]
M_ff	Mass fuel fraction	[-]
n_ult	Ultimate load factor	[-]
Nb	Number of spanwise vortices	[-]
Nc	Number of chordwise vortices	[-]
Nw	Number of wing sections	[-]
r	Radius	[m]

Re	Reynolds number	[-]
S	Wing planform area	[m^2]
S_{ref}	Reference surface area	[m^2]
V	Velocity	[m/s]
V	Volume	[m^3]
W	Weight	[N] or [kg]
$\boldsymbol{x}$	Design vector	[-]

Greek symbols

α	Angle of attack	[°]
α_{eff}	Effective angle of attack	[°]
ϵ	Twist angle at local wing section	[°]
η	Dimensionless spanwise position from root till tip (0...1)	[-]
λ	Wing taper	[-]
μ	Dynamic viscosity	[kg/s/m]
ρ	Air density	[kg/m^3]
σ	Standard deviation	[-]
ξ_c	Constant chord percentage of sweep line	[-]
Λ	Sweep angle	[°]

Subscripts

$\perp$	Normal to sweep line (constant chord percentage line over wing)
∞	Free-stream
av	Available
ave	Average
$c/2$	Mid-chord
$c/4$	Quarter-chord
calc	Calculated
des	Design
eff	Effective
f	Fuel
f	Friction
g	Gaussian
i	Induced
in	Inner wing
k	Kink
out	Outer wing
p	Pressure
prof	Profile
r	Root

ref	Reference
rest	Rest
t	Tip
to	Take-off
w	Wing or wave (compressibility)
zf	Zero-fuel

Abbreviations

AAO	All-at-once
BLISS	Bi-Level Integrated System Synthesis Method
CAD	Computer Aided Design
CFD	Computations Fluid Dynamics
CO	Collaborative Optimization
CPU	Central Processing Unit
CSSO	Concurrent Subspace Optimization
DNS	Direct Numerical Solution
FDM	Finite Difference Method
FEM	Finite Element Method
FVM	Finite Volume method
GSE	Global sensitivity equation
IDF	Individual Design Feasible
LLT	Lifting-Line Theory
LOCSMOOTH	Local Optima Smoothing for Global Optimization
NS	Navier-Stokes
MDA	Multidisciplinary Design Analysis
MDF	Multidisciplinary Design Feasible
MDO	Multidisciplinary Design Optimization
RANS	Reynolds-averaged Navier-Stokes
SQP	Sequential Quadratic Programming
VLM	Vortex Lattice Method

Chapter 1

Introduction

Aircraft wing design using Multidisciplinary Design Optimization (MDO) techniques is a complex task which involves different disciplines, mainly aerodynamic and structure. Different levels of analysis are used for wing design and optimization. Typically simple empirical methods are used in the earliest stages of the concept design. The design task proceeds towards the final design by increasing the complexity of the analysis methods. For instance, a variety of methods are available for aerodynamic analysis of a wing; from a simple lifting line theory or a vortex lattice method up to complex Euler and Reynolds-Average Navier-Stokes methods. Similarly for structural weight estimation, various methods with different levels of fidelity are available. The difficulty lies in the quest or development of analysis methods which are sufficiently simple to be used thousands of times during the optimization. At the same time, these methods should be sophisticated enough to capture changes in the local geometry. In this chapter, the thesis objective is stated, followed by the approach used to answer the thesis objective.

As different analysis methods come with different levels of fidelity, this can influence the optimization results using multidisciplinary design optimization techniques. Therefore, the main objective of this thesis is to:

Investigate the effect of using different weight estimation methods on the outcome in a wing design task using multidisciplinary design optimization techniques.

This main objective is accompanied by the following sub-goals:

- Develop a quasi-three-dimensional aerodynamic solver to calculate the wing aerodynamic characteristics.
- Compare the different weight estimation methods, with low and medium levels of fidelity, by analyzing their accuracy and sensitivity to wing parameters.
- Implement a global optimization algorithm that uses gradient-based techniques and local optima smoothing. The results using this algorithm are also compared to a local solution.

The thesis context of this research is discussed in Chapter 2. This context presents a brief discussion on aircraft design processes, the role of MDO in aircraft design and wing design. In Chapter 3, an optimization algorithm to find the global solution is presented. Several test cases were used to show the effectiveness and robustness of this algorithm. The development of the quasi-three-dimensional aerodynamic solver is presented in Chapter 4. Different weight prediction methods are discussed and compared to each other in Chapter 5. Consequently, Chapter 6 presents an aircraft wing shape design task using MDO techniques. In this design task, different weight estimation methods were used and the optimization results were compared. The MDO results generated using a local optimization algorithm are presented and compared to the results found using the global optimization algorithm. Finally, in Chapter 7 the conclusions and recommendations of this thesis research are presented.

Chapter 2

Thesis background

Based on the thesis objectives, a preliminary research was done. This research serves as background information for this thesis. First, the different phases of aircraft design processes are discussed in Section 2-1. Multidisciplinary design optimization (MDO) has shown great benefits for aerospace applications in the past decades. In Section 2-2 the role of multidisciplinary design optimization (MDO) in aircraft design processes is discussed. The objective of this thesis concerns a wing design task using MDO techniques. In Section 2-3, the different aerodynamic methods for calculating the aerodynamic properties of a wing are discussed together with different wing weight prediction techniques.

2-1 Aircraft design process

The complete aircraft design process goes through three distinct phases that are carried out in a sequence. These phases are, in chronological order: conceptual design, preliminary design and detailed design. Discrimination between the three above-mentioned design phases is related to the differences in activities, tools, amounts of people and expertise, time scales, etc. that take place in each process.

Conceptual design

The design process starts with a set of specifications (mission requirements) for a new aircraft. There is a rather concrete goal to which the designers are aiming. The first steps towards achieving that goal constitute the conceptual design phase. Here, within a certain design freedom, the overall shape, size, weight and performance of the new design are determined. The product of the conceptual design phase is a layout (on paper or computer) of the aircraft configuration. This concept might still be slightly changed during the second design phase. However, the conceptual design phase determines such fundamental aspects as the the shape of the wings (swept back, forward sweep or straight), the location of the wings relative to the fuselage, shape and location of the horizontal and vertical tail, use of canard surface or not, etc. The major drivers during the conceptual design process are aerodynamics, propulsion, weights and flight performance [1]. Typical questions designers will have to answer in this phase can be:

- What is actually driving the design?

- What are the most critical requirements?
- Can the design meet the specifications?
- Which of the possible aircraft configurations has the highest potential?

These questions are answered in the conceptual design phase by using tools primarily from aerodynamics, structures, propulsion and flight performance. No part of the design process is ever carried out in a total vacuum unrelated to the other parts.

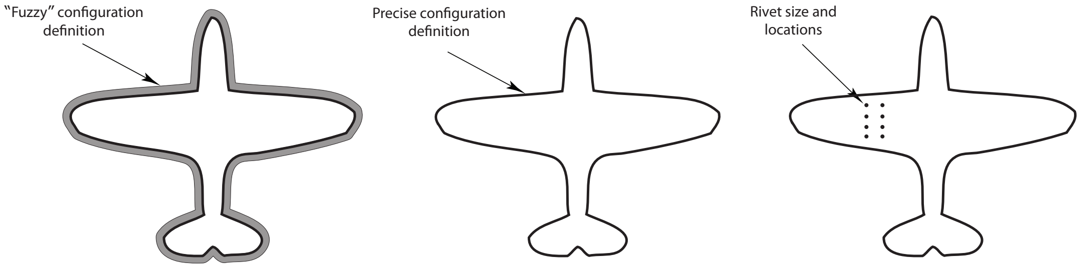
Preliminary design

The preliminary design phase starts at that point when the major changes in the design solutions are over. The preliminary phase uses the baseline configuration that was elaborated and selected during the conceptual phase. The purpose of this phase is to further develop and mature the baseline design, until sufficient understanding (with confidence) of the design quality is achieved. At that point, the design can be frozen and the detail design phase can start. In the preliminary design phase, only minor changes are made to the conceptual design. Questions such as wether to use a canard or an aft tail have been resolved. If major changes were demanded during this phase, the conceptual design process would have been flawed to begin with. The purpose of this phase is to further develop and mature the baseline design, until sufficient understanding (with confidence) of the design quality is achieved. During this phase major computational fluid dynamic (CFD), structural and stress calculations of the aircraft configuration will be made. Additionally, substantial wind tunnel testing will be carried out. It is possible that the wind tunnel tests and/or the CFD calculations will uncover some undesirable aerodynamic interference, or some unexpected stability problems, which will promote changes to the configuration layout. The drawing process called "lofting" is carried out which mathematically models the precise shape of the outside skin of the aircraft, making certain that all sections of the aircraft fit together. Lofting is a term adopted from ship design, where shipbuilders designed the shape of the hull in the loft (an area located above the shipyard floor). At the end of the preliminary design phase, the configuration is frozen and precisely defined. Moreover, the end of this phase brings a major decision – to commit the aircraft to the manufacturer or not. The importance of this decision for modern aircraft manufacturers cannot be understated, considering the tremendous costs involved in the design and manufacture of a new aircraft.

Detailed design

The detail design phase is literally the "nuts and bolts" phase of aircraft design. The aerodynamic, structural, propulsion, performance and flight control analyses have all been finished at the preliminary design phase. In this phase, the design of each individual spar, web, skin, panels, etc. can now take place. The size, number and location of fasteners (rivets, joints, etc.) are determined. Manufacturing tools and jigs are designed. At the end of this phase, the aircraft is ready to be fabricated.

Figure 2-1 is intended to visualize, in a very simple manner, the distinction between the products of the three design phases in aircraft design. The product of conceptual design is represented in Figure 2-1a. Here, the basic configuration of the aircraft is determined within a certain (hopefully small) "fuzzy" latitude. Figure 2-1b shows the product of the preliminary design phase (with precise dimensions). Finally, the product of the detailed design is represented in Figure 2-1c. Here, the precise fabrication details are determined, represented by the rivets sizes and locations.


"Fuzzy" configuration definition
Precise configuration definition
Rivet size and locations

(a) Product of conceptual design

(b) Product of preliminary design

(c) Product of detailed design

Figure 2-1: Schematic illustrating the difference between the design phases. (Adapted from [1])

2-2 Role of MDO in aircraft design

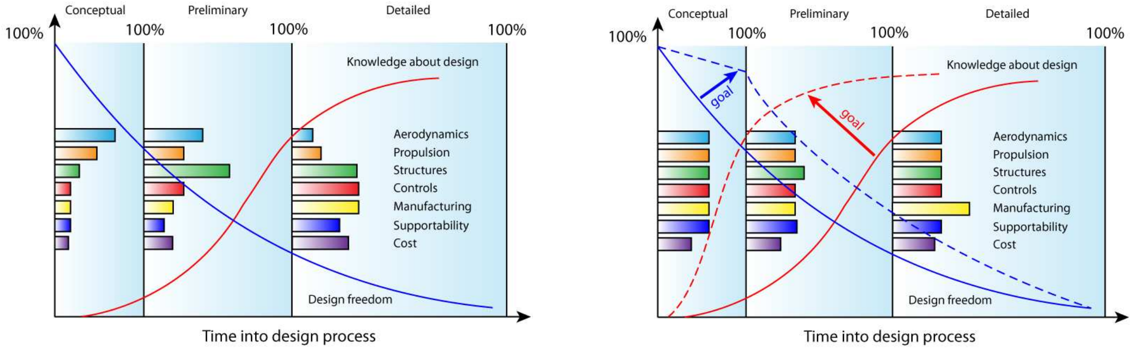
The traditional approach in aircraft design consists roughly out of 20% creative work and 80% repetitive work [2]. The repetitive work load is too high compared to the creative work in engineering. How can the designers productivity be improved while simultaneously reducing the whole design process duration? Multidisciplinary design optimization (MDO) provides a good solution to improve this. In this section, it is explained why this methodology provides a good solution and what different strategies there are. This methodology is also compared to the traditional process in aircraft design, and what improvements it could bring.

Multidisciplinary Design Optimization or MDO, is a methodology for the design of systems in which strong interaction between disciplines motivates designers to simultaneously manipulate variables in several disciplines [3]. The interdisciplinary coupling is inherent in MDO and shows tougher computational and organizational challenges than single-discipline optimization. The goal of MDO, according to Kroo [4], is to provide a more consistent, formalized method for complex system design than is found in traditional approaches.

Aircraft design optimizations are complex processes because they contain a lot of design variables and interdisciplinary operations. Here, the use and necessity of MDO in aircraft design is highlighted. The traditional aircraft design process for conventional airplanes is shown in Figure 2-2a. In order to cope with a more complex design procedure, the need of a different design process arises. A new methodology is needed, that is able to facilitate the application of new technologies for aircraft design. This methodology should also allow the user to acquire more design freedom and more knowledge about the design during the conceptual design phase. Figure 2-2b shows graphically the difference between the traditional design process and the preferred process. In order to establish this methodology, the new design process will enhance MDO techniques.

2-2-1 MDO strategies

The MDO architectures those are used to solve a problem, can be divided into two classes: monolithic formulations and multilevel formulations [6]. The term architecture refers to how the multidisciplinary system is decomposed and the optimization formulation employed to meet design requirements. Monolithic formulations, which among others include the multidisciplinary design feasible (MDF), individual discipline feasible (IDF) and all-at-once (AAO), use a single-system level optimizer for the whole problem. In all single level system architectures, the design task (also-called the decision authority) is in the hands of the optimizer, that means that the optimizer controls the design variables [2]. These approaches are the most straightforward to implement for small



(a) Traditional aircraft design process

(b) Preferred aircraft design process

Figure 2-2: Aircraft design processes. (Source: [5])

problems. However, for real life design processes, it is often harder to implement these approaches, which is partly due to the presence of centralized decision authority (a single system optimizer that controls all the design variables). As the system scale increases, this problem becomes more apparent [2].

Multilevel architectures such as collaborative optimization (CO), concurrent subspace optimization (CSSO) and bi-level integrated systems synthesis (BLISS) make use of subspace optimization to promote discipline autonomy [6]. This means that the system-level optimizer is then responsible for managing the interactions between the discipline optimizations. Multilevel strategies are more advanced compared to single system-level because they employ disciplinary optimizers in addition to the system optimizer.

MDO can also be defined as a methodology together with a set of tools for assistance in the design of complex coupled systems, that is, systems whose behavior is governed by many distinct but interacting physical phenomena [4].

Initial applications of multidisciplinary design optimization involve the direct integration of multiple disciplinary analyses and an optimizer [7] .

Multidisciplinary Feasible Design

The multidisciplinary feasible design (MDF) has the simplest formulation for solving MDO problems. The MDF formulation links a multidisciplinary design analysis (MDA) with an optimizer (see Figure 2-3) to find the optimal global z and local variables x, for a given objective function and constraints. The system optimizer guides the MDA and reaches a multidisciplinary feasible state for an entire set of disciplines. In this method, the word feasible refers to consistency. A disadvantage of this approach is that the solution of the system could be expensive and it does not exploit the potentially weak coupling between some of the disciplines that would enable the division into different analyses modules that might run in parallel [8]. The MDF approach can be stated as:

$$\begin{aligned} \min_{z,x} \qquad & f\left(z,x,y\left(x,y,z\right)\right) \\ \text{subject to} \qquad & c\left(z,x,y\left(x,y,z\right)\right) \leq 0 \end{aligned}$$

where f represents the objective function and c all the global and local system constraints.

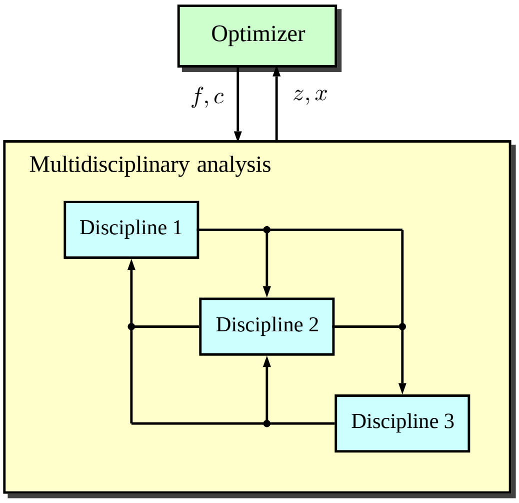


Figure 2-3: Multidisciplinary feasible method

Individual discipline feasible

The main idea of the individual discipline feasible strategy is to use the optimizer to enforce inter-disciplinary compatibility. Instead of iterating the multidisciplinary analysis to converge the coupling variables y, these coupling variables are given by the optimizer as a guess, y'. The optimization problem can then be stated as follows,

$$\begin{aligned} \min_{z,x,y'} \quad & f\left(z, y\left(x, y', z\right)\right) \\ \text{subject to} \quad & c\left(z, y\left(x, y', z\right)\right) \leq 0 \\ & y' - y(x, y', z) = 0 \end{aligned}$$

The number of design variables has increased and equals the number of original design variables plus the number of coupling variables. On the one side this increases the size of the optimization problem, but on the other side it conveniently decouples all the analyses. This decoupling enables it to solve the problem in parallel without intercommunication. Note that if the optimizer is gradient-based, the gradient $\partial f/\partial y'$ and $\partial c/\partial y'$ must also be calculated. This method is particularly advantageous for cases where there is a small number of coupling variables.

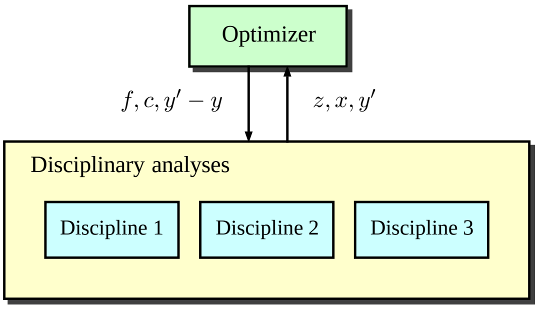


Figure 2-4: Individual discipline feasible method

All-At-Once

The all-at-once architecture (also known as simultaneous analysis and design – SAND) goes a step beyond IDF. It decomposes the multidisciplinary problem further by setting the governing equation for each discipline as equality constraints in the optimization problem. The AAO strategy can be written as a single optimization problem:

$$\begin{array}{ll} \min\limits_{z,x,y} & f\left(z,x,y\left(z,x\right)\right) \\ \text{subject to} & c\left(z,x,y\left(z,x\right)\right) \leq 0 \\ & \mathcal{R}_i\left(z,x_i,y_i(z,x,y_i)\right) = 0 \qquad i = 1,\ldots,n \end{array}$$

where n represents the number of disciplines and $\mathcal{R}_i$ the residuals of the governing equations for each discipline. In general, this architecture is impractical for MDO that involves large sets of governing equations due to the excessive number of design variables that it adds to the optimization problem [6]. Another drawback to this architecture is that it evaluates the residuals of the analysis equations, rather than solving some set of equations [9].

Collaborative Optimization

The collaborative optimization architecture is designed to provide disciplinary autonomy while achieving interdisciplinary compatibility. The optimization problem is decomposed into a number of independent optimization subproblems, each corresponding to a discipline. The objective of each subproblem is to agree on the values of the coupling variables with the other disciplines. Each discipline is given control over its design variables and is responsible for satisfying its constraints. The CO formulation at system level can be stated as:

$$\begin{array}{ll} \min\limits_{z,y} & f\left(z,y\right) \\ \text{subject to} & J_i^*\left(z,z^*,y,y^*\left(x_i^*,y,z_i^*\right)\right) = 0, \qquad i = 1,\ldots,n \end{array}$$

where J_i^* represents the measure of the interdisciplinary discrepancy for the i-th discipline after solving the disciplinary subproblem:

$$\begin{array}{ll} \min\limits_{z,y_i,z_i} & J_i\left(z,z_i,y,y_i\left(x_i,y,z_i\right)\right) = \sum\limits_{i=1}^{n}\left(z-z_i\right)^2 + \sum\limits_{i=1}^{n}\left(y-y_i\right)^2 \\ \text{subject to} & c\left(x_i,z_i,y_i\left(x_i,y,z_i\right)\right) \leq 0 \end{array}$$

Figure 2-5 shows the CO method. From this figure it can be seen that there is no direct communication for each discipline with other disciplines to enforce the governing equations. Albeit that CO does has several benefits, it also comes with some drawbacks. As the number of coupling variables increases, the dimensionality of the system level problem increases as well as the number of variables involved with the calculation of the system level compatibility constraints. For this reason, CO tends to be most effective on problems having a low dimensionality of coupling.

Concurrent Subspace Optimization

The Concurrent Subspace Optimization (CSSO) is also a decomposition-based strategy that allows for the disciplines to run decoupled from each other. Also for this approach, the multiple subspace optimization problems are driven by a system-level optimizer that provides overall coordination.

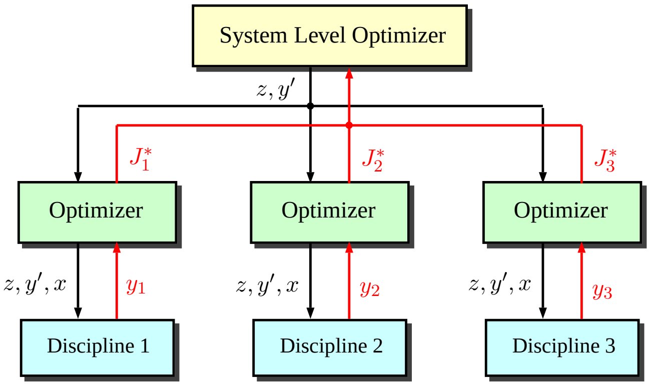


Figure 2-5: Collaborative Optimization method

Each subproblem in CSSO uses approximations to non-local disciplinary coupling variables to estimate the influence of these variables on the system-level objective and constraints. The subspace optimization problem for the i-th discipline is given by

$$\begin{array}{ll} \min\limits_{z,x_i,y_i,\tilde{y}_j} & f\left(z, y_i\left(x_i, \tilde{y}_j, z_i\right), \tilde{y}_j\right) \\ \text{subject to} & c\left(x_i, z, y_i\left(x_i, \tilde{y}_j, z_i\right)\right) \leq 0 \end{array}$$

where $\tilde{y}_j = \tilde{y}_j\left(z, x_j\right)$ represent the approximations to coupling variables (or states) of the other disciplines. The system-level optimizer solves the following problem,

$$\begin{array}{ll} \min\limits_{z,\tilde{y}} & f\left(z, \tilde{y}\right) \\ \text{subject to} & c\left(x_i, z, y_i\left(x_i, \tilde{y}_j, z_i\right)\right) \leq 0 \end{array}$$

After each iteration of the system-level optimizer, a multidisciplinary analysis (MDA) is performed to update the model which gives the approximate response of all coupling variables $\tilde{y}$.

Bi-Level Integrated System Synthesis

The bi-level integrated system synthesis (BLISS) method is a decomposition of the global sensitivity equations (GSE) method. It calculates the total derivative of the coupling values y with respect to local sensitivities. Each discipline is optimized by varying their local variables x, while holding the global variables z constant and simultaneously minimizing the disciplinary objective under local constraints. The global variables are utilized by the system level optimization only. The total derivatives (obtained from GSE) are used to predict the effects of each set of variables on the objective function. The optimization of the i-th discipline takes the form:

$$\begin{array}{ll} \min & d\left(f, x_i\right)^T \Delta x_i \\ \text{subject to} & g_i\left(x_i\right) \leq 0 \end{array}$$

where $d\left(f, x_i\right)^T$ is the local derivative of the objective function with respect to the local variables and disciplines. It includes the indirect effects of these variables on other disciplines. The term $d\left(f, x_i\right)^T \Delta x_i$ corresponds to the first order predicted objective function change caused by the change in x_i. The system level objective in the BLISS formulation is strongly related to the objective functions of the disciplines and it is expressed in terms of a first order Taylor series expansion:

$$\begin{array}{ll} \min & \Phi = d\left(y_{1,i}, x_i\right)^T \Delta x_1 + d\left(y_{1,i}, x_2\right)^T \Delta x_2 + d\left(y_{1,i}, x_3\right)^T \Delta x_3 + \ldots \\ \text{subject to} & g\left(z, y\left(x, z\right), x\right) \end{array}$$

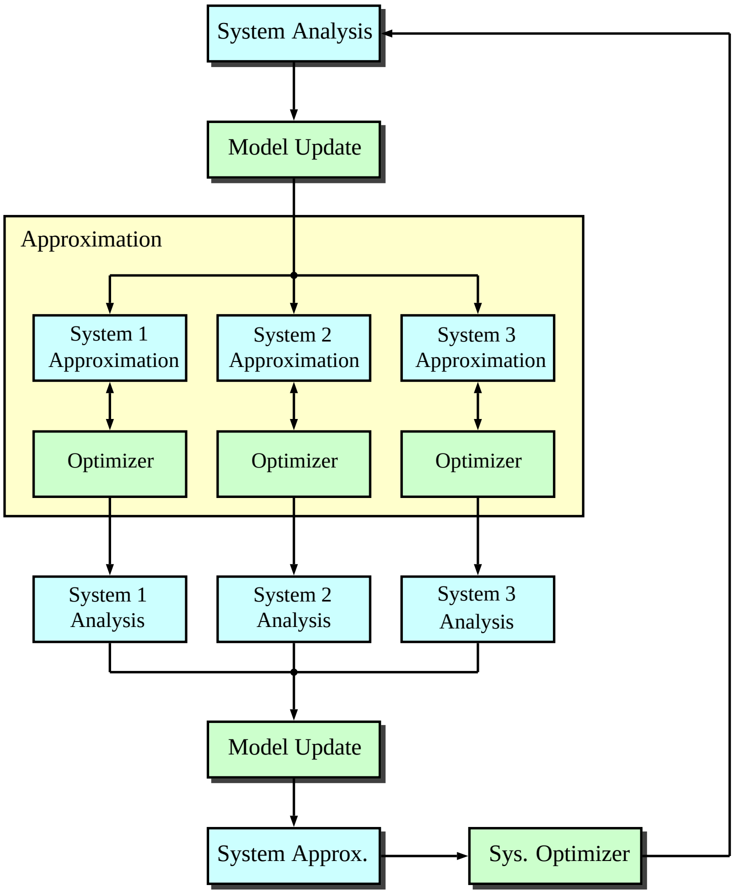


Figure 2-6: Concurrent Subspace Optimization method

2-2-2 Comparison of MDO strategies

As for any optimization problem, the choice of the MDO strategy (method) depends strongly on the problem that is to be solved. It is almost impossible to say in advance which strategy should be used, but nevertheless there are some guidelines which are helpful to decide which strategy would be most likely better to use.

Multilevel strategies are often better suited for use in large complex design problems, because they allow distribution decision authority. However, the multilevel strategies are more prone to convergence issues. The following aspects have been identified that are important for successful implementation in large-scale projects [10]. An ideal MDO strategy should have the following characteristics (based on [2]):

- *Disciplinary design autonomy:* The strategy should allow the use of available expertise and legacy design tools. Local decision authority should be respected.
- *Flexibility:* The strategy should be easily adaptable to a specific organization.
- *Mathematical rigor:* The strategy should yield reliable and consistent results, and the optimality of the results should be provable.
- *Efficiency:* The strategy should lead to an optimal solution in a minimum number of iterations and it should minimize the design time (e.g. by the concurrency of tasks).

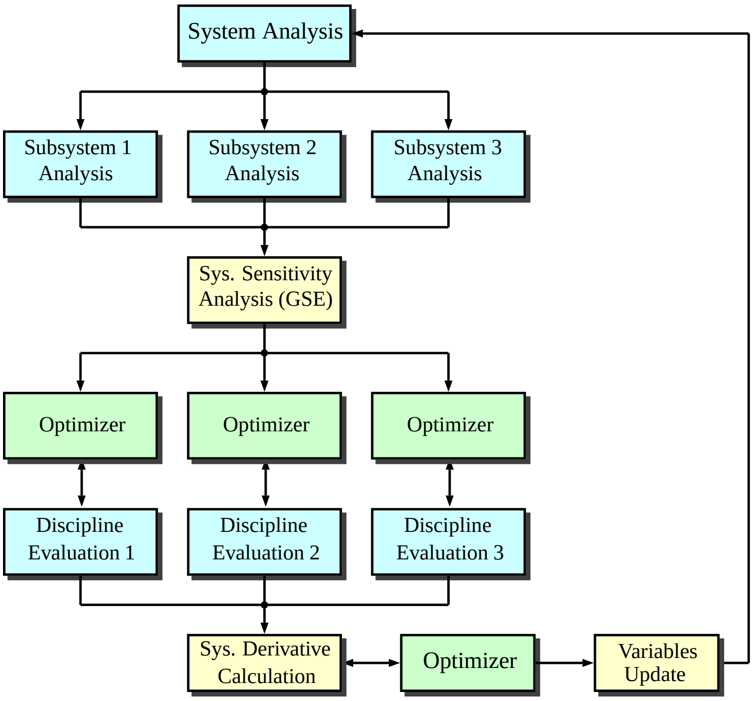


Figure 2-7: Bi-Level Integrated System Synthesis method

During the discussion of the MDO strategies, the optimizer itself was treated as a black box. However, the choice of the optimizer also influences the success of the MDO strategy. The optimization algorithm decides how to move through the design space.

2-2-3 Optimization algorithms

Optimization methods can often yield good designs and sometimes even optimal designs. This rather because they provide a systematic framework for considering some of the many decisions designers are charged with taking. Therefore it is very helpful when looking at computational approaches for designing to have a thorough understanding of the various classes of optimization methods. Also for how they work and more importantly, where they are likely to fail.

When bringing together multiple disciplines in one single computational environment, including the subject of design search and optimization, attention must be paid on the subject of formal optimization methods and their underlying theory. The (numerical) optimization algorithms can be divided into two main categories, the gradient-based and non gradient-based algorithms. In this subsection, these algorithms are discussed. Furthermore, there also exist combinations of gradient-based and non gradient-based, so-called hybrid methods.

Gradient-based algorithms

The gradient-based optimization methods use gradient information of the objective functions to drive the design into direction of improvement. Using the gradient information of the objective functions, the direction of the steepest descent can be determined. Then a single move or an entire line search is performed in the identified steepest direction obtained from the gradient information of the objective function. Following this, the direction of the steepest descent is again determined and the process repeated until optimization criteria is satisfied.

A major drawback of these methods is that they always (directly) steer towards a local minimum, while the local minimum might not be the same as the global optimum (as illustrated in Figure 2-8). Only convergence to a local minimum is guaranteed [11]. Therefore, it is of great importance to chose the initial points in the vicinity of the global minimum. Although, for complex multidisciplinary design optimization processes (such as aircraft design problems) it is not straightforward where this global optimum is to be found within the design space. The method requires existence of continuous first derivatives of the objective function and possibly higher derivatives [11]. The technique used to obtain the gradient information is very important as it has a a large influence the required number of function evaluations for the optimization [2]. Examples of these techniques are steepest descent, conjugate gradients, Newton methods, Quasi-Newton methods, etc. A drawback of the gradient-based method is the inability to explore the design space. This implies that many of the gradient-based methods yield good performance in theory, but less good in practice [12]. This method is significantly faster than non-gradient based algorithms and easier to implement.

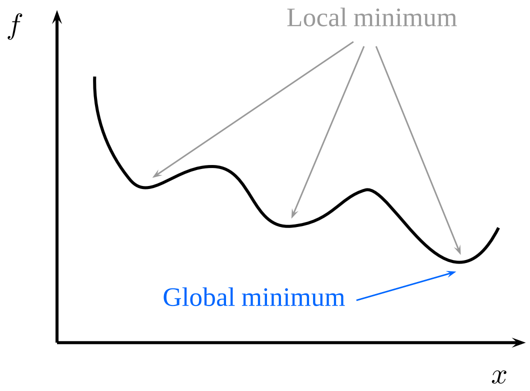


Figure 2-8: Graphical representation of difference between *Local minimum* and *Global optimum*.

Non-gradient algorithms

Non-gradient algorithms overcome the problem related to the complexity of the computation of the partial derivatives of the objective functions. Therefore they are more suitable for global optimization problems. In contrary with the gradient-based methods, this method is able to explore the whole design space and (depending on optimization criteria) to find the global optimum but requires a large number of function evaluations [11]. Usually it is one or two orders of magnitude more expensive (and slower) than gradient-based [13]. Another disadvantage of non-gradient algorithms is that they usually show slow convergence. For aerospace applications, this method has been proven to effective but time-consuming.

2-3 Wing design

In this research an optimization of aircraft wings is conducted. The different methods to predict the wing aerodynamics and to calculate the structural wing weight, are discussed in this section.

2-3-1 Aerodynamics

Aerodynamic drag

Drag is the aerodynamic force that opposes an aircraft's motion through the air. The aim of aerodynamicists (aerospace engineers) always has been to minimize the drag by direct and in-direct methods. The drag force is mathematically defined as

$$D = \frac{1}{2}\rho C_D V^2 S \tag{2-1}$$

where V is the aircraft's velocity, ρ the free-stream air density, S_{ref} the reference surface area and C_D the dimensionless drag coefficient. Drag is what drives the aerodynamic design because it is a measure of how much power is required from propulsion to overcome this drag during its cruise stage (from Eq. (2-1), it can be deduced that the drag increases quadratically with the velocity). Drag is thus directly related the amount of power required, the fuel consumed and resultantly the overall weight of the aircraft.

The drag coefficient C_D is a dimensionless term that quantifies the drag or resistance a particular aircraft has. The lower this coefficient, the "cleaner" the design. Next, an overview of the different types of drag is given.

Types of drag

The drag forces can be divided into subsonic drag and supersonic drag. Figure 2-9 shows a schematic overview of how the different types of drag are divided. An additional drag type that occurs in supersonic and transonic flights is the so-called *wave drag*. Wave drag is the pressure drag that arrises due to the effect of expansion and compressibility waves due to the body shape. In subsonic flight, the drag may be divided into two major categories: *profile drag* and *induced drag*. The profile drag can be subdivided into *skin friction drag*, *parasite drag* and *pressure drag*. Induced drag is caused by lift producing surfaces such as the wing. Those parts of the aircraft that do not produce lift produce non-lift dependent drag also known as *parasite drag*.

These different drag types form the total drag and can the be written in the following mathematical coefficient form

$$C_D = C_{D_{\text{prof}}} + C_{D_i} + C_{D_w} \tag{2-2}$$

where $C_{D_{\text{prof}}}$, C_{D_i} and C_{D_w} are respectively the profile drag, induced drag and wave drag coefficients. The profile drag is a direct function of aircraft wetted area and its aerodynamic contouring. Meanwhile, induced drag account for the losses associated with lift generation.

2-3-2 Aerodynamic flow models

The most fundamental basis of computational fluid dynamics (CFD) problems are the *Navier-Stokes* (NS) equations. The NS equations completely describe the aerodynamics of a fluid (except for the chemical-reaction effects at high temperatures). Though the NS seem straightforward enough, they cannot be analytically solved for any useful flow conditions [15]. While "direct numerical solution" (DNS) codes are the beginning to solve the full NS equations for simplified geometries and conditions. For aircraft design however, there are currently no practical codes to solve the full NS equations. This is mainly due to the difficulty in mathematically analyzing the aerodynamic phenomena turbulence.

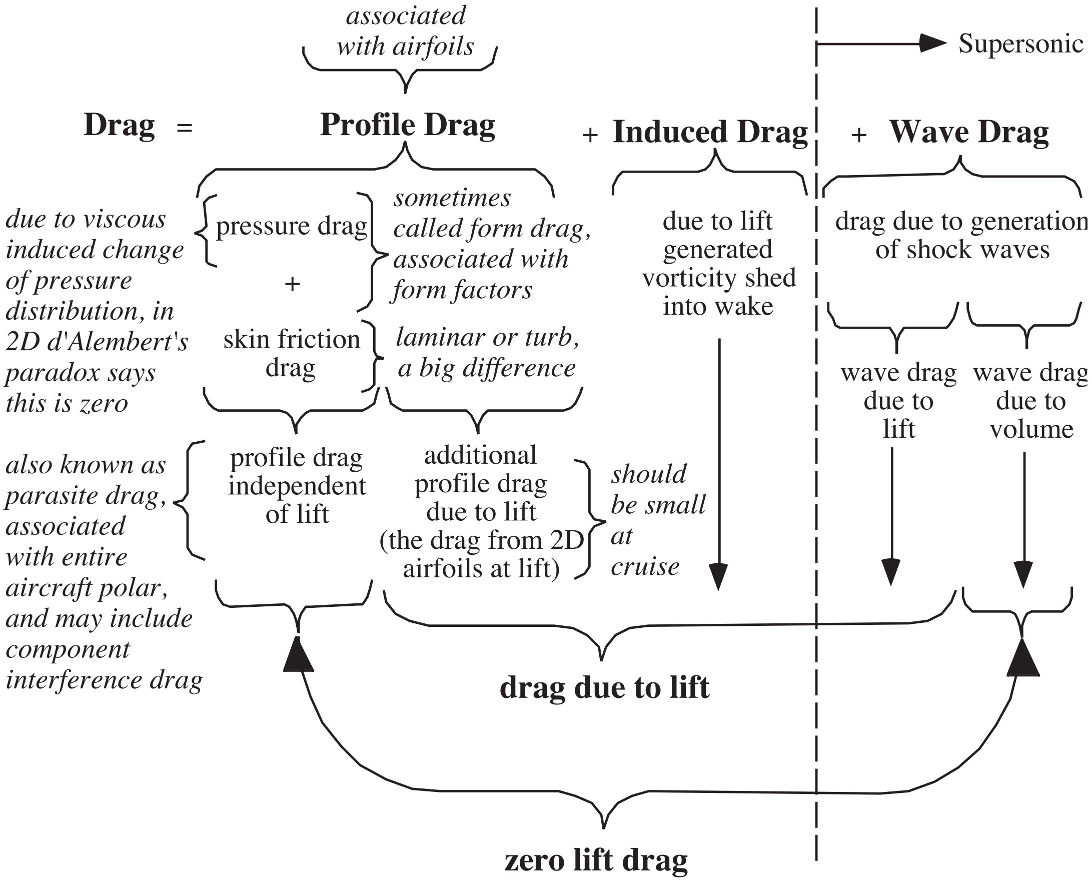

associated with airfoils
Supersonic
Drag = Profile Drag + Induced Drag + Wave Drag
due to viscous induced change of pressure distribution, in 2D d'Alembert's paradox says this is zero
pressure drag
+
skin friction drag
sometimes called form drag, associated with form factors
laminar or turb, a big difference
due to lift generated vorticity shed into wake
drag due to generation of shock waves
wave drag due to lift
wave drag due to volume
also known as parasite drag, associated with entire aircraft polar, and may include component interference drag
profile drag independent of lift
additional profile drag due to lift (the drag from 2D airfoils at lift)
should be small at cruise
drag due to lift
zero lift drag

Figure 2-9: Different types of drag, for both subsonic and supersonic flight regimes [14].

