Internal Systems Design for Smart Fixed Wing Technologies using Knowledge Based Engineering

M.A.B. van den Berg BSc.
August 23, 2010
Internal Systems Design for Smart Fixed Wing Technologies using Knowledge Based Engineering

Master of Science Thesis

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

M.A.B. van den Berg BSc.

23-08-2010

Faculty of Aerospace Engineering - Delft University of Technology
The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance a thesis entitled “Internal Systems Design for Smart Fixed Wing Technologies using Knowledge Based Engineering” by Merijn Alexander Brand van den Berg in partial fulfilment of the requirements for the degree of Master of Science.

Dated: _______

Head of Aerodynamics Department: ________________________________
prof.dr.ir. drs. H. Bijl

Supervisor: ________________________________
ir. R.E.C. van Dijk

Reader: ________________________________
ir. G. La Rocca

Reader: ________________________________
dr.ir. L.L.M. Veldhuis
Abstract

The growing awareness of our environment demands the aviation industry to produce eco-friendly and more efficient aircrafts. To support this the Clean Sky Joint Technology Initiative (JTI) is developing and improving breakthrough technologies, that will be demonstrated on a flying prototype. The Clean Sky JTI is split up into six technology domains; one of these is the Smart Fixed Wing Aircraft (SFWA). This domain aims to develop and test a new wing design that makes use of passive and active flow and load control technologies. Department of Systems Engineering and Aircraft Design (SEAD) at Delft University of Technology has been asked to develop a simulation framework to support the internal systems design on the Smart Fixed Wing Aircraft.

During this thesis work a parametric model and framework has been created for two flow control technologies and their pneumatic supply system. The model is based on Knowledge Based Engineering (KBE) techniques, which aim to increase the productivity of engineers and allow more detailed and fair concepts trade-offs.

The two selected ‘smart technologies’ are Hybrid Laminar Flow Control (HLFC) and Fluidic Actuated Flow Control (FAFC). The first extends laminar flow using a combination of Natural Laminar Flow and Boundary Layer Suction. The second delays separation, stalling and buffeting by reenergising the boundary layer using pulsating or synthetic jets.

The parametric model creates a 3D representation of various internal system concepts, which is used to evaluate the Internal Aerodynamics, External Aerodynamics (HLFC only), weight and cost. The internal aerodynamic analysis uses handbook relations and has been incorporated into the KBE environment. It has been validated for conceptual design with limited success. External aerodynamics is performed with a link to Xfoil-suc. The cost and weight are currently based on component and (sheet-)material price and weight.

The product model created in this thesis work can evaluate multiple concepts of the internal system for both ‘smart technologies’ and thereby allow a more detailed and fair trade-off. Due to the lack of available input specifications, e.g. aircraft-type and external aerodynamic data, a trade-off between the concepts could unfortunately not yet be made.
Acknowledgements

First of all I would like to thank my family for all their love and support throughout my study. A special thanks to my girlfriend for her support and taking care of me during these last intensive weeks.

I would like to thank my friends and tennis buddies who helped me get relaxed and get my mind away from my thesis work from time to time. But also for their encouragements when things did not worked out the way I planned to.

I would like to thank Reinier van Dijk for all his support and expertise of KBE, Jonas for his extensive literature research and Fred & Gianfranco for their support with my report. Also thanks to Arne Seitz for his support on HLFC.

Delft, The Netherlands

23-08-2010

M.A.B. van den Berg
# Contents

Abstract ........................................................................................................................................... v

Acknowledgements ....................................................................................................................... vii

Abbreviations ................................................................................................................................. xiii

List of Symbols ............................................................................................................................... xv

Chapter 1 Introduction ................................................................................................................... 1
  1.1 Technical challenges ............................................................................................................... 2
  1.2 Clean Sky JTI ......................................................................................................................... 3
  1.3 Industry Challenges ............................................................................................................. 4
  1.4 IT to manage knowledge ....................................................................................................... 6
  1.5 Smart Technologies ............................................................................................................. 7
  1.6 Previous Work .................................................................................................................... 8
  1.7 Goals ......................................................................................................................................... 9
  1.8 Document Layout ................................................................................................................ 9

Chapter 2 KBE design approach .................................................................................................. 11
  2.1 What is KBE? ......................................................................................................................... 11
  2.1.1 KBS in Engineering .......................................................................................................... 11
  2.1.2 Combining KBE and geometry based analysis ................................................................. 12
  2.1.3 Process cycle ..................................................................................................................... 13
  2.1.4 Design Engineering Engine ........................................................................................... 14
  2.1.5 Object oriented programming .......................................................................................... 15
  2.2 Design Approach .................................................................................................................. 16
  2.2.1 Discipline selection .......................................................................................................... 16
  2.2.2 Knowledge management steps ......................................................................................... 18
  2.2.3 Product Model requirements ........................................................................................... 19

Chapter 3 Hybrid Laminar Flow Control ..................................................................................... 21
  3.1 Technology principles .......................................................................................................... 21
  3.1.1 Boundary layers ............................................................................................................... 21
  3.1.2 Natural Laminar Flow ....................................................................................................... 22
  3.1.3 Boundary Layer Suction ................................................................................................. 24
  3.1.4 Hybrid Laminar Flow control ........................................................................................ 25
3.2 HLFC Requirements ...................................................... 26
3.3 HLFC internal systems .................................................. 26
    3.3.1 Perforated surface .................................................. 28
    3.3.2 Skin-chambers and manifold ..................................... 29
    3.3.3 Pipe-system inlet ................................................. 30
3.4 HLFC Concepts ............................................................ 30

Chapter 4 Fluidic Actuated Flow Control .................................. 33
4.1 Technology Principle ..................................................... 33
    4.1.1 Continuous jets / pulsating jets ................................. 34
    4.1.2 Synthetic jets ....................................................... 35
4.2 Actuator requirements .................................................. 36
4.3 Actuator Selection ........................................................ 36

Chapter 5 Pneumatic system .................................................. 39
5.1 Pneumatic principles ..................................................... 40
    5.1.1 General pneumatic relations ..................................... 40
    5.1.2 Relations for straight pipe segments ............................ 41
    5.1.3 Relations for bend segments ..................................... 42
    5.1.4 Relations for Junctions .......................................... 43
    5.1.5 Relations for Pumps .............................................. 48
    5.1.6 Relations for the inlet/outlet slot ............................... 49
5.2 Requirements and Criteria ............................................. 50
5.3 Pneumatic Concepts ....................................................... 51

Chapter 6 Knowledge structure .............................................. 55
6.1 Global overview .......................................................... 56
6.2 HLFC structure ............................................................ 57
6.3 FAFC structure ........................................................... 58
6.4 Pipe assembly structure ............................................... 58
6.5 Air inlet/outlet assembly structure .................................. 59