Several simplifications can be made to the NS equations. The *Reynolds-Averaged Navier-Stokes* (RANS) equations are the time-averaged equations of motion for fluid flows which models the turbulence statistically. RANS codes are used for many projects to solve particular design problems where no other methods provide correct results. Unfortunately, it still is really expensive to setup and run the RANS codes.

By ignoring all viscous terms and assuming steady flow, the *Euler equations* can be derived from the NS equations. Euler codes are cheaper in use for simulations (runs) than RANS codes, and are used quite often. Moreover, the inviscid assumption performs well outside the boundary layer. They also can handle vortex formations, and by adding a separate boundary-layer code, they can also realistically estimate viscous and separation effects.

The Euler equations can be simplified further by ignoring rotational terms, yielding the *potential flow equations*. This simplification prevents the analysis of vortex flow, which is important at higher angle of attacks and of less importance at cruise conditions. The potential flow equations have the advantage that they can handle transonic shock formation which makes it very useful for transonic design compared to the linearized methods. Although these equations imply some simplifications, they are widely used in aerodynamic codes that treat the entire flow field instead of just the surface conditions [16].

The *linearized potential flow equations* are obtained by neglecting the higher-order terms in the potential flow equations. Due to the linearization, these equations do not perform well at transonic speeds since it neglects the non-linear terms.

Figure 2-10 shows the hierarchy of the above mentioned aerodynamic solvers. In order to solve the partial differential equations of the aerodynamic models, use can be made of numerical solution techniques.

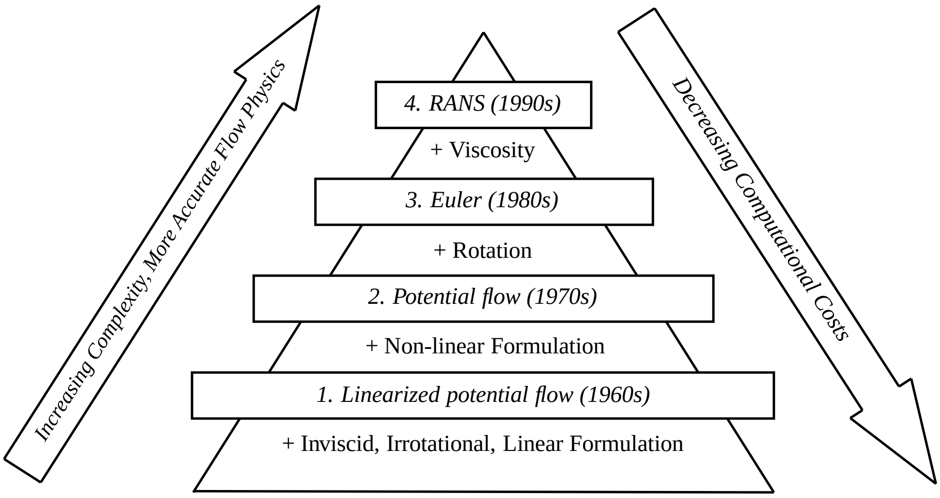

Increasing Complexity, More Accurate Flow Physics
4. RANS (1990s)
+ Viscosity
3. Euler (1980s)
+ Rotation
2. Potential flow (1970s)
+ Non-linear Formulation
1. Linearized potential flow (1960s)
+ Inviscid, Irrotational, Linear Formulation
Decreasing Computational Costs

Figure 2-10: Hierarchy of fluid flow models. (Source: [17])

2-3-3 Numerical methods for solving the fluid flow models

The governing equations (partial differential equations) from the aerodynamic models can be solved using numerical solution techniques. The techniques differ for nonlinear of linear simulations, as will be explained in the next sections. Besides the linear and non-linear methods, semi-empirical models (handbook models and DATCOM) can also be used [18]. Note that non-linear methods usually require more detailed input geometries than linear methods.

Nonlinear 'field' methods

The nonlinear CFD methods can be used for predicting complex flow fields, such as those associated with transonic or separated flows. The nonlinear CFD methods are used in RANS, Euler and (full) potential solvers. The numerical methods that are used most frequently for nonlinear CFD simulations are the finite volume method (FVM), the finite element method (FEM) and the finite difference method (FDM). Among these, there are several more available techniques, although the three techniques mentioned have the broadest applicability (about 95%) [19]. These spatial discretization techniques will be briefly discussed below. FEM, FDM and FVM are called "field methods", because they all discretize the whole fluid domain (field).

Finite difference method Historically, this method is the oldest of the three. The finite difference method is the method that is most used and it was the first one applied to the numerical solution of differential equations. This method is directly applied to the differential form of the governing equations. The first and second derivatives of the differential equations are approximated by a difference formula that can be (easily) derived from a Taylor series expansion [20, 21]. An important advantage of the finite difference methodology is its simplicity for numerical implementation [19]. Another advantage is the possibility to obtain high-order approximations and thus achieving high-order accuracies of the spatial discretization [22]. On the other hand, the method requires a structured grid, which restricts the range of application. And there is no conservation of momentum, energy and mass on coarse grids [19]. Figure 2-11 shows an example of a finite difference mesh and a corresponding stencil (scheme) that can be used for discretization of the nonlinear fluid flow equations.

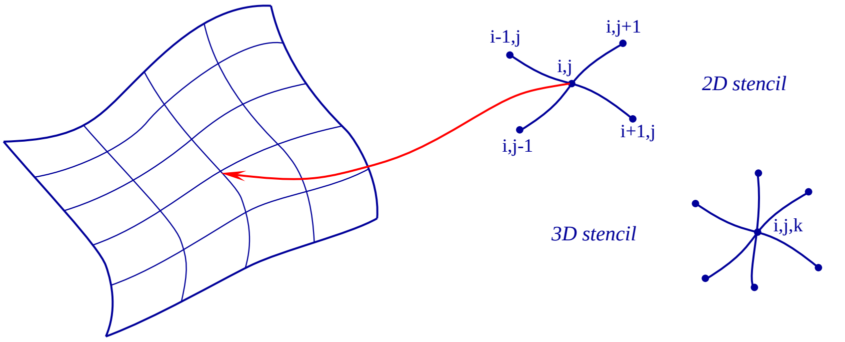


Figure 2-11: Finite Difference mesh and examples of a 2D and 3D stencil

Finite Volume Method This method divides the problem domain into a set of finite volumes (or cells, see Figure 2-12) and the resulting statement expresses the (exact) conservation of relevant properties for each finite size cell (volume). The basic advantage of this method over FDM is that it is applicable for any type of mesh (both structured and unstructured) where mass, momentum and energy are conserved [19, 22]. The clear relationship between the numerical algorithm and the underlying physical conservation principle forms one of the main attractions of the FVM. Therefore, its concepts are much more simple to understand by engineers than the FEM [23]. This is why this method is widely adopted for solving fluid flow problems and implemented in CFD tools. This method is very flexible and so it can be rather easily implemented on structured grids as well as on unstructured grids [22]. Just as the other methods, this method also has some disadvantages. False diffusion often occurs in numerical predictions with FVM, especially when simple numerics are engaged. Besides that, it is also difficult to develop schemes with higher than second-order schemes accuracy for multidimensional problems.

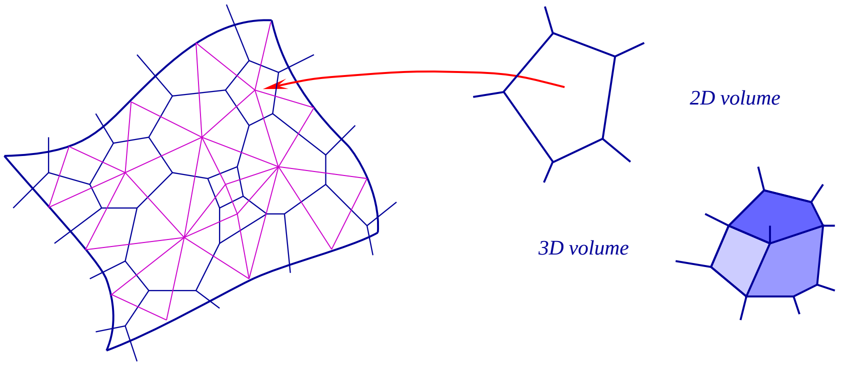


Figure 2-12: Finite Volume mesh and examples of a 2D and 3D volume blocks

Finite Element Method FEM is mostly used for analyzing structural mechanics problems. However, it also found its applicability for fluid flow analyses. In general it is applied to the solution of Euler and NS equations, and the physical space (domain) is subdivided into triangular (in 2D) or into tetrahedral (in 3D) elements (see Figure 2-13). The classical FEM discretization is based on these elements (refer to Figure 2-13) and a piecewise representation of the solution in terms of basis functions. For improved numerical stability, more convincing finite element procedures can be applied [21]. The strength of FEM is its ability to deal with arbitrary geometry using different shapes and orders of element that are usually formed based on a set of unstructured mesh. And just like the FVM, the discretization of the conservation laws allows the treatment of discontinuous solutions such as shocks. FEM can give the highest accuracy on coarse mesh among the three traditional numerical methods [19]. Although it can be shown that in certain cases the method is mathematically equivalent to the FVM, the effort for the numerical implementation is much higher [22]. Another advantage is that it is very effective for diffusion-dominated problems (viscous flow) [19]. On the other hand, the coding of FEM is much more complex when it is used for solving turbulent fluid flow problems [19, 21].

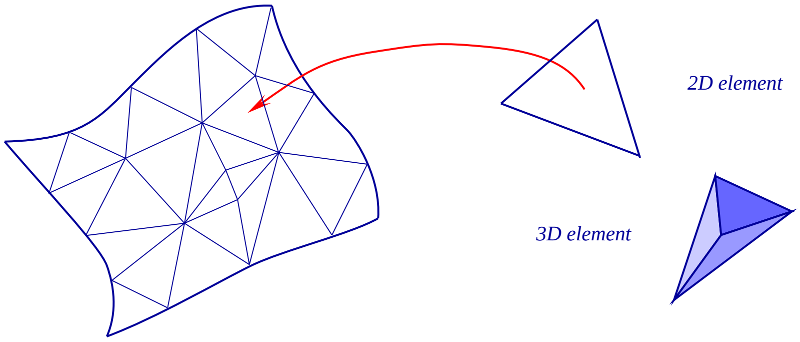


Figure 2-13: Finite Element mesh and examples of a 2D and 3D element

Linear methods

The simplest CFD solver methods are the linear solvers. Among the linear solvers, we have the panel codes, the vortex lattice method (VLM) and the lifting-line method. These method make use of singularity element methods and are not valid for compressible flows. Several corrections can be applied such that these methods are valid within certain flight conditions.

Lifting-line method The simplest three-dimensional wing theory is that based on the concept of Prandtl's lifting line theory [24] and on linearized potential flow theory. In this theory, the wing is replaced by a lifting line. The circulation about the wing associated with the lift is replaced by a vortex filament. This vortex filament lies along the straight line. At each spanwise station of this filament, the strength of the of the vortex is proportional to the local intensity of the lift. The variation of the vortex strength along the straight line is therefore assumed to result from the superposition of a number of horseshoe-shaped vortices, as shown in Figure 2-14. The portions of the vortices lying along the the span are called the bound vortices, and the portions extending downstream are called the trailing vortices (or wake). The effect of trailing vortices corresponding to a positive lift is to induce a downward velocity component at and behind the wing. The downward velocity is called downwash. From this rotation of the flow the effective angle of attack reduces, and correspond in a rotation of the lift vector and produces on its turn an additional drag component in stream-wise direction. The induced drag, downwash velocity, lift distribution, and vorticity distribution can be calculated using this theory. This theory has several limitations: it does not take compressibility, viscous flow and unsteady flow effects into account. Another important drawback is that is does not account for the effect of wing sweep. The method is only valid for thin lifting surfaces at small angles of attack and sideslip.

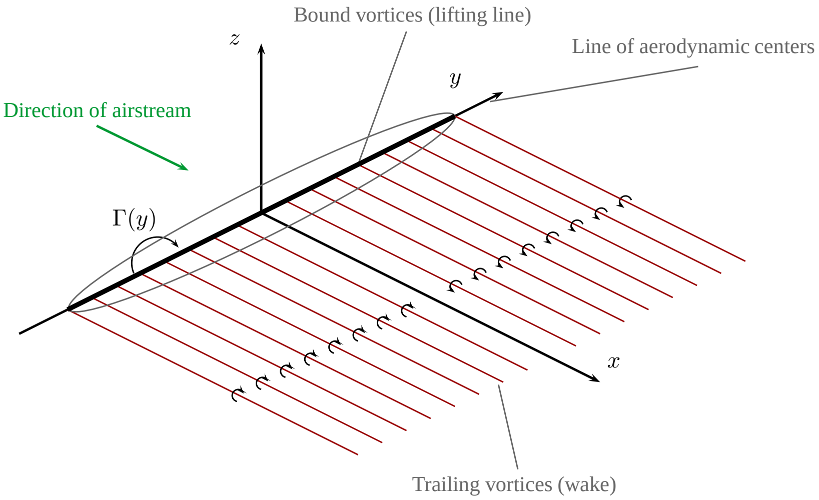


Figure 2-14: Lifting-line model consisting of multiple horseshoe vortices

Vortex Lattice Method Prandtl's classical lifting line theory gives reasonable results for straight wings at moderate to high aspect ratio. However, for low-aspect-ratio straight wings, swept wings and delta wings, the lifting line theory is inappropriate [25]. For such planforms, more sophisticated method models must be used. The vortex lattice method (VLM) is an extension of the lifting line theory. The VLM places a series of lifting lines on the plane of the wing at different chordwise stations. The wing is then represented by a lifting surface without thickness and discretized in quadrilateral element, so-called panels. Figure 2-15 illustrates a swept wing with these panels. A vortex ring is associated with each panel, placed on the quarter-chord line. The spanwise vorticity is lumped on each panel into a discrete vortex along its quarter-chord line. This results in a system of horseshoe vortices, one for every panel on the surface. Classical VLM formulations ignore the thickness of the wing. With the addition of compressibility correction in the flow direction, the VLM can also be used to a limited extent to compressible flow.

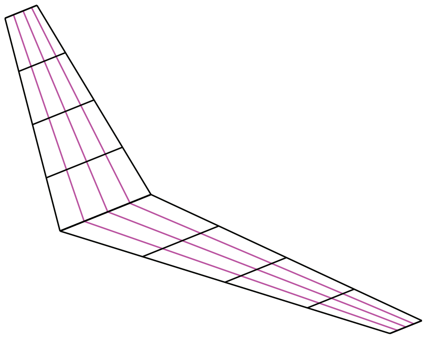


Figure 2-15: Schematic of a lifting surface

Panel method In the two above-mentioned methods, the singularities are located inside the body. Unfortunately, an arbitrary body shape cannot be created using the singularities placed inside the body. A more sophisticated method has to be used to determine the potential flow over arbitrary shapes. Panel methods solve the linearized potential equations for inviscid, irrotational, incompressible flow. Since the equation is linear, superposition of the solutions can be used. The most familiar singularities are the point source, doublet and vortex. With the addition of a compressibility correction in the free-stream flow direction (e.g. Prandtl-Glauert, Karman-Tsien), panel methods can also be applied to a limited extent to compressible flow around slim bodies. In general, with compressibility corrections, panel methods are limited to free-stream Mach number of less than 0.7 [12]. For higher subsonic Mach numbers with small disturbance to the freestream flow, the Prandtl-Glauert Equation can be used (see [26, 14] for more details).

Aerodynamic panel methods generally use quadrilateral panels to define the surface. Since three points determine a plane, a quadrilateral may not necessarily define a consistent flat surface. In practice however, the methods actually divide panels into triangular elements to determine an estimate of the outward normal [14]. It is important that all the edges fit such that there is no gap (or leakage) in the panel model representation of the surface.

Higher-order panel methods (advanced) use singularity distributions that are not constant on the panel. Moreover, the may use panels which are non-planar. Higher-order methods were found to be crucial in obtaining accurate solutions for the Prandtl-Glauert Equation at higher subsonic and supersonic speeds. At these speeds, the Prandtl-Glaurt Equation becomes a wave equation (hyperbolic) [14]. Such equation require more accurate numerical solutions than the subsonic case in order to avoid pronounced errors. In theory, good results can be obtained using fewer panels with higher-order methods [14]. More information about higher-order methods can be found in [27, 26].

2-3-4 Aerodynamic methods comparison

The most widely used CFD methods in industry and research are the RANS solver, Euler solver, the panel method and VLM. These methods are compared to each other based on differences, such as compressibility effects, shockwave prediction, etc. The comparison is shown in Table 2-1. A multi-fidelity model in function of the CFD method and the geometry detail of the model, is graphically represented in Figure 2-16. The geometry detail increases when the level of CFD method increases. Euler and RANS solvers require computer aided design (CAD) models for meshing.

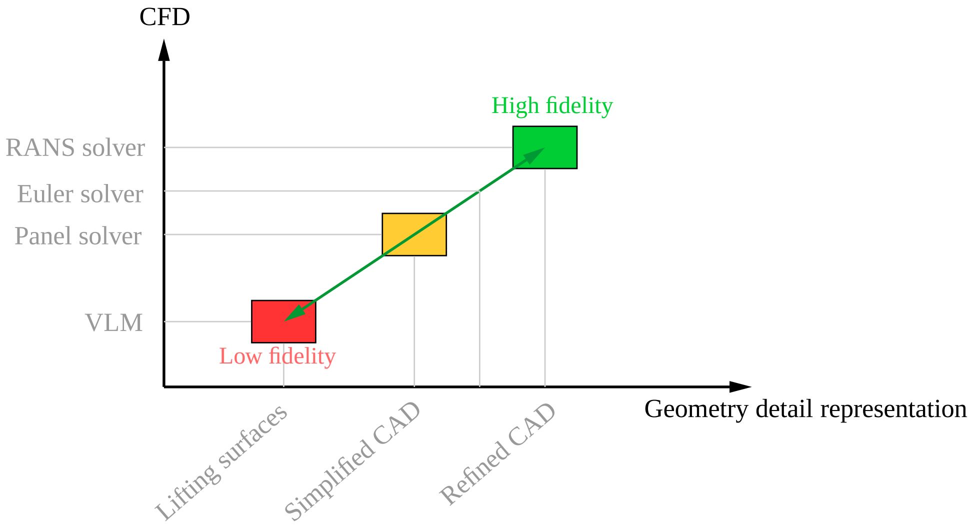


Figure 2-16: Multi-fidelity model in function of CFD method and geometry detail. (Adapted from: [28])

2-3-5 Structural weight estimation methods

Weight estimation methods used in aircraft design processes can be grouped based on their level of fidelity. In general the lower the fidelity, the more the method is based on statistics. Higher class methods rely less on statistical estimations and use more physics based calculations. These calculations require more detailed input parameters. In a wing design task, three class of weight estimation methods are used: statistical, quasi-analytical and analytical methods.

Statistical weight prediction techniques These methods use statistical coefficients, developed based on existing aircraft, to estimate the wing weight as a function of several significant geometrical parameters. A number of conceptually similar methods, tailored to different classes of aircraft, are presented by Howe [29], Torenbeek [30], Shevell [31] and Raymer [15]. Implementing these methods results in a fast and robust weight estimation with minimum amount of required data. The main problem is the level of accuracy. Another problem is the validity margin, as these method are only valid within the bounds of the original data. Note that such methods are limited when considering new concepts that lie outside the original database.

Quasi-analytical weight estimations Another class of weight prediction techniques are the quasi-analytical methods. These methods are based on elementary strength/stiffness analyses of aircraft structures, augmented by empirical factors and statistical data. This approach enables weight engineers to develop more accurate and design-sensitive results. Additionally they have a wider validity margin compared to statistical methods. These methods are based on the 'stationary analysis' of a simplified structure. In such methods the primary wing box weight is computed by calculating the amount of material required to resist the applied loads. Contributions of the secondary structures (i.e., flaps, slats, ailerons, etc.) are still estimated based on statistical methods. Examples of wing weight estimations based on this approach are presented by Torenbeek [32], Macci [33] and Elham [34].

Analytical methods These methods rely on finite element methods (FEM) to size various components of the primary wing structure and compute their weight using the material density. Examples of such methods are demonstrated by Bindolino [35] and Laban [36]. Generally, these methods are more accurate than those mentioned above. However their implementation requires a large amount of detailed geometrical information as well as structural and material data. This is usually not available in early design stages. Furthermore, the time required to implement a finite element model and perform the analysis, is significantly longer than computing the simple empirical equations of a statistical method. This makes these methods unsuitable for both early design phases and weight predictions within high level MDO processes.

	Vortex-lattice method	Panel solver	Full potential solver	Euler solver	RANS solver
Governing equations	Linearized potential flow equations	Linearized potential flow equations	Full potential flow equations	Euler equations	RANS equations
Compressibility	using compressibility corrections	using compressibility corrections	'Exact'	'Exact'	'Exact'
Lift coefficient	Yes	Yes	Yes	Yes	Yes
Shockwave prediction	No	No	Inaccurate for strong shocks	Yes	Yes
Pressure distribution on surface	Maybe (inaccurate at leading edge)	Yes	Yes	Yes	Yes
CPU calculation time per case [28]	5 sec. - 1 min.	1 min. - 15 min.	5 min. - 1 hr.	1 - 15 hrs.	Multiple days

Table 2-1: CFD methods comparison

Chapter 3

Local optima smoothing for global optimization

In this chapter, a novel optimization algorithm to find the global solution is presented. This principle of algorithm is extensively discussed and several benchmark test functions and a classic aerodynamic problem were used to prove its effectiveness and accuracy.

First, a brief discussion is given on the formulation of optimization problems in Section 3-1. In the same section, the need for the global solution is highlighted. In Section 3-2, a global optimization algorithm is explained that uses local optima smoothing. Finally, in Section 3-3 several test cased were used to prove the effectiveness, robustness and accuracy of the global optimization algorithm.

3-1 Optimization problem

The following mathematical formulation is used for a general constraint optimization problem:

$$\min_{\boldsymbol{x}} \quad f(\boldsymbol{x}) \tag{3-1}$$

$$\text{subject to} \quad g_i(\boldsymbol{x}) \leq 0 \qquad i = 1, \ldots, m \tag{3-2}$$

$$h_j(\boldsymbol{x}) = 0 \qquad j = 1, \ldots, n \tag{3-3}$$

where $f(\boldsymbol{x})$ is the optimization objective function and $\boldsymbol{x}$ the design vector. Function $g_i(\boldsymbol{x})$ stands for the inequality constraints and $h_j(\boldsymbol{x})$ for the equality constraints. There are different optimization algorithms that can solve problem (3-1). In general gradient-based optimization algorithms only seek local solutions. However, the search for global solutions is an important task for engineering purposes, mainly for the following reasons (as addressed by Rizzo and Frediani [37]):

(a) The objective functions and constraints can include “black boxes”. This means that explicit expressions for the objective function and constraints might not be available. The values of these functions at point x can be (partly) provided by numerical code (so-called “black boxes”). Therefore, the general properties of these functions might not be known in advance. (See Figure 3-13 for an example of the discontinuity of a “black box”)

(b) Numerical issues associated to the evaluation of the objective and constraints introduce numerical noise. This is related to previous point, where the numerical evaluations of the function values are often approximated. Approximations superimpose numerical errors that may affect the function by high-frequency oscillations with local peaks (local minima).

(c) When applying optimization algorithms in the study of new projects, a good exploration property of algorithms is mandatory. The effectiveness of the algorithm can be influenced by the optimizing designs that have an almost unknown design domain. Furthermore, the initial point for the optimization might lie far from the global solution.

Two algorithms are used in this research: one to find the local optimum and another one to find the global solution. For the local optimization algorithm, the Sequential Quadratic Programming (SQP, gradient-based) algorithm was selected. A brief overview of the SQP algorithm can be found in Appendix A. The global optimization algorithm is explained in the following section.

3-2 An algorithm for local optima smoothing

Many algorithms to find the global optimum have been devised. Examples of global optimization algorithms are: genetic algorithm, A*, dynamic programming, etc. The optimization algorithm (framework) proposed by Addis et al. [38, 39] is capable of finding the global optima using a local solver and it has shown good performances in terms of robustness and time consumption. This algorithm is called “Local Optima Smoothing for Global Optimization” (LOCSMOOTH) and is discussed here. The global algorithm is also used to compare the local optima results for a multidisciplinary design optimization of aircraft wings in Chapter 6.

3-2-1 Local optima smoothing principle

The basic idea of the LOCSMOOTH algorithm is explained here. Suppose that the objective function has the form represented by the solid blue line in Figure 3-1 and that it has an underlying function like the dashed line. The objective function can then be viewed as the underlying function plus some perturbation (noise) around it. A function with this property is known as a function having a funnel structure.

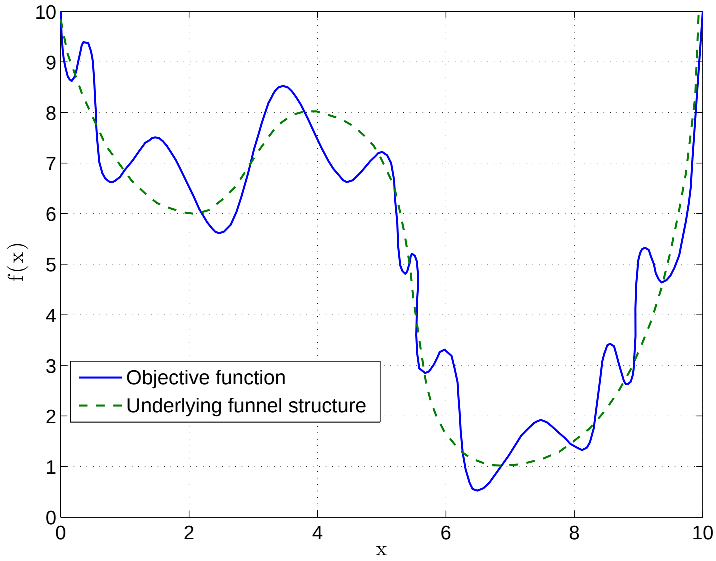


Figure 3-1: Example of optimization function with a funnel structure.

When a local optimizer is used to find the optimum starting from a point in interval $[a, b]$ in Figure 3-2, then it will return x_1 as the solution. Only if the starting point belongs to the $[c, d]$

interval, the global solution will be found. This is a typical characteristic of gradient-based optimization algorithms. In Figure 3-1, the thickest line represents the local minima found by a local optimization algorithm starting from different points. It can be deduced that the global optimum is the minimum of the enveloping function containing all local minima (thickest, red line). The idea of the LOCSMOOTH algorithm is to use this information from the local optimizations starting from different points to construct the enveloping curve to find the global minimum. This (piecewise constant) function is also known as step function.

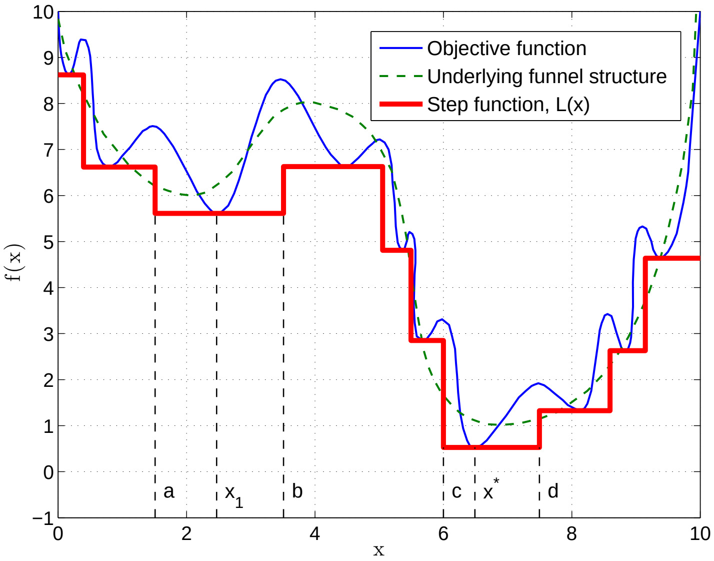


Figure 3-2: Local and global optimization.

Gradient-based optimization algorithms (such as SQP) have problems with step functions. In order to build a more reliable method for funnel-type global optimization problems, sampling the optimization function coupled with smoothing of the step function (denoted by $L(x)$) might provide a good strategy [38]. Looking at Figure 3-2, it can immediately be noticed that a very good smoothing effect has already been achieved by simple observing $L(x)$, the results of local optimizations, instead of the objective function $f(x)$. However, in order to fully exploit the funnel structure, a smoothing method should be applied to $L(x)$. This way, the piecewise constant structure of $L(x)$ will be replaced by a smooth function which contains information of descent directions. It is evident that in a piecewise constant function, no local information is available to provide guidelines for gradient-based optimizations. Though, if smoothing were possible, the smoothed function could help in finding appropriate descent directions and thus guide the search towards a global optimum.

The following smoothing function was derived by Addis et al. (see [38] for the complete derivation):

$$\hat{L}_g^B(x) = \frac{\sum_{i=1}^{K} L(y_i) g\left(||y_i - x||\right)}{\sum_{i=1}^{K} g\left(||y_i - x||\right)} \tag{3-4}$$

where $g(x)$ represents the Gaussian kernel, K the number of local minima in interval B. The 'kernel' for smoothing, defines a shape of the function that is used to take the average of the neighboring points. A Gaussian kernel is a kernel with the shape of a Gaussian (normal distribution) curve, see Figure 3-3.

The basic process of smoothing is simple, as it proceeds through the data point by point. For each data point, a new value is generated that is some function of the original value at that point and

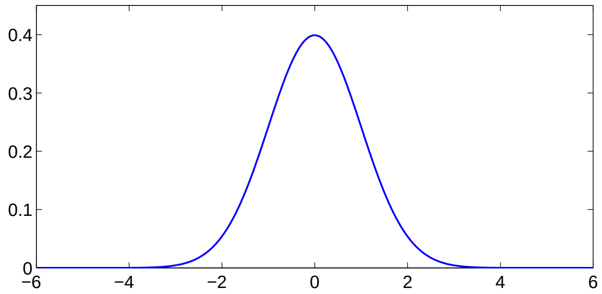


Figure 3-3: Normal distribution curve (Gaussian), with standard deviation $\sigma = 1$.

the surrounding data points. The function that is used within the Gaussian kernel is the Gaussian (or normal distribution) curve. In the standard statistical way, the width of the Gaussian shape is defined by the standard deviation or σ (additional notes on σ are given in the following subsection). The Gaussian kernel is defined as

$$g(z) = e^{\frac{-z^2}{(2\sigma)^2}} \tag{3-5}$$

An example of smoothing the step function $L(x)$ is given by Figure 3-4. Now, the optimum of the smoothed function can be found using a local optimization algorithm. At this minimum, a new local search is performed on the real optimization function $f(x)$ in order to find the global optimum.

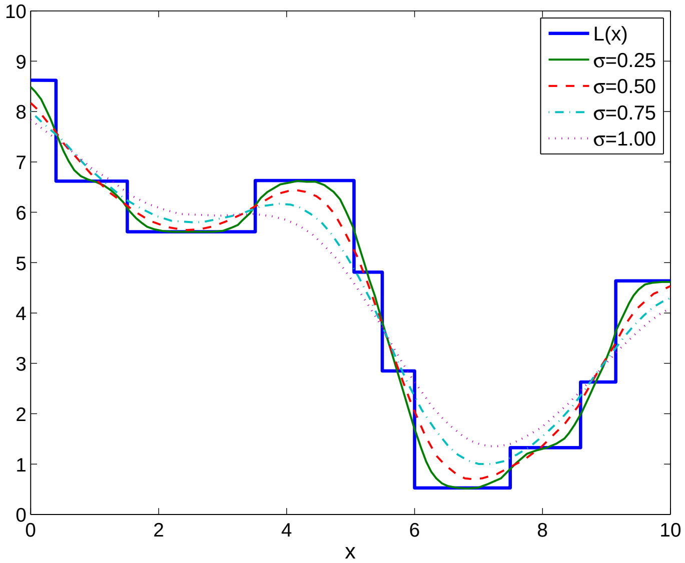


Figure 3-4: Gaussian filtered step function $L(x)$, with different standard deviations σ.

3-2-2 The LOCSMOOTH framework

The main idea of the LOCSMOOTH algorithm is to take information from the local minima in order to build a smoothed function giving a direction on the basin of attraction. When the starting point changes, the minimum will generally change as well. By sampling multiple points and determine their minima, a step function can be constructed.

The LOCSMOOTH algorithm is organized in two phases: an approximation phase and a displacement phase. In the approximation phase, the local minimum $L(x_h)$ starting from point x_h is calculated. Then, K point are random uniformly sample inside a sphere $B(x_h, r)$ with radius r center in x_h, producing K observations. The step function $L(x)$ can then be constructed based on the minima of these observations. $L(x)$ is then used to build the smoothed function $\hat{L}_g^B(x)$. Consequently, the smoothed function is minimized and its minimum gives directions on the basis of attraction. The minimum of $\hat{L}_g^B(x)$ is then taken as the next current point and the procedure is iterated (displacement phase). The whole procedure is interrupted at that point where no improvements (*MaxNoImp*) are found.

In order to speed up this process, each time a *record* (the best local optimum observed so far) is obtained, the procedure sets the *current* point to the *record*.

The definition for the standard deviation for the Gaussian operator is given by the following expression:

$$\sigma = r/\sqrt[n]{K} \tag{3-6}$$

By this definition, the whole volume of the sphere is covered by the Gaussian weight. In order to obtain an equal coverage for different radii values, the number of samples must be $K = r^n/\sigma^n$ [38]. This choice for standard deviation is less effective when the variables have a different range of variation, which can lead to deterioration of the convergence speed of this algorithm, as addressed by Rizzo [40]. This problem can be avoided by choosing different radii for the variables. Hence, instead of sphere, the samples are chosen inside an ellipsoid. Equation (3-6) is then modified into the following expression [40]:

$$\sigma = \left(\frac{\prod_{i=1}^{K} r_i}{K} \right)^{1/n} \tag{3-7}$$

where r_i represents the radii along different axes. For more details on this modified standard deviation, one can refer to reference [40]. The number of observations (samples) to be made are problem dependent. Parameter n influences the quality of the approximated function, but it is rather time-consuming because a local optimization is performed for each sample. The variable r_i affects the exploration aptitude of the algorithm; smaller radii limit the search in a neighborhood of the current solution, whereas larger radii could give too dispersed data with deterioration of the approximated function [38, 40].

Below, the complete procedure of the LOCSMOOTH algorithm with the initialization, based on the psuedo-code provided by Addis et al. [38], is discussed. Additionally, a visualization of the complete procedure is shown in Figure 3-5.

Initialization

To start the optimization procedure, several initial user inputs are required: the initial starting point x_0, the radius of the sphere r, the number of observations to perform in the current sphere K and the maximum number of no improvement *MaxNoImp*. First, set *NoImp* $= 0$ and choose a starting point x_0 in which the sphere $B(x_0, r)$ is centered with radius r. Choose a random uniform sample point x in the sphere $B(x_0, r)$ and calculate the minimum of the real-valued function $f(x)$ (objective function) resulting in $x^* = \min_x f(x) = LS(x)$. Then the *current* is set to *current* $= f(x^*) = L(x)$ and the *record* point to *current*. Note that $LS(x)$ denotes the local search of the objective function starting from point x.

Procedure

The iterative procedure of the algorithm is given by the following steps:

1. When *NoImp* $<$ *MaxNoImp*, then set $i = 0$ and continue with step 2.
2. If $i < K$ and *record* $\leq$ *current*, then set $i = i + 1$ and observe a random uniform sample point y_i inside the sphere $B(x^*, r)$ centered in x^*. Calculate the local minimum $y_i^* = LS(x)$ and set *current* $= f(y_i^*) = L(y_i)$.
3. **If** a new *record* has been found while sampling in $B(x^*, r)$ (*current* $<$ *record*), then this is set to *record* $=$ *current*. Consequently, the center of the sphere is moved to y_i^* $(x^* = y_i^*)$ and set *NoImp* $= 0$. At this point, the procedure starts again at step 1.
   Else *NoImp* $=$ *NoImp* $+ K$ and develop the smoothed function $\hat{L}_g^B(x)$ using the stored local minima y_i^*. Next, the smoothed function is optimized to find its minimum $x = \arg \min\limits_{x \in B(x^*, r)} \hat{L}_g^B(x)$. Consequently, the local minimum $y = LS(x)$ is to be found starting from x and set *current* $= f(y) = L(x)$. The procedure is continued with step 4.
4. **If** *current* $<$ *record*, then a new *record* has been found and set to *record* $=$ *current*. Furthermore, set $x^* = y$ and *NoImp* $= 0$. **Else**, set $x^* = x$ and go to step 1.

The algorithm terminates when *NoImp* $=$ *MaxNoImp*. As soon as a new *record* is obtained, the center of the sphere is moved. The LOCSMOOTH algorithm showed great performances in finding global solutions as it is show in the following test cases.

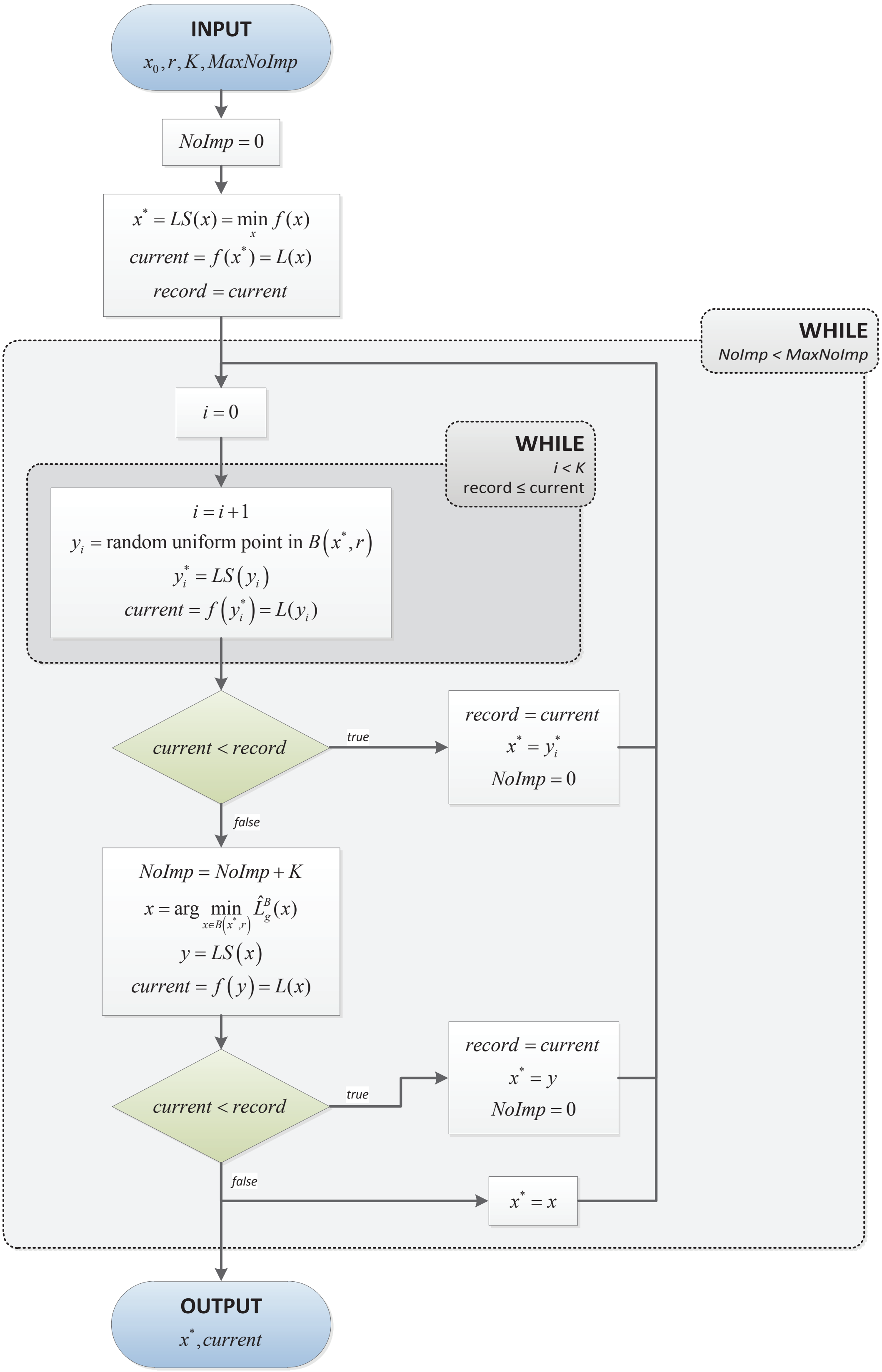


Figure 3-5: Roadmap of the LOCSMOOTH algorithm.

3-3 Test cases

In this section, some tests of the global search algorithm are performed by solving some benchmarking problems taken from literature [41, 42]. These problems include very noisy objective functions without constraints. Another problem that was constituted, is a constrained classical aerodynamic problem. All the test cases were solved by means of the global optimization algorithm LOCSMOOTH using a gradient-bad based algorithm for local searches. The user-defined parameters for the LOCSMOOTH algorithm used throughout these test cases (unless stated otherwise) are:

- Sphere radius $r = (ub - lb)/3$, where ub and lb are the lower and upper bounds respectively
- Number of sampled points K within the sphere
- Maximum number of no improvements, *MaxNoImp=3K*

For the local search, the SQP algorithm of the Matlab Optimization Toolbox is used. All the test case use the default settings of this algorithm.

3-3-1 Ackley's function (unconstrained)

A classical benchmarking function to test optimization algorithms, is the Ackley's function. A 3D illustration of this figure is shown in 3-6. This function presents a deep basin (see Figure 3-7) of attraction and is very noisy in minimum region. High frequency waves are superimposed over the whole domain. Ackley's function is defined by

$$f(x) = -20e^{-0.2\sqrt{\frac{1}{2}\sum_{i=1}^{n} x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} + 20 - e \tag{3-8}$$

where n represents the number of variables. The design space for this problem is bounded by $-32.768 \leq x_i \leq 32.768$ and the theoretical solution of the global optimum is $f(x^*) = 0$ where $x_i^* = 0$ for $i = 1, \ldots, n$.

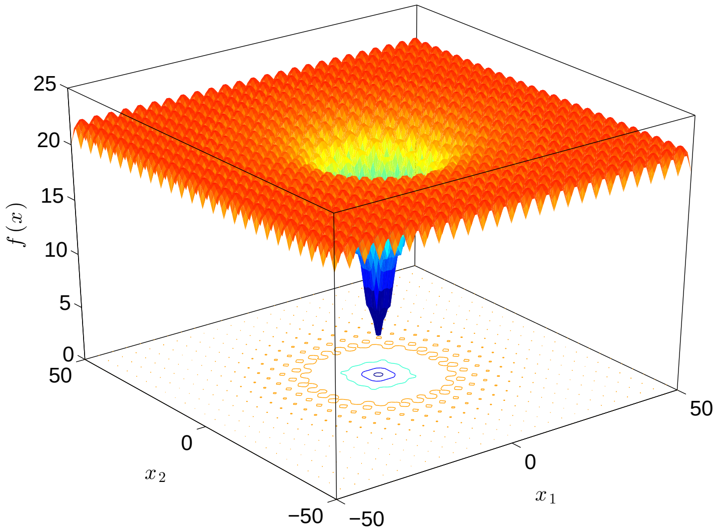


Figure 3-6: 3D plot of the Ackley's function for $n = 2$ variables.

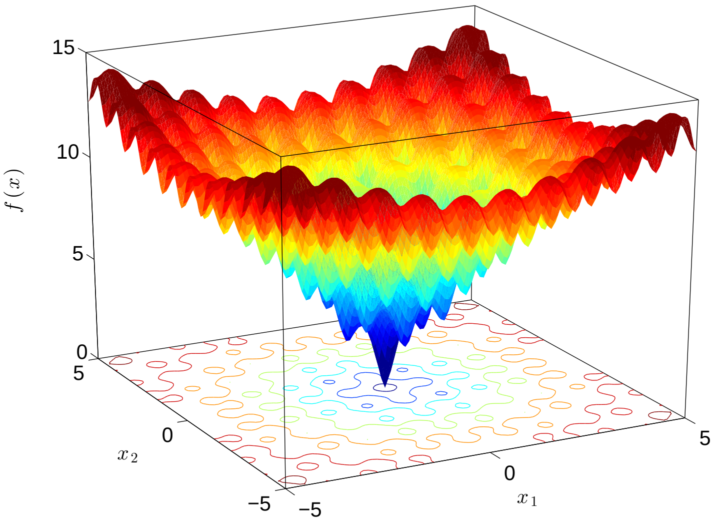


Figure 3-7: A zoom near the origin of the Ackley's function for $n = 2$ variables.

Initially, the radii of the spheres were defined by $r = (ub - lb)/3$ which gave results far from the exact solution $f(x^*) = 0$. As mentioned before, larger radii could give too dispersed data with deterioration of the approximated function. Therefore, the radii for the problem were redefined to $r = (ub - lb)/5$. In Table 3-1 the minima calculated by the global optimization algorithm are summarized.

Table 3-1: Numerical solutions of Ackley's function with sphere radii $r_i = (ub - lb)/5$, LOCSMOOTH algorithm.

n	$f(x^*)$	*NFval*
2	1.0454×10^{-6}	3062
5	1.8150×10^{-6}	8663
10	2.8787×10^{-5}	30496
15	1.0664×10^{-4}	48754
20	1.6683×10^{-4}	45268

It can be observed that in general the number of function evaluations *NFval* increase and the solutions deteriorate as the space dimensions grow.

3-3-2 Rastrigin's function (unconstrained)

This is typical example of a non-linear multimodal function (function with many local minima). A typical funnel structure is present (see Figure 3-1) and consequently the LOCSMOOTH procedure is expected to be effective.

$$f(x) = 10n + \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i)\right] \tag{3-9}$$

The design space for this problem is bounded by $-5.12 \leq x_i \leq 5.12$ and the theoretical solution of the global optimum is $f(x^*) = 0$ where $x_i^* = 0$ for $i = 1, \ldots, n$. Table 3-2 provides the minima found using LOCSMOOTH algorithm.

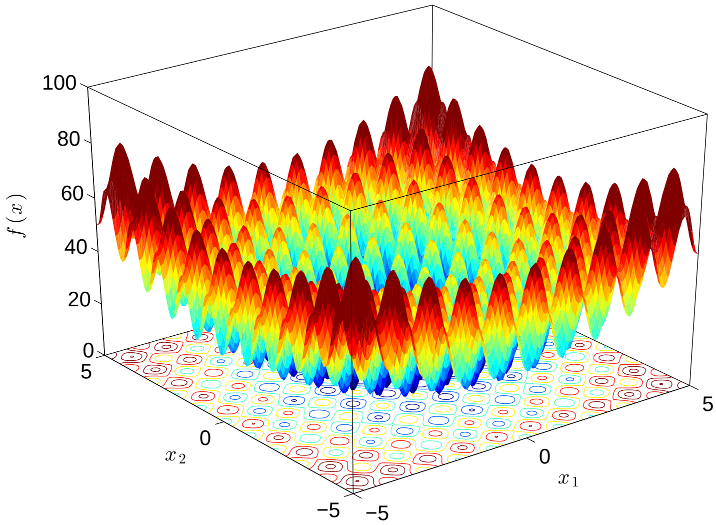


Figure 3-8: 3D plot of the Rastrigin's function for $n = 2$ variables.

Table 3-2: Numerical solutions of Rastrigin's function, LOCSMOOTH algorithm.

n	$f(x^*)$	*NFval*
2	2.1316×10^{-14}	1745
5	8.0291×10^{-13}	5039
10	1.0328×10^{-10}	12628
15	0	21682
20	0	33511

3-3-3 Rosenbrock's function (unconstrained)

Rosenbrocks's valley is a classic optimization problem, also known as the "banana function". A 3D plot of this function is given in Figure 3-9. The global optimum of this function lies inside a long, narrow, parabolic-shaped flat valley. Finding the basin of attraction is a trivial task, since the convergence of the global optimum is more difficult. The Rosenbrock's function is defined as:

$$f(x) = \sum_{i=1}^{n} \left[100 \left(x_{i+1} - x_i^2 \right)^2 + (1 - x_i)^2 \right] \tag{3-10}$$

The design space for this problem is bounded by $-2.48 \leq x_i \leq 2.48$ and the theoretical solution of the global optimum is $f(x^*) = 0$ where $x_i^* = 1$ for $i = 1, \ldots, n$. The results using the LOCSMOOTH algorithm are summarized in Table 3-3.

3-3-4 Schwefel's function (unconstrained)

Schwefel's function is deceptive in that the global minimum is geometrically distant from the next best local minima. For this reason, search algorithm are potentially prone to convergence into the wrong direction. The Schwefel's function is illustrated by Figure 3-10 and given by the following expression:

$$f(x) = \sum_{i=1}^{n} \left[-x_i \sin \left(\sqrt{|x_i|} \right) \right], \tag{3-11}$$

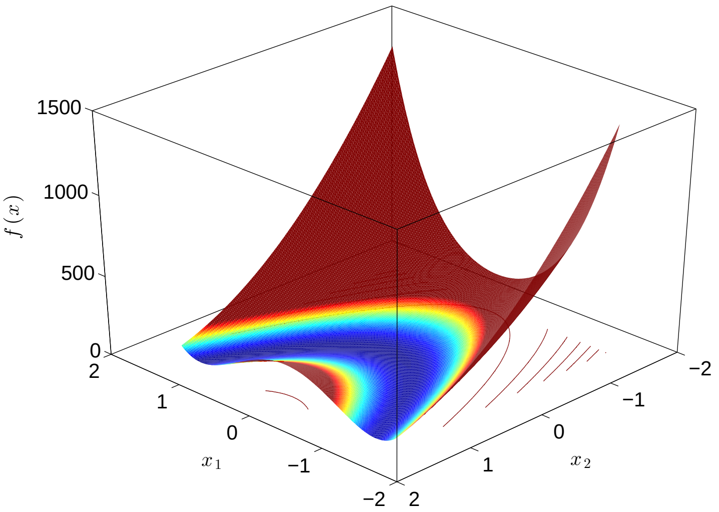


Figure 3-9: 3D plot of the Rosenbrock's function for $n = 2$ variables.

Table 3-3: Numerical solutions of Rosenbrock's function, LOCSMOOTH algorithm.

n	$f(x^*)$	*NFval*
2	2.1384×10^{-12}	5963
5	3.5132×10^{-11}	11794
10	4.0188×10^{-11}	51358
15	4.8604×10^{-11}	89519
20	5.7358×10^{-11}	63424

where n is the number of variables. The test area of this function is usually restricted to a hypercube, bounded by $-500 \leq x_i \leq 500$ for $i = 1, ..., n$. The global minimum is $f(x^*) = -418.9829 \cdot n$ where $x_i^* = 420.9686$ for $i = 1, \ldots, n$.

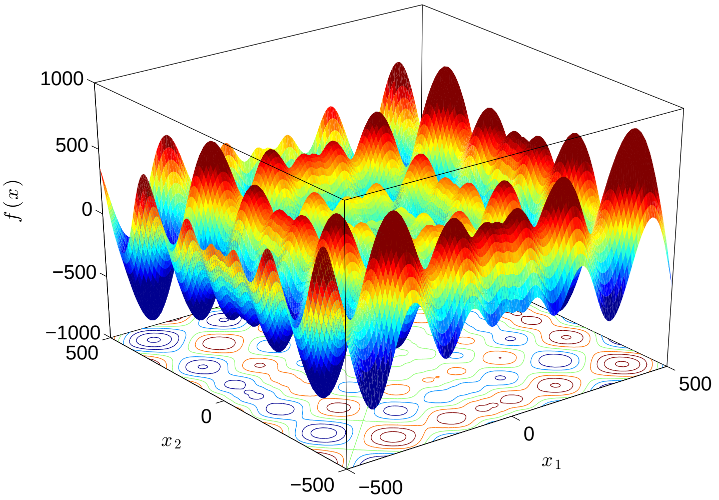


Figure 3-10: 3D plot of the Schwefel's function for $n = 2$ variables.

The initial settings for the spheres' radii was $r_i = (ub - lb)/3$ which gives results far from the theoretical global optimum. By upscaling this radii to $r_i = (ub - lb)/2$ and downscaling the number of samples to $K = 5$, the algorithm was able to find solutions close to the theoretical minimum.

Table 3-4 summarizes the numerical results of the optimization.

Table 3-4: Numerical solutions of Schwefel's function, number of sampled points $K = 5$, sphere radii $r_i = (ub - lb)/2$, LOCSMOOTH algorithm.

n	$f(x^*)$	$f(x^*) - f(x^*)_{\text{theor.}}$	*NFval*
2	-837.966	2.5455×10^{-5}	922
5	-2094.91	6.3637×10^{-8}	2138
10	-4189.83	1.2728×10^{-4}	8555
15	-6284.74	1.9091×10^{-4}	4222
20	-8379.66	2.5455×10^{-4}	6670

3-3-5 Minimum induced drag of a wing

A classical problem of aerodynamics is to find the lift distribution over an isolated wing with the minimum induced drag. It is well known that the optimum lift distribution is elliptical shaped. The optimization problem considers an unswept wing. The wing span is divided into certain number of trunks (or bays), see Figure 3-11.

Due to symmetry of the wing, only half the wing is considered for the optimization. The optimization problem is to achieve a chord distribution to minimize the induced drag D_i with a constraint on the global lift L. The problem can be formulated as

$$\begin{aligned} \min_{x} \quad & D_i(\boldsymbol{x}) \\ \text{subject to} \quad & L(\boldsymbol{x}) = W \\ & lb \leq \boldsymbol{x} \leq ub \end{aligned}$$

where W is the weight, lb the vector of the lower boundaries and ub the vector of the upper boundaries. The variable $\boldsymbol{x}$ is the design vector containing the design variables. The starting geometry has 5 bays and thus 6 equally spaced ribs. The design variables are the chord lengths at each rib. The induced drag is defined as

$$D_i = \tfrac{1}{2} C_{D_i} \rho V^2 S \tag{3-12}$$

This configuration will be optimized at sea level conditions with a free-stream velocity $V = 50$ m/s and Mach number $M = 0.2$. Both parameters S and C_{D_i} are of importance for D_i and will be mentioned as well in the results. The weight W is fixed to 5000 N and the chords may vary between 0.1 and 3 m. Each wing section has a NACA 0014 airfoil shape. The wing aerodynamics

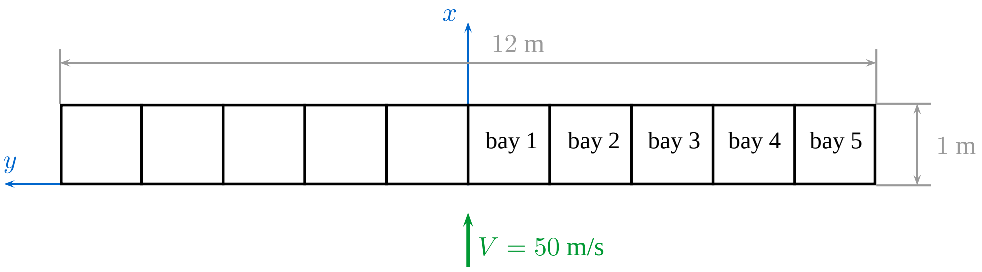


Figure 3-11: Starting geometry of the wing with 5 bays, top view.

are calculated using AVL [43] (a vortex-lattice method, see Subsection 2-3-3), that is interfaced with Matlab. This interface enables information exchange between AVL and the optimizer. The starting aerodynamics of the initial wing configuration are shown in Table 3-5. AVL uses the Trefftz plane (far-field analysis) to calculate the dimensionless induced drag (C_{D_i}). The Trefftz plane result for the starting geometry is given in Figure 3-12.

Table 3-5: Starting aerodynamics of isolated wing.

α	C_L	C_{D_i}	S	D_i
$1.2196°$	0.2721	0.00209	12 m^2	38.036 N

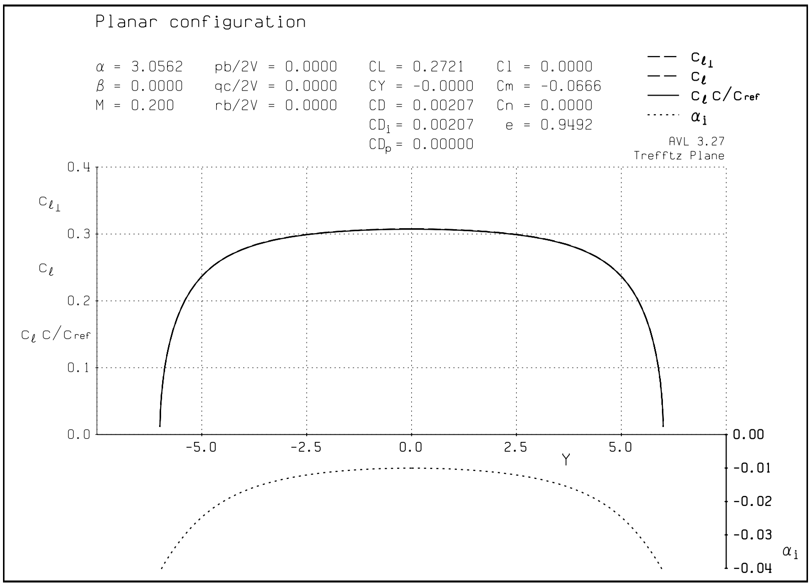


Figure 3-12: Lift distribution on the Trefftz's plane for the initial rectangular wing.

The following optimization algorithms are used for this optimization problem:

- SQP algorithm, implemented in the optimization toolbox of Matlab
- LOCSMOOTH algorithm: algorithm based on local optima smoothing for global optimization

As was mentioned before, gradient-based optimization algorithms have problems calculating the optimum of step functions. Low fidelity aerodynamic solvers, such as AVL ("black box"), can imply noise or non-smooth functions for varying input parameters. An example of a piecewise constant function is given in Figure 3-13. The function represents the induced drag coefficient as a function of wing span. This graph was generated using the starting geometry and flight conditions, except the wing span variable and the lift coefficient is fixed at $C_L = 0.2$.

First, the wing is optimized using the SQP algorithm. The optimization results are summarized in Table 3-6 and its lift distribution is shown in Figure 3-14. The optimized chord lengths of each spanwise rib are given in Table 3-7, where the spanwise position $\eta = y/(b/2)$ indicates the (dimensionless) distance from root to tip section.. It can be noted that the optimized wing configuration is close to and elliptical shaped wing.

Consequently, the same optimization problem is solved using the LOCSMOOTH algorithm. The optimization results are summarized in Table 3-8. The design vector and the trefftz plane of the optimized configuration are given by Table 3-9 and Figure 3-15.

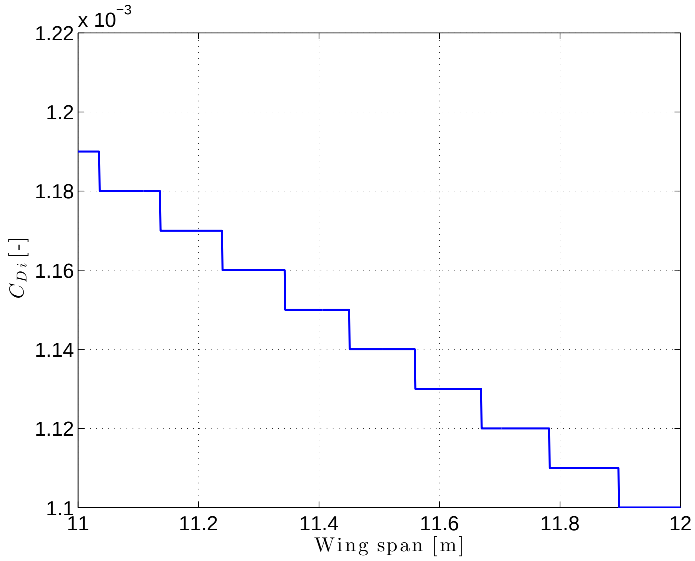


Figure 3-13: Discontinuity in induced drag coefficient C_{D_i} calculated by AVL, fixed $C_L = 0.2$.

Table 3-6: Wing optimization results for 5 bay geometry using SQP algorithm.

	Initial	SQP
C_{D_i}	0.00207	0.00186
C_L	0.2721	0.2566
D_i [N]	38.036	36.245
S [m^2]	12	12.726
α [°]	3.0562	2.8128

Table 3-7: Optimized chord lengths of rectangular wing for minimum induced drag using the SQP algorithm.

Spanwise location	$\eta = 0.0$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.8$	$\eta = 1.0$
Initial	1	1	1	1	1	1
SQP	1.2574	1.4296	1.0955	1.0955	0.7596	0.5535

Table 3-8: Wing optimization results for 5 bay geometry using LOCSMOOTH algorithm.

	Initial	SQP	LOCSMOOTH
C_{D_i}	0.00207	0.00186	0.00183
C_L	0.2721	0.2566	0.2538
D_i [N]	38.036	36.245	36.055
S [m^2]	12	12.726	12.867
α [°]	3.0562	2.8128	2.7845

Figure 3-16 illustrates the optimized wing configuration using both optimization algorithms. Worth of notice is that the optimized wing planform using the LOCSMOOTH algorithm has an elliptical shape providing an elliptical lift distribution.

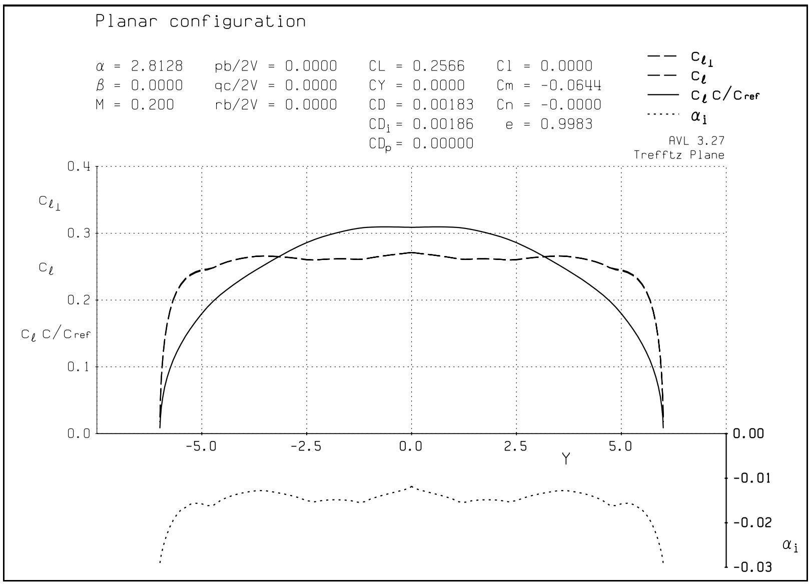


Figure 3-14: Lift distribution on the Trefftz's plane of optimized wing using SQP.

Table 3-9: Optimized chord lengths of rectangular wing for minimum induced drag.

Spanwise location	$\eta = 0.0$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.8$	$\eta = 1.0$
Initial	1	1	1	1	1	1
SQP	1.2574	1.4296	1.0955	1.0955	0.7596	0.5535
LocSmooth	1.3240	1.2495	1.2402	1.0867	0.8901	0.4654

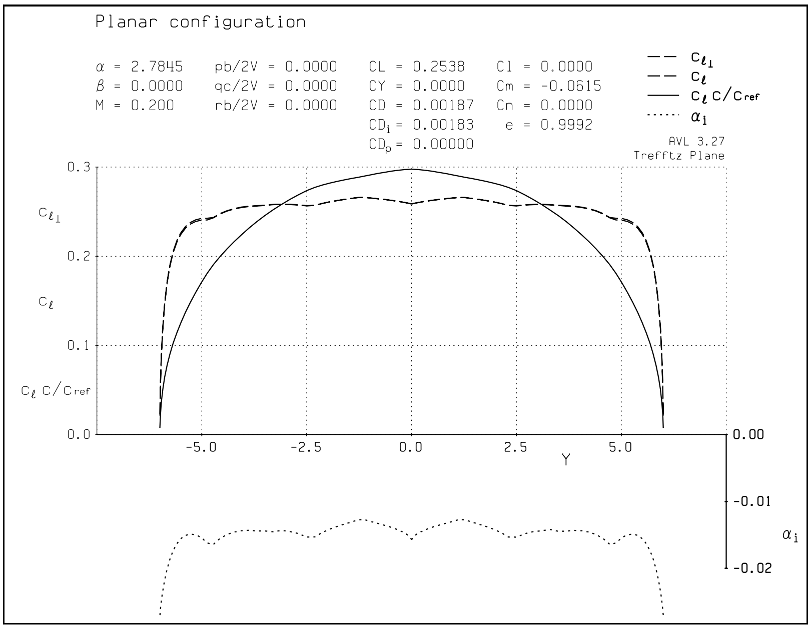


Figure 3-15: Lift distribution on the Trefftz's plane of optimized wing using LOCSMOOTH algorithm.

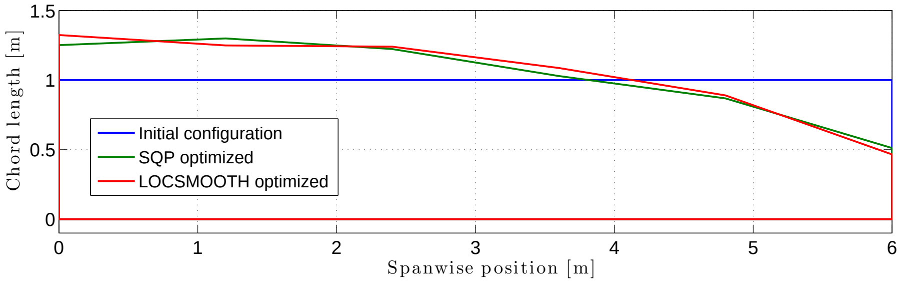


Figure 3-16: Initial and optimized wing configurations.

Chapter 4

Quasi-3D aerodynamic solver

The total drag produced by a wing exists of induced drag and profile drag (see Figure 2-9). Profile drag can be subdivided into friction drag and pressure drag. Friction drag is the component of the resulting forces tangential to the wing surface acting in the direction of the flow. The pressure drag is the component of the forces normal to the surface, in the flow direction.

Optimization of aircraft wings requires many repetitive aerodynamic evaluations of different planforms, which leads to time-consuming processes. Therefore, it is of great interest to find a fast way for calculating the total wing drag. An interesting way for establishing such an approach, is by combining (two-dimensional) viscous airfoil data with an inviscid three-dimensional wing lift calculation. In reference [44], such a method has been developed and validated. Viscous airfoil data can be found in experiments, or can be calculated using two-dimensional viscous airfoil calculation tools. Three-dimensional lift distributions from wing configurations can be found using a vortex lattice code.

First, an application of the strip method is presented in Section 4-1, because of its importance for this research. A quasi-three-dimensional aerodynamic solver was developed, using the same principle as the strip method in [44] extended with the simple sweep theory. The simple sweep theory is discussed in Section 4-2. The developed solver is capable to analyze tapered swept wings. The taper implementation into the simple sweep theory is discussed in Section 4-3. Consequently, the procedure of the developed quasi-3D aerodynamic solver is elaborated in Section 4-4. As mentioned before, the quasi-3D method can also used viscous airfoil calculation tools instead of performing interpolations on experimental data. The different tools used by the developed solver are discussed in Section 4-5. In Section 4-6, the selection of different input parameters regarding panel density and number of strips are discussed. Finally, the developed tool was validated using low-speed and high-speed test cases in Sections 4-7 and 4-8.

4-1 Strip method

The viscous drag of a wing can be found using a strip method. The strip method presented here combines the strip theory with a vortex lattice method. The strip theory, also known as blade element theory [45, 46], concerns dividing an aircraft wing geometry into discrete segments and computing aerodynamic forces and moments on those segments based on their local velocities. A more detailed explanation of the combination of strip theory and vortex lattice models can be found in [47].

In reference [44], an application of such a strip method is given. The principle of this method forms the basis of the aerodynamic solver that was developed. This procedure combines two-dimensional viscous airfoil data with a three-dimensional lift distribution over a rectangular wing, and was validated using CFD results [44]. The principle of this method is of importance for the aerodynamic solver that was developed and will therefore be presented here.

First, the three-dimensional lift distribution of the wing is calculated using a vortex lattice code. Having the lift distribution along the span, the spanwise drag distribution can be determined by calculating the drag at each strip. The process of calculating the drag at each strip (or wing section), is explained using Figure 4-1.

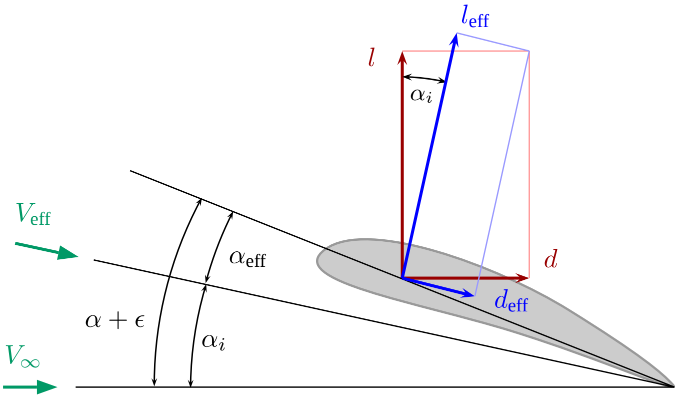


Figure 4-1: Three-dimensional lift (l), three-dimensional drag (d), local lift (l_{eff}) and local drag (d_{eff}) of a strip.

The relationship between the three-dimensional lift force l (computed by the VLM), the three-dimensional drag d, the local lift l_{eff} and the local drag d_{eff} is given in Figure 4-1. The local lift and drag forces represent the viscous airfoil forces at the effective angle of attack α_{eff}. Note that the three-dimensional forces, shown in this figure, represent the forces at a particular section which are deduced from the lift distribution obtained by the VLM. Based on this lift distribution, the two-dimensional viscous drag force can be calculated. The following steps are required to calculate this viscous drag at a given spanwise strip:

1. Set $\alpha_i = 0$ and $d_{\text{eff}} = 0$
2. $l_{\text{eff}} = (l + d_{\text{eff}} \sin \alpha_i) / \cos \alpha_i$
3. Determine local velocity $V_{\text{eff}} = V_\infty / \cos \alpha_i$ and compute the local Reynolds number Re_{eff}
4. Find α_{eff} by two-dimensional interpolation of experimental data using l_{eff}, V_{eff} and Re_{eff}
5. Let $\alpha_i = \alpha + \epsilon - \alpha_{\text{eff}}$
6. Find d_{eff} by two-dimensional interpolation of experimental data using α_{eff}, V_{eff} and Re_{eff}
7. Repeat steps 3-6 until α_i converged
8. $d = d_{\text{eff}} \cos \alpha_i + l_{\text{eff}} \sin \alpha_i$

The total drag force of the wing is the sum over all strips computed spanwise three-dimensional drag (d) distribution. This drag force is an approximation of the total drag of the wing including induced drag, which is the drag due to the tip vortex on the wing. The strip method combines experimentally measured profile drag and computationally predicted induced drag. This method may give inaccurate results for flow regimes with dominant tip vortices. By adding the vortex drag to the drag predicted by the strip method, one assumes that there is no influence of the tip vortex suction on induced and profile drag (see [44] for more details).

4-2 Simple sweep theory

The simple sweep theory, described in reference [48], is presented here. This theory was derived for wings with sweep and high aspect ratios, which is typical for most modern transport aircraft. Such wings are often designed such that the flow (except in root and tip regions) is of quasi-two-dimensional nature, in the sense that it varies sluggish across the span [48]. In these regions, the semi-mid-span regions, the idealized concept of an "infinite yawed wing" is a useful starting point that relates the swept wing flow to an equivalent two-dimensional flow over a transformed airfoil section, taken normal to the sweep line of the actual wing (as indicated in Figure 4-2).

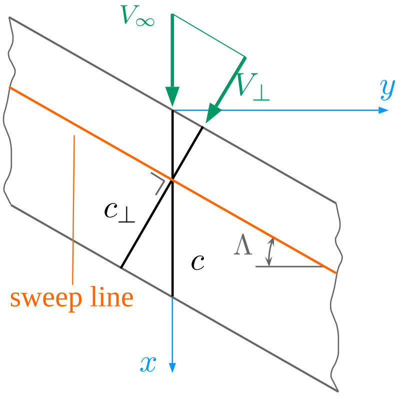


Figure 4-2: Simple sweep theory of an infinite wing (untapered wing)

The following relations were derived from geometric considerations:

$$c_\perp = c\cos\Lambda, \tag{4-1}$$

$$z_\perp = z,$$

$$\text{so} \qquad \left(\frac{z}{c}\right)_\perp = \left(\frac{z}{c}\right)\sec\Lambda$$

$$\text{and} \qquad \left(\frac{t}{c}\right)_\perp = \left(\frac{t}{c}\right)\sec\Lambda. \tag{4-2}$$

where z represents the ordinates perpendicular to the $x-y$ plane. From the aerodynamics of the flow cases, the following was derived:

$$M_\perp = M_\infty\cos\Lambda, \tag{4-3}$$

$$V_\perp = V_\infty\cos\Lambda, \tag{4-4}$$

$$C_{p_\perp} = C_p\sec^2\Lambda. \tag{4-5}$$

Equation (4-5) can be derived from the pressure perturbations which are normalized by the dynamic pressure. Based on the same equation, it can be deduced that the lift coefficients are related by

$$C_{l_\perp} = C_l\sec^2\Lambda \tag{4-6}$$

while the pressure drag coefficients are related by

$$C_{d_{p_\perp}} = C_{d_p}\sec^3\Lambda. \tag{4-7}$$

For the friction drag coefficient, it is assumed that they are about equal in free-stream direction and in direction normal the sweep line (this assumption is based on reference [49], where the same assumption has been made).

$$C_{d_{f_\perp}} = C_{d_f} \tag{4-8}$$

The profile drag, consisting of friction drag and pressure drag, is found by summing equations (4-7) and (4-8). For the profile drag in free-stream flow direction, this yields:

$$\begin{aligned} C_d &= C_{d_f} + C_{d_p} && \text{(4-9a)} \\ &= C_{d_{f_\perp}} + C_{d_{p_\perp}} \cos^3 \Lambda && \text{(4-9b)} \end{aligned}$$

Thus the air flow on a swept wing with sweep angle Λ, may be compared with the flow of an equivalent airfoil at a lower free-stream Mach number (factor $\cos \Lambda$), which is thicker (factor $\sec \Lambda$) and at a higher lift coefficient (factor $\sec \Lambda$).

As the Reynolds number changes with varying chord length, the actual Reynolds number acting on the airfoil section perpendicular to the sweep line needs to be calculated. The Reynolds number is defined as follows:

$$Re_\infty = \frac{\rho V_\infty \bar{c}}{\mu} \tag{4-10}$$

where $\bar{c}$ and μ represent respectively the mean aerodynamic chord (MAC) and the dynamic viscosity. Only the velocity and the chord length vary in (4-10), whereas the density ρ and μ are altitude dependent and are constant. Knowing that, the Reynolds number $Re_\perp$ can be scaled by

$$Re_\perp = Re_\infty \frac{V_\perp}{V_\infty} \frac{c_\perp}{\bar{c}} \tag{4-11}$$

It is important to note that the simple sweep theory is based on infinite wings and since it relates the swept wing flow to an equivalent two-dimensional flow, it does not include real three-dimensional flow characteristics such as boundary layer movements and others [48, 50].