Chapter 7 Classes and routines ............................................... 61
7.1 Common storage structure for flow conditions .................... 61
7.2 HLFC classes .............................................................. 61
    7.2.1 Skin-chamber distribution ...................................... 62
    7.2.2 Perforated surface ............................................... 63
    7.2.3 Skin-chamber ...................................................... 63
    7.2.4 Suction-assembly ............................................... 64
    7.2.5 Flow-regulation-through-orifice .............................. 64
    7.2.6 Manifold ........................................................... 65
7.3 FAFC classes .............................................................. 66
    7.3.1 Actuator-area ...................................................... 66
    7.3.2 Actuator ............................................................ 66
7.4 Pipe-assembly classes ................................................ 67
    7.4.1 Pipe-path .......................................................... 67
    7.4.2 Pipe-route ........................................................ 68
E.1 Obstruction ................................................................. 115
E.2 Reducing the pipe diameter ........................................... 117

Appendix F Comparison between AFT Arrow and the horizontal momentum equations for equal area junctions ........................................ 119

Appendix G YADD console ...................................................... 121

Appendix H Suction analysis at different flight conditions ................. 127

Appendix I Smart Fixed Wing Manual ........................................ 129
  I.1 Packages ........................................................................... 129
  I.2 Inputs ................................................................................. 130
    I.2.1 Wing and cp-profile ...................................................... 130
    I.2.2 Fixed parameters ............................................................. 131
    I.2.3 SFWA inputs ................................................................. 131
    I.2.4 Other inputs ................................................................. 132
  I.3 Outputs ............................................................................. 135
  I.4 Pumps .............................................................................. 135
  I.5 Actuators .......................................................................... 136
  I.6 Final Remarks ................................................................... 136
# Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AOA</td>
<td>Angle of Attack</td>
</tr>
<tr>
<td>BLS</td>
<td>Boundary Layer Suction</td>
</tr>
<tr>
<td>CAA</td>
<td>Computer Aided Analysis</td>
</tr>
<tr>
<td>CAD</td>
<td>Computer Aided Design</td>
</tr>
<tr>
<td>CFD</td>
<td>Computational Fluid Dynamics</td>
</tr>
<tr>
<td>DEE</td>
<td>Design Engineering Engine</td>
</tr>
<tr>
<td>FAFC</td>
<td>Fluidic Actuated Flow Control</td>
</tr>
<tr>
<td>GDL</td>
<td>General-Purpose, Declarative, Language</td>
</tr>
<tr>
<td>HLFC</td>
<td>Hybrid Laminar Flow Control</td>
</tr>
<tr>
<td>ID</td>
<td>Inner Diameter</td>
</tr>
<tr>
<td>ITD</td>
<td>Integrated Technology Demonstrator</td>
</tr>
<tr>
<td>JTI</td>
<td>Joint Technology Initiative</td>
</tr>
<tr>
<td>KBE</td>
<td>Knowledge Based Engineering</td>
</tr>
<tr>
<td>KBS</td>
<td>Knowledge Based Systems</td>
</tr>
<tr>
<td>LE</td>
<td>Leading Edge</td>
</tr>
<tr>
<td>NLF</td>
<td>Natural Laminar Flow</td>
</tr>
<tr>
<td>MMG</td>
<td>Multi Model Generator</td>
</tr>
<tr>
<td>SEAD</td>
<td>Systems Engineering and Aircraft Design</td>
</tr>
<tr>
<td>SFWA</td>
<td>Smart Fixed Wing Aircraft</td>
</tr>
<tr>
<td>TE</td>
<td>Trailing Edge</td>
</tr>
<tr>
<td>TRL</td>
<td>Technology Readiness Level</td>
</tr>
</tbody>
</table>
List of Symbols

- $c$: Chord [m]
- $C_d$: Drag coefficient
- $C_{aq}$: Drag coefficient due to the pneumatic system
- $C_p$: Pressure coefficient
- $C_q$: Suction coefficient
- $D$: Diameter [m]
- $D_h$: Hydraulic Diameter [m]
- $f$: Friction factor
- $g$: Gravitational acceleration [m/s$^2$]
- $h_l$: Head loss [m]
- $h_p$: Pump head [m]
- $K$: Pressure loss coefficient
- $l_e$: Equivalent length [m]
- $L$: Length [m]
- $\dot{m}$: Mass flow [kg/s]
- $p$: Pressure [Pa]
- $P_{in}$: Input power [W]
- $P_{out}$: Output power [W]
- $q$: Mass-flow ratio
- $Q$: Flow rate [m$^3$/s]
- $R$: Perfect Gas constant
- $R_c$: Bend radius [m]
- $Re$: Reynolds Number
- $S$: Cross-sectional area [m$^2$]
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>Temperature</td>
<td>[K]</td>
</tr>
<tr>
<td>$u$</td>
<td>Speed</td>
<td>[m/s]</td>
</tr>
<tr>
<td>$V_\infty$</td>
<td>Free-stream airspeed</td>
<td>[m/s]</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Bend angle</td>
<td>[°]</td>
</tr>
<tr>
<td>$\Delta p$</td>
<td>Pressure drop</td>
<td>[Pa]</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Absolute roughness</td>
<td>[m]</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Branch angle</td>
<td>[°]</td>
</tr>
<tr>
<td>$\theta^*$</td>
<td>Corrected branch angle</td>
<td>[°]</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Density</td>
<td>[kg/m$^3$]</td>
</tr>
<tr>
<td>$\eta_p$</td>
<td>Pump efficiency</td>
<td></td>
</tr>
<tr>
<td>$\mu$</td>
<td>Dynamic viscosity</td>
<td>[Pa*s]</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Area-ratio</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

In our current society and economy aviation has an important role. More than 45,000 people in the Netherlands work at airports, airliners or elsewhere associated to the aviation industry. Together the Dutch airports handle more than 45 million passenger manoeuvres and 1.3 million ton of cargo. The net turnover of aviation related activities in 2007 was almost eleven billion euro and which resulted into around 650 million euro of taxes. These figures clearly show the importance of the aviation industry on our economy.