4-3 Wing taper implementation

The simple sweep theory, needs some small modifications to account for taper ($\lambda \neq 1$). The two major modifications to be made are: the selection of a proper sweep line and the interpolation of the airfoil shape normal to this sweep line. The chord length of the airfoil shape normal to the sweep line must also be derived.

The sweep line is taken at a constant chord percentage over the wing. Since the value of the constant chord position defines the sweep line and thus the aerodynamic components and the shape of the airfoil perpendicular to this line, it is important to make a proper selection of this value. According to Obert [51], the velocity component perpendicular to the quarter-chord line with sweep angle Λ can be used in practice. For transonic flows, it is more acceptable to chose this sweep line such that it coincides with the shockwave line [49]. The reason for this is that the pressure drag acts perpendicular to the isobars (or shockwave line), as is explained in [50]. Figure 4-3 shows an infinite swept wing with the shock line taken as the sweep line for the sweep theory. Since the selection for a proper chordwise position for this sweep line, an additional input variable was introduced, This variable defines the chord percentage of the sweep line and is denoted by ξ_c. For this research, the quarter-chord sweep line ($\xi_c = 0.25$) for subsonic conditions and mid-chord sweep line ($\xi_c = 0.50$) for transonic conditions were assumed to be reasonable.

Having a proper chord percentage for the sweep line of a tapered swept wing, the chord length $c_\perp$ and airfoil shape perpendicular to this line can be determined. In Appendix C, it is explained how $c_\perp$ can be calculated. The airfoil shape is determined by interpolation between the surrounding defined airfoil shapes. This interpolation is illustrated in Figure C-3 (see Appendix C).

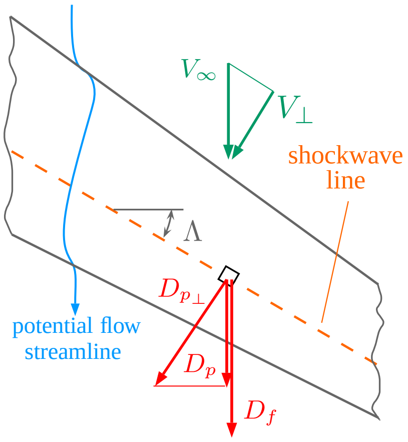


Figure 4-3: Friction and pressure drag forces on a swept tapered wing (adopted from [49])

4-4 Quasi-3D aerodynamic solver

The quasi-3D aerodynamic solver is a tool that was developed during this thesis. It is based on the strip method, see Section 4-1, and combined with the simple sweep theory (Section 4-2). Additional modifications for tapered wings are also implemented in this tool. Note that the implementation of the sweep theory implies the limitation to model three-dimensional flow characteristics. In this section, the procedure of the developed quasi-3D solver is presented. The overall procedure of the developed aerodynamic solver, can be broken down into the three stages:

1. Run the vortex lattice code to calculate the lift distribution, the wing lift coefficient C_L and the wing induced drag coefficient C_{D_i} for the given wing.
2. Calculate the viscous drag distribution using the extended strip method.
3. Compute the total wing drag.

Instead of using experimental two-dimensional viscous airfoil data, two-dimensional viscous airfoil analysis tools are used to ensure a wide range of applicability. A short overview of selected tools for the developed quasi-3D solver is given in Section 4-5.

Next, the procedure of each stage is discussed. The second stage is explained in more detail, as it combines the strip method with the sweep theory.

Stage 1

In the first stage, a vortex lattice method is used to calculate the three-dimensional inviscid lift distribution of the given wing for a given angle of attack α or lift coefficient C_L. The vortex lattice method calculates the C_L/α, C_{D_i} and lift distribution for the given wing configuration. Vortex lattice methods calculate the induced drag at the Trefftz plane ("far-field" analysis) [27, 52]. The induced drag calculated in this stage is used for the determination of the total wing drag. Note that the induced drag cannot be determined separately in "near-field" method, which are used in the strip method to calculate the profile drag.

Stage 2

In this stage, the strip method is extended using the simple sweep theory and wing taper effect. The implementation of the simple sweep theory into the strip method is described by the following steps, for the viscous drag calculation at one spanwise strip:

1. Interpolation on the three-dimensional lift distribution of the wing (calculated by the VLM) to find the lift coefficient C_l at the particular wing section (strip).
2. Apply sweep theory (considering taper effects) to find the geometric and aerodynamic characteristics of the wing section normal to the sweep line. Appendix C provides more information on the determination of the chord length $c_\perp$ and the corresponding airfoil shape. The aerodynamic characteristics of the section perpendicular to the sweep line (with angle Λ) are defined as:

$$M_\perp = M_\infty \cos\Lambda, \tag{4-3}$$
$$V_\perp = V_\infty \cos\Lambda, \tag{4-4}$$

   And the corresponding lift coefficient is found using (4-6) and the three-dimensional lift distribution calculated by the VLM:

$$C_{l_\perp} = C_l \sec^2\Lambda \tag{4-12}$$

3. Next, the induced angle of attack α_i is found by using an iterative process:
   (a) Set $\alpha_i = 0$ and $C_{d_\text{eff}} = 0$
   (b) The viscous lift coefficient can be derived using Figure 4-4:

$$l_\text{eff} = \frac{l_\perp + d_\text{eff}\sin\alpha_i}{\cos\alpha_i} \tag{4-13}$$

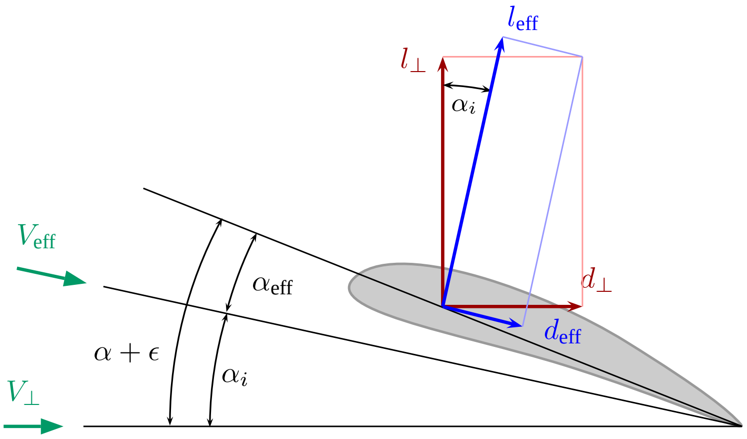


Figure 4-4: Definition of the forces and angles used to determine the inviscid downwash angle

This method uses the lift and drag coefficients instead of the lift/drag forces. Therefore, the following force-coefficient relations can be substituted in equation (4-13) together with the effective velocity conversion:

$$l_{\text{eff}} = \tfrac{1}{2}\rho V_{\text{eff}}^2 S C_{l_{\text{eff}}} \tag{4-14}$$

$$d_{\text{eff}} = \tfrac{1}{2}\rho V_{\text{eff}}^2 S C_{d_{\text{eff}}} \tag{4-15}$$

$$l_{\perp} = \tfrac{1}{2}\rho V_{\perp}^2 S C_{l_{\perp}} \tag{4-16}$$

$$V_{\text{eff}} = \frac{V_{\perp}}{\cos\alpha_i} \tag{4-17}$$

which yields

$$C_{l_{\text{eff}}} = \frac{C_{l_{\perp}}\cos^2\alpha_i + C_{d_{\text{eff}}}\sin\alpha_i}{\cos\alpha_i} \tag{4-18}$$

(c) The effective Reynolds number can be calculated using (4-11)

$$Re_{\text{eff}} = Re_{\infty}\frac{V_{\text{eff}}}{V_{\infty}}\frac{c_{\perp}}{\bar{c}} \tag{4-19}$$

(d) Find the effective angle of attack α_{eff} and the effective drag coefficient $C_{d_{\text{eff}}}$ of the airfoil using a viscous airfoil analysis tool and the calculated values Re_{eff}, $M_{\perp}$ and $C_{l_{\text{eff}}}$. The aerodynamic tools calculate besides the total drag coefficient, the friction drag coefficient $C_{d_{f_{\text{eff}}}}$ and pressure drag coefficient $C_{d_{p_{\text{eff}}}}$ which will be used later on.

$$\left[\alpha_{\text{eff}}, C_{d_{\text{eff}}}, C_{d_{p_{\text{eff}}}}, C_{d_{f_{\text{eff}}}}\right] = \text{airfoil analysis tool } \left(C_{l_{\text{eff}}}, Re_{\text{eff}}, M_{\perp}\right)$$

(e) The induced angle of attack can now be recalculated using α_{eff}.

$$\alpha_i = -\alpha_{\text{eff}} + (\alpha + \epsilon)\cos\Lambda \tag{4-20}$$

where the $\cos\Lambda$ factor accounts for decrease of the inflow angle when the sweep angle increases.

(f) Repeat steps (b)-(e) until α_i has converged.

4. The viscous drag component normal the sweep line (in direction $V_{\perp}$) can now be calculated. First, the profile drag force can be derived from Figure 4-4, which yields

$$d_{\perp} = d_{\text{eff}}\cos\alpha_i + l_{\text{eff}}\sin\alpha_i \tag{4-21}$$

As this expression has been derived for the force, it can be rewritten for coefficients (analogue to equation (4-18)):

$$C_{d_{\perp}} = \underbrace{C_{d_{\text{eff}}}\frac{\cos\alpha_i}{\cos^2\alpha_i}}_{\text{Profile drag}} + \underbrace{C_{l_{\text{eff}}}\frac{\sin\alpha_i}{\cos^2\alpha_i}}_{\text{Induced drag}} \tag{4-22}$$

The drag coefficient $C_{d_{\text{eff}}}$ is the sum of both the friction and pressure drag coefficients. The induced drag term is not taken into consideration here, as it is calculated by the VLM (see Stage 1).

5. In order to find the profile drag coefficient in streamwise direction of the wing section, the simple sweep theory can be used:

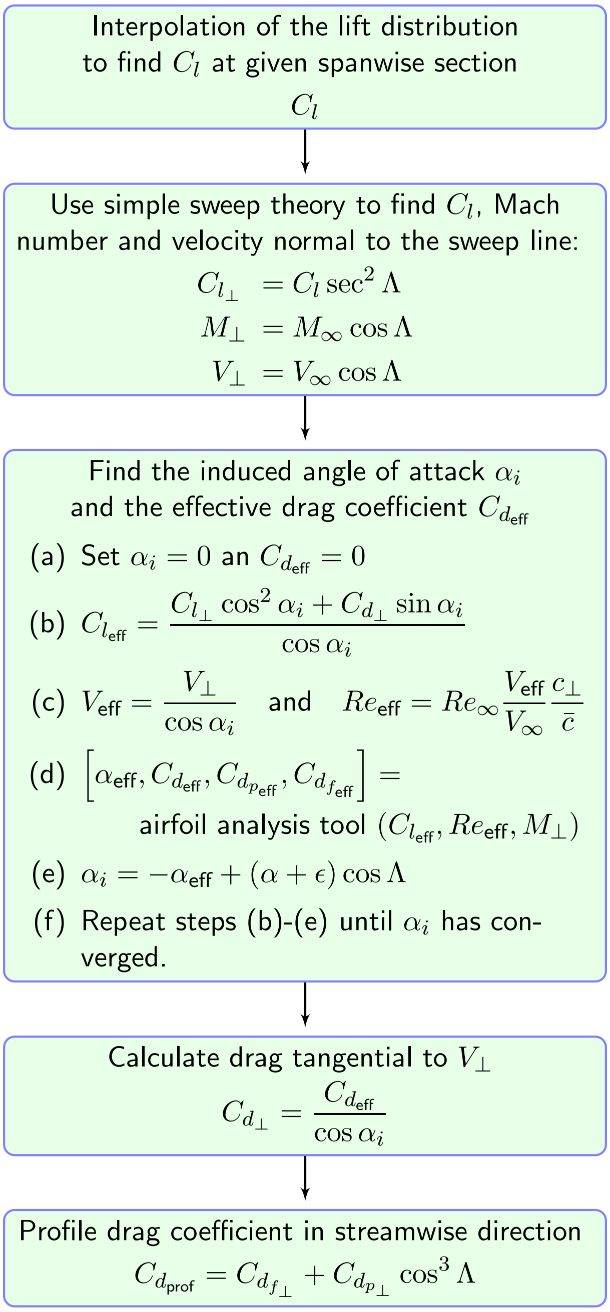
$$\left.\begin{aligned} C_{d_f} &= C_{d_{f_{\perp}}} \\ C_{d_p} &= C_{d_{p_{\perp}}}\cos^3\Lambda \end{aligned}\right\} \quad C_{d_{\text{prof}}} = C_{d_f} + C_{d_p} \tag{4-23}$$

Once all the profile drag coefficients on each wing section (Nw sections) are calculated, they can be integrated over the wing in order to find the three-dimensional profile drag coefficient.

$$C_{D_{\text{prof}}} = \frac{2}{S}\int_{0}^{b/2} C_{d_{\text{prof}}} c \; dy \tag{4-24}$$

where S and c represent the wing planform area and the chord length at a wing section.

Figure 4-5 gives an overview of the simple sweep theory implementation into the strip method.



Interpolation of the lift distribution to find C_l at given spanwise section

C_l

Use simple sweep theory to find C_l, Mach number and velocity normal to the sweep line:

$$\begin{aligned} C_{l_\perp} &= C_l \sec^2 \Lambda \\ M_\perp &= M_\infty \cos \Lambda \\ V_\perp &= V_\infty \cos \Lambda \end{aligned}$$

Find the induced angle of attack α_i and the effective drag coefficient $C_{d_{\text{eff}}}$

(a) Set $\alpha_i = 0$ an $C_{d_{\text{eff}}} = 0$

(b) $C_{l_{\text{eff}}} = \dfrac{C_{l_\perp} \cos^2 \alpha_i + C_{d_\perp} \sin \alpha_i}{\cos \alpha_i}$

(c) $V_{\text{eff}} = \dfrac{V_\perp}{\cos \alpha_i}$ and $Re_{\text{eff}} = Re_\infty \dfrac{V_{\text{eff}}}{V_\infty} \dfrac{c_\perp}{\bar{c}}$

(d) $\left[\alpha_{\text{eff}}, C_{d_{\text{eff}}}, C_{d_{p_{\text{eff}}}}, C_{d_{f_{\text{eff}}}}\right] =$ airfoil analysis tool $(C_{l_{\text{eff}}}, Re_{\text{eff}}, M_\perp)$

(e) $\alpha_i = -\alpha_{\text{eff}} + (\alpha + \epsilon) \cos \Lambda$

(f) Repeat steps (b)-(e) until α_i has converged.

Calculate drag tangential to $V_\perp$

$$C_{d_\perp} = \frac{C_{d_{\text{eff}}}}{\cos \alpha_i}$$

Profile drag coefficient in streamwise direction

$$C_{d_{\text{prof}}} = C_{d_{f_\perp}} + C_{d_{p_\perp}} \cos^3 \Lambda$$

Figure 4-5: Overview of the sweep theory implementation in the strip method for one strip

Stage 3

The lift coefficient and induced drag coefficient of the given wing configuration are calculated by the vortex lattice method at Stage 1. The profile drag is calculated as Stage 2 using a strip method extended by the simple sweep theory. Finally, to conclude the procedure of the developed quasi-3D aerodynamic solver, the total drag coefficient is determined by combining the induced drag (Stage 1) and the profile drag (Stage 2).

$$C_D = C_{D_i} + C_{D_{\text{prof}}} \tag{4-25}$$

Finally, a complete overview of the inputs and outputs is given in Appendix E.

4-5 Aerodynamic tools

Viscous airfoil data can be found in literature, e.g. see reference [53, 54], or from two-dimensional viscous airfoil calculation tools. Selected airfoil calculation tools for the development of the quasi-3D solver are

XFOIL is a program developed for the design and analysis of low Reynolds number airfoils (single-element airfoils), see reference [55, 56]. It calculates the viscous aerodynamics using a higher order panel method coupled with a viscous/inviscid interaction method. It uses the Karman-Tsien compressibility correction, which extend the validity limit of free-stream Mach numbers up to 0.7 [12].

VGK calculates the viscous aerodynamics of a single-element airfoil and is based on the full potential equations combined with integral boundary layer equations. The full potential flow equations are accurate for shock-free flows, but are only approximate for flows with shocks of appreciable strength, since the regions across the shock are generally rotational (and therefore non-isentropic). For a good representation of such flows, solutions of the Euler equations are required. However with a suitable choice of several parameters (refer to [57]), the solutions yield surface pressure distributions that correspond closely to those of true physical shock waves.

MSES is a coupled viscous/inviscid Euler method that features the design and analysis of single and multi-element airfoils at low Reynolds numbers and transonic Mach numbers. Additionally, MSES can also predict flows with transitional separation bubbles, shock waves, trailing edge and shock-induced separations [58].

At subsonic flows, the aerodynamic solver uses XFOIL for viscous airfoil calculations. XFOIL shows accurate results within this range and it is the fasted tool among the listed tools. For transonic flows, the quasi-3D solver uses VGK or MSES. As MSES is an Euler based tool, it provides more accurate results than VGK (full-potential) and comes at the cost of computation time.

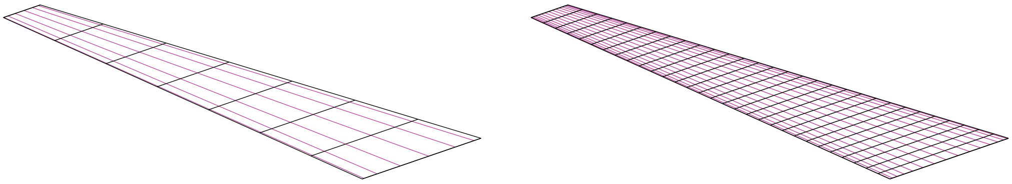
The three dimensional inviscid lift calculation of the wing is performed by means of vortex lattice method. **AVL** [43] was selected as VLM for this research.

4-6 Selection panel density and number of strips

The panel density (in vortex lattice code) and the number of strips (for strip method) have to be selected properly. The panel density has an influence on the calculation of the induced drag, whereas the number of strips affects the profile drag prediction. In this section, a proper set for the density and number of strips is derived based on convergency of these drag components.

4-6-1 Vortex lattice grid size

Vortex lattice codes model lifting surfaces (panels) by dividing them into a number of chordwise and spanwise horseshoe vortices. Both these numbers can be changed independently, and in general the results become more accurate by increasing the number of spanwise and chordwise vortices. Increasing the number of horseshoe vortices comes at a cost: the amount of computational time increases significantly, as well as the memory usage. Figure 4-6 shows a wing with different grid sizes. At a certain certain point an increase in number of vortices will hardly effect the final results, while the computational time will still increase. Therefore a number of runs was done, in which the effect of the number of horseshoe vortices in spanwise and chordwise direction on computational time and induced drag convergency was investigated. As a measure for accuracy, the induced drag was normalized by the average of the four finest grids. Then the selection of the grid size was based upon a balance between accuracy and computational time.



(a) Coarse grid **(b)** Fine grid

Figure 4-6: Half wing with different grid sizes

Figure 4-7 shows the influence of the number of spanwise (Nb) and chordwise vortices (Nc) on the normalized induced drag coefficient. From this figure it can be deduced that increasing Nc yields a more accurate induced drag coefficient. On the other hand, when Nb increases, one can deduce that by increasing Nb, the resulting induced drag coefficient converges rather well. In Figure 4-8, the time needed to compute each point is plotted against the grid size.

From the figures above, a grid size was selected to be $Nb = 24$ and $Nc = 13$ (indicated by a small black box in Figure 4-7 and 4-8, with its corresponding values displayed). The selection implies a deviation of 0.15% on the induced drag coefficients and requires about one fifth of the computational times for the finest grid tested ($Nb = 30$ and $Nc = 20$).

In addition to Figure 4-7, the convergency plots for the number of spanwise and chordwise elements are shown separately in Figure 4-9.

4-6-2 Number of strips

Using different number of strips strongly affects the computational times and final results. Consequently a number of runs was done, in which the effect of the number of strips on computational time and profile drag convergency was investigated. The number of strips only affects the profile drag in the developed aerodynamic solver. As a measure for accuracy, the profile drag coefficient was normalized by the average of the three configurations having the highest number of strips. Figure 4-10 shows the effect of the number of strips on the normalized profile drag coefficient. One can deduce that when the number of strips (Nw) increases, the resulting profile drag coefficient converges rather well. In Figure 4-11, the computational times for each number of strips is plotted.

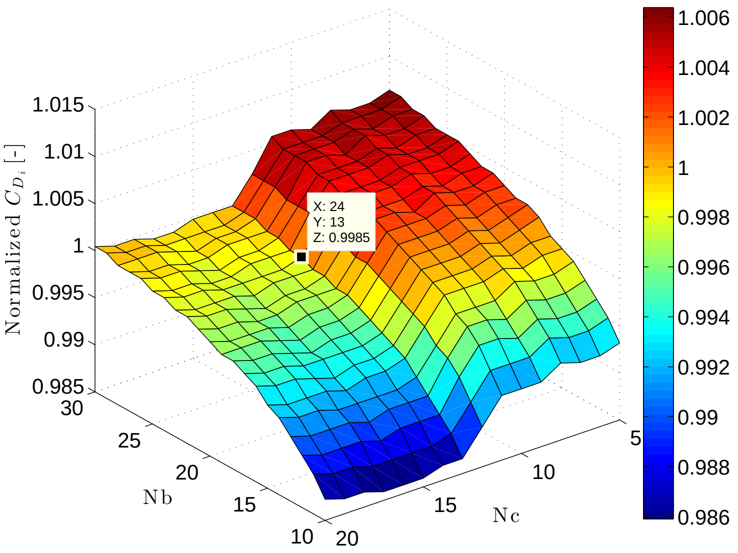


Figure 4-7: Effect of number of spanwise (Nb) and chordwise (Nc) vortices on normalized induced drag coefficient.

From these figures, a strip number was chosen to be $Nw = 8$ (indicated by a black box in Figure 4-10 and 4-11, with their values shown). This selection yields a deviation of the profile drag coefficient smaller than 0.05% with a corresponding computational time of 2.25 seconds.

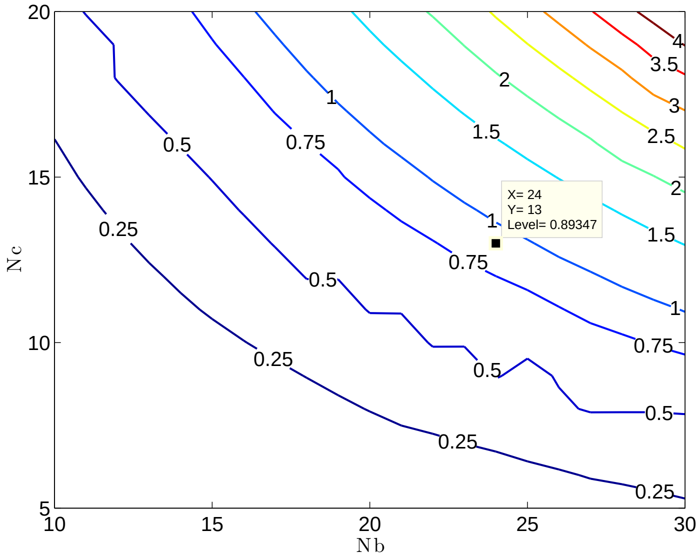


Figure 4-8: Effect of number of spanwise (Nb) and chordwise (Nc) vortices on computation time. Each level indicates the computational time seconds.

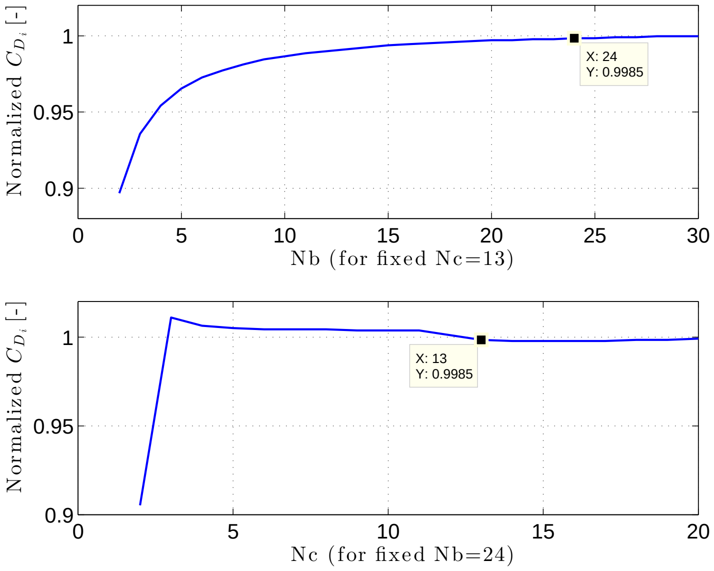


Figure 4-9: Effect of number of spanwise and chordwise elements on normalized induced drag coefficient (separate plots).

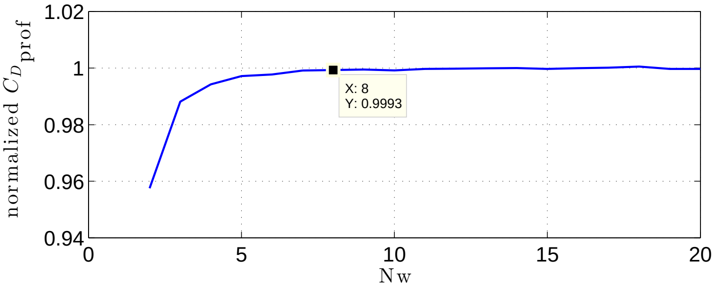


Figure 4-10: Effect of number of strips (Nw) on normalized profile drag coefficient.

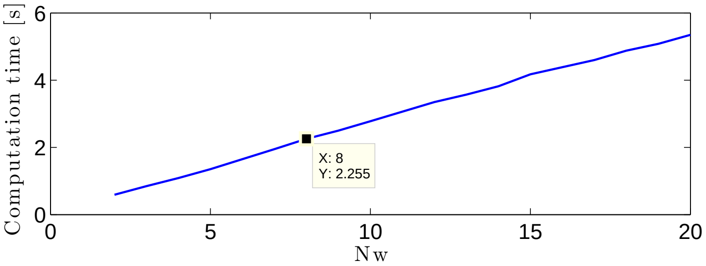


Figure 4-11: Effect of number of strips (Nw) on computational time.

4-7 Validation of the quasi-3D aerodynamic solver at low speeds

Validation of the developed aerodynamic solver is done using experimental data from the NACA Technical Report No. 572 [59]. This technical report contains experimental data of four wing configurations having different sweep angles and twisted tip sections, as shown in Figure 4-12. The results using the quasi-3D solver are also compared to two other aerodynamic solvers: VSAERO [60, 61] and MATRICSV [62].

VSAERO is a commercial aerodynamic solver that calculated the linearized potential flow equation. Compressible flow is analyzed by applying either a Karman-Tsien correction or a Prandtl-Glauert linearization to the compressible flow. MATRICSV is a full potential transonic flow solver and it can calculate both laminar and turbulent boundary layers. More information on VSAERO and MATRICS can be found in references [61, 62].

Table 4-1 shows the characteristics that define each wing configuration. In the NACA24-AA-BB wings denotation, the AA represents the quarter-chord sweep angle and BB the twist angle multiplied by 10 at the tip section (positive downward). Both the sweep angle and the twist angle are defined in degrees. All NACA wing configurations were tested at free-stream Reynolds number of 3,090,000 and freestream velocity of 69.9 ft/s.

Table 4-1: NACA wing characteristics

Parameter	Value	Unit
Æ	6	-
S	150	in^2
λ [-]	0.5	-
c_r	6.6667	ft

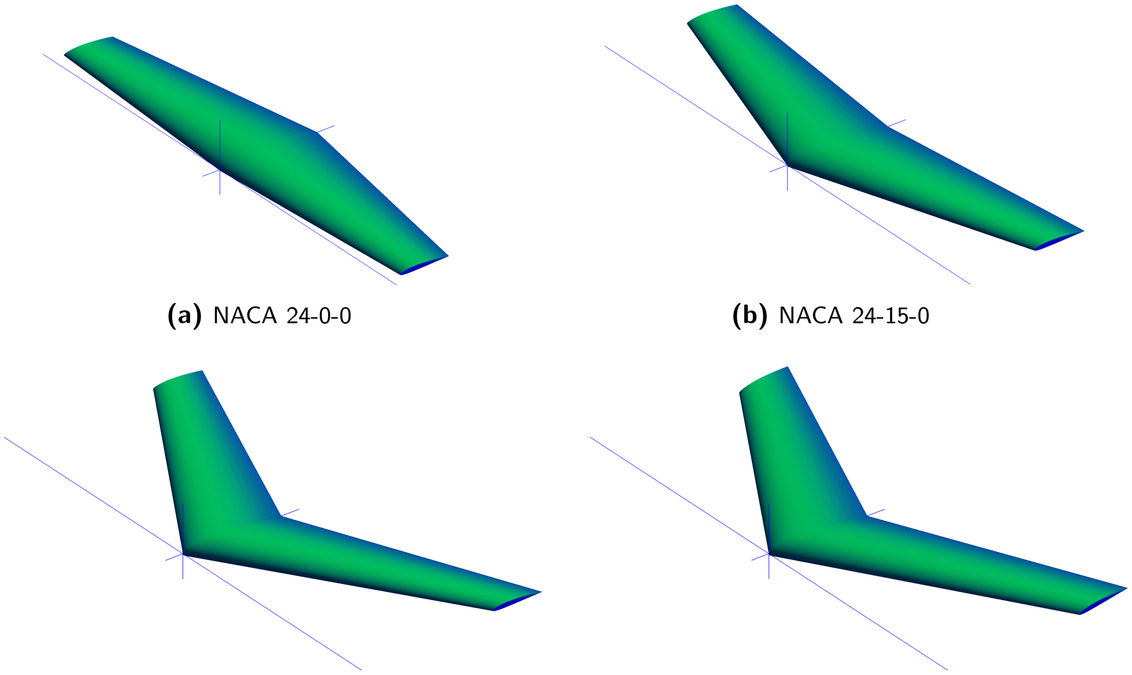
Only the validation results of the NACA 24-30-0 wing are discussed in this section. The results for the other wing configurations are similar and can be found in Appendix D.

4-7-1 NACA 24-30-0 wing

The drag and lift coefficient curves as a function of the angle of attack for the NACA 24-30-0 wing, are given by Figure 4-13 and 4-14. Figure 4-15 shows the drag coefficient curves as a function of the lift coefficient.

Comparing the C_L curves of the quasi-3D aerodynamic solver to experimental data and to results of other solves, it can be concluded that the lift is slightly over-predicted. This over-prediction in lift originates from the vortex lattice code. The drag coefficient on the other hand, is under-predicted by the quasi-3D solver. MATRICSV, the three-dimensional CFD solver based on the full potential equations, shows rather good results compared to the experimental data. Therefore, the induced and profile drag coefficient curves of the quasi-3D solver are compared to those from MATRICSV (as these data were available).

At the lower range angles of attack, the induced drag is about equal for both the quasi-3D solver and MATRICSV. Figure 4-16 shows both the induced, profile and total drag coefficient curves of the quasi-3D and the MATRICSV solver. At higher angles of attack however, the quasi-3D solver shows higher values for induced drag coefficients. On the other hand, the profile (viscous) drag is overall under-predicted compared to the MATRICSV data. Especially at higher α values, this under-prediction becomes more significant.



(a) NACA 24-0-0

(b) NACA 24-15-0

(c) NACA 24-30-0

(d) NACA 24-30-85

Figure 4-12: Graphical representation of the wing configurations used for low speed validation of the quasi-three-dimensional aerodynamic solver

The numeric values values from the $C_D - C_L$ curves are provided by Table 4-2. The error in drag coefficient prediction (denoted by ΔC_D) represents the difference between the predicted and the experimental values. The error in drag prediction by MATRICSV is smaller than those from quasi-3D solver. However, the quasi-3D solver has a wider range of validity than MATRICSV. Table 4-3 provides the computational times for a single case for each aerodynamic solver. Note that all runs were performed on computers with similar hardware performance.[1] Based on Table 4-2 and Figure 4-15, one can conclude that the developed solver shows slightly better results than the 3D panel method VSAERO for the NACA 24-30-0 wing. Moreover, the quasi-3D is significantly faster than VSAERO and MATRICSV.

[1] Using computers with better performance will reduce these computational times.

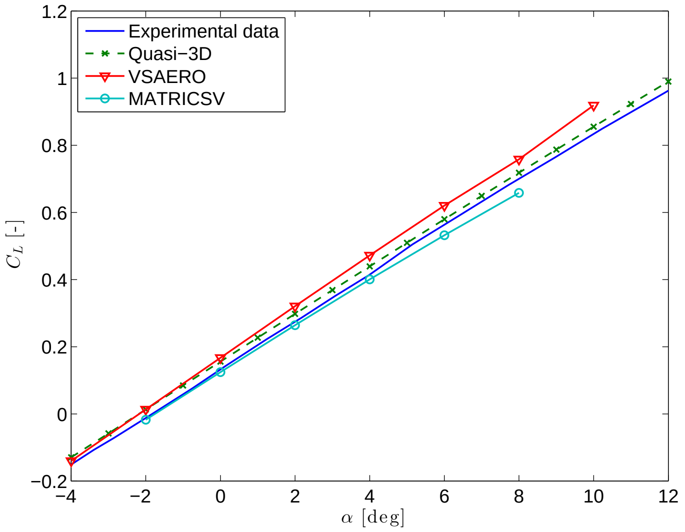


Figure 4-13: $C_L - \alpha$ curve of the NACA 24-30-0 wing

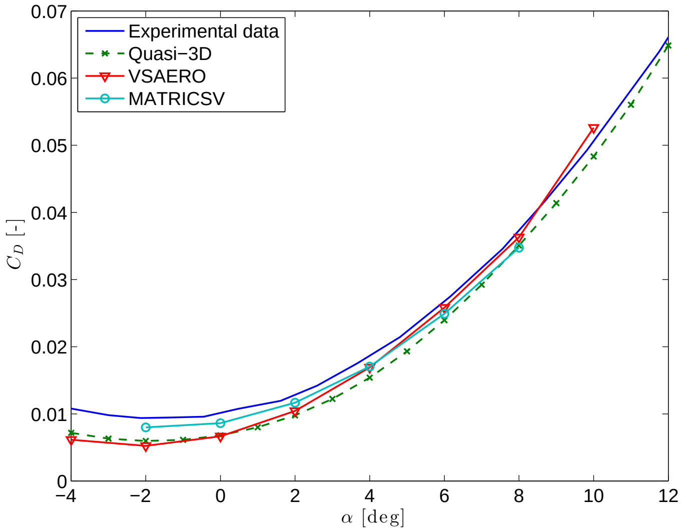


Figure 4-14: $C_D - \alpha$ curve of the NACA 24-30-0 wing

Table 4-2: Error analysis of different aerodynamic solver for the wing drag coefficient of the tapered NACA 24-30-0 wing

	Experimental	Quasi-3D		VSAERO		MATRICSV	
C_L	C_D	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$
-0.129	0.0104	0.0072	-3.226	0.0060	-4.403	-	-
0.013	0.0094	0.0060	-3.420	0.0052	-4.147	0.0079	-1.467
0.156	0.0104	0.0068	-3.627	0.0065	-3.931	0.0091	-1.315
0.298	0.0136	0.0098	-3.837	0.0097	-3.906	0.0128	-0.853
0.439	0.0199	0.0154	-4.462	0.0153	-4.573	0.0191	-0.769
0.580	0.0279	0.0240	-3.899	0.0231	-4.705	0.0284	0.532
0.718	0.0390	0.0351	-3.920	0.0330	-5.966	-	-
0.855	0.0530	0.0483	-4.667	0.0455	-7.490	-	-
0.990	0.0693	0.0649	-4.445	0.0617	-7.602	-	-

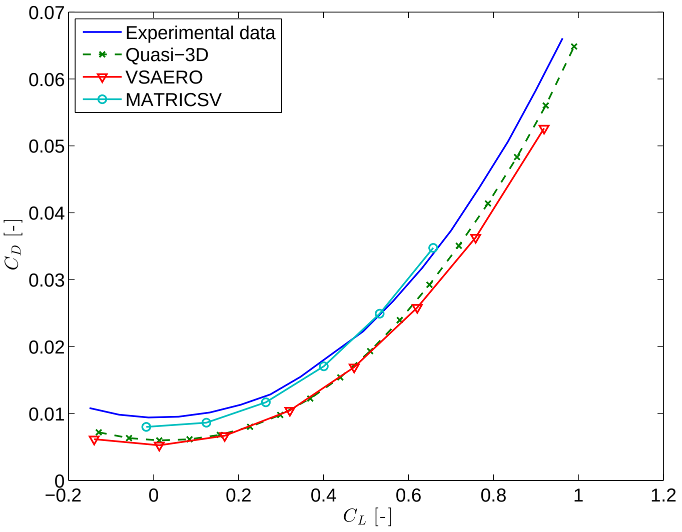


Figure 4-15: $C_D - C_L$ curve of the NACA 24-30-0 wing

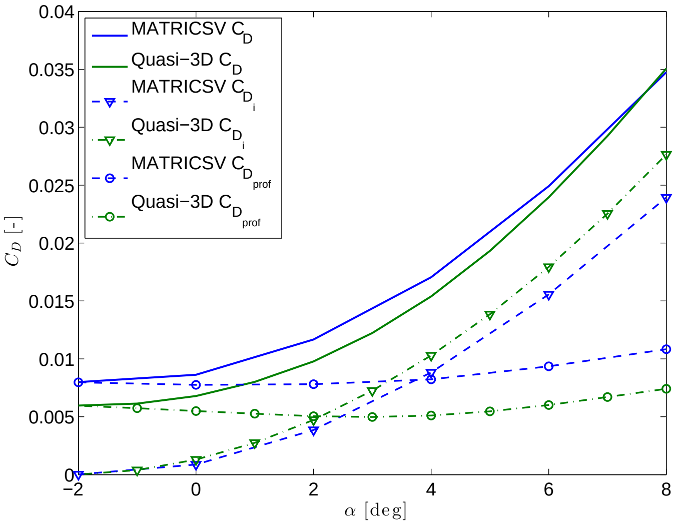


Figure 4-16: Drag differences of NACA 24-30-0 wing for quasi-3D solver and MATRICSV

Table 4-3: Computational time per case

Solver	Computational time
Quasi-3D	20 sec. - 50 sec.
VSAERO* [63]	5 min. - 10 min.
MATRICSV	4 min. - 5 min.

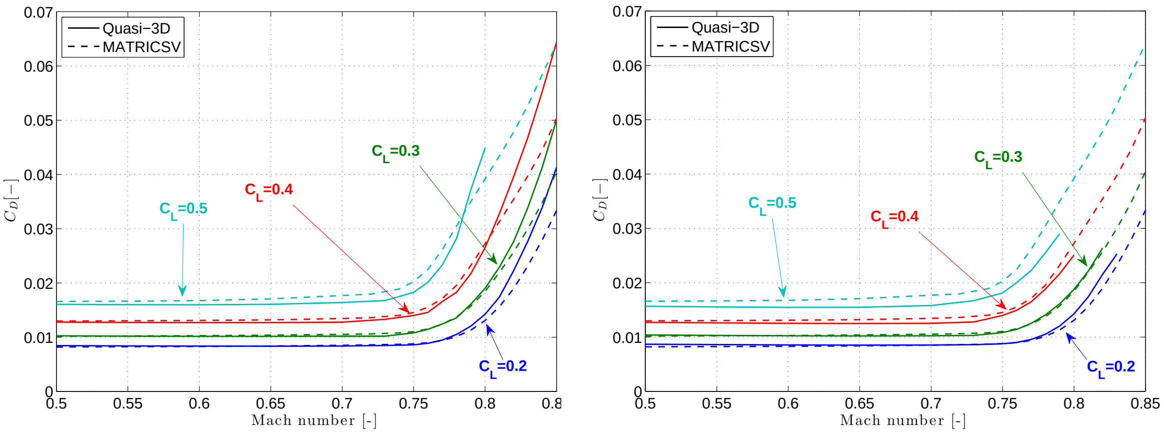
* *from which about 4 min. were required for input file generation and output processing*

4-8 Validation of the quasi-3D aerodynamic solver at high speeds

In addition to the low speed validation, the developed solver is also validated using a test case at transonic conditions. The Fokker 100 wing is used for validation of the solver at high speeds. Both the results of the solver using AVL-MSES and AVL-VGK are both illustrated and compared to the results of MATRICSV. Note that the results of the developed solver are compared to a three-dimensional CFD tool, as experimental data was not available.

4-8-1 Drag coefficients comparison

The Fokker 100 wing is analyzed at an altitude of 37,000 ft using different Mach numbers and lift coefficient combinations. VSAERO is not valid in these flight regimes. As experimental data of the flying wing was not available, a comparison was made using results from MATRICSV solver. Figure 4-17 shows the resulting drag coefficients using both AVL-VGK and AVL-MSES combination. Note that for all aerodynamic solvers, a fixed transition at 5% chord was used for the high-speed cases to ensure convergency.



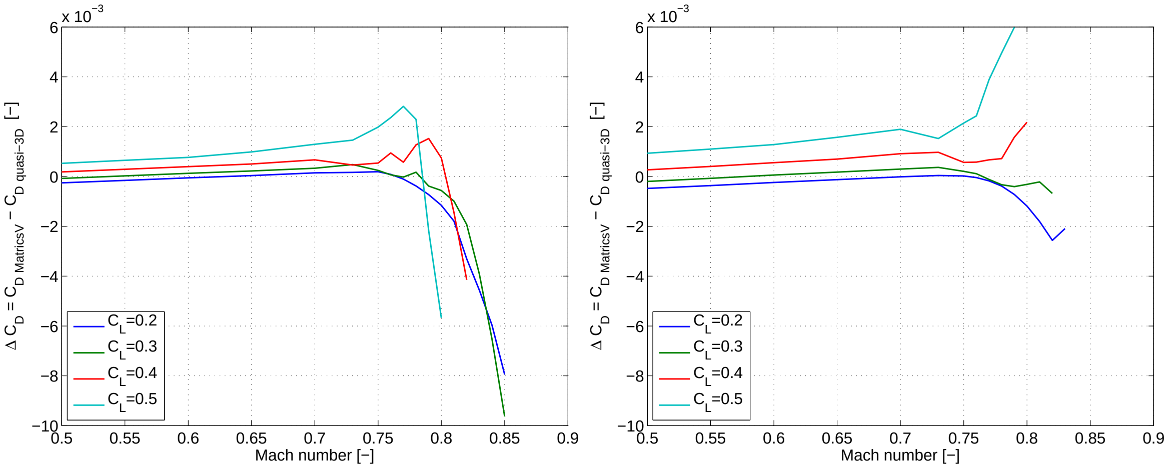
(a) AVL-MSES

(b) AVL-VGK

Figure 4-17: C_D versus Mach comparison with MATRICSV

In general, the drag coefficient remains relatively constant all the way up to about $M = 0.75$. As the Mach number increases slightly further, the drag usually rises slowly at first. By increasing the Mach number even further, a point will be reached where the drag coefficient suddenly starts to increase. This is the point where the wave drag (or compressibility drag) appears. For increasing C_L, it can be deduced that the under-prediction in comparison with the MATRICSV results becomes larger. The drag coefficient of the quasi-3D solver using AVL-MSES, increases more than the MATRICSV data at higher Mach numbers (M=0.75-0.78). For the quasi-3D solver using AVL-VGK, this higher drag increase is only present for the lower C_L values whereas for higher C_L values, this drag increase is negative. Figure 4-18 shows these differences between the solver and the MATRICSV data. The difference in drag prediction between the quasi-3D solver and MATRICSV increase rapidly at Mach numbers higher than 0.75.

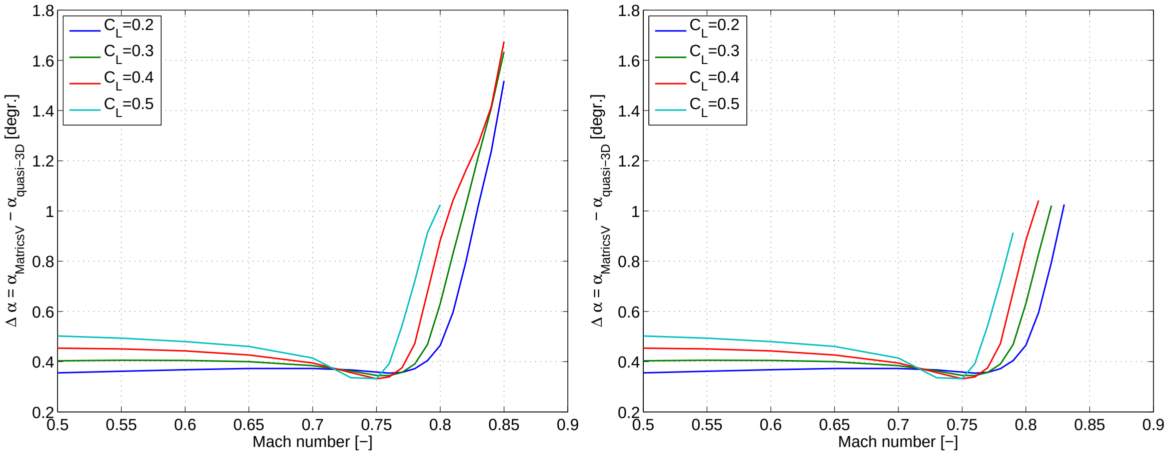
Figure 4-19 shows the differences in the angle of attack between MATRICSV and the quasi-3D solver (for both AVL-MSES and AVL-VGK). In general, the angles of attack of the quasi-3D solver are under-predicted compared to MATRICSV. This means that for the same angle of attack, AVL slightly over-predicts the lift. At the low-speed validation of the developed aerodynamic tool (see Section 4-7), the lift was also over-predicted compared to both the experimental data and the MATRICSV results.



(a) ΔC_D from AVL-MSES

(b) ΔC_D from AVL-VGK

Figure 4-18: C_D difference between AVL-MSES/AVL-VGK and MATRICSV



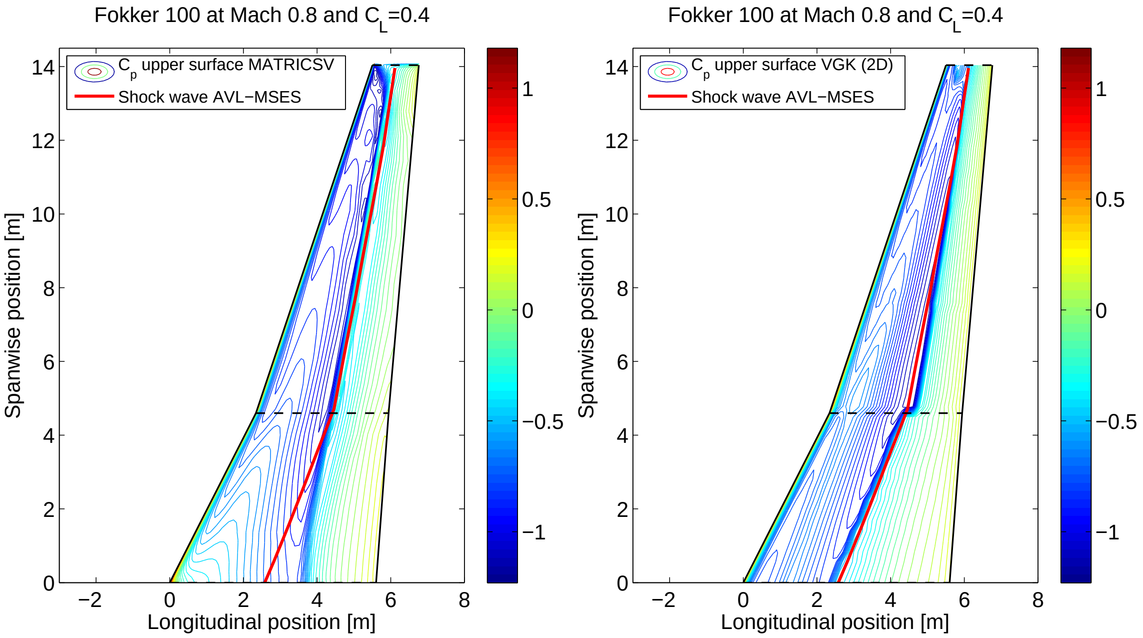
(a) $\Delta\alpha$ from AVL-MSES

(b) $\Delta\alpha$ from AVL-VGK

Figure 4-19: α difference between AVL-MSES/AVL-VGK and MATRICSV

4-8-2 Pressure distribution comparison

Three-dimensional aerodynamic effects, i.e., root and tip effect, are not modeled by the quasi-3D solver (see Section 4-2). In this subsection the flow characteristics of the Fokker 100 wing flying at $M_\infty = 0.8$ at $C_L = 0.4$ are shown for MATRICSV (full 3D CFD) and the quasi-3D solver. The pressure contour plot of the upper wing surface from MATRICSV is provided by Figure 4-20a. This figure shows the bending of the isobars near the root and tip section, which illustrates the so-called root and tip effect. Note that the dark blue line indicates the position of the pressure drop (or shock wave line). Figure 4-20b shows the pressure contour results from AVL-VGK, in which no root or tip effect is present. Both contour plots present the shockwave line (in red) calculated by AVL-MSES. MATRICSV shows a more realistic representation of the actual flow over the wing. The lack of these three-dimensional effect is due to the use of the simple sweep theory. As can be deduced from Figure 4-20b, the shockwave lines of both VGK and MSES are quite similar. Note the aft movement of the shockwave position near the kinked wing section in Figure 4-20b. This aft movement can be explained by the effect of the different sweep angle (for the inner and outer wing) on the simple sweep theory.



(a) Comparison of MSES shockline with MATRICSV data

(b) Comparison of MSES shockline with VGK data

Figure 4-20: Upper surface pressure coefficient distributions from MATRICSV (full potential 3D) and VGK (full potential 2D) with MSES shockwave locations along span

The minimum pressure along the half wing is shown in Figure 4-21. It can be deduced that the minimum pressure near the tip station differs from the MATRICSV results. The minimum pressure of MATRICSV towards the tip section decreases strongly in strength. This is caused by the tip effect (three-dimensional aerodynamic effect) captured by MATRICSV.

Figure 4-22 shows the pressure distributions of each solver for 3 wing sections: one near the root, a midspan section and one near the wing tip. As can be seen for the pressure distribution at the near root section, the shock position (shock) for the MATRICSV solver is positioned more aft than those from the quasi-3D solver. This is in agreement with Figure 4-20 and is caused by the three-dimensional root effect.

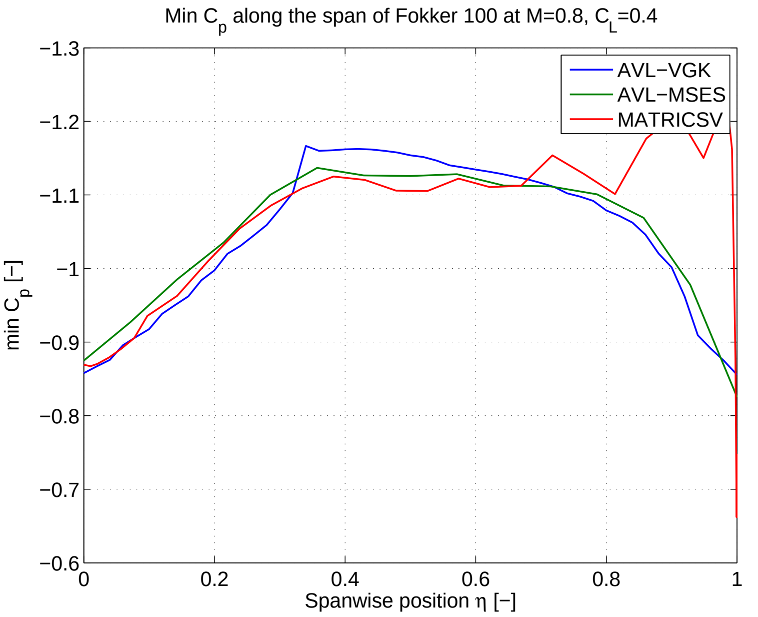
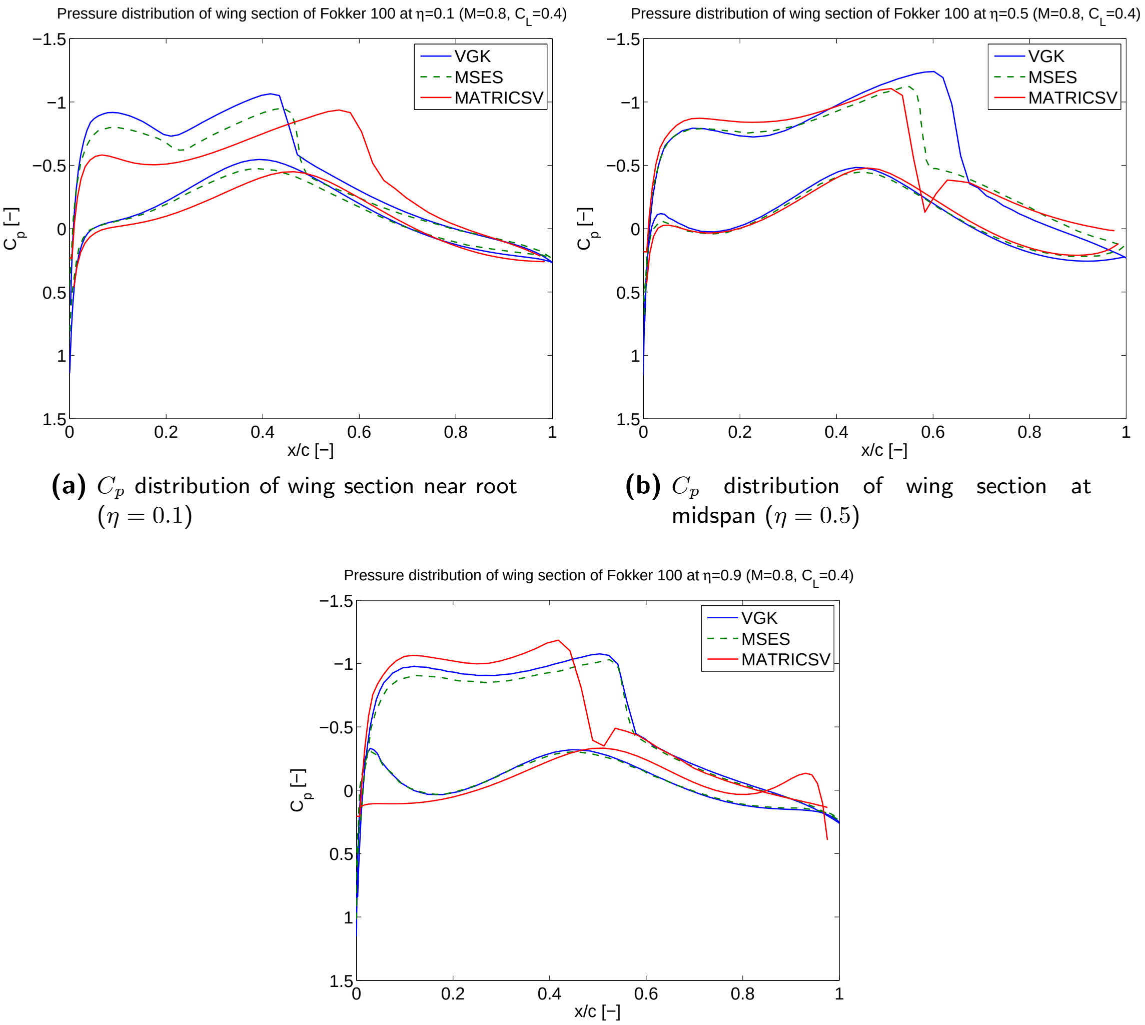


Figure 4-21: Minimum C_p distribution along span



(a) C_p distribution of wing section near root $(\eta = 0.1)$

(b) C_p distribution of wing section at midspan $(\eta = 0.5)$

(c) C_p distribution of wing section near tip $(\eta = 0.9)$

Figure 4-22: Pressure coefficient distribution comparison for three wing sections

Chapter 5

Wing weight estimation methods

In this chapter several existing statistical wing weight estimation methods are explained and compared with each other. This is done by analyzing their accuracy and sensitivity to different wing design variables. These methods are compared with a quasi-analytical method (EMWET), based on accuracy, design sensitivity and computational time.

5-1 Different methods

5-1-1 Torenbeek (1) method

Torenbeek has developed several weight estimation methods, of which two statistical methods were used in present research. The first method is a simplified approximation, derived from a weight estimation method by Torenbeek [30, 64]. This method can be used to predict the structural wing weight of civil airplanes with cantilever wings and is presented by the following equation:

$$W_{\mathsf{w}} = W_G \cdot k_{\mathsf{w}} \cdot b_{\mathsf{s}}^{0.75} \left[1 + \sqrt{\frac{b_{\mathsf{ref}}}{b_{\mathsf{s}}}} \right] n_{\mathsf{ult}}^{0.55} \left(\frac{b_{\mathsf{s}} / (c_r \cdot (t/c)_r)}{W_G / S} \right)^{0.30} \tag{5-1}$$

where b_s is the structural wing span. The factor of proportionality, k_w, is as follows (for metric units):

- For light aircraft, $W_{\mathsf{to}} \leq 5670$ kg: $k_w = 4.90 \times 10^{-3}$ and $W_G = W_{\mathsf{to}}$ in kg.
- Transport category aircraft, $W_{\mathsf{to}} > 5670$ kg: $k_w = 6.67 \times 10^{-3}$ and $W_G = W_{\mathsf{zf}}$ in kg.

The weight given by Eq. (5-1) includes the weight of high-lift devices and ailerons. If spoilers and speed brakes are incorporated an additional weight equal to 2% of the total wing weight should be added. In the case of wing-mounted engines the wing weight must be reduced by 5% and 10% for 2 or 4 engines respectively. Additional reductions can be made for braced wings or in the case that the main undercarriage is not mounted to the wing.

In this method, the primary wing structural weight is determined based on the requirement that in a specified critical flight condition the bending moment due to the wing lift must be resisted by the wing box structure. The weight of high-lift devices is estimated based on the critical loading conditions at the flap design speed. The accuracy of this method has been assessed using the actual wing weight of 46 aircraft (dating from about 1940-70s). A standard deviation of 9.64 percent was achieved [30, 64]. Application of the method to several high-subsonic short-haul airplanes led to the conclusion that the original method underestimates the wing weight. Main reasons for this underestimation can be assumed to be caused by (a) the extra weight (up to 20%) required to provide adequate stiffness against flutter and (b) the weight penalty due to long service life[64]. Since a simplified expression is used in this paper, a larger deviation is to be expected than the indicated standard deviation.

5-1-2 Torenbeek (2) method

Another statistical method, also developed by Torenbeek, was used for this research. A practical wing weight formula presented by Torenbeek [65] was derived from a quasi-analytical method described by the same author [32]:

$$W_{\mathsf{w}} = 0.06\frac{w}{\sigma_{\mathsf{r}}}n_{\mathsf{ult}}\sqrt{W_{\mathsf{to}} \cdot W_{\mathsf{zf}}}\frac{b^3}{S}\frac{1}{(t/c)_{\mathsf{ave}}\cos^2\Lambda}\frac{1+2\lambda}{1+\lambda} + wt_{\mathsf{ss}}S \tag{5-2}$$

In this expression the term $\frac{w}{\sigma_r}$ represents the weight of the materials in the upper and lower wing panels. These panels are required to withstand the bending load due to lift, which is represented by $n_{\mathsf{ult}}\sqrt{W_{\mathsf{to}} \cdot W_{\mathsf{zf}}}$. The factor 0.06 is a non-dimensional constant of proportionality. This value accounts for the average values of the bending relief due to e.g. wing and engine masses, structural efficiency, spar locations, etc. The design weight W_{des} and ultimate load factor n_{ult} refer to critical combinations of manoeuvre/gust loads, weight distributions and operating speeds. The meaning of the other values is as follows:

- The term $\frac{w}{\sigma_r}$ is the ratio of the specific wing material weight (w) to the mean stress level at the wing root σ_{r}. Torenbeek [65] suggested that a statistical average value based on a Al-alloy structure can be used:

$$\frac{w}{\sigma_{\mathsf{r}}} = 40\left[1 + 1.10\left(W_{\mathsf{to}}/10^6\right)^{-1/4}\right] \times 10^{-6} \tag{5-3}$$

- The average thickness-to-chord ratio t/c is found by suitable weighing of the wing thickness in spanwise direction.
- The sweep angle Λ is theoretically related to the elastic axis. However, since its location is usually unknown, it is for this method approximated by the mean quarter-chord line. Note that textbooks usually indicate that the elastic axis lies at about 40% chord [66, 67].
- The smeared skin thickness t_{ss} is defined by

$$t_{\mathsf{ss}} = 0.004\left(1 + \sqrt{W_{\mathsf{to}}/10^6}\right) \tag{5-4}$$

As it is mentioned before, this method was derived from a more sophisticated method [32]. The original method is characterized as follows:

- The wing structural weight is divided into functional components. The weight of each component is estimated by implementing either a statistical or a semi-analytical equation.

- The wingbox structure is sized in such a way to have minimum amount of material required to resist the bending moment in the upper and lower equivalent panels and the minimum amount of material to resist the shear force in the spar webs. These values are obtained implementing an analytical structural analysis method.
- The effect of variation in the thickness-to-chord ratio and the compressive stresses (buckling criteria) have been taken into account.
- A series of semi-empirical equations have been developed to estimate the ribs weight, the non-optimum weight penalties, the weight of the fixed leading and trailing edge structures as well as the weight of the high-lift devices and the control surfaces.
- The mean stress level is computed as a function of the aircraft maximum take-off weight using an empirical equation. This equation is valid for Al-alloy structures. A user-selected design stress levels is also possible for alternative material applications.

The original method was validated using several transport aircraft. The results are shown in Table 5-1. Although the small number of aircraft considered does not permit to determine the mean prediction error. Nevertheless the table indicates that the original method yields good results for the size effect. The results from the simplified method will most-likely deviate more than the original one.

Table 5-1: Estimated versus actual wing weight for several transport aircraft [65].

		Wing weight		
Aircraft	W_{to} [kN]	computed [kN]	actual [kN]	error [%]
Boeing 747-100	3158	391.6	384.4	$+1.9$
Airbus A340	2486	345.0	340.9	$+1.2$
Fokker F-28 Mk4000	315.8	32.03	33.28	-3.8
Cessna Citation II	59.16	5.950	5.730	$+3.8$

5-1-3 Shevell method

Shevell [31] developed a statistical method based on a wing weight index, which is related to the weight of the fully-stressed wing box. This weight index is defined as follows (in imperial units):

$$I_{\text{w}} = \frac{n_{\text{ult}} b^3 \sqrt{W_{\text{to}} \cdot W_{\text{zf}}}}{(t/c)_{\text{ave}} S \cos^2 \Lambda} \frac{(1+2\lambda)}{(1+\lambda)} 10^{-6} \tag{5-5}$$

From Eq. (5-5) one can observe that the weight index is proportional to the compressive and tensile loads due to the bending moments at the root of the wing in the skin and the spar caps. Figure 5-1 shows the variation of the ratio of the wing weight to the square foot of the wing planform area with the wing weight index. Based on this figure the total wing weight is estimated as follows [31, 68]:

$$\frac{W_{\text{w}}}{S} = 4.22 + 1.642 I_{\text{w}} \tag{5-6}$$

The first term on the right hand side of this equation is a constant and represents the secondary wing weight. The second term varies in proportion to the amount of material required to resist the applied bending loads and represents the primary wing weight.

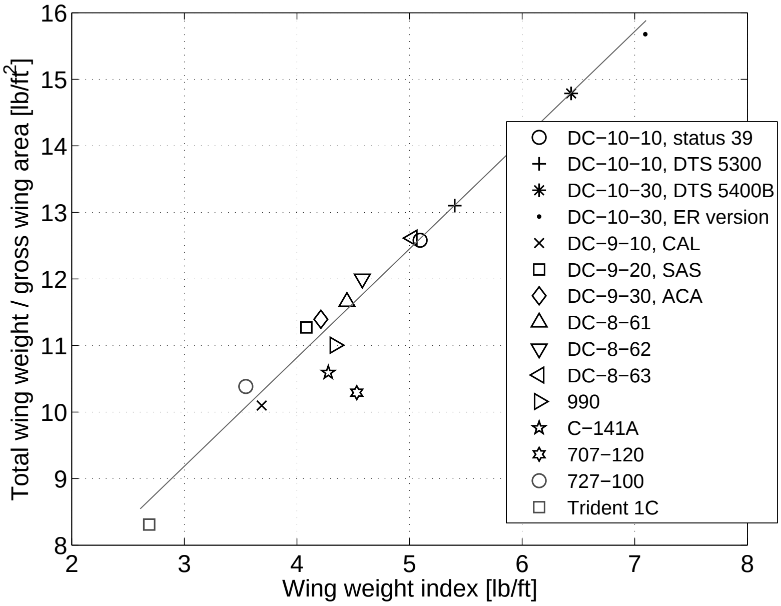


Figure 5-1: Statistical wing weight correlation (adopted from [31]).

5-1-4 Howe method

A statistical weight estimation method was developed by Howe[29], in which the wing weight is derived based on the number of passengers or the payload weight for a particular airplane. In this fundamentally empirical method, the wing weight is calculated as a function of the main geometric and operational parameters. The estimated wing weight is correlated to some actual wing weight values. The equation for wing weight estimation is as follows:

$$W_{\mathsf{w}} = g\frac{C_1}{C_5}\left[\text{Æ}^{0.5}S^{1.5}\sec\Lambda\left(\frac{1+2\lambda}{3+3\lambda}\right)\frac{W_{\mathsf{to}}}{gS}n_{\mathsf{ult}}\left(\frac{V_{\mathsf{D}}}{(t/c)_r}\right)^{0.5}\right]^{0.9} \tag{5-7}$$

Coefficient C_1 in this expression is defined as follows:

a) Long range, $s > 5000$ km

$$C_1 = 0.00072 - 0.0005(270 + 0.05s)PAX \times 10^{-6} \tag{5-8}$$

b) Short/medium range, $W_{\mathsf{to}} \leq 46,000$ kg

$$C_1 = 0.00167 - 0.016(370 + 0.03s)PAX \times 10^{-6} \tag{5-9}$$

c) Turboprop, $W_{\mathsf{to}} \leq 46,000$ kg

$$C_1 = 0.00149 - 5.8PAX \times 10^{-6} \tag{5-10}$$

d) Long range jet freight aircraft

$$C_1 = 0.00072 - 0.0005(2.08 + 0.00038s)W_{\mathsf{pay}} \times 10^{-6} \tag{5-11}$$

e) Turboprop freight aircraft

$$C_1 = 0.00077 - 0.00053(2.08 + 0.00038s)W_{\mathsf{pay}} \times 10^{-6} \tag{5-12}$$

Table 5-2: Coefficient C_5 in equation (5-7).

Type of aircraft	C_5
Tailless delta	1.10
Long haul jet transports	1.16
Short/medium haul jet transports	1.20
Executive jet aircraft	1.30
All other types	1.24

The original weight estimation equation described above calculates the weight of all lifting surfaces. Dividing this total weight by the lifting surface factor C_5 (see Table 5-2) yields the individual wing weight [29].

This method was tested on 118 aircraft for which sufficient data was available or could be reasonably assumed [69]. These tested aircraft have take-off weight range from about 700 kg (Cessna 150) to more than 377000 kg (Boeing 747-200). Figure 5-2 shows the results of the civil aircraft that were used as the test cases, together with ±10% boundaries. For 76% of those aircraft, the error of the wing weight prediction is less than 10%.

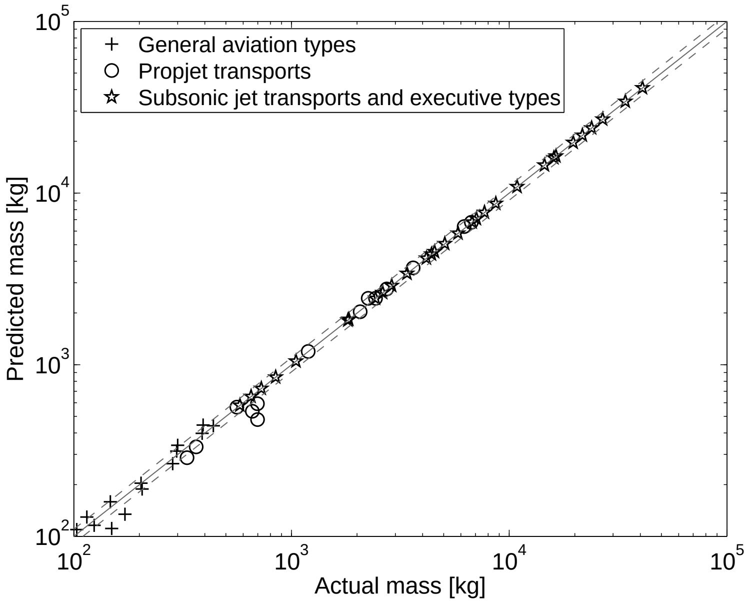


Figure 5-2: Howe method – actual and predicted wing masses for civil aircraft [69].

5-1-5 LTH method

The last statistical method which is used in this research is the empirical equation developed by Dorbath on behalf of LTH (Luftfahrttechnisches Handbuch) organization [70]. The equation is presented below:

$$W_{\text{w}} = 2.20013g \cdot 10^{-4} \left[401.146S^{1.31} + \left(\frac{W_{\text{to}}}{g}\right)^{1.1038}\right] \cdot (t/c)_{\text{rep}}^{-0.5} \mathcal{R}^{1.5} \frac{1}{\cos\Lambda} \tag{5-13}$$

In this equation the representative thickness-to-chord ratio $(t/c)_{\text{rep}}$ is defined as $0.6(t/c)_r + 0.3(t/c)_k + 0.1(t/c)_t$. Table 5-3 shows the range of the values for which this method is valid.

The validation results of this method are shown in Figure 5-3. This method has a standard deviation of 6.2%.

Table 5-3: Range of values for using the LTH method

Parameter	Range of values	Unit
Wing mass	4,100 – 50,300	[kg]
Reference wing area	75 – 550	[m^2]
Maximum take-off mass	40,000 – 400,000	[kg]
Representative t/c ratio	0.10 – 0.15	[-]
Aspect ratio	6.9 – 9.6	[-]
Quarter-chord sweep angle	15.0 – 37.5	[°]

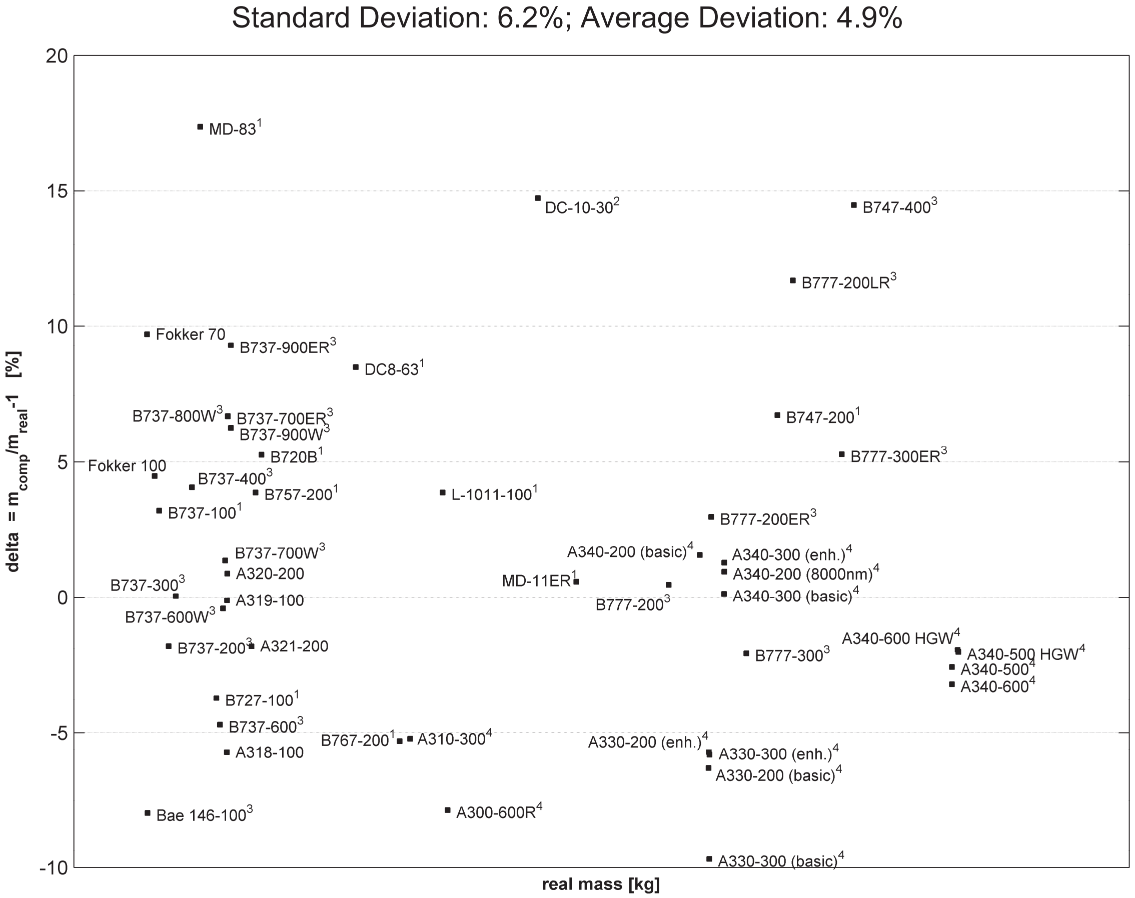


Figure 5-3: Validation results of the LTH method [70].

5-1-6 Elham Modified Weight Estimation Technique (EMWET)

In addition to the statistical methods, a quasi-analytical weight estimation method is also used for this research. The student version of the EMWET (Elham Modified Weight Estimation Technique) tool [34] is used as a quasi-analytical weight estimation method. This tool makes use of an analytical approach to size the wingbox primary structure. The structural and material parameters, the applied loads and the geometrical data of the wing planform and relative airfoils are required to implement this method. A realistic aerodynamic loads distribution over the wing span is computed using a commercial Vortex Lattice Method tool, AVL [43]. Several passenger aircraft from different manufacturers (Fokker, Boeing and Airbus) have been used for validation. The validation study done by Elham [34] has demonstrated an average accuracy in the order of 2% as shown in Table 5-4. This level of accuracy is comparable to the higher class methods such as the FEM based weight prediction methods, while the computational time is in the same order of magnitude as the empirical methods.

Table 5-4: Validation results of EMWET.

Aircraft	W_{to} [kg]	Error [%]
Fokker 50	20,820	-0.72
B737-200	52,390	$+0.15$
B727-300	95,028	-2.71
A300-600R	170,500	$+1.86$
A330-300	217,000	-2.18
B777-200	242,670	$+2.66$

The EMWET tool calculates the primary and secondary weights separately. However, the student version of EMWET uses a regression to find the total wing weight (W_{w}) based on the analytically computed wing box weight (W_{calc}), as shown in Figure 5-4. Applying a power regression results in the following equation:

$$W_{\text{w}} = 10.147 W_{\text{calc}}^{0.8162} \quad \text{with} \quad R^2 = 0.9982 \tag{5-14}$$

Note that by using the student version instead of the full version of EMWET the accuracy of the results might slightly deviate from the average error of 2%, as will be shown later when comparing the accuracy of the treated methods.