With an exception to 2009, due to the financial crisis, the number of passengers and tons of cargo has been continuously increasing in the past, shown in Figure 1-1. Even though 2009 shows a decrease, the figures are expected to rise again for the coming decades and follow the trend prior to the crisis. Among others this is shown by the orders made ($44.7 billion worth of business) during the Farnborough International Airshow [Farnborough International Airshow, 2010]. In 2008 Boeing and Airbus together sold 1517 aircraft, but in 2009 only 573. This year already 444 aircraft have been sold and the year is only half way. [NOS, 2010]

![Figure 1-1 Dutch aviation figures of passengers and tons of cargo (source: CBS)](image_url)
1.1 Technical challenges

On a level of performance improvement the air transport sector is, however, reaching the end of its second S-curve (Figure 1-2). The current type of aircraft has seen many improvements and is optimized for commercial benefits. Little room is left for commercial improvement. Because of the growing awareness on the environmental impacts, the next generation aircraft will have to handle more transport manoeuvres and carrying more passengers whilst being more sustainable than current aircraft. In order to make this growth sustainable new solutions will have to be created. Which technologies will enable such substantial growth and an entry into the Age of Sustainable Growth is not yet clear.

The challenges for the aerospace world arising from this are outlined in ‘European Aeronautics: a Vision for 2020’ [2001]. A selection of these challenges, relevant to the work in this thesis, is listed in Table 1-1. Together with industry the European Union has initiated the Clean Sky Joint Technology Initiative (JTI) to develop and validate the technology breakthroughs needed to meet these, and other, challenges. The next section will provide a short summary of the Clean Sky JTI and the position of this thesis within the project.

Figure 1-2 Civil air transport performance S-curve’s [van Tooren, et al., 2007]
Table 1-1 ACARE Visions 2020 goals relevant to this thesis work

<table>
<thead>
<tr>
<th>Environment</th>
<th>European Aeronautics industry</th>
</tr>
</thead>
<tbody>
<tr>
<td>• A 50% cut in CO2 emissions per passenger kilometre (which means a 50% cut in fuel consumption in the new aircraft of 2020)</td>
<td>• A new framework that permits and encourages companies to work together more effectively in setting and achieving their industrial priorities. This will strengthen competitiveness and improve responses to changing market conditions.</td>
</tr>
<tr>
<td>• An 80% cut in nitrogen oxide emissions.</td>
<td>• Halve the &quot;time to market&quot; for new products with the help of advanced electronic analytical, design, manufacturing and maintenance tools, methods and processes.</td>
</tr>
<tr>
<td>• A reduction in perceived noise to one half of current average levels.</td>
<td></td>
</tr>
</tbody>
</table>

1.2 Clean Sky JTI

The Clean Sky JTI is one of the largest European research projects ever and has a budget of €1.6 billion, equally shared between the European Commission and industry, research institutes, and academia. It is planned to run over a time span of seven years, from 2008 to 2015. This public-private partnership aims to speed up technological breakthrough developments and shorten the time to market for new solutions tested on Full Scale Demonstrators.

The Clean Sky JTI is divided into six technology domains, shown in Figure 1-3, that each will deliver full-scale Integrated Technology Demonstrators (ITD). These ITDs will be created with a focus on the integration of advanced technologies and validation of results in a multidisciplinary approach. The technologies allowing a potential step change in aircraft performance are concurrently developed, integrated and validated.

Figure 1-3 the six technology domains of the Clean Sky JTI (Source: Clean-sky.eu)
The TU Delft has different involvements in the Clean Sky JTI. The department of Systems Engineering and Aircraft Design (SEAD) was asked to develop a simulation framework for the Smart Fixed Wing Aircraft (SFWA) ITD. The goal is to create a tool to support integration of the internal systems for the Smart Technologies. The SFWA aims to reach the following goals for high speed business jets, regional-, and long-range transport aircraft [Smart Fixed Wing Aircraft - Integrated Technology Demonstrator, 2008]:

- Reductions in fuel consumption – and hence in CO2 & NOx emissions – in the order of 20%
- Noise reduction by 5 to 10dB
- Successful and efficient integration of innovative power-plant and empennage concepts
- Enhance the comfort and safety of future transport including aircraft agility and flight trajectory flexibility

Note that the reduction in fuel consumption excludes the reduction coming from the engines.

No aircraft type has yet been selected within the SFWA project; the integration tool should thus be flexible and able to adjust to different aircraft shapes. One aircraft that has been suggested within the SFWA ITD is the Dassault Falcon 7X, a long range business tri-jet (Figure 1-4).

---

**1.3 Industry Challenges**

The entry into the Age of Sustainable Growth is not the only challenge the Aviation industry is facing. As van Tooren [Inaugural speech, 2003] indicated, less technical factors create challenges as well. Three challenges the Aerospace Industry in western countries faces are shown in Figure 1-5. Although some of these challenges are caused by technical aspects, such as increasing complexity, their impacts and solutions are less technical.
1. Introduction

The first challenge lies in the reducing availability of engineers, due to decreasing birth rates in western countries and decreasing number of technical master graduates (Figure 1-6). Fortunately aerospace still attracts young people to choose a technical study, such that this decrease is not occurring at the faculty of Aerospace Engineering. It does however mean that the aviation industry will have to share its engineers with other technical areas.

The second challenge involves the increasing complexity of products and processes in aerospace. Aircraft are a very complex means of transportation, which leads to very high development, production and maintainability cost. Development cost is the major determining factor for accessibility and creates large business risks for the companies involved. One project shutdown can seriously challenge the sustainability of a company. The Clean Sky JTI can be seen as an attempt to minimize these risks and increase the accessibility of new technologies.

The third challenge is a reduction in experience of the aerospace engineer, due to the reduction in aircraft development programs and the duration of each program. This effect is shown Figure 1-7. The average engineer used to be involved in five programs during its career, in contrast to current engineers being involved in less than two programs.

Figure 1-5 challenges of the aerospace industry

Figure 1-6 Decrease in Technical Graduates in The Netherlands [CBS Statline, 2010]
Together these challenges result in the need for less people to do more in the future. Engineers will thus have to improve their productivity considerably.

Information Technology (IT) was long believed to automatically increase the productivity of engineers. Instead it has increased the demand for information from e.g. airworthiness authorities, resulting in an increase in engineers per project. [van Tooren M. J., 2003]

Finally, with the current aircraft being extremely complex there is a major interest in multi-disciplinary design and optimisation, as is also identified in the Clean Sky JTI. This requires engineers of different trades to work together on one single project, engineers often working for different companies and located in different places on the world. Sharing and managing knowledge in such a project often turns out to be a large challenge.

1.4 IT to manage knowledge

To overcome the challenges described in section 1.3, a significant decrease of cost and development time is needed. To do this Drucker [2002] claims we should use IT to clearly record, manage, and re-use existing knowledge. To record engineering knowledge IT must combine geometry manipulation with programming. This is known as Knowledge Based Engineering (KBE). Geometry is no longer drawn by an engineer but is created by the KBE system, based on knowledge stored within. This knowledge can be applied on different sets of input data to generate new solutions, thereby reducing the amount of repetitive work to be performed by the engineers. Managing knowledge in this way is also thought to enable engineers to efficiently work together and create a truly multi-disciplinary design. The time saved by the KBE approach will free intellectual resources and improve creativity.