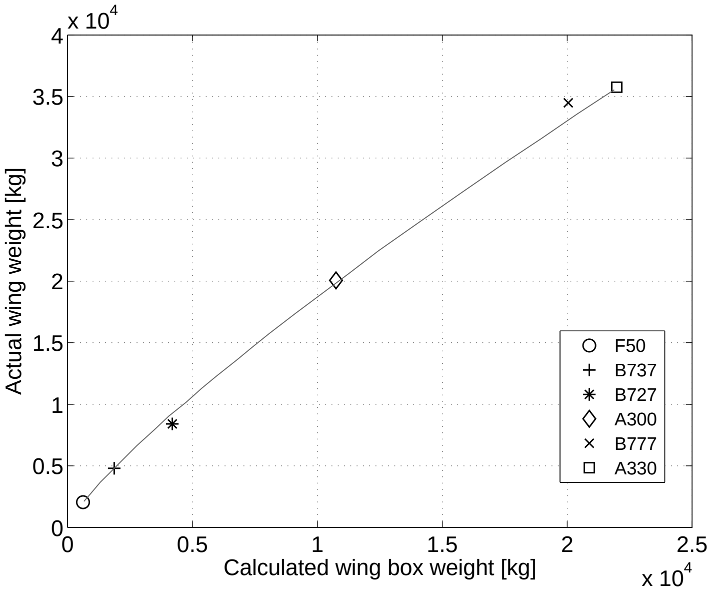


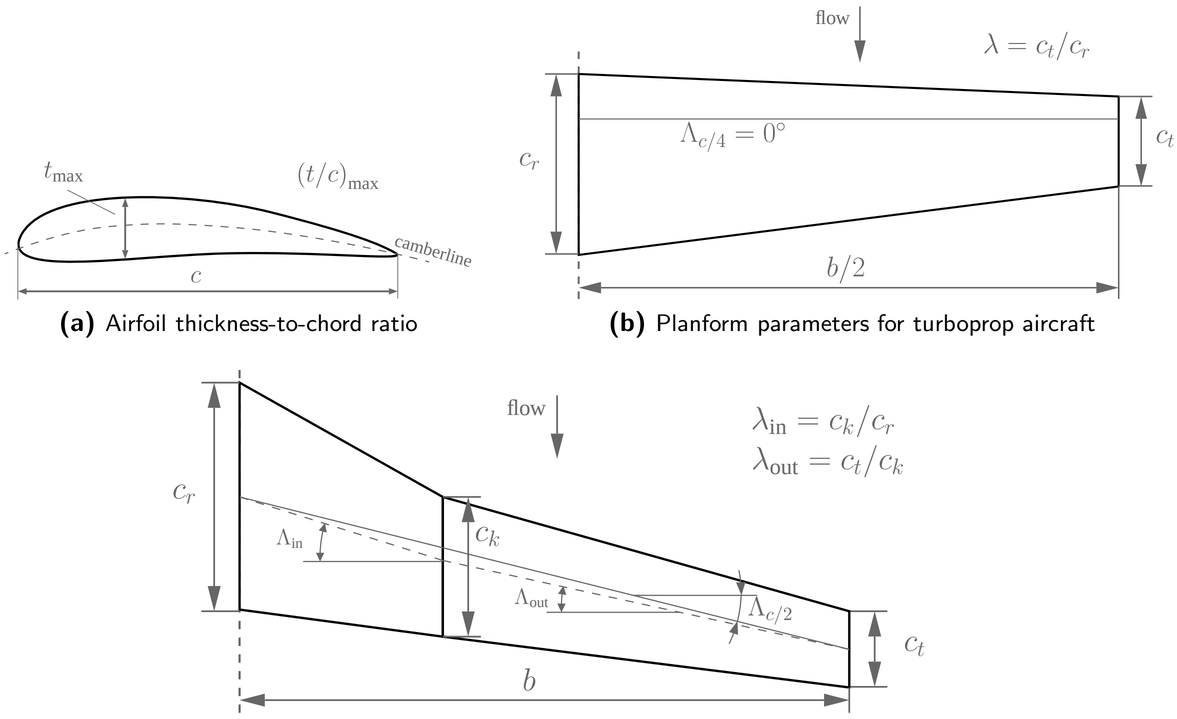
Figure 5-4: Actual total weight of the wing versus the analytically computer wing box weight (ribs and non-optimum weight excluded) [34].

5-2 Method comparison

The above mentioned weight estimation methods are compared by analyzing their accuracy and sensitivity to several independent wing parameters. A distinction can be made among the independent wing parameters for the sensitivity analyses:

- wing span b
- root chord length c_r
- taper ratio λ (tip chord length divided by root chord length)
- thickness-to-chord ratio at root $(t/c)_r$ and tip section $(t/c)_t$
- twist angle at tip section ϵ_t (downward negative)
- thickness-to-chord ratio at kink section $(t/c)_k$, only for turbojet aircraft
- mid-chord sweep angle $\Lambda_{c/2}$, only for turbojet aircraft

Note that the quarter-chord sweep angle of the turboprop aircraft is near 0° and consequently the sensitivity to sweep for the turboprop aircraft is not considered.



(a) Airfoil thickness-to-chord ratio

(b) Planform parameters for turboprop aircraft

(c) Planform parameters for turbojet aircraft

Figure 5-5: Wing parameters.

Figure 5-5 illustrates the wing parameters, of both the turboprop and turbojet aircraft, that are used for the method comparison. Table 5-5 gives an overview of the dependency of each method to those mentioned parameters. Each method depends on the maximum take-off weight, wing span, sweep angle, taper ratio and the root thickness-to-chord ratio. Investigating the method presented by Howe (Eq. (5-7)) closely, it can be concluded that the root chord length is canceled out by the reference wing area and the aspect ratio (as they include c_r). Hence, the Howe method is the only method that does not depend on c_r. Another remarkable aspect, analyzing the parameter dependency of the method, is that only the Torenbeek (2) and EMWET methods require the maximum zero-fuel weight as an input. Torenbeek (1) and Howe are the only methods independent of the tip thickness-to-chord ratio.

Table 5-5: Wing weight estimation method sensitivity to wing parameters.

	Wing weight as a function of ...									
Method	W_{to}	W_{zf}	b	c_r	Λ	λ	$(t/c)_r$	$(t/c)_k$	$(t/c)_t$	ϵ_t
Torenbeek (1)		✓	✓	✓	✓	✓	✓			
Torenbeek (2)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Howe	✓		✓		✓	✓	✓			
EMWET	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Shevell	✓		✓	✓	✓	✓	✓	✓	✓	
LTH	✓		✓	✓	✓	✓	✓	✓	✓	

5-2-1 Accuracy analysis

The accuracy of all methods is analyzed using the actual wing weight of a turboprop and a turbojet passenger aircraft. Fokker 50 is selected as the turboprop aircraft and Fokker 100 is selected as the turbojet aircraft. The actual wing weight of those aircraft is available in [51]. Table 5-6 shows the errors of both the turboprop and the turbojet aircraft. The error was calculated as follows:

$$\epsilon = \frac{W_{\text{calc}} - W_{\text{ref}}}{W_{\text{calc}}} \tag{5-15}$$

Table 5-6: Wing weight estimation errors.

Method	Fokker 50 [%]	Fokker 100 [%]
Torenbeek (1)	8.89	-19.70
Torenbeek (2)	6.45	-19.69
Howe	0.59	-4.46
EMWET	-3.30	2.30
Shevell	12.20	-17.53
LTH*	-	-3.77

* *this method is not valid for F50 class aircraft*

From Table 5-6 one can observe that the quasi-analytical method under-predicts the wing weight for the turboprop aircraft compared to the other methods. However for the turbojet aircraft this is the opposite. The quasi-analytical method over-predicts the wing weight, while all the statistical methods under-predict that.

In general, the weight prediction of Fokker 50 is for most methods within 10%. However, the Shevell method predicts the wing mass with an error of 12.2% with respect to actual wing mass. A possible cause of this slightly larger deviation can be explained by the fact that the relation between the total wing weight and the weight index (Eq. (5-6)) was derived using only turbojet aircraft. The mass estimated using EMWET, is slighty different than the one presented in Table 5-4. This difference can be explained by the fact that the student version of EMWET method uses regression to estimate the total weight based on the primary wing weight. In other words, in the student version the secondary wing weight is estimated as a function of the primary wing weight. The Howe method shows a good accuracy with an error of 0.59%, which can be clarified by the fact that this method includes a term that depends on the airplane type.

As the quasi-analytical method showed accurate results for the turboprop aircraft, this is also the case for the turbojet aircraft with an over-prediction of 2.30%. For the same reason as explained before, the Howe method also estimates the weight accurately for the turbojet aircraft. On the other hand, the Torenbeek (1), Torenbeek (2) and Shevell methods show an under-prediction of about 18-20%. Application of these methods for short-haul airplanes could lead to underestimations of the wing weight. As mentioned by Torenbeek [64], reasons for these underestimations can be caused by the extra weight (up to 20%) required to provide adequate stiffness against flutter and weight penalty due to long service life cost.

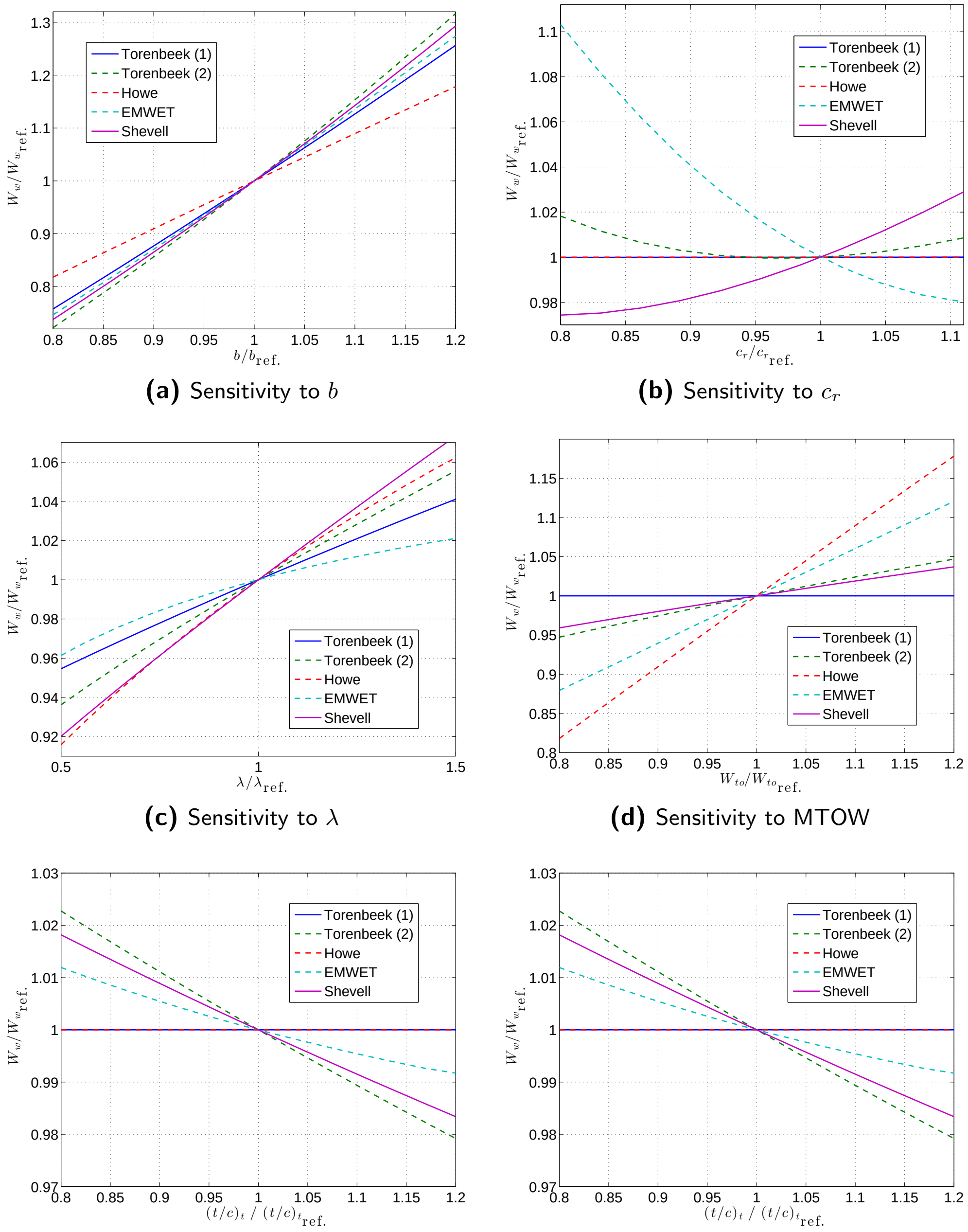
5-2-2 Sensitivity analysis

A study has been performed to investigate the sensitivity of the above mentioned weight estimation methods to the design parameters. Consequently, gaining more insight about their behavior can also serve as a foundation to explain their impact on optimizations. Both the turboprop and the turbojet aircraft are used for sensitivity analysis.

The sensitivity study is performed for the independent wing parameters for which the wing weight shows the greatest changes. The independent parameters are: wing span, root chord length, taper ratio and sweep angle. Wing sweepback is (mainly) applied to increase the critical Mach and drag divergence Mach numbers of the wing at transonic condition [31]. By increasing the critical Mach number, the airplane can reach higher speeds. The considered turbojet aircraft flies at transonic speeds. However, this is not the case for the turboprop aircraft. Therefore, there is no sensitivity investigation of the sweep angle applied for the turboprop aircraft.

Figure 5-6 and 5-7 illustrate the sensitivity analysis of the mentioned parameters. From Figure 5-6a and 5-7a it can be concluded that a 10% decrease in the wing span results in a weight reduction of about 10-15% for the turboprop aircraft and about 10-20% for the turbojet aircraft. In Table 5-5 it is indicated that the Howe method is independent of c_r, which can be clearly seen in Figure 5-6b and 5-7c.

Observing the sensitivity graphs for both turboprop and turbojet aircraft, one can find a remarkable different behavior for some design variables. For both test cases, the weight prediction methods have a similar trend for sensitivity with respect to the wing span. The LTH method for the turbojet case shows the largest change in the wing weight by varying the wing span, see Figure 5-7a. For the turboprop case, the Howe method is insensitive to the root chord length, as can be derived from Eq. (5-7). This explains the insensitivity to chord length of the Howe method illustrated in Figure 5-6b. At the first sight the Torenbeek (1) methods looks insensitive to the root chord length, although it is not the case as can be clearly seen for the turbojet aircraft in Figure 5-7c. Analysis of the sensitivity to the wing taper ratio for the turboprop aircraft shows the same trend as the wing span (see Figure 5-6c), in which the wing weight increases by increasing the taper ratio. In Figure 5-7d, the wing weight sensitivity analysis with respect to the wing taper for turbojet aircraft indicates a different trends compared to the turboprop case. To be more specific, the EMWET and LTH method show opposite behavior than the other methods and to what was observed from the turboprop aircraft. The sensitivity to the quarter-chord sweep shows a similar trend for all methods, except the EMWET method is more prone to varying with the sweep angle. The reason for the different behavior of the quasi-analytical method is that the design variables also affect the aerodynamic loads required by the EMWET method. From the sensitivity to tip thickness-to-chord ratio graphs (Figure 5-6f and 5-7h), it can be deduced that the Torenbeek (1) and Howe method are insensitive for $(t/c)_t$. For the turboprop, Figure 5-7g shows that also the Torenbeek (1) and Howe methods are insensitive for $(t/c)_k$. The sensitivity to MTOW is also analyzed for both aircraft, as shown by Figure 5-6d and 5-7e. As was indicated by Table 5-5, the Torenbeek (1) is not sensitive to the MTOW as can be seen in the figures.



(a) Sensitivity to b

(b) Sensitivity to c_r

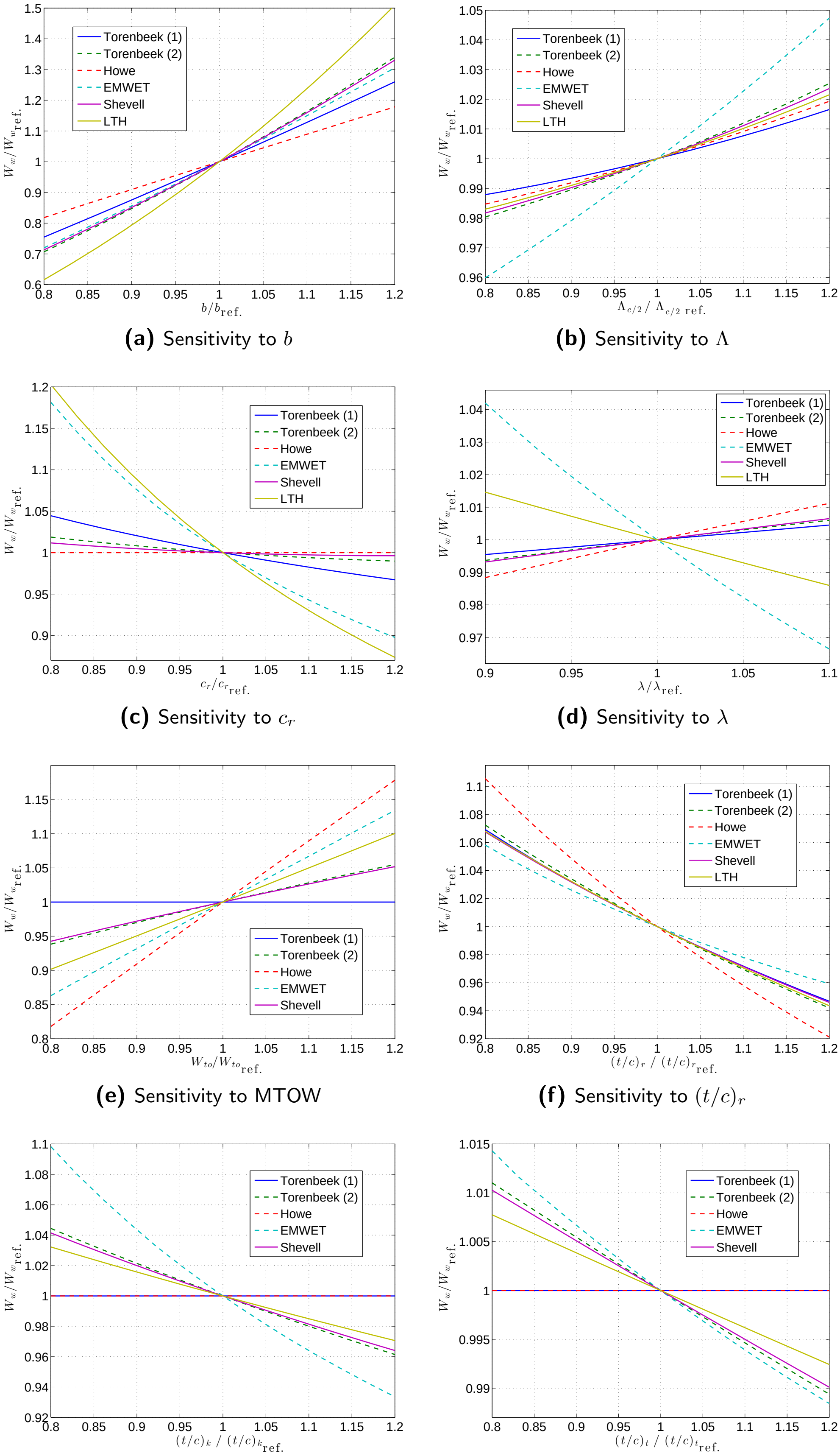
(c) Sensitivity to λ

(d) Sensitivity to MTOW

(e) Sensitivity to $(t/c)_r$

(f) Sensitivity to $(t/c)_t$

Figure 5-6: Sensitivity analysis for turboprop aircraft.



(a) Sensitivity to b

(b) Sensitivity to Λ

(c) Sensitivity to c_r

(d) Sensitivity to λ

(e) Sensitivity to MTOW

(f) Sensitivity to $(t/c)_r$

(g) Sensitivity to $(t/c)_k$

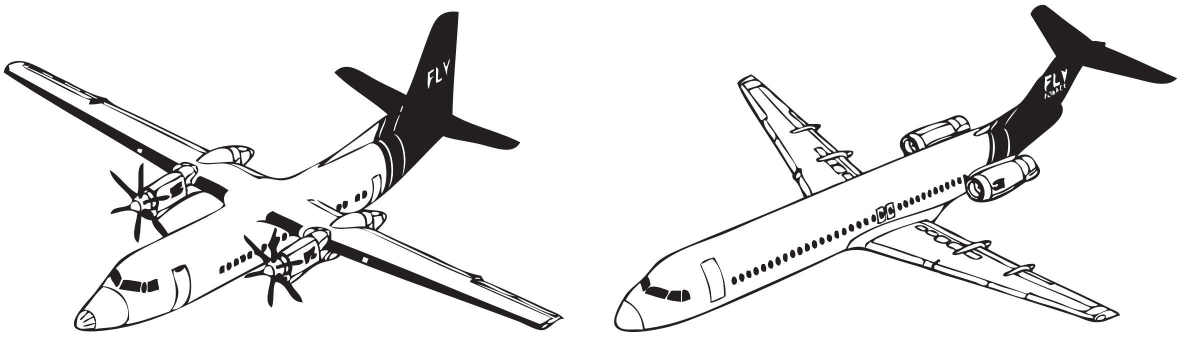
(h) Sensitivity to $(t/c)_t$

Figure 5-7: Sensitivity analysis for turbojet aircraft.

Chapter 6

MDO of aircraft wings

A series of wing MDO assignments have been performed to evaluate the effect of using different weight estimation on aircraft wing design and optimization. The turboprop and turbojet aircraft mentioned in the previous chapter are used as test cases. In Figure 6-1 these test cases are shown. The MDO system consists of three disciplines: aerodynamics, performance and weight. The sequential quadratic programming (SQP) algorithm from Matlab software is used as the optimizer. In addition to that, the LOCSMOOTH algorithm (see Chapter 3) was used in an MDO process to investigate the difference between the global optima and the local optima (using the SQP algorithm).



(a) Fokker 50 **(b)** Fokker 100

Figure 6-1: Turboprop and turbojet aircraft test cases.

First, the objective function for the scope of this research is introduced together with the design vectors (different for each test case) and constraints. Then the MDO strategy selected for this assignment is discussed in Section 6-2. Having discussed the whole MDO procedure, the optimization results are shown and discussed in Section 6-3. In the same section a recommendation was made to implement additional constraints. In Section 6-4, several additional constraints are examined and implemented. Consequently, new optimization were performed on the same test cases with the implemented additional constraints. In Section 6-5 the MDO results are shown which are generated using the additional constraints. Finally, the wing shape was also optimized using the LOCSMOOTH algorithm and the results were compared in Section 6-6.

6-1 Objective function, design vector and constraints

The optimization problem can be written in the following generic mathematical form:

$$\begin{aligned} \min_{x} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & g_i(\boldsymbol{x}) \leq 0 \;\; i = 1, \ldots, m \\ & h_j(\boldsymbol{x}) = 0 \;\; j = 1, \ldots, n \end{aligned}$$

The aircraft maximum take-off weight (MTOW) is considered as the objective function. The design vectors for two different test cases test cases are as follows:

$$\text{turboprop aircraft:} \quad \boldsymbol{x} = [\, b, c_r, \lambda, (t/c)_r\,, (t/c)_t\,, \epsilon_t \,] \tag{6-1}$$

$$\text{turbojet aircraft:} \quad \boldsymbol{x} = [\, b, c_r, \lambda_{\text{in}}, \lambda_{\text{out}}, \Lambda_{\text{in}}, \Lambda_{\text{out}}, (t/c)_r\,, (t/c)_k\,, (t/c)_t\,, \epsilon_t \,] \tag{6-2}$$

Important to note is that the turbojet aircraft has a kinked wing configuration, as illustrated in Figure 5-5c, and therefore two taper ratios and two sweep angles are introduced to size the inner and outer wing. The variable values are allowed to be changed according to engineering expertise (see Tables 6-1 and 6-2) and must be restrained in order to generate a realistic design. The quarter-chord sweep angle is kept constant ($\Lambda_{c/4} = 0°$) for the turboprop case.

The MTOW is considered to be the sum of three terms: the wing weight, the mission fuel weight and the rest weight.

$$W_{\text{to}} = W_{\text{w}} + W_{\text{f}} + W_{\text{rest}} \tag{6-3}$$

The rest weight is initialized based on the reference aircraft actual weight and remains constant throughout the optimization. The wing weight is estimated using different weight estimation methods. The fuel weight is calculated using the aircraft range performance equation (including the aircraft aerodynamic efficiency) and a series of statistical data.

Table 6-1: Design vector for turboprop aircraft.

Design variables	b	c_r	λ	$(t/c)_r$	$(t/c)_t$	ϵ_t
	[m]	[m]	[-]	[-]	[-]	[°]
Initial values	29.00	3.56	0.40	0.21	0.15	-2.0
Lower bounds	20	2.5	0.3	0.25	0.25	-5
Upper bounds	40	5	0.5	0.10	0.10	5

Table 6-2: Design vector for turbojet aircraft.

Design variables	b	c_r	λ_{in}	λ_{out}	Λ_{in}	Λ_{out}	$(t/c)_r$	$(t/c)_k$	$(t/c)_t$	ϵ_t
	[m]	[m]	[-]	[-]	[°]	[°]	[-]	[-]	[-]	[°]
Initial values	28.08	5.60	0.64	0.35	16.24	11.90	0.12	0.12	0.09	-2.00
Lower bounds	20	3.5	0.5	0.3	5	5	0.08	0.08	0.08	-5
Upper bounds	40	6.5	0.8	0.5	22	35	0.15	0.15	0.15	5

Optimizing the aircraft wing for cruise condition may result in a unrealistic value for wing loading because the value of wing loading is affected by some other segments of flight such as take-off and landing (the effect of wing loading on the aircraft stall speed). Hence, a constraint is defined to keep the wing loading smaller or equal to that of the reference aircraft.

$$\left(\frac{W_{\text{to}}}{S}\right) \leq \left(\frac{W_{\text{to}}}{S}\right)_{\text{ref.}} \tag{6-4}$$

The flight conditions at which the turboprop and turbojet aircraft operate, are given in Table 6-3. Based on the flight altitude, the other required parameters (such as air density and viscosity) can be derived.

Table 6-3: Flight conditions of test cases.

Parameter	Turboprop aircraft	Turbojet aircraft
Cruise altitude [m]	7620	10675
Mach number at cruise	0.55	0.77

The following tolerances were defined for all optimizations performed using the `fmincon` function in Matlab:

- Termination tolerance on the normalized objective function value of 1e-4 (default 1e-6).
- Violation tolerance on normalized constraints of 1e-3 (default 1e-6).

6-2 MDO strategy

The Individual Discipline Feasible (IDF) strategy [71] is selected for this MDO assignment. Using this strategy the coupling between different disciplines is removed. In addition, the optimizer is made responsible for the consistency of the coupling variables.

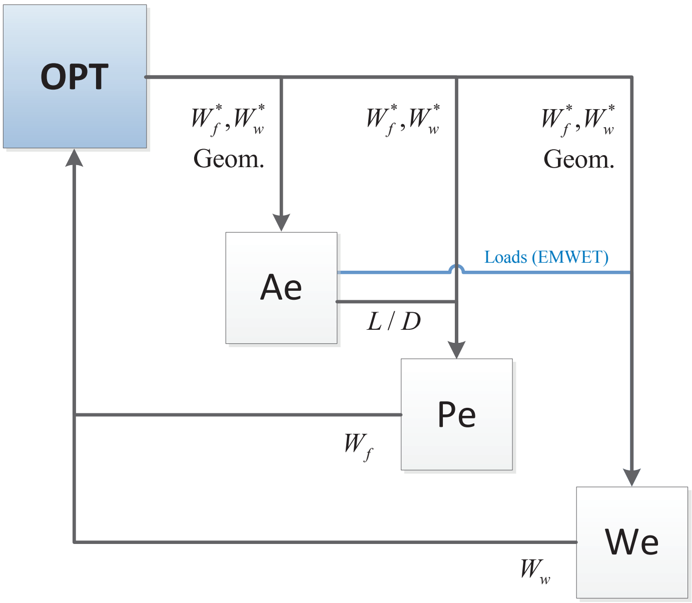


Figure 6-2: Design structure matrix of MDO system with aerodynamics, performance and weight modules.

The design structure matrix of the MDO system is illustrated in Figure 6-2 with its three modules: aerodynamics (Ae), performance (Pe) and weight (We). In this workflow, W_f^* and W_w^* represent the coupling variables (also referred to as the surrogate values). Using IDF there is no feedback between the disciplines and therefore there is no necessity for an iterative procedure to converge the multidisciplinary system. A drawback of using this strategy is that additional design variables are introduced. The optimization problem using IDF can be stated as follows:

$$\begin{aligned} \text{min} \quad & W_{\text{to}}(\boldsymbol{X}) \\ \text{w.r.t} \quad & \boldsymbol{X} = (\boldsymbol{x}, W_{\text{f}}^{*}, W_{\text{w}}^{*}) \\ \text{subject to} \quad & \left(\frac{W_{\text{to}}}{S}\right) \leq \left(\frac{W_{\text{to}}}{S}\right)_{\text{ini}} \\ & W_{\text{f}}^{*} = W_{\text{f}} \\ & W_{\text{w}}^{*} = W_{\text{w}} \end{aligned}$$

As mentioned before, quasi-analytical weight prediction methods require the aerodynamic loads in order to calculate the forces and moments applied to the structure. These aerodynamic loads are calculated by the aerodynamics module. They are then passed to the weight module when the EMWET is used for wing weight estimation. The flow diagram of each module is given in Appendix F.

Aerodynamics module

The strip theory [44] is used to develop an aerodynamic solver. The developed quasi-3D aerodynamic solver, see Chapter 4, is used to accurately calculate the total wing drag. Once the total lift coefficient and the total wing drag coefficient are known, the wing lift-to-drag ratio can be calculated. In order to calculate the total lift to drag ratio of the aircraft, the drag of the other parts of the aircraft (e.g. fuselage) should be added to the drag of the wing. The drag of the aircraft without wing is considered as the "rest drag". The rest drag of the reference aircraft is determined and it is normalized using the wing area and dynamic pressure to obtain the drag coefficient of the aircraft minus the wing. At each objective function evaluation, this coefficient is added to the wing drag coefficient calculated by the aerodynamic solver.

Performance module

The required fuel for the flight mission is calculated using the method presented by Roskam [72]. In this method the required fuel for the cruise is calculated using the Bréguet range equation [1, 73], while some statistical factors are used to estimate the fuel weight of the other segments of the flight mission, see Table 6-4. Each fuel weight fraction M_{ff_i} indicates the ratio of the total aircraft weight at the end of the flight segment divided by the total aircraft weight at the beginning of the segment, see Figure 6-3.

The total fuel weight fraction indicates the consumed fuel as a ratio of the total aircraft weight at the end of the flight mission divided by the fuel weight at the beginning.

$$M_{\text{ff}} = M_{\text{ff}_1} \cdot M_{\text{ff}_2} \cdot \ldots \cdot M_{\text{ff}_n} \tag{6-5}$$

Hence, the fuel weight can be determined including a 5% of the total fuel weight as reserve fuel using the following equation:

$$W_{\text{f}} = 1.05\,(1 - M_{\text{ff}})\,W_{\text{to}} \tag{6-6}$$

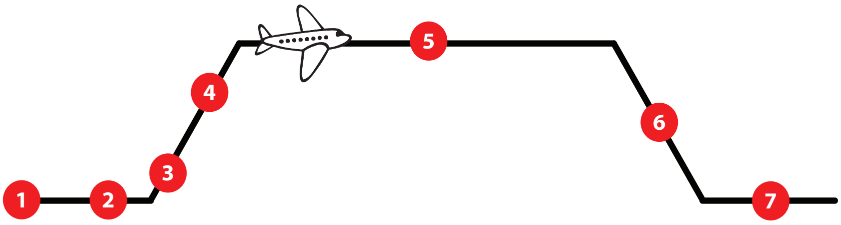


Figure 6-3: Normal flight mission definition with flight segments.

Table 6-4: Fuel fraction for each segment in simple flight mission, suggested values from Roskam [72].

Fuel weight fraction, M_{ff_i}	Turboprop aircraft	Turbojet aircraft
(1) Start & warm-up	0.990	0.990
(2) Taxi	0.995	0.990
(3) Take-off	0.995	0.995
(4) Climb	0.985	0.980
(5) Cruise	*Calculated*	*Calculated*
(6) Descent	0.985	0.990
(7) Landing, taxi & shutdown	0.995	0.992

Weight module

In the weight module, the weight of the wing is predicted using one of the proposed wing weight estimation methods in Chapter 5.

6-3 MDO results

Tables 6-5 and 6-6 provide the results for the optimized wings for the turboprop and turbojet aircraft. Each table also contains the percentage of the reduction in MTOW compared to the reference design. The achieved reduction in MTOW is up to 3.45% for the turboprop aircraft and up to 8.08% for the turbojet aircraft. The planforms of the optimized wings are shown in Figure 6-4.

Table 6-5: Results of wing optimization for turboprop aircraft.

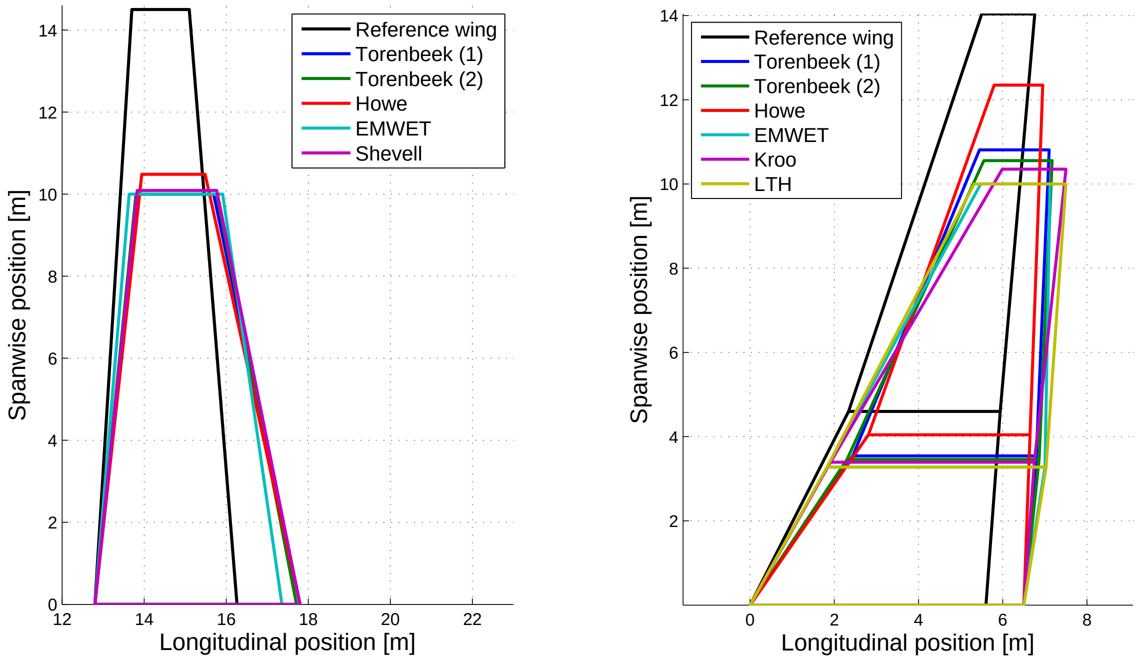
Method	Reduction	b	c_r	λ	$(t/c)_r$	$(t/c)_t$	ϵ_t
	W_{to} [%]	[m]	[m]	[-]	[-]	[-]	[°]
Reference wing	-	29.00	3.46	0.40	0.21	0.15	-2.00
Torenbeek (1)	3.45	20.00	5.00	0.37	0.25	0.19	-1.70
Torenbeek (2)	2.83	20.00	4.91	0.40	0.25	0.25	-0.70
Howe	2.76	20.97	5.00	0.31	0.25	0.15	-0.64
EMWET	2.51	20.00	4.56	0.50	0.15	0.16	-2.45
Shevell	1.93	20.18	5.00	0.39	0.25	0.22	-1.54

Table 6-6: Results of wing optimization for turbojet aircraft.

Method	Reduction	b	c_r	λ_{in}	λ_{out}	Λ_{in}	Λ_{out}	$(t/c)_r$	$(t/c)_k$	$(t/c)_t$	ϵ_t
	W_{to} [%]	[m]	[m]	[-]	[-]	[°]	[°]	[-]	[-]	[-]	[°]
Reference wing	-	28.08	5.60	0.64	0.35	16.24	11.90	0.12	0.12	0.09	-2.00
Torenbeek (1)	4.44	21.62	6.50	0.67	0.38	21.13	13.58	0.14	0.08	0.10	-2.15
Torenbeek (2)	4.41	21.11	6.50	0.70	0.36	21.10	13.78	0.14	0.08	0.10	-2.15
Howe	4.54	24.69	6.50	0.59	0.30	20.09	14.30	0.15	0.08	0.08	-2.51
EMWET	4.84	20.01	6.49	0.80	0.32	19.75	11.58	0.13	0.08	0.08	-2.01
Shevell	4.03	20.71	6.50	0.74	0.31	17.64	15.92	0.12	0.09	0.12	-2.45
LTH	8.08	20.00	6.50	0.80	0.42	19.78	12.33	0.15	0.08	0.08	-1.82

A noteworthy aspect of the optimization results, is that the wing span has been decreased towards its lower bound value. Sensitivity analyses of the weight estimation methods demonstrated that the wing span has a large impact on the wing weight. The objective function is affected by both the wing weight and the fuel weight, where the fuel weight is a function of the aerodynamic efficiency and the aircraft total weight. Increasing the wing span increases the wing aspect ratio (for a constant chord length), which results in a lower induced drag but higher friction drag (as the wetted area increases). Figure 6-5 shows the variation of the wing weight and the fuel weight of the turbojet aircraft with the wing span. It is remarkable that the aerodynamic module shows little changes in aerodynamic efficiency for varying wing span. The figure indicates that the wing weight shows a greater impact on the objective function than the aerodynamic efficiency. This explains the fact that the optimizer drives the wing span towards its lower bound. Other design variables (i.e. wing taper, root chord length, kink thickness-to-chord ratio) are also driven towards lower/upper bounds by the optimizer. In contrast to the statistical weight methods, the quasi-analytical method (EMWET) does not drive the root thickness-to-chord ratio towards its upper bound. Note that the EMWET method, is the only tool used that is sensitive to the wing twist. Hence, the value of the tip twist is purely driven by the aerodynamic aspects in case of using one of the statistical weight estimation methods.

As the optimization results show unrealistic configurations, additional constraints are required to model a more realistic situation.