The Clean Sky JTI is a clear example of a multi-disciplinary and geographical border crossing program. Managing its knowledge streams and combining its disciplines is usually a difficult but important task. KBE has the capability to support the knowledge
management and support the integration of systems. This thesis work will focus on using the KBE design methodology to support the integration of the internal systems for the SFWA.

1.5 Smart Technologies

The SFWA ITD will develop and evaluate a large number of technologies. Figure 1-8 shows some examples of these. Time restraints did not allow all technologies to be worked out during this thesis. Two were selected on the available knowledge, the relevance to the project and applicability to the KBE approach.

The first technology is Hybrid Laminar Flow Control (HLFC). A technology that combines Natural Laminar Flow and Boundary Layer Suction to achieve extended laminar flow on the wing surface even on aircraft with high Reynolds numbers and large wing sweep. HLFC has already undergone extensive research and has been applied to the vertical tail of an Airbus A320 in the ELFIN II project, shown in Figure 1-9. This technology is selected because of its available knowledge and the large influences of the internal systems on its performance. The goal for this technology is to capture the available knowledge using KBE and apply this to the Leading Edge (LE) of the main wing.

The second technology is Fluidic Actuated Flow Control (FAFC), a technology that reenergises the boundary layer, with pulsating or synthetic jets, to delay separation, stalling and decrease buffeting. This is a less matured technology, however one that shows great potential. This technology still has many design steps to go through and therefore many unknown specifications to its supply systems. Using KBE in this case can have its advantages. By making a division into functions and components a conceptual design tool can be created. The physical objects, e.g. actuators (Figure 1-10), are still under development. Their functions and type of supply systems however, are known. A KBE design will allow replacement of one actuator by another, without having to redesign the entire model.

Both technologies address a different part of the flight and use different methods to alter the flow over the wing surface. However, both need an internal system for air guidance. Where HLFC needs a source for suction, FAFC needs a source for blowing. To deliver this to the parts of each technology located at the wing surface a set of pipes, pumps and valves are needed. Using the KBE approach the knowledge relevant to this pneumatic system is stored only once and then used for both technologies.
1.6 Previous Work

The TU Delft has been active in developing KBE product models over the last years, such as for wind turbine blades [Chiciudean, et al., 2008] and aircraft wiring harnesses [van der Elst, et al., 2008]. Recently a multi-element wing model has been created using GDL, the DARwing. This product model is able to create multi-element wing shapes and other aircraft components that closely match those of wings found on current aircraft, but also wings for new aircraft configuration such as the Blended Wing...
1. Introduction

Body. Examples of the outcomes are shown in Figure 1-11. The DARwing will be used in this thesis to create the wing of the SFWA, which will form the basis of the rest of the work. More information on the outer surface of the DARwing can be found in the thesis work of T. van den Berg [2009] and J.H. Koning [2010].

Figure 1-11 Fokker 100 and Blended Wing Body aircraft modelled with the DARwing model [van den Berg T., 2009]

1.7 Goals

KBE is one of the main research fields at the department of Systems Engineering and Aircraft Design (SEAD) from the faculty of Aerospace Engineering. The internal systems design for the smart technologies, described above, is a good candidate for KBE. Together this leads to the following goals for this thesis work:

1. Support the internal systems design for HLFC and FAFC on the Smart Fixed Wing Aircraft using Knowledge Based Engineering
2. Demonstrate the capabilities and benefits of Knowledge Based Engineering in a multi-disciplinary conceptual design by creating an autonomous KBE application.

Because this thesis work will end before the completion of the SFWA design continuity is also of importance. The KBE tool that is created should therefore be easily extended and improved during later stages of the design. Recommendations will also be to how this can be achieved.

1.8 Document Layout

In Chapter 2 the design approach for this thesis work is identified and described. HLFC, FAFC and the pneumatic system are discussed in Chapter 3, Chapter 4 and Chapter 5, respectively. These chapters will each start with an overview of the knowledge involved and end with a selection of applicable concepts.
The knowledge structure created for the SFWA equipped with HFLC and FAFC is detailed in Chapter 6. The individual classes and analysis routines that fill up the structure are described in Chapter 7. For the external aerodynamic analysis an interface is created to Xfoil-suc (Chapter 8). Verification of the analysis routines is shown in Chapter 9.

At last the knowledge stored can be used to create design solution and assess their performance. Chapter 10 will shows results for an example case obtained from the product model. Finally, in Chapter 11, the conclusions based on this research are given as well as recommendations for future work.
Chapter 2

KBE design approach

This chapter aims to give more information on the design approach and scope of the design. As indicated in the thesis goals in section 1.7, KBE will be used to support the design of the internal systems of the smart fixed wing. To identify which steps and actions are needed to perform this design first the basic principles of KBE are explained (2.1). Next the scope of the design is determined and the steps to be taken are described (2.2.2). At last the requirements imposed on the KBE application are given (2.2.3).

2.1 What is KBE?

La Rocca [2009] indicates that KBE finds its origins in Knowledge Based Systems (KBS), which have been in use since the 1970’s. These systems consist of a database of knowledge that is stored in the form of a set of rules. This knowledge is used by the system to reason a solution to a posed problem. Because the system generates solutions from knowledge it can explain how and why it generates a specific solution. This solution will always be consistent with the rules specified in the knowledge database. In case of missing information the system will not be able to generate a solution, instead it will ask for extra information. These aspects enable a user to apply the knowledge in an automated process, and benefit from the repetitive capabilities of IT. Successful examples of these have been developed for fields as blood infection and analysis of chemicals [La Rocca, 2009], but they are currently used in much more fields.

2.1.1 KBS in Engineering

Although KBS has been around for a long time it was never used on a large scale for engineering purposes. The main reason for this is that engineering is largely dependent on geometry manipulation and analysis, which was not possible in a KBS environment. Well known products that help an engineer with the geometry manipulation are computer aided design (CAD) systems1.

---

1 Examples are: CATIA of Dassault Systems and NX of Siemens PLM Software
2.1 What is KBE?

KBE combines CAD with KBS (Figure 2-1). It is actually a special kind of KBS system. La Rocca [2009] states:

“Knowledge based engineering (KBE) is a technology based on the use of dedicated software tools (i.e. KBE systems) that are able to capture and reuse product and process engineering knowledge.”

The main objective of KBE is the reduction of time and costs of product development by means of the following:

- Automation of repetitive, non-creative, design tasks
- Support of multidisciplinary integration as from the conceptual phase of the design process.

Different KBE system are available, for this thesis work the Genworks GDL environment is used to create the product models for HLFC and FAFC. GDL is widely used at the TU Delft, is has the capability to manipulate geometry directly using standardised objects and routines.

2.1.2 Combining KBE and geometry based analysis

In engineering the geometries created with the CAD programs will have to be analysed for its properties, this often performed with the help of Computer Aided Analysis (CAA) software.

The GDL KBE system can analyse geometry to a limited extend; primarily dimensions, area and volume. To perform a more detailed analysis, e.g. aerodynamic and cost, extensions are needed to the KBE system. Two main approaches can be identified to combine the KBE system with other geometry based analysis. One is to create an interface between the KBE environment and an existing CAA tool, the other approach is to incorporate analysis routines into the knowledge based engineering system.

---

1 Examples are: Dassault Systems KnowledgeWare, Siemens Knowledge Fusion, Genworks GDL and Technosoft AML

2 Examples are: AMI VSAERO™ and ANSYS Structural
2. KBE design approach

Figure 2-2 two approaches to combine CAD and KBS

CAA tools often require their input in a very specific form. In most cases this requires complex pre-processing of the geometry model. When an automated interface is used between KBE and the CAA tool, the KBE system is required to perform this pre-processing. To do this the pre-processing knowledge has to be captured first into the KBE system, which can be a complex task. This approach, however, requires no or little changes to the analysis tool and will not impact its fidelity. It does limit the freedom of the designer, caused by the limits of the interface or the tool itself. It can also decrease the speed of the system as data may have to be converted and send back and forth multiple times between the tool and the knowledge environment.

Incorporating the knowledge of the analysis into the KBE environment excludes the need for an interface and pre-processing. This allows a much more direct interaction with the analysis. The knowledge is available to create a solution, rather than only evaluate a solution. The fidelity however has to be proven again. Also the implementation can be a cost and time consuming process.

2.1.3 Process cycle

Knowledge Based Engineering enables a new design process cycle [La Rocca, et al., 2005]. This cycle is shown in Figure 2-4 next to the traditional design process. Both cycles start with a set of requirements and input specifications. With these a concept is created based on engineering knowledge. Relevant data is collected and a geometry model is created. The created concept is analysed and its performance is compared to the initial requirements. If the design solution is not acceptable the inputs are adapted and the loop must be repeated. This iterative process is repeated until an acceptable design solution is found.

In a traditional design process the gathering of data, the creation of geometry and the manipulation of geometry for analysis is performed by the engineers. These time-consuming tasks form bottle-necks in the design process. They keep the engineers occupied, such that they are less involved with the creative parts of the design process.

In the KBE design process the product model performs a pivoting role. Engineering knowledge is stored within and used to generically create a design solution. The performance of the created solution is analysed using routines within the model or through automated interface to other computer aided tools.
This design process involves little repetitive work for the engineer; this is instead performed by the KBE product model. The generic abilities of the model give the designer more time to spend on creative tasks, allowing multiple concepts to be investigated in more detail before selection has to be made.

2.1.4 Design Engineering Engine

To automate KBE process cycle a Design and Engineering Engine (DEE) is used. The DEE consists of a set of interconnected toolboxes automating as far as possible multi-disciplinary design and optimization. A schematic overview of a DEE is shown in Figure 2-4.

The heart of the DEE is the parametric product model (or Multi Model Generator (MMG)); it creates the geometry and provides discipline specific data. With this the various disciplines perform their analysis. The analysis data of all routines together forms the overall performance of the solution.

Other parts are the initiator and the converger & evaluator. The first provides the initial values for the product model. The second checks whether the solution is converged and if the requirements are met. If the solution is not converged the parameter values are adjusted and run through the product model again. When the solution is converged but does not meet the requirements, the inputs are reconsidered and the entire process is run again. When the solution does meet the requirements the DEE is finished and the solution is presented.

The modular build-up of the DEE allows analysis tools to be easily added and replaced during a complete development program. Such that low fidelity tools, used in the conceptual phase, can be replaced by higher fidelity tools in the preliminary phase.
2. KBE design approach

2.1.5 Object oriented programming

To support the use, reuse and addition of engineering knowledge in the product model a clear knowledge structure is needed. KBE uses an object oriented structure. Instead of using a list of if-then-else rules, the knowledge is stored into classes. Each class has its own slots with information, rules and formula’s. They can be seen as building blocks that, in an assembly, together create the complete solution.

The classes can be divided into three types: physical classes, functional classes, and property classes. Figure 2-5 shows on the left a structure tree with purely physical classes, the right shows a tree with property and functional classes. Mixes of both trees are often found in KBE applications. The physical classes carry knowledge about the physical shape, e.g. the spar class carries information regarding the size, thickness etc. of a spar. The functional classes carry knowledge to a specific function, they compute parameters and values usually used by physical objects, e.g. the Provide Control will determine where and how many control surface are needed on the wing. The property classes evaluate the physical objects, e.g. the weight class determines total weight of the wing.
2.2 Design Approach

Now that the principles of the KBE design are identified, the design approach used for this thesis work can be determined. Because the design of internal systems for the Smart Fixed Wing is a multi-disciplinary design, it is important to start by selecting the disciplines to incorporate and how to use them in a DEE (2.2.1). Next the steps taken to create the product model are discussed (2.2.2) and at last requirements to the product model are specified (2.2.3)

2.2.1 Discipline selection

During a brainstorm session with the involved people at the SEAD department the disciplines involved in the integrated design of the smart technologies were identified together with their priority. The result of this brainstorm is shown in Figure 2-6 and briefly described below.

1. Because both technologies rely on flow control aerodynamic analysis (AERO) is given the highest priority. External aerodynamic analysis will determine the operating conditions and performance of the smart technologies, while internal aerodynamic analysis determines the losses and preferred layout of the pneumatic system.

2. The second priority is awarded to the energy consumption (PWR) of the smart technologies, to improve efficiency it is important to keep this as low as possible.

3. By combining the first two analyses the performance (PERF) of the system can be determined. Because this will give an indication to the benefits of the smart technologies, this is awarded the third priority.

4. The fourth priority is given to the weight of the system; heavier systems will decrease the maximum payload or fuel load of the aircraft and thereby reduce its commercial revenue.

5. The fifth priority is contamination (CONTAM). This is only applicable to HLFC as debris or ice can significantly decrease its performance [Thiede, 2000].

6. The sixth is the space, it is an important factor to the supply systems, but because the concept aircraft is yet unknown it is currently awarded with a relatively low priority.

7-12. The next six disciplines are safety and reliability, structure (STRUC), control, manufacturing (MANUF), maintainability (MAINT) and cost. These disciplines go into more detail and are thought to have less influence on the conceptual design. Therefore these were given a low priority at the moment.