(a) Turboprop aircraft

(b) Turbojet aircraft

Figure 6-4: Planform geometry of the optimized wings.

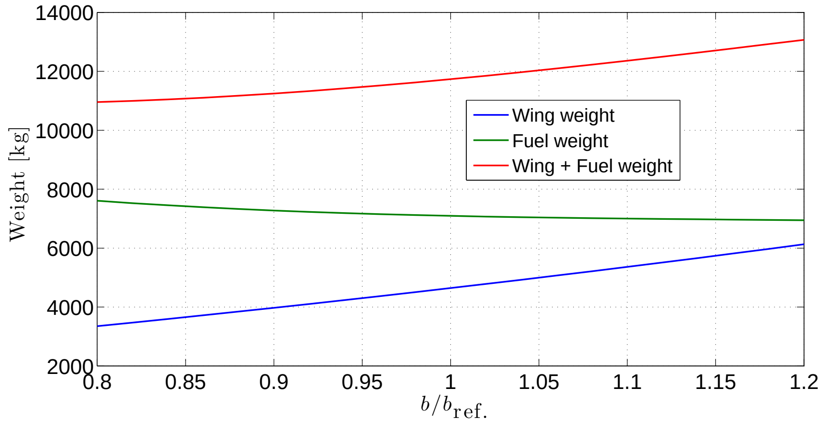


Figure 6-5: Sensitivity of the wing weight (calculated using EMWET) and the fuel weight with respect to the wing span.

6-4 Additional constraints

In order to obtain more realistic wing geometry, two additional constraints are applied. A fuel volume constraint is used to ensure that the required fuel for the flight mission can be stored in the wing fuel tanks (available fuel volume). The fuel volume is calculated using the mission fuel weight and the Jet-A fuel density (according to Air BP [74]).

$$(V_f)_{\text{req.}} \leq (V_f)_{\text{av.}} \tag{6-7}$$

Another constraint is applied to keep aspect ratio $\AE$ larger than typical values for turboprop and turbojet aircraft categories [75, 76]. This constraint is implemented in order to take the performance requirement for the second climb segment into account.

$$
Æ \geq \begin{cases} 10 & \text{for turboprop aircraft} \quad (6\text{-}8) \\ 8 & \text{for turbojet aircraft} \quad (6\text{-}9) \end{cases}
$$

With these additional constraints, the optimization problem can be rewritten:

$$
\begin{aligned}
\text{min} \quad & W_{\text{to}}(\boldsymbol{X}) \\
\text{w.r.t} \quad & \boldsymbol{X} = (\boldsymbol{x}, W_{\text{f}}^{*}, W_{\text{w}}^{*}) \\
\text{subject to} \quad & \left(\frac{W_{\text{to}}}{S}\right) \leq \left(\frac{W_{\text{to}}}{S}\right)_{\text{ref.}} \\
& Æ \geq \begin{cases} 10 & \text{for turboprop} \\ 8 & \text{for turbojet} \end{cases} \\
& (V_f)_{\text{req.}} \leq (V_f)_{\text{av.}} \\
& W_{\text{f}}^{*} = W_{\text{f}} \\
& W_{\text{w}}^{*} = W_{\text{w}}
\end{aligned}
$$

6-5 MDO results using additional constraints

Tables 6-7 and 6-8 provide the MDO results for the turboprop and turbojet aircraft. Both cases show smaller reductions in MTOW using three constraints instead of one.

Table 6-7: Results of wing optimization for turboprop aircraft using new set of constraints.

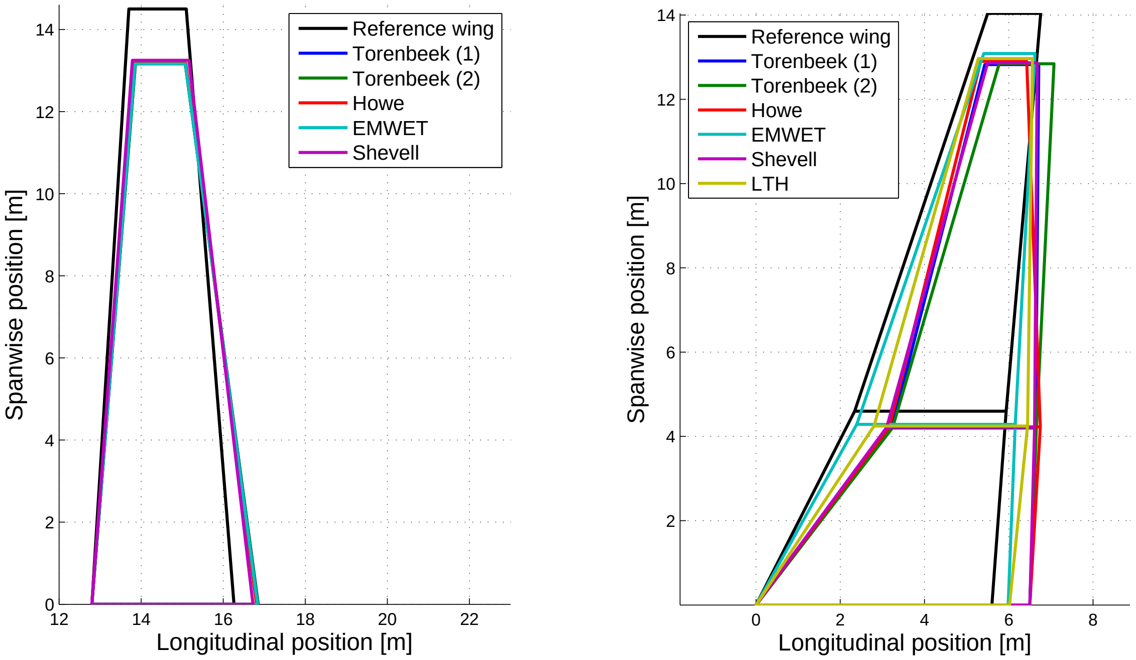
Method	Reduction W_{to} [%]	b [m]	c_r [m]	λ [-]	$(t/c)_r$ [-]	$(t/c)_t$ [-]	ϵ_t [°]
Reference wing	-	29.00	3.46	0.40	0.21	0.15	-2.00
Torenbeek (1)	1.97	26.35	4.05	0.30	0.25	0.10	-2.18
Torenbeek (2)	2.14	26.34	4.05	0.30	0.25	0.25	-1.36
Howe	1.76	26.36	4.01	0.31	0.25	0.17	-2.41
EMWET	1.38	26.33	4.04	0.30	0.22	0.19	-2.07
Shevell	1.61	26.51	3.93	0.35	0.25	0.25	1.11

The optimizations convergence history is shown in Figure 6-7. Computational times of the MDO processes are shown in Table 6-9. All optimizations were performed using Matlab R2010b on a computer having an Intel Core2Duo E4400 (2.00Ghz), with 2Gb RAM and Windows 7 (64-bit version). For the turboprop aircraft, the optimizer requires most iterations for the Torenbeek (1) and Howe methods. Whereas, for the turbojet case most iterations are required for EMWET and Howe method.

Each function evaluation using a statistical method takes about 39 seconds, whereas the EMWET method takes 51 seconds. The additional time required for the quasi-analytical method is due to the aerodynamic load calculation (see Figure 6-2).

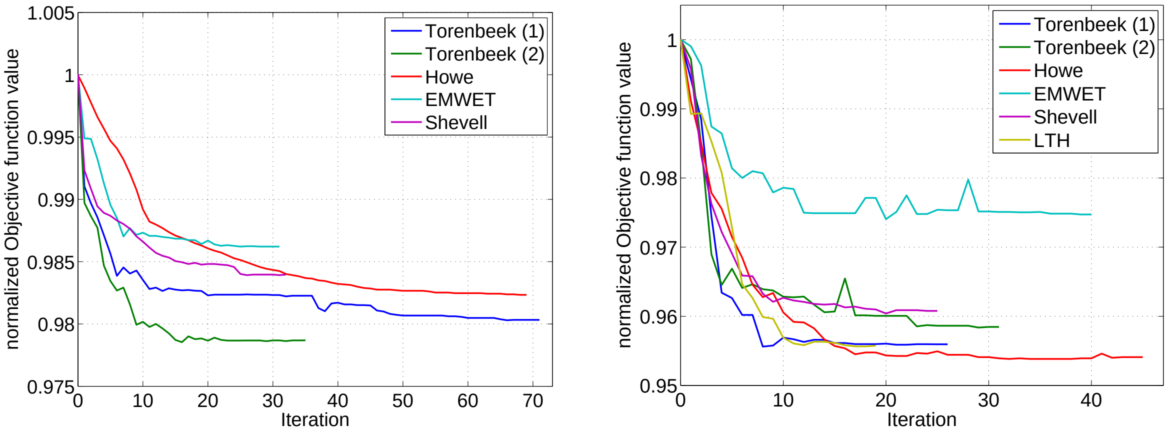
Table 6-8: Results of wing optimization for turbojet aircraft using new set of constraints.

Method	Reduction	b	c_r	λ_{in}	λ_{out}	Λ_{in}	Λ_{out}	$(t/c)_r$	$(t/c)_k$	$(t/c)_t$	ϵ_t
	W_{to} [%]	[m]	[m]	[-]	[-]	[°]	[°]	[-]	[-]	[-]	[°]
Reference wing	-	28.08	5.60	0.64	0.35	16.24	11.90	0.12	0.12	0.09	-2.00
Torenbeek (1)	4.40	25.65	6.50	0.53	0.38	21.82	13.21	0.14	0.08	0.11	-2.13
Torenbeek (2)	4.17	25.68	6.50	0.53	0.38	22.00	15.39	0.15	0.09	0.10	-2.05
Howe	4.60	25.80	6.50	0.55	0.31	22.00	11.40	0.15	0.08	0.10	-2.07
EMWET	2.53	26.17	5.99	0.63	0.32	16.62	11.52	0.13	0.09	0.11	-2.62
Shevell	3.91	25.73	6.50	0.54	0.34	20.87	12.97	0.15	0.08	0.12	-2.55
LTH	4.43	25.93	6.02	0.61	0.35	20.82	11.39	0.15	0.08	0.10	-1.89



(a) Turboprop aircraft **(b)** Turbojet aircraft

Figure 6-6: Planform geometry of the optimized wings using new set of constraints.



(a) Turboprop aircraft **(b)** Turbojet aircraft

Figure 6-7: Optimization convergency.

Table 6-9: MDO computational times using the new constraint set.

Method	Turboprop [hrs]	Turbojet [hrs]
Torenbeek (1)	7.61	6.82
Torenbeek (2)	3.88	7.18
Howe	10.32	10.43
EMWET	5.21	14.48
Shevell	8.28	4.85
LTH	/	4.06

6-6 MDO results using the LOCSMOOTH algorithm

In this section, the optimized turbojet aircraft wing configurations using the SQP and LOCSMOOTH algorithms are compared. Table 6-10 provides the MDO results using the wing loading constraint for both optimization algorithms. The MDO results using the additional constraints are shown in Table 6-11. The optimization results in this section were all generated using the EMWET wing weight estimation method. From these tables, it can be deduced that the LOCSMOOTH algorithm found higher reductions than the SQP algorithm. For the 1 constraint case (the wing loading constraint) the SQP algorithm yields a reduction in MTOW of 4.84% and the LOCSMOOTH algorithm a reduction of 5.08%. The difference 0.24% between these two reduction results in a difference of 103 kg. Whereas for the 3 constraints case (wing loading constraint plus additional constrains), this difference is 0.42% resulting in a difference of 173 kg.

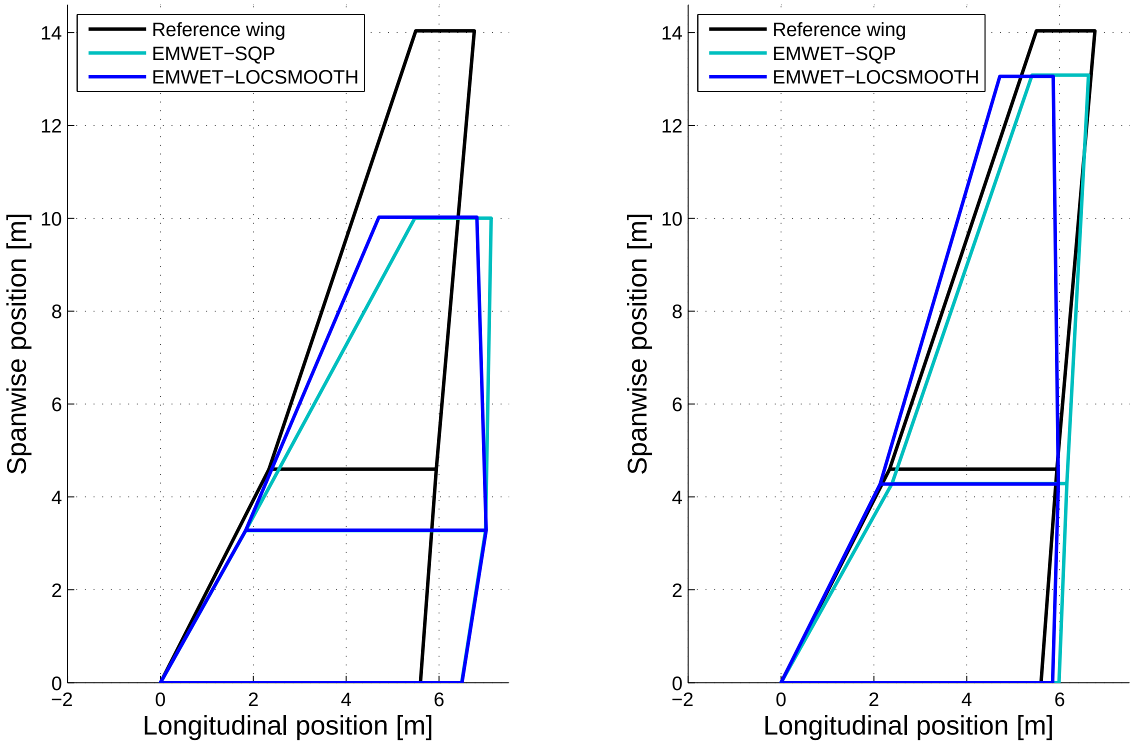
Table 6-10: Comparison MDO results of turbojet aircraft with 1 constraint using SQP and LOCSMOOTH.

Method	Reduction W_{to} [%]	b [m]	c_r [m]	λ_{in} [-]	λ_{out} [-]	Λ_{in} [°]	Λ_{out} [°]	$(t/c)_r$ [-]	$(t/c)_k$ [-]	$(t/c)_t$ [-]	ϵ_t [°]
Reference wing	-	28.08	5.60	0.64	0.35	16.24	11.90	0.12	0.12	0.09	-2.00
EMWET:											
– SQP	4.84	20.01	6.49	0.80	0.32	19.75	11.58	0.13	0.08	0.08	-2.01
– LOCSMOOTH	5.08	20.05	6.49	0.80	0.41	19.79	7.02	0.13	0.08	0.13	-2.61

Table 6-11: Comparison MDO results of turbojet aircraft with 3 constraints using SQP and LOCSMOOTH.

Method	Reduction W_{to} [%]	b [m]	c_r [m]	λ_{in} [-]	λ_{out} [-]	Λ_{in} [°]	Λ_{out} [°]	$(t/c)_r$ [-]	$(t/c)_k$ [-]	$(t/c)_t$ [-]	ϵ_t [°]
Reference wing	-	28.08	5.60	0.64	0.35	16.24	11.90	0.12	0.12	0.09	-2.00
EMWET:											
– SQP	2.53	26.17	5.99	0.63	0.32	16.62	11.52	0.13	0.09	0.11	-2.62
– LOCSMOOTH	2.95	26.11	5.85	0.66	0.30	14.79	6.68	0.14	0.09	0.10	-4.59

Figure 6-8 shows the resulting wing planforms for both algorithms using both constraint cases. It can be deduced that the LOCSMOOTH algorithm was able to find a better reduction (higher reduction in MTOW) compared to the SQP algorithm. However, these further reductions come at a cost: the computational time increases significantly, as indicated by Table 6-12.



(a) Using 1 constraint

(b) Using 3 constraints

Figure 6-8: Difference MDO results using SQP and LOCSMOOTH.

Table 6-12: MDO Computational times with EMWET using SQP and LOCSMOOTH.

EMWET	1 constraint	3 constraints
	[hrs]	[hrs]
SQP	10.02	14.48
LOCSMOOTH	179.80	330.92

Chapter 7

Conclusions & recommendations

Based on the research and developments presented in the preceding chapters, different conclusions are drawn. Additionally, several recommendations can be given for further developments of these tools and further research on these topics.

7-1 Conclusions

The conclusions presented are based on the objectives stated in Chapter 1. The main objective of this thesis research was to

> *Investigate the effect of using different weight estimation methods on the outcome in a wing design task using multidisciplinary design optimization techniques.*

which is accompanied by the following sub-goals:

- Develop a quasi-three-dimensional aerodynamic solver to calculate the wing aerodynamic characteristics.
- Compare the different weight estimation methods, with low and medium levels of fidelity, by analyzing their accuracy and sensitivity to wing parameters.
- Implement a global optimization algorithm that uses gradient-based techniques and local optima smoothing. The results using this algorithm are also compared to a local solution.

First, the conclusions drawn from the sub-goals are addressed. When all the sub-goals are discussed, the main objective can be elaborated as well.

Quasi-3D aerodynamic solver A strip method was combined with a vortex lattice method and the simple sweep theory to develop an aerodynamic solver. The developed solver combines combines the results of a three-dimensional vortex lattice method with the results of a viscous two-dimensional airfoil analyzer to accurately calculate the total wing drag. The solver was validated against experimental data and data from other CFD tools. From these validations, it was shown that the quasi-three-dimensional shows good agreements with the experimental data and other aerodynamic solvers. Except, at higher speeds (especially at free-stream Mach numbers higher than 0.75) the developed solver is not able to simulate 3D aerodynamic effects (such as root and tip effect). This difference of these effects was shown in Subsection 4-8-2 and is due to the use of the simple sweep theory.

Weight estimation methods Both a quasi-analytical and statistical wing weight predictions methods were compared to each other by analyzing their accuracy and sensitivity to different wing parameters. Most methods show results close to the reference weight for the turboprop aircraft class. For the turbojet aircraft class, on the other hand, most statistical methods under-predict the wing weight. This under-prediction of about 18-20% can be explained by the fact that short-haul airplanes require extra weight (up to 20%) to provide appropriate stiffness against flutter and a weight penalty due to long service lift cost (according to Torenbeek [30]). Furthermore, it was shown that the statistical methods are insensitive to some design variables.

Local smoothing for global optimization The "local optima smoothing for global optimization" algorithm was capable of finding the theoretical optimum for several benchmark test functions. A classical aerodynamic problem was also tested, at which the global optimizer was able to find the global optimum. It can be concluded that this hybrid algorithm, an appropriate mixture of local approximation and global exploration, is efficient and robust in solving global optimization problems. Another important is that the input settings (sphere radii and number of sampled points) for this algorithm are problem dependent. For example, very noisy functions mostly require smaller sphere radii.

Different wing weight estimation methods were implemented in a wing design task using MDO techniques. Two test cases, a turboprop and a turbofan passenger aircraft, were used. Comparison of the implemented weight prediction methods, based on their accuracy and sensitivity to wing shape parameters, gave better insight into each method. MDO results for both test cases were shown using one constraint (on wing loading), yielding optimized configurations where some design variables were driven towards their lower/upper bounds. These results suggest that implementing additional constraints to the optimization problem will yield a more realistic optimization. Doing so, two more constraints concerning the aspect ratio and fuel volume were implemented. The MDO results using three constraints, showed that the problem from the one constraint case was partially solved. Realistic optimization problems contain significantly more constraints. However, the number of constraints used was sufficient for the scope of this research.

When a sufficient geometry is available in early design stages, it is of great interest to use methods with higher design-sensitivity. This allows engineers to gain more knowledge about the design and the design-sensitivity. The statistical methods that were used for the wing weight calculations, were insensitive to some design variables (tip twist, tip thickness-to-chord ratio and root chord). EMWET, the quasi-analytical method, is sensitive to all used variables. However, using EMWET (quasi-analytical weight predictions) comes at the cost of increasing the computational time for each function evaluation by 30%.

7-2 Recommendations

Based on the work performed, several recommendations are given for further research and further improvements that can be made.

Exploit parallel computing On a single computer, a multidisciplinary design optimization of an aircraft wing takes several of hours. Especially, when using the global optimization algorithm (LOCSMOOTH), this process can take up to two weeks. It is expected that the runtime for these optimizations can be significantly reduced by using parallel computing capabilities.

To run the optimization process in parallel, several modification in the program are required. Parallel computing can be applied in the quasi-3D aerodynamic solver, where the viscous airfoil drag calculations can be done parallel.

Quasi-3D aerodynamic solver The quasi-3D aerodynamic solver could be further developed, such that it can calculate more aerodynamic characteristics. Useful additions might be: calculations of maximum lift coefficient, stall characteristics, etc. Furthermore, the viscous airfoil calculation that were used are capable of calculating the aerodynamics of wing sections with high lift devices. Therefore, this tool can be further extended to analyze wings with high lift devices.

Multi-stage MDO Multi-stage frameworks can be used to obtain high-fidelity design results. An application of a multi-stage framework for this research might be to first optimize the wing planform (first stage) and consequently to optimize the airfoil shapes (second stage). While optimizing one stage, the design variables of the other stage are “frozen”.

More realistic optimizations For the scope of this research, a small amount of constraints were used. For more realistic optimization, many more constraints should be implemented. For example, additional constraints on performance, stability, control, etc.

Bibliography

[1] Anderson, J. D., *Aircraft Performance and Design*, WCB/McGraw-Hill, 1999.

[2] van Tooren, M., Steenhuizen, D., van Gerwen, D., La Rocca, G., and Schroijen, M., *Advanced Design Methodologies (TU Delft – Course notes, Chapter 2)*, Delft University of Technology, 2009.

[3] Sobieszczanski-Sobieski, J. and Haftka, R. T., "Multidisciplinary aerospace design optimization: survey of recent developments," *Structural and Multidisciplinary Optimization*, Vol. 14, 1997, pp. 1–23.

[4] Alexandrov, N. M. and Hussaini, M. Y., editors, *Multidisciplinary Design Optimization: State of the Art*, SIAM, 1997.

[5] American Institute of Aeronautics and Astronautics, *White Paper on Industrial Experience with MDO*, AIAA Technical Committee on Multidisiplinary Design Optimization (MDO), 1999.

[6] Martins, J. R. R. A. and Marriage, C. J., "An Object-Oriented Framework for Multidisciplinary Design Optimization," *3rd AIAA Multidisciplinary Design Optimization Specialist Conference*, Vol. 36, No. 4, 2009, pp. 1–25.

[7] Kroo, I., "Distributed Multidisciplinary Design and Collaborative Optimization," *Lecture series on Optimization Methods and Tools for Multicriteria/Multidisciplinary Design, The von Karman Institute*, November 2004.

[8] Alonso, J. J. and Fike, J., *Introduction to Multidisciplinary Design Optimization (Course material, Chapter 7: MDO Architectures)*, Stanford University, http://adl.stanford.edu/aa222/Home.html, 2010.

[9] Cramer, E. J., Dennis, J. E. J., Frank, P. D., Lewis, R. M., and Shubin, G. R., "Problem Formulation for Multidisciplinary Optimization," *SIAM Journal on Optimization*, Vol. 4, No. 4, 1994, pp. 754–776.

[10] Sobieszczanski-Sobieski, J. and Haftka, R. T., "Multidisciplinary aerospace design optimization: survey of recent developments," *Structural and Multidisciplinary Optimization*, Vol. 14, 1997, pp. 1–23.

[11] Nadarajah, S., *Review of numerical optimization methods, Brachistochrone problem*, McGill University, October 2004.

[12] Keane, A. J. and Nair, P. B., *Computational Approaches for Aerospace Design: The Pursuit of Excellence*, John Wiley & Sons Ltd., 2005.

[13] Whitney, E. J., Gonzalez, L. F., and Periaux, J., *Multidisciplinary Methods for Analysis Optimization and Control of Complex Systems*, Vol. 6 of *Mathematics in Industry*, Springer Berlin Heidelberg, 2005.

[14] Mason, W., *Applied Computational Aerodynamics, Volume 1: Foundations and Classical Pre-CFD Methods*, Department of Aerospace and Ocean Engineering - Virginia Polytechnic Institute and State University, http://www.dept.aoe.vt.edu/~mason/Mason_f/CAtxtTop.html, 1992–1995.

[15] Raymer, D. P., *Aircraft Design: a Conceptual Approach*, AIAA (American Institute of Aeronautics and Astronautics) Education Series, Reston, VA (USA), 4th ed., 2006.

[16] Mason, W., "On the Use of the Potential Flow Model for Aerodynamic Design at Transonic Speeds," *AIAA paper 95-0741*, 33rd Aerospace Sciences Meeting & Exhibit, January 1995, pp. 1–6.

[17] Jameson, A., *Encyclopedia of Computational Mechanics (Chapter 11:Aerodynamics)*, John Wiley & Sons Ltd., 2004.

[18] Kroo, I., *Aerodynamic Modeling for Simulation and Control (Course material, Chapter 3: Approaches to Aerodynamic Modeling)*, Stanford University, http://adg.stanford.edu/aa208/amsc.html, 2002.

[19] Bakker, A., *Computational Fluid Dynamics (Course material, Lecture 5: Finite volume solvers)*, Dartmout College, http://www.bakker.org/dartmouth06/engs150/05-solv.pdf, 2006.

[20] Kuzmin, D., *Introduction to Computational Fluid Dynamics (Course material, Lecture 3: Discretization techniques)*, Institute of Applied Mathematics – University of Dortmund, http://www.mathematik.uni-dortmund.de/~kuzmin/cfdintro/lecture3.pdf, 2010.

[21] Liu, G. and Xu, G. X., "A gradient smoothing method (GSM) for fluid dynamics problems," *International Journal for Numerical Methods in Fluids*, Vol. 58, No. 10, March 2008, pp. 1101–1133.

[22] Blazek, J., *Computational Fluid Dynamics: Principles and Applications*, Elsevier Science Ltd., Oxford (UK), 2001.

[23] Versteeg, H. and Malalasekera, W., *An introduction to computational fluid dynamics: The finite volume method*, Longman Scientific & Technical, Essex (England), 1st ed., 1995.

[24] Prandtl, L., "Applications of modern hydrodynamics to aeronautics," Tech. rep., NACA Rep. 116 (National Advisory Committee for Aeronautics), 1923.

[25] Anderson, J. D., *Fundamentals of Aerodynamics*, McGraw Hill, 4th ed., 2007.

[26] Moran, J., *An Introduction to Theoretical and Computational Aerodynamics*, Aeronautical Engineering Series, Dover Publications, 1984.

[27] Katz, J. and Plotkin, A., *Low-Speed Aerodynamics*, Cambridge Aerospace Series, Cambridge University Press, 2001.

[28] Tomac, M., *Adaptive-fidelity CFD for Predicting Flying Qualities in Preliminary Aircraft Design*, Master's thesis, KTH School of Engineering Sciences, Stockholm (Sweden), January 2011.

[29] Howe, D., *Aircraft Conceptual Design Synthesis*, Professional Engineering Publishing, London and Bury St Edmunds (UK), 2000.

[30] Torenbeek, E., *Synthesis of Subsonic Airplane Design*, Delft University Press, Delft (Netherlands), 1976.

[31] Shevell, R. S., *Fundamentals of Flight*, Prentice-Hall, Michigan, US, 1983.

[32] Torenbeek, E., "Development and Application of a Comprehensive, Design-sensitive Weight Prediction Method for Wing Structures of Transport Category Aircraft," Report LR-693, Delft University of Technology, Delft (Netherlands), September 1992.

[33] Macci, S. H., *Semi-Analytical Method for Predicting Wing Structural Mass*, 54th Annual Conference, Society of Allied Weight Engineers, Inc., SAWE Paper No. 2282, May 1995.

[34] Elham, A., van Tooren, M. J. L., and La Rocca, G., *An Advanced Quasi-Analytical Weight Estimation Method for Airplane Lifting Surfaces*, 71st International Conference on Mass Properties, Society of Allied Weight Engineers, Inc., SAWE Paper No. 3571, May 2012.

[35] Bindolino, G., Ghiringhelli, G., Ricci, S., and Terraneo, M., "Multilevel Structural Optimization for Preliminary Wing-Box Weight Estimation," *Journal of Aircraft*, Vol. 47, No. 2, March-April 2010.

[36] Laban, M., Arendsen, P., Rouwhorst, W., and Vankan, W., "Multidisciplinary Design Optimization for a Blended Wing Body Transport Aircraft with Distributed Propulsion," *AIAA paper 2001-5446*, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, September 2002, pp. 1–11.

[37] Buttazzo, G., Frediani, A., Rizzo, E., and Frediani, A., “Application of Optimisation Algorithms to Aircraft Aerodynamics,” *Variational Analysis and Aerospace Engineering*, Vol. 33 of *Springer Optimization and Its Applications*, Springer New York, 2009, pp. 419–446.

[38] Addis, B., Locatelli, M., and Schoen, F., “Local optima smoothing for global optimization,” *Optimization Methods and Software*, Vol. 20, No. 4-5, August 2005, pp. 417–437.

[39] Addis, B. and Leyffer, S., “A Trust-Region Algorithm for Global Optimization,” *Computational Optimization and Applications*, Vol. 35, No. 3, June 2006, pp. 287–304.

[40] Rizzo, E., *Optimization Methods Applied to the preliminary design of innovative non conventional aircraft configurations*, Ph.D. thesis, Pisa University, June 2009.

[41] Yang, X. S., *Engineering Optimization: An Introduction with Metaheuristic Applications*, Wiley, 2010.

[42] Sarker, R., Mohammadian, M., and Yao, X., *Evolutionary Optimization*, International Series in Operations Research & Management Science, Kluwer Academic Publishers, 2002.

[43] Drela, M. and Youngren, H., *AVL (Athena Vortex Lattice)*, Massachusetts Institute of Technology, http://web.mit.edu/drela/Public/web/avl/, Last accessed: May 2011.

[44] Cosyn, P. and Vierendeels, J., “Numerical Investigation of Low-Aspect-Ratio Wings at Low Reynolds Numbers,” *Journal of Aircraft*, Vol. 43, No. 3, 2006, pp. 713–722.

[45] Bramwell, A., *Helicopter Dynamics*, John Wiley & Sons, New York (US), 1976.

[46] Gessow, A. and Myers, G., *Aerodynamics of the Helicopter*, the Macmillan Company, New York (US), 1952.

[47] C.J.Sequira, D.J.Willis, and J.Peraire, “Comparing aerodynamic models for numerical simulation of dynamics and control of aircraft,” *AIAA paper 2006-1254*, 44th AIAA Aerospace Sciences Meething, 2006, pp. 1–20.

[48] Holt, D. R., “Introduction to transonic aerodynamics of aerofoils and wings,” Tech. rep., ESDU (Engineering Sciences Data Unit) 90008, April 1990.

[49] Drela, M., “N+3 Aircraft Concept Designs and Trade Studies – Volume 2 (Appendices): Design Methodologies for Aerodynamics, Structures, Weight, and Thermodynamic Cycles (Appendix A: TASOPT – Transport Aircraft System OPTimization),” Final Report NASA/CR-2010-216794/VOL2, NASA Glenn Research Center, December 2010.

[50] Desktop Aeronautics, Inc., *Oblique Flying Wings: An Introduction and White Paper*, http://www.desktop.aero/library/ofwwhitepaper.pdf, June 2005.

[51] Obert, E., Slingerland, R., Leusink, D., van den Berg, T., Koning, J., and van Tooren, J., *Aerodynamic Design of Transport Aircraft*, IOS Press, Amsterdam (The Netherlands), 2009.

[52] Kroo, I., “DRAG DUE TO LIFT: Concepts for Prediction and Reduction,” *Annual Review of Fluid Mechanics*, Vol. 33, No. 1, 2001, pp. 587–617.

[53] Selig, M. S., Lyon, C. A., and Giguere, P., *Summary of Low-Speed Airfoil Data*, Vol. 2, SoarTech Publications, Virgnia Beach, VA (US), 1995.

[54] Abbott, I. and Von Doenhoff, A., *Theory of Wing Sections: Including a Summary of Airfoil Data*, Dover Books on Physics and Chemistry, Dover Publications, 1959.

[55] Drela, M. and Youngren, H., *XFOIL: Subsonic Airfoil Development System*, Massachusetts Institute of Technology, http://web.mit.edu/drela/Public/web/xfoil/, Last accessed: May 2011.

[56] Drela, M., “XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils,” *Conference on Low Reynolds Number Airfoil Aerodynamics*, University of Notre Dame, 1989, pp. 1–12.

[57] The Royal Aeronautical Society, “VGK method for two-dimensional aerofoil sections (Part 1: principles and results),” Tech. rep., ESDU (Engineering Sciences Data Unit) 96028, October 1996.

[58] Drela, M., *MSES: Multi-Element Airfoil Design/Analysis Software*, Massachusetts Institute of Technology, http://raphael.mit.edu/drela/msessum.ps, Last accessed: May 2011.

[59] Anderson, R. F., "Determination of the characteristics of tapered wings," Tech. Rep. 572, National Advisory Commitee for Aeronautics, 1940.

[60] Maskew, B., *PROGRAM VSAERO: A computer program for calculating the non-linear aerodynamic characteristics of arbitrary configurations: User's manual*, NASA contractor report: CR-166476, NASA, December 1982.

[61] Nathman, J. K., *VSAERO: A Computer Program for Calculating the Nonlinear Aerodynamic Characteristics of Arbitrary Configurations – Users' Manual version 7.2*, ANALYTICAL METHODS, INC., Washington (US), September 2007.

[62] van der Wees, A., van Muijden, J., and van der Vooren, J., *A Fast and Robust Viscous-Inviscid Interaction Solver for Transonic Flow about Wing/Body Configurations on the Basis of Full Potential Theory*, Technical Publication 93214 U, NLR – National Aerospace Laboratory of the Netherlands, 1993.

[63] Koning, J. H., *Development of a KBE application to support aerodynamic design and analysis: Towards a Next-Generation Multi-Model Generator*, Master's thesis, Delft University of Technology, Delft (The Netherlands), May 2010.

[64] Torenbeek, E., "Prediction of wing group weight for preliminary design," *Aircraft Engineering and Aerospace Technology*, Vol. 43, No. 7, July 1971, pp. 16–21.

[65] Torenbeek, E., "Optimum Wing Area, Aspect Ratio and Cruise Altitude for Long Range Transport Aircraft," Report LR-775, Delft University of Technology, Delft (Netherlands), October 1994.

[66] Wright, J. and Cooper, J., *Introduction to Aircraft Aeroelasticity and Loads*, Vol. 18 of Aerospace Series of *AIAA education series*, John Wiley, 2008.

[67] Bisplinghoff, R., Ashley, H., and Halfman, R., *Aeroelasticity*, Dover books on physics, Courier Dover Publications, 1996.

[68] Kroo, I. and Shevell, R., *Aircraft design: Synthesis and analysis*, Digital Textbook Version 1.2, Desktop Aeronautics, Inc., Stanford, CA (US), http://adg.stanford.edu/aa241/AircraftDesign.html, September 2006.

[69] Howe, D., "The prediction of aircraft wing mass," *Proc. Instn Mech. Engrs, Part G, Journal of Aerospace Engineering*, 1996.

[70] Dorbath, F., "Large Civil Jet Transport (MTOM $>$ 40t) Statistical Mass Estimation," DLR-LY/Airbus, LTH MA 401 12-01, 2011.

[71] Martins, J. R. and Marriage, C., "An Object-Oriented Framework for Multidisciplinary Design Optimization," *AIAA Paper 2007-1906*, 3rd AIAA Multidisciplinary Design Optimization Specialist Conference, April 2007, pp. 1–15.

[72] Roskam, J., *Airplane Design: Component Weight Estimation (Part V)*, Airplane Design, Design, Analysis and Research Corporation (DARcorporation), Lawrence, KS (USA), 1999.

[73] Ruijrok, G., *Elements of Airplane Performance*, Delft University Press, 2009.

[74] Air BP Ltd., *Handbook of products*, http://www.bp.com/liveassets/bp_internet/aviation/air_bp/STAGING/local_assets/downloads_pdfs/a/air_bp_products_handbook_04004_1.pdf, Last accessed: June 2012.

[75] Jenkinson, L., Rhodes, D., and Simpkin, P., *Civil jet aircraft design*, AIAA education series, American Institute of Aeronautics and Astronautics, 1999.

[76] Jenkinson, L., Rhodes, D., and Simpkin, P., "Civil Jet Aircraft Design – Aircraft Data Set," http://www.elsevierdirect.com/companions/9780340741528/appendices/data-a/default.htm, Last accessed: June 2012.

[77] Bazaraa, S., Sherali, H., and Shetty, C., *Nonlinear Programming: Theory And Algorithms*, Wiley-Interscience, 3rd ed., 2006.

[78] Fletcher, R., *Practical Methods of Optimization: Unconstrained Optimization*, Wiley, 2nd ed., 1987.

Appendix A

SQP algorithm

Sequential (or successive) Quadratics Programming (SQP) is a well-known and popular method for non-linear constrained optimization. In addition to that, it is one of the robust methods. The algorithm exists of two steps in which a quasi-Newton method and a line search are used iteratively to find the best possible solution. The quasi-Newton method is used to locally, around the current location, approximate the real problem with a quadratic function. The line connecting the solution to this quadratic problem and the current location, is then used to determine the new-current location, and is referred to as the line search.