During this thesis the first four disciplines are worked out. The last, cost, is also incorporated because it was found to show some similarities with the weight. Although the Clean Sky JTI does not identify the TU Delft to perform external aerodynamic analysis, it is identified as one of the main disciplines to be worked out. This is to create a more autonomous product model.
**Discipline approach**

As indicated in section 2.1.2 there are two approaches to incorporate the analysis of the various disciplines. The external aerodynamic analysis requires CFD computations that are already available in existing CAA tools. The analysis will therefore be accessed using an interface to the CAA tool. The internal aerodynamic analysis and energy consumption can be roughly predicted using pneumatic handbook relations. They are grouped together under pneumatic analysis. These relations are relatively easy to use in the KBE environment and will therefore be incorporated. The cost and weight disciplines are based on rules and databases which will also be incorporated into the KBE environment.

**Smart fixed wing DEE**

With the disciplines and their approaches identified a DEE, shown in Figure 2-7, is created for the Smart Fixed Wing design of this thesis. The DEE shows how the pneumatic analysis, weight and cost disciplines are incorporated in the KBE environment. Only external aerodynamics is performed outside and thus requires an interface. Parts of the analysis data is directly fed back to the product model and is used to create parts of the geometry. Together to analysis results give an indication on the performance of the created solution.
2.2 Design Approach

2.2.2 Knowledge management steps

To create the parametric model and the analysis routines the following steps are identified. With these the knowledge for both smart technologies and the pneumatic system is managed.

1. Research the working principles, boundaries and requirements for the internal systems

2. Sketch concepts

3. Create structure to store knowledge

4. Create individual classes for the most viable concepts

5. Create analysis classes for
   a. Pneumatic Analysis
   b. External Aerodynamic Analysis
   c. Weight
   d. Cost

Figure 2.7 DEE for the Smart Fixed Wing
6. Create solutions

7. Trade-off between concepts

2.2.3 Product Model requirements

The main deliverable of this thesis is the KBE product model, therefore requirements for this model are identified. They are divided into two groups: performance (to allow easy use of the model) and documenting (to allow understanding, expanding and improvements to the model by other users).

Product model performance

- Allow .dat files as input source
- Output is easily accepted by a spread sheet program, such as Microsoft Excel
- The ability to create multiple instances of the model automatically
- Checking inputs for errors and correcting them where possible while always informing the user.
- Providing useful feedback on errors, instead of the often complex errors of the GDL software package.

Documenting

- Commenting with the Genworks YADD console on class inputs, outputs and functions
- Extra comments in the class files to explain knowledge steps
- An overall manual to describe the workings and how to operate it.
Chapter 3

Hybrid Laminar Flow Control

Hybrid Laminar Flow Control (HLFC) is a technology that combines Natural Laminar Flow (NLF) with Boundary Layer Suction (BLS). In this chapter the relevant information to this technology is gathered. The working principles behind NLF and BLS are described (3.1) and an explanation is given why a combination will be used. From this a set of requirements for the internal systems is created (3.2). Next the working principle of the internal systems is explained (3.3) and at last concepts for the system architecture are drafted and selected for implementation into the product model (3.4).

3.1 Technology principles

Hybrid Laminar Flow Control aims to reduce the friction and pressure drag of the wing surface, by preventing or delaying boundary layer transition. To understand this principle first the different types of boundary layers are addressed (3.1.1), next the principles of NLF and BLS are explained (3.1.2 and 3.1.3). At last the combination is explained (3.1.4).

3.1.1 Boundary layers

Due to the friction of the solid material of the wing surface and the air flowing over the wing a thin region of retarded flow occurs. This small layer of air that is slowed down is called the boundary layer. A typical flow velocity distribution in this boundary layer is shown in Figure 3-1.

![Figure 3-1 Velocity profile through a boundary layer [Anderson, 2000]](image-url)
In viscous flow there are two basic types, each having a different boundary layer.

1. Laminar flow, in which streamlines are smooth and regular and a fluid element moves smoothly along a streamline
2. Turbulent flow, in which the streamlines break up and a fluid element moves in a random, irregular, and tortuous fashion

The different profiles of these flow types are shown in Figure 3-2. It can be seen that the turbulent flow achieves higher velocities close to the surface. This results in an increase in friction. Also the turbulent boundary layer is generally thicker which increases pressure drag.

A standard airfoil (as shown in Figure 3-3) is characterised by a region of laminar flow near the Leading Edge (LE), a short transition region and a turbulent flow region to the Trailing Edge (TE).

![Figure 3-2 Velocity profiles for Laminar and Turbulent Flow](Anderson, 2000)

![Figure 3-3 Standard airfoil with boundary layer profile (based on Campe, 2004)]

### 3.1.2 Natural Laminar Flow

A NLF airfoil is shaped to prevent transition to turbulent flow, thereby reducing the friction and pressure drag. Figure 3-4 shows a standard airfoil and a NLF airfoil with their pressure distribution on the upper side. For the standard airfoil it can be seen that the minimum pressure occurs near the LE, behind this point a long stretch of increasing pressure occurs up till the TE. This positive pressure gradient destabilises the laminar boundary layer and thus encourages the transition from laminar to turbulent flow. The NLF airfoil however has the minimum pressure point near the TE. The negative pressure gradient before this point stabilises the boundary layer and thus encourages
the laminar layer to sustain. As a result the friction drag of an NLF airfoil is significantly lower.

Although the shape of the NLF airfoil encourages laminar flow, it is still very sensitive for disruptions. The slightest roughness of the airfoil surface causes instabilities with transition to turbulent flow as a result. Typical causes of disruptions are protruding rivets, imperfections in machining, bug spots and ice accumulations.

![Figure 3-4 pressure profile for a standard and laminar airfoil](Anderson, 2000)

Wings made from composite structure can achieve a very smooth surface, thus enabling NLF. Currently they are used in the certified personal aircraft segments, an example is shown in Figure 3-5. The high speed end of general aviation, however, does not use NLF airfoils. The main reasons for this are the following:

- Current LE devices. The junctions behind LE devices, such as slats and de-icing devices, create disruptions on the surface.
- Increased shock waves. Flying at transonic Mach numbers shockwaves occur on the upper surface of the wing. These shock waves are increased by NLF, because the minimum pressure point lies closer to the TE, which requires a more rapid flow deceleration. As a consequence the wave drag increases. [Stanewsky, et al., 2002]
- Cross-flow. Swept wings have cross-flows over the wing, this increases the instability of the boundary layer [White, et al., 2005]
- High Reynolds numbers. Boundary layer instabilities increase with the Reynolds number, thus encouraging transition.
- Ice build-up. Because these aircraft fly at high altitudes, ice build-up on LE occurs. De-icing devices often remove ice at time intervals; in the intermediate period ice is present on the LE and thus creates disruptions on the airfoil.
3.1 Technology principles

3.1.3 Boundary Layer Suction

Another method to maintain laminar flow is Boundary Layer Suction. This method uses a perforated skin to suck away the lowest (and slowest) part of the boundary layer on the wing surface. This layer is replaced by the layer directly above, having a slightly higher velocity. This effect is shown in Figure 3-6; the boundary layer thickness increases slightly but transition is prevented. The result is a decrease in friction and pressure drag. Because the boundary layer for high speed aircraft is thin, they would primarily benefit from the decrease in friction drag.

In a BLS design the suction speeds form a critical factor. Too little suction will not affect the boundary layer enough and still allow transition. Too much suction can create local 3D disturbances in the boundary layer that trigger transition [Young, et al., 2001]. It also requires more energy, decreasing the overall performance of the suction. The suction distribution, together with an internal structure to create the suction, should therefore be optimized.

Previous work

BLS was invented by Ludwig Prandtl in 1904. Since then a lot of research has been conducted to the technology. It has been applied and tested on various types of air-
and rotorcraft\(^1\). The TU Delft has also conducted research to BLS on gliders under the lead of ir. L.M.M. Boermans [2006]. This has led to the founding of Actiflow, a company that has successfully applied BLS on race and sports cars [Guns, 2006].

For the high speed end of aviation the ELFIN II project is worth mentioning. In this project BLS has been applied to the front section of the vertical tail plane of an Airbus A320. Tests have shown a reduction of 38% skin friction drag on the tail plane, resulting in fuel reductions between 1 and 1.5 % [Thiede, 2000].

### 3.1.4 Hybrid Laminar Flow Control

As explained in chapter 1.5, HLFC combines the technologies of NLF and BLS. It uses a NLF airfoil shape, thus having a decreasing pressure gradient over a large stretch of the wing. But to overcome the problems for high speed airliners, discussed at the end of section 3.1.2, the wing will also be equipped with a BLS system near the LE.

BLS in a HLFC system mainly aims to

- Counteract the destabilising effects of cross-flow over the swept wing
- Prevent transition due to instabilities at the attachment-line\(^2\) [Heeg, 1998], by applying extra high suction at the LE

A typical suction profile for HLFC is shown in Figure 3-7. The suction coefficient \(C_\Omega\) determines the suction speed \(u_s = C_\Omega u_\infty\).

If the BLS system performs as intended, the flow will reach the end of the suction area in a laminar state. From there the flow will follow the smooth surface of the middle part of the wing without suction. Because of the negative pressure gradient over this part the flow is encouraged to remain laminar. The results in the boundary layer are shown in Figure 3-8.

\[^1\] Examples are: the Vampire Aircraft by the British RAE, the B-18 by NASA, the F-94 and the X-21A by Northrop [Braslow, 1999].

\[^2\] The attachment-line is the front region of the wing where the free-stream air flow first meets the wing surface.
3.2 HLFC Requirements

To create the required suction profile an internal system is needed. To keep this system efficient, low risk and affordable the following requirements are identified

- Minimum internal losses. Energy losses due to the internal system will decrease energy savings realised by the extended laminar flow.
- Flow regulation. The system should be able to create the suction distribution as close as possible to the specified distribution.
- Low complexity. A simple systems layout will increase reliability, decrease risks and costs.
- Minimum weight. Because the system does not affect the take-off and landing performance, any weight introduced by the system will decrease the maximum payload of the aircraft.
- Minimize cost. To make the technology affordable for new aircrafts to use the costs of the whole system should be kept to a minimum.
- Leave space for other devices. Because the nose of the wing is equipped with multiple systems, e.g. slats and de-icing devices, the BLS system will have to leave enough space to accommodate them.

3.3 HLFC internal systems

During many previous applications of BLS the suction distribution was created using a complex system of valves, pipes and pumps. The ALTTA project, set up in February 2000, had multiple targets; one of them was aimed to reduce the complexity of a HLFC system for a vertical tail plane of an Airbus A320 [ALTTA Synthesis Report, 2003].
A simplified system, shown in Figure 3-9, uses a perforated surface, a set of skin-chambers, a suction duct, a pump and an air outlet on the wing. The flow regulation in this system is fixed to one setting, aside from the pump which can be turned on or off. It is designed to create a specific suction distribution during cruise conditions. In other flight conditions it is not able to deliver the same distribution, or a distribution that would optimal for these conditions. As result the performance of the HLFC system will decrease.

The flight manoeuvres for the SFWA types initially aimed at [Smart Fixed Wing Aircraft - Integrated Technology Demonstrator, 2008] have a relative long cruise state. This, together with the simplicity of the system, makes the ALTTA layout principle very suitable for the SFWA, except for the size of the manifold. Unlike the vertical tail plane, the nose of transport aircrafts wing houses various leading edge devices, such as slats and associated hydraulics and wiring. Therefore a different suction duct or manifold is needed, that leaves enough space for other equipment. To maintain a constant pressure inside the thinner manifold more connection points to one or more pumps might be needed. Figure 3-10 shows an example of the HLFC system on an aircraft wing. A 3D view of a suction section is shown in Figure 3-11. In this figure the manifold is a thin section following the shape of the skin-chamber and is connecting to the main-duct with two connectors.
Figure 3-11 Example of the suction system

Figure 3-12 shows a schematic overview of the flows through the system. The impacts on the flow are explained below. Starting with the effects of the perforated surface (3.3.1) to the flow regulation by the skin-chambers and the manifold (3.3.2) and ending with the effects of inlet from the manifold to the pipe-system (3.3.3). The effects of the pipes, pump and outlet will be discussed in Chapter 5.

Figure 3-12 schematic view of the flow through the system, with annunciations

3.3.1 Perforated surface
To create the right suction distribution, a set of skin-chambers is placed beneath the perforated surface. The pressure difference between the air in the boundary layer and the air in the skin-chambers determines the suction velocity through the perforated surface, estimated by the Goldstein Method (3-1) [Seitz, 2010]

$$\Delta p_s = A \frac{\mu_s}{\mu_0} u_s + B \frac{D_s}{\rho_0} u_s^2$$  \hspace{1cm} (3-1)

Where $\rho$ stands for the density and $u$ the suction speed. $A$ and $B$ are constants depending on the hole diameters and the material of the perforated skin. The fraction $\mu_s / \mu_0$ follows from the Sutherland formula (3-2).

$$\frac{\mu}{\mu_0} = \left( \frac{T}{T_0} \right)^{\frac{3}{2}} \frac{T_0 + 110}{T + 110}$$  \hspace{1cm} (3-2)

Where $T$ stands for the Temperature in Kelvin.

A schematic overview of this section is shown in Figure 3-13 together with the relations between the pressure and the assumptions for the density and temperature [Seitz, 2010].