Quadratic programming subproblem

The fundamental idea of sequential quadratic programming is to approximate the computationally extensive full Hessian matrix using a quasi-Newton updating method. Subsequently, this generates a subproblem of quadratic programming (a so-called QP subproblem) at each iteration. The solution to this subproblem can be used to determine the search direction and next trial solution. Using the Taylor expansion, the above problem can be approximated at each iteration, as the following problem:

$$\min_{x} \quad \frac{1}{2}\mathbf{s}^T \nabla^2 \mathcal{L}\left(\boldsymbol{x}_k\right)\mathbf{s} + \nabla f\left(\boldsymbol{x}_k\right)^T \mathbf{s} + f\left(\boldsymbol{x}_k\right) \tag{A-1}$$

$$\text{subject to} \quad \nabla g_i\left(\boldsymbol{x}_k\right)^T \mathbf{s} + g_i\left(\boldsymbol{x}_k\right) = 0, \qquad i = 1, \ldots, m \tag{A-2}$$

$$\nabla h_j\left(\boldsymbol{x}_k\right)^T \mathbf{s} + h_j\left(\boldsymbol{x}_k\right) \leq 0, \qquad j = 1, \ldots, n \tag{A-3}$$

where the Lagrangian function (also called merit function), is defined by

$$\mathcal{L}\left(\boldsymbol{x}\right) = f\left(\boldsymbol{x}\right) + \sum_{m}^{i=1} \lambda_i g_i\left(\boldsymbol{x}\right) + \sum_{n}^{j=1} \lambda_j h_j\left(\boldsymbol{x}\right) \tag{A-4}$$

$$= f\left(\boldsymbol{x}\right) + \boldsymbol{\lambda}^T \boldsymbol{g}(\boldsymbol{x}) + \boldsymbol{\mu}^T \boldsymbol{h}(\boldsymbol{x}) \tag{A-5}$$

where $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_m)^T$ is the vector of Lagrange multipliers, and $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)^T$ is the vector of KKT multipliers. The terms $\boldsymbol{g}$ and $\boldsymbol{h}$ represent $\boldsymbol{g} = (g_1(\boldsymbol{x}), \ldots, g_m(\boldsymbol{x}))^T$ and $\boldsymbol{h} = (h_1(\boldsymbol{x}), \ldots, h_n(\boldsymbol{x}))^T$.

To approximate the Hessian $\nabla^2 \mathcal{L}\left(\boldsymbol{x}_k\right)$ by a positive definite symmetric matrix $\boldsymbol{H}_k$, the standard Broydon-Fletcher-Goldfarbo-Shanno (BFGS) approximation of the Hessian can be used (see reference [41] for more details), which leads to

$$\boldsymbol{H}_{k+1} = \boldsymbol{H}_k + \frac{\boldsymbol{v}_k \boldsymbol{v}_k^T}{\boldsymbol{v}_k^T \boldsymbol{u}_k} + \frac{\boldsymbol{H}_k \boldsymbol{u}_k \boldsymbol{u}_k^T \boldsymbol{H}_k^T}{\boldsymbol{u}_k^T \boldsymbol{H}_k \boldsymbol{u}_k} \tag{A-6}$$

where

$$\boldsymbol{u}_k = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k \tag{A-7}$$

and

$$\boldsymbol{v}_k = \nabla \mathcal{L}\left(\boldsymbol{x}_{k+1}\right) - \nabla \mathcal{L}\left(\boldsymbol{x}_{ks}\right) \tag{A-8}$$

Line search

The QP subproblem is solved to obtain the search direction

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha \boldsymbol{s}_k \tag{A-9}$$

using a line search method by minimizing a penalty function,

$$\Phi(\boldsymbol{x}) = f(\boldsymbol{x}) + \rho \left[\sum_{i=1}^{m} |g_i(\boldsymbol{x})| + \sum_{j=1}^{n} \max\left\{0, h_j(\boldsymbol{x})\right\} \right] \tag{A-10}$$

where ρ is the penalty parameter. Any SQP method requires a good choice of $\boldsymbol{H}_k$ as the approximate Hessian of the Lagrangian $\mathcal{L}$. Obviously, if $\boldsymbol{H}_k$ is exactly calculated as $\nabla^2 \mathcal{L}$, the SQP essentially becomes Newton's method for solving the optimality condition. A popular way to approximate the Lagrangian Hessian is to use a quasi-Newton scheme as we used the BFGS formula mentioned earlier.

The following psuedo-code summarizes the procedure of the sequential quadratic programming.

```
repeat $k = 1, 2, \ldots$
    Solve a QP subproblem: QP$_k$ to get the search direction $\boldsymbol{s}_k$
    Given $\boldsymbol{s}_k$, find $\alpha$ so as to determine $\boldsymbol{x}_{k+1}$
    Update the approximate Hessian $\boldsymbol{H}_{k+1}$ using BFGS scheme
    $k = k + 1$
until stop criterion
```

The built-in optimization routine of Matlab, `fmincon` uses the SQP algorithm to find a solution for a constrained minimization problem. It can handle both equality and inequality constraints, and bounds imposed on the design variables. The `fmincon` also uses BGFS to update the approximation to the Hessian of the Lagrangian ($\boldsymbol{H}_k$).

Appendix B

Psuedo-code of LOCSMOOTH algorithm

The psuedo-code used to program the LOCSMOOTH algorithm is given on the next page. This code is taken from reference [38, 39].

```
Procedure ALSO($r$, MaxNoImprove, $K$)
  // ALSO: an Algorithm based on Local Smoothing for Optimization
  // see also: http://alanine.dsi.unifi.it/also
  // MaxNoImprove: stopping criterion
  // $K$: number of observations to perform in the current sphere
  // $r$: radius used in local perturbation of the current point
  NoImprove = 0;
  $x$ = random uniform point in $S$;
  $x^* = \mathcal{LS}(x)$;
  current= $f(x^*) = L(x)$;
  record = current;
  while (NoImprove < MaxNoImprove)
    $i = 0$;
    while ($i < K$ and record $\leq$ current)
      $i = i + 1$;
      $y_i$ = random uniform point in $B(x^*, r)$;
      $y_i^* = \mathcal{LS}(y_i)$;
      current = $L(y_i)$;
    end while
    if (current < record)
      // a new record has been found while sampling in $B(x^*, r)$
      record = current
      $x^* = y_i^*$
      NoImprove = 0;
    else
      NoImprove = NoImprove + $K$;
      // Major iteration: optimization of the
      //approximate smoothing based upon the observations placed in $y_1, \ldots, y_K$
      $x = \arg\min_{x \in B(x^*, r)} \hat{L}_g^B(x)$;
      // The current point is moved
      $y = \mathcal{LS}(x)$
      current = $L(x)$;
      if (current < record)
        // a new record has been found
        record = current;
        NoImprove = 0;
        $x^* = y$
      else
        $x^* = x$
      end if
    end if
  end while
end Procedure
```

Appendix C

Taper implementation for simple sweep theory

The calculation of the chord length perpendicular to the quarter-chord sweep line for swept wings without taper, can be done using equation (4-1). For tapered swept wings however, this calculation is slight more complex. Based on Figure C-1, it is explained how the chord length perpendicular to the sweep line, at given constant chord percentage (denoted by ξ_c), can be found.

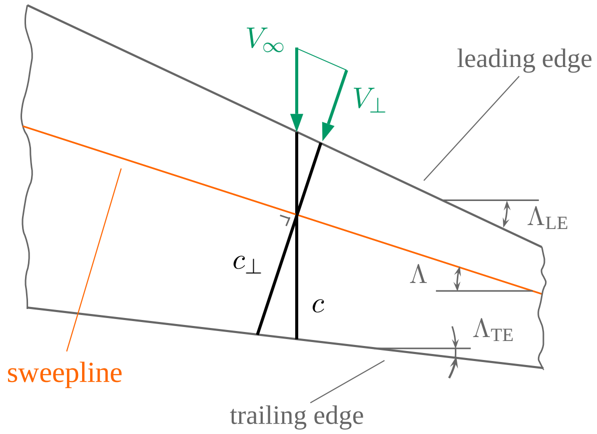


Figure C-1: Airfoil section perpendicular to sweep line of a tapered wing

The chord length of the airfoil section perpendicular to the sweep line is denoted by $c_\perp$. The calculation of $c_\perp$ consists of out two parts, the front part (between leading edge and sweep line) and aft part (between sweep line and trailing edge).

$$c_\perp = c_{\text{front}} + c_{\text{aft}} \tag{C-1}$$

The front part can be derived using Figure C-2 and the law of sines, which yields the following expressions for both the front part and the aft part (which can be calculated using the same procedure as for the front part):

$$c_{\text{front}} = \frac{\sin(90^\circ - \Lambda_{\text{LE}})}{\sin(90^\circ + \Lambda_{\text{LE}} - \Lambda)} c \cdot \xi_c \tag{C-2a}$$

$$c_{\text{aft}} = \frac{\sin(90^\circ - \Lambda_{\text{TE}})}{\sin(90^\circ + \Lambda_{\text{TE}} - \Lambda)} c \cdot (1 - \xi_c) \tag{C-2b}$$

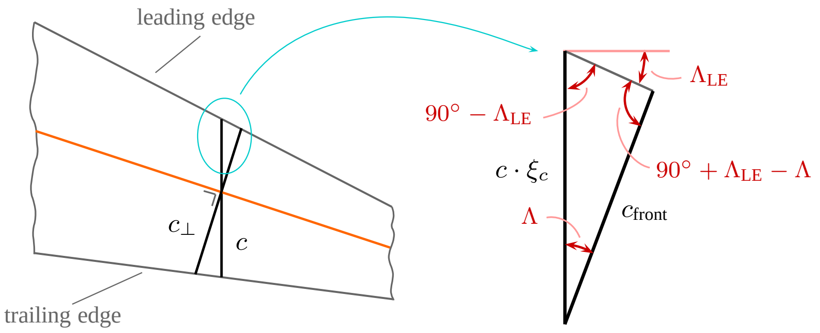


Figure C-2: Front part of the airfoil section perpendicular to sweep line

The airfoil shape at a given section normal to the sweepline, is interpolated between the neighboring user-defined airfoils. Figure C-3 shows how this interpolation is done.

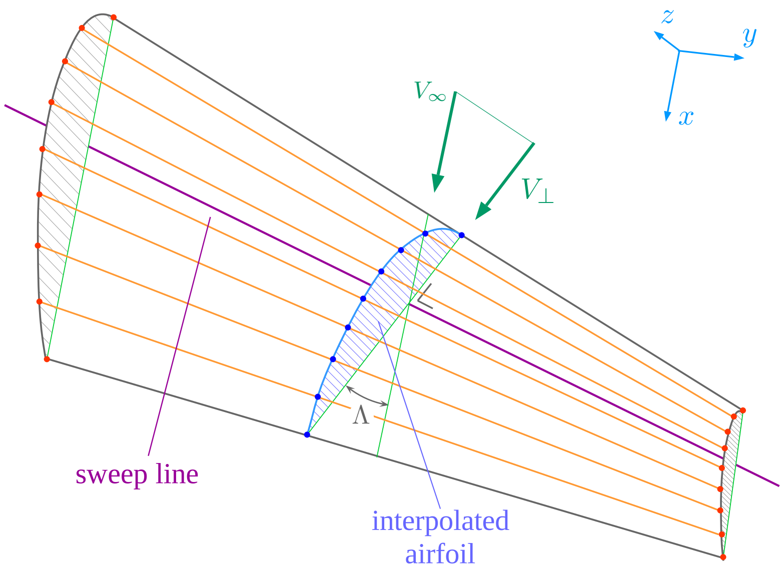


Figure C-3: Upper curve airfoil shape determination based on interpolation of coordinates

Appendix D

Validation quasi-3D aerodynamic solver at low speed

D-1 NACA24-0-0 wing

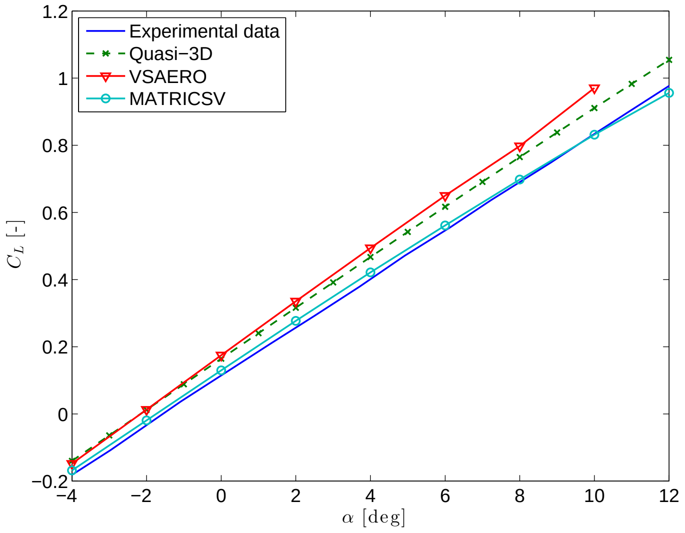


Figure D-1: $C_L - \alpha$ curve of the NACA 24-0-0 wing

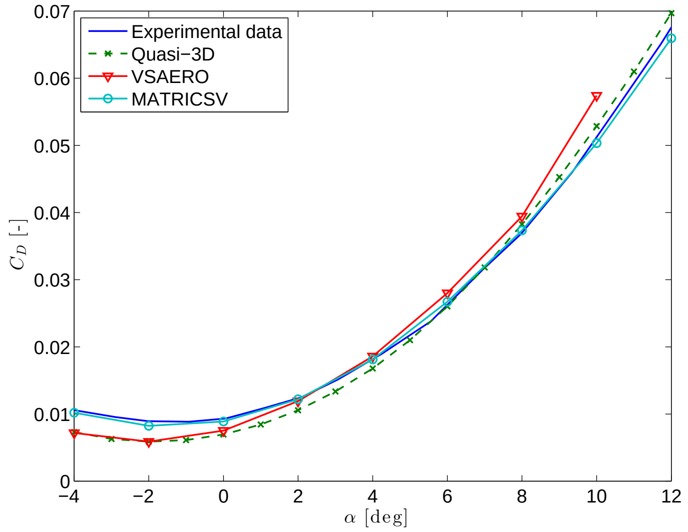


Figure D-2: $C_D - \alpha$ curve of the NACA 24-0-0 wing

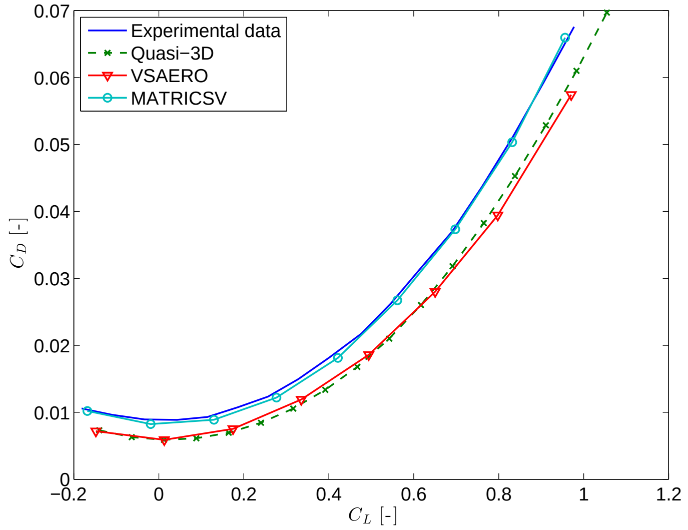


Figure D-3: $C_D - C_L$ curve of the NACA 24-0-0 wing

Table D-1: Error analysis of different aerodynamic solver for the wing drag coefficient of the tapered NACA 24-0-0 airfoil

	Experimental	Quasi-3D		VSAERO		Matrix-V	
C_L	C_D	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$
-0.140	0.0100	0.0073	-2.647	0.0070	-2.922	0.0096	-0.332
0.012	0.0088	0.0059	-2.899	0.0059	-2.905	0.0082	-0.629
0.164	0.0100	0.0069	-3.100	0.0073	-2.722	0.0094	-0.627
0.316	0.0144	0.0106	-3.877	0.0113	-3.168	0.0135	-0.909
0.467	0.0216	0.0168	-4.794	0.0173	-4.315	0.0206	-0.934
0.617	0.0311	0.0260	-5.078	0.0258	-5.326	0.0308	-0.331
0.765	0.0440	0.0383	-5.741	0.0366	-7.362	0.0435	-0.546
0.911	0.0599	0.0529	-7.012	0.0505	-9.365	0.0598	-0.080
1.054	0.0766	0.0697	-6.895	0.0686	-7.940	-	-

D-2 Tapered NACA24-15-0 wing

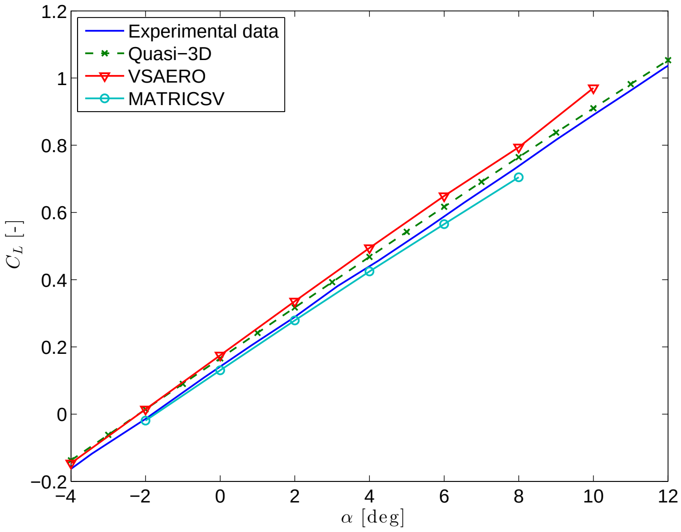


Figure D-4: $C_L - \alpha$ curve of the NACA 24-15-0 wing

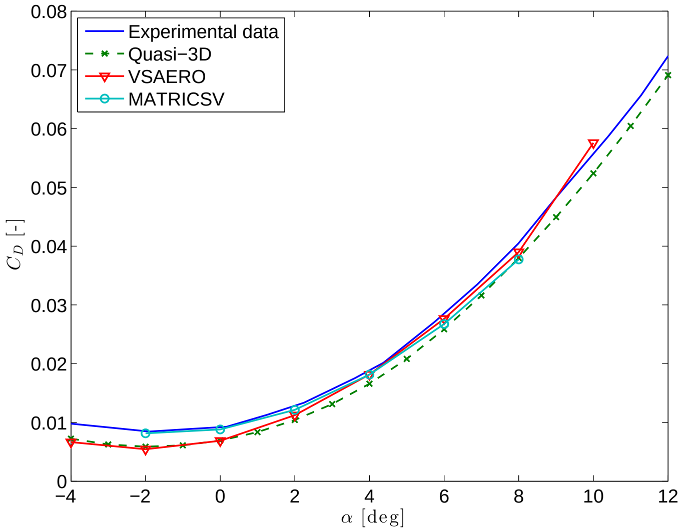


Figure D-5: $C_D - \alpha$ curve of the NACA 24-15-0 wing

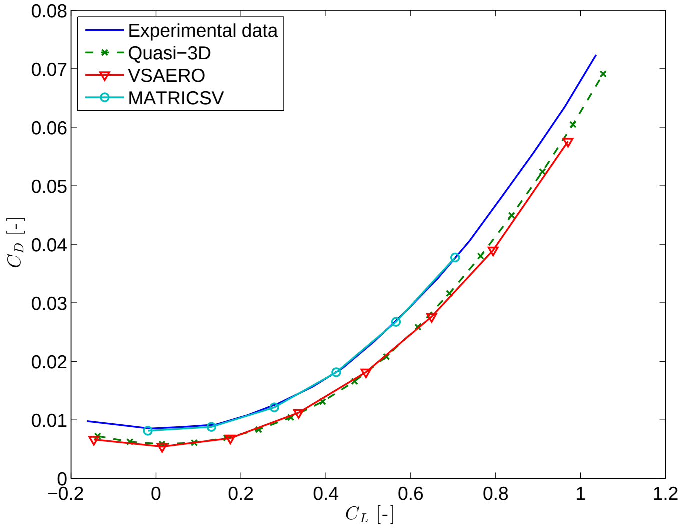


Figure D-6: $C_D - C_L$ curve of the NACA 24-15-0 wing

Table D-2: Error analysis of different aerodynamic solver for the wing drag coefficient of the tapered NACA 24-15-0 airfoil

	Experimental	Quasi-3D		VSAERO		Matrix-V	
C_L	C_D	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$
-0.1375	0.00952	0.00723	-2.28849	0.00653	-2.99415	-	-
0.0142	0.00844	0.00587	-2.56521	0.00544	-2.99975	0.00807	-0.36780
0.1659	0.00962	0.00691	-2.70861	0.00672	-2.89563	0.00936	-0.26057
0.3172	0.01376	0.01043	-3.32916	0.01060	-3.15609	0.01344	-0.31479
0.4678	0.02043	0.01659	-3.84575	0.01679	-3.64417	0.02050	0.06698
0.6170	0.03066	0.02586	-4.80759	0.02546	-5.20263	0.03058	-0.08568
0.7646	0.04295	0.03800	-4.95374	0.03644	-6.51448	-	-
0.9101	0.05793	0.05241	-5.51947	0.05049	-7.44015	-	-
1.0532	0.07431	0.06911	-5.20098	0.06882	-5.48862	-	-

D-3 NACA 24-30-85 airfoil

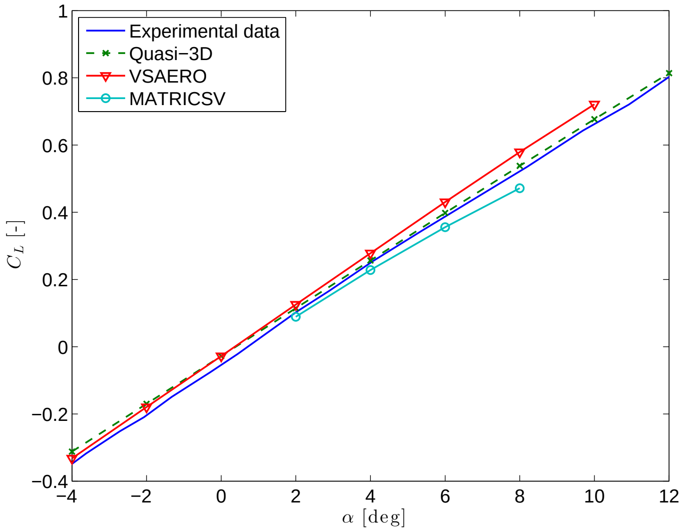


Figure D-7: $C_L - \alpha$ curve of the NACA 24-30-85 wing

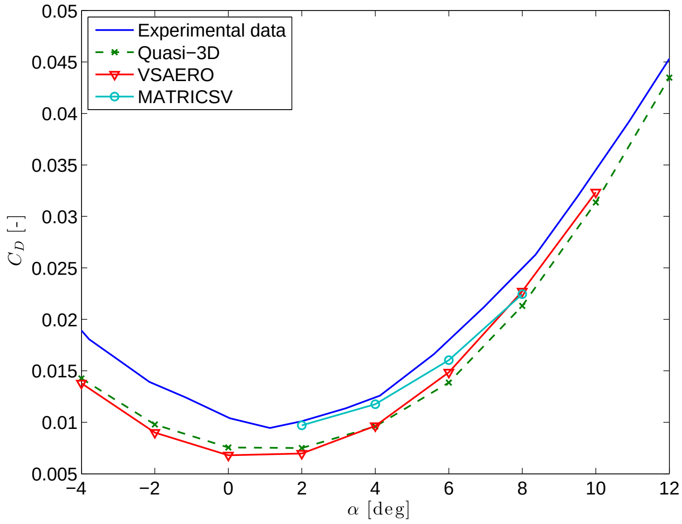


Figure D-8: $C_D - \alpha$ curve of the NACA 24-30-85 wing

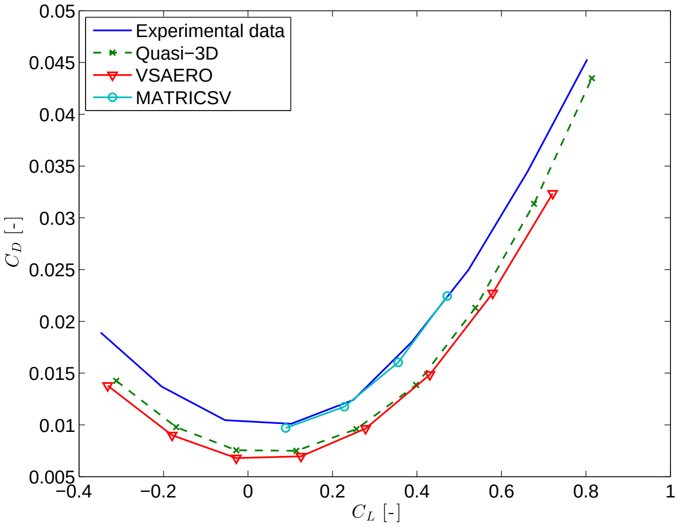


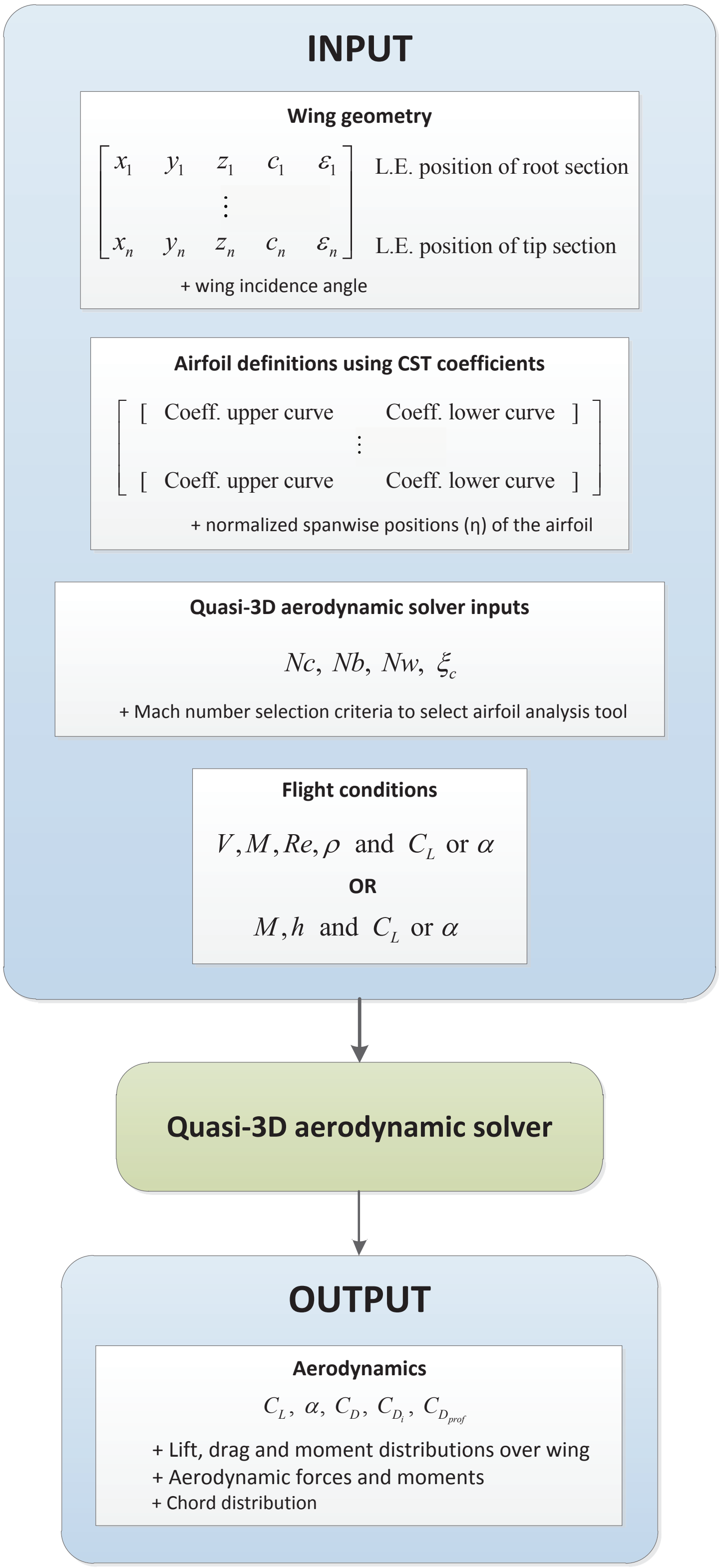
Figure D-9: $C_D - C_L$ curve of the NACA 24-30-85 wing

Table D-3: Error analysis of different aerodynamic solver for the wing drag coefficient of the tapered NACA 24-30-85 wing

	Experimental	Quasi-3D		VSAERO		MATRICSV	
C_L	C_D	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$	C_D	ΔC_D $(\times 1000)$
-0.312	0.0174	0.0143	-3.192	0.0130	-4.456	-	-
-0.170	0.0127	0.0098	-2.944	0.0088	-3.944	-	-
-0.028	0.0102	0.0075	-2.644	0.0068	-3.394	-	-
0.115	0.0102	0.0075	-2.685	0.0069	-3.318	0.0099	-0.280
0.257	0.0126	0.0096	-3.032	0.0091	-3.517	0.0125	-0.135
0.398	0.0185	0.0138	-4.643	0.0135	-4.986	0.0181	-0.417
0.538	0.0260	0.0213	-4.659	0.0203	-5.663	-	-
0.677	0.0356	0.0314	-4.244	0.0292	-6.432	-	-
0.814	0.0462	0.0435	-2.666	-	-	-	-

Appendix E

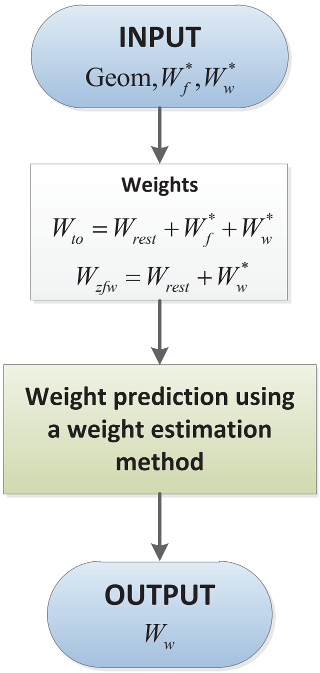
Quasi-3D aerodynamic solver inputs and outputs


INPUT
Wing geometry
$\begin{bmatrix} x_1 & y_1 & z_1 & c_1 & \varepsilon_1 \\ & & \vdots & & \\ x_n & y_n & z_n & c_n & \varepsilon_n \end{bmatrix}$
L.E. position of root section
L.E. position of tip section
+ wing incidence angle
Airfoil definitions using CST coefficients
[Coeff. upper curve Coeff. lower curve]
⋮
[Coeff. upper curve Coeff. lower curve]
+ normalized spanwise positions (η) of the airfoil
Quasi-3D aerodynamic solver inputs
Nc, Nb, Nw, ξ_c
+ Mach number selection criteria to select airfoil analysis tool
Flight conditions
V, M, Re, ρ and C_L or α
OR
M, h and C_L or α
Quasi-3D aerodynamic solver
OUTPUT
Aerodynamics
$C_L,\ \alpha,\ C_D,\ C_{D_i},\ C_{D_{prof}}$
+ Lift, drag and moment distributions over wing
+ Aerodynamic forces and moments
+ Chord distribution

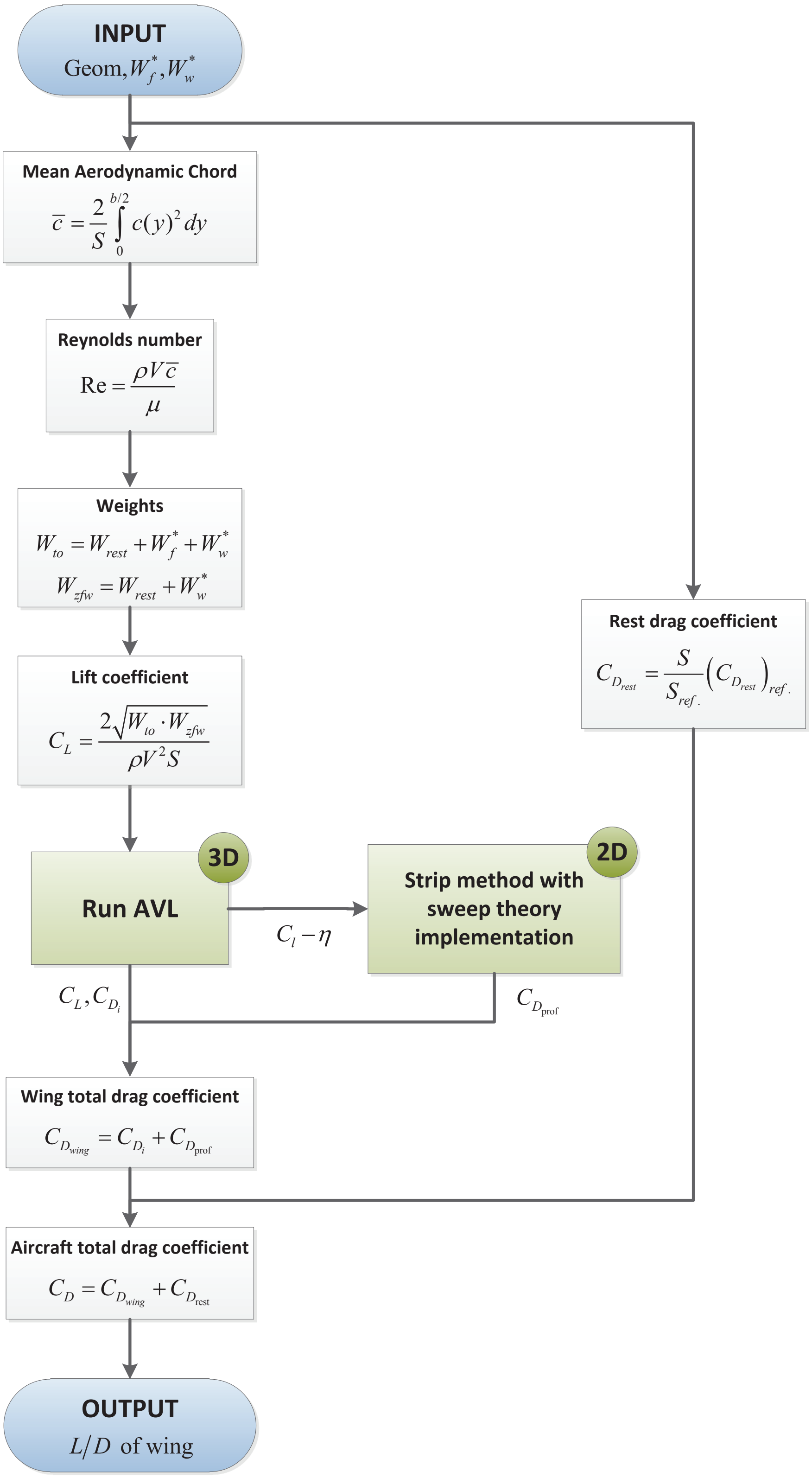
Appendix F

Multidisciplinary design optimization modules

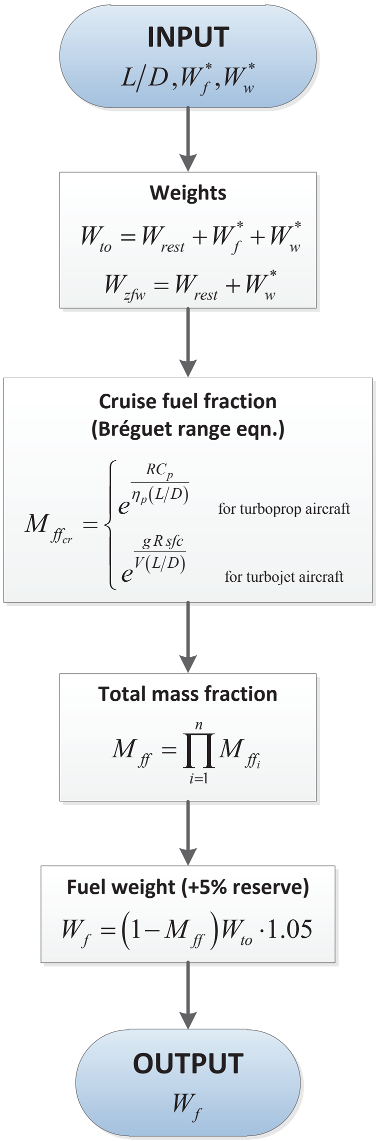
F-1 Weight module (We)


INPUT
Geom, W_f^*, W_w^*
Weights
$W_{to} = W_{rest} + W_f^* + W_w^*$
$W_{zfw} = W_{rest} + W_w^*$
Weight prediction using a weight estimation method
OUTPUT
W_w

F-2 Aerodynamic module (Ae)


INPUT
$\mathrm{Geom}, W_f^*, W_w^*$
Mean Aerodynamic Chord
$\bar{c} = \frac{2}{S}\int_0^{b/2} c(y)^2\,dy$
Reynolds number
$\mathrm{Re} = \frac{\rho V \bar{c}}{\mu}$
Weights
$W_{to} = W_{rest} + W_f^* + W_w^*$
$W_{zfw} = W_{rest} + W_w^*$
Lift coefficient
$C_L = \frac{2\sqrt{W_{to} \cdot W_{zfw}}}{\rho V^2 S}$
3D
Run AVL
$C_l - \eta$
2D
Strip method with sweep theory implementation
C_L, C_{D_i}
$C_{D_{\mathrm{prof}}}$
Rest drag coefficient
$C_{D_{rest}} = \frac{S}{S_{ref.}}\left(C_{D_{rest}}\right)_{ref.}$
Wing total drag coefficient
$C_{D_{wing}} = C_{D_i} + C_{D_{\mathrm{prof}}}$
Aircraft total drag coefficient
$C_D = C_{D_{wing}} + C_{D_{\mathrm{rest}}}$
OUTPUT
L/D of wing

F-3 Performance module (Pe)


INPUT
$L/D, W_f^*, W_w^*$
Weights
$W_{to} = W_{rest} + W_f^* + W_w^*$
$W_{zfw} = W_{rest} + W_w^*$
Cruise fuel fraction
(Bréguet range eqn.)
$M_{ff_{cr}} = \begin{cases} e^{\frac{RC_p}{\eta_p(L/D)}} & \text{for turboprop aircraft} \\ e^{\frac{g\,R\,sfc}{V(L/D)}} & \text{for turbojet aircraft} \end{cases}$
Total mass fraction
$M_{ff} = \prod_{i=1}^{n} M_{ff_i}$
Fuel weight (+5% reserve)
$W_f = (1 - M_{ff}) W_{to} \cdot 1.05$
OUTPUT
W_f