$^1$ Typical values for A and B are 14000 and 27000 Ns/m$^3$ respectively [Seitz, 2010]
3. Hybrid Laminar Flow Control

3.3.2 Skin-chambers and manifold

Because both the suction distribution and the outside pressure vary along the suction area, each skin-chamber will need a specific pressure inside. The amount and distribution of skin-chambers determines to what extend the required suction-distribution is realised.

Behind the skin-chambers lies a manifold, which is assumed to have a near uniform pressure inside. This pressure must be lower than the pressure of all skin-chambers, thus limited to the chamber with the minimum pressure. The air inside each skin-chamber is sucked into the manifold through an orifice. The diameter of this orifice determines the pressure drop between both sides, predicted by the Borda-Carnot equation (3-6). This diameter is used to create the required pressure in each skin-chamber individually.

\[
\Delta p_m = \frac{\rho_{sc}}{2} u^2 \left( \frac{S_s}{\alpha S_h} - 1 \right)^2
\]

(3-6)

\[
\alpha = 0.6 + 0.4 \left( \frac{S_h}{S_s} \right)^2
\]

In this relation \( S_h \) stands for the area of orifice and \( S_s \) for the area of the sheet, shown in Figure 3-14.

A schematic overview of this is given in Figure 3-15 together with the equations, where \( \dot{m} \) stands for the mass-flow. Note that every skin-chamber has its own pressure, density and temperature. Because multiple airflows, each having different conditions, come together in the manifold a weighted average over the mass-flow is taken for \( \rho \) and \( T \).
3.3.3 Pipe-system inlet

After entering the manifold the air is sucked away by the pump through a pipe-system. At the entry of the pipe pressure losses occur, these are dependent on the shape and the size of the inlet. The pressure drops to be expected for two types of inlets are shown below, where $S$ refers to the area [Idelchik, 1994].

Table 3-1 pressure drop relations for various inlet shapes

<table>
<thead>
<tr>
<th>Inlet Shape</th>
<th>Pressure Drop Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta \leq 45^\circ$</td>
<td>$\Delta p = \frac{0.5 \left( 1 - \frac{S_{pipe}}{S_m} \right) \rho_m u_m^2}{\left( \frac{S_{pipe}}{S_m} \right)^2}$ (3-11)</td>
</tr>
<tr>
<td>$\theta &gt; 45^\circ$</td>
<td>$\Delta p = \frac{0.8 \sin \frac{\theta}{2} \left( 1 - \frac{S_{pipe}}{S_m} \right) \rho_m u_m^2}{\left( \frac{S_{pipe}}{S_m} \right)^2}$ (3-12)</td>
</tr>
</tbody>
</table>

3.4 HLFC Concepts

With this system layout principle, various concept layouts are sketched. The sketches are all chord section views. The span views will be discussed in Chapter 5, because it is primarily affecting the pneumatic-system. The concepts with a short description are listed in Table 3-2. The number of skin-chambers, pipe-connections, and main ducts (if multiple) as well as the size of each part can vary in the product model, due to the generic capabilities of KBE.
### Table 3-2 Concepts

<table>
<thead>
<tr>
<th>Concept #</th>
<th>Sketch</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1. | ![Base layout](image1.png) | - One single manifold at constant offset  
- One pipe-connection to a single main duct |
| 2. | ![Base layout, multiple connections](image2.png) | - One single manifold at constant offset  
- Multiple pipe-connections to one single main duct |
| 3. | ![Double layer](image3.png) | - 2 manifold layers at constant offset  
- One pipe-connection to a single duct (could also be multiple) |
| 4. | ![Single layer, Divisions](image4.png) | - Multiple manifolds along chord direction at constant offset  
- All connected by individual pipe connection to a single main duct |
| 5. | ![Single layer, Divisions, Multiple main ducts](image5.png) | - Multiple manifolds along chord direction at constant offset  
- Each connected via a connection pipe to its own main duct |
| 6. | ![No manifold](image6.png) | - No manifold  
- Each skin-chambers is connected to a single main duct via individual pipe-connections |
| 7. | ![No manifold, multiple main ducts](image7.png) | - No manifold  
- Skin-chambers are connected to multiple main ducts via single pipe connections |
| 8. | ![Manifold, no pipe-connections](image8.png) | - Single manifold, with constant offset  
- No pipe-connections, main duct lies against/in the manifold |
| 9. | ![Manifold against front spar](image9.png) | - Manifold uses front spar as side  
- No pipe-connections, main duct lies against/in the manifold |
| 10. | ![Manifold with aerodynamic shape](image10.png) | - Manifold is shaped to guide airflow with minimum losses to the main duct  
- No pipe-connections |
To save development time of the product model, a priority analysis is made between these concepts. The selected concepts are modelled in the KBE-system, which will eventually allow a more detailed and fair trade-off between the selected concepts. For the priority analysis the concepts are graded against the requirements set out in section 3.2 with one extra topic: the ease of implementation into KBE. In general having more parts increases the complexity and is therefore negatively awarded for complexity. Flow regulation is thought to be better performed when more regulating steps are present, e.g. with a double layer. More flow regulating steps and sharp angles are judged to increase internal losses. The final score in Table 3-3 represents the sum of all grades with a uniform weight. The last column indicates if the concept is selected to implement into the product model.

Although some concepts receive a positive or neutral final score (i.e. concept 8 and 9) they will not be implemented into the KBE product model. Concept 9 is very similar to the design of the ALTTA project and it exceeds the available space in the nose of the wing. It would leave no space for other devices and is therefore considered not applicable. Concept 8 was finally discarded because it shows less similarity with the other concepts that are implemented. Therefore relatively more work would have to be performed on this concept, while at the same time, the expected effectiveness of the concept is lower than that of the others.

<table>
<thead>
<tr>
<th>Concept #</th>
<th>Complexity</th>
<th>Flow regulation</th>
<th>Internal losses</th>
<th>Space left</th>
<th>Weight</th>
<th>Cost</th>
<th>KBE</th>
<th>final score</th>
<th>implemented</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+</td>
<td>o</td>
<td>o</td>
<td>++</td>
<td>+</td>
<td>+</td>
<td>++</td>
<td>↑</td>
<td>☑</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>↑</td>
<td>☑</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>+</td>
<td>-</td>
<td>0</td>
<td>o</td>
<td>+</td>
<td>+</td>
<td>↑</td>
<td>☑</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>+</td>
<td>0</td>
<td>o</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>↑</td>
<td>☑</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>++</td>
<td>+</td>
<td>0</td>
<td>0</td>
<td>+</td>
<td>+</td>
<td>↑</td>
<td>☑</td>
</tr>
<tr>
<td>6</td>
<td>--</td>
<td>o</td>
<td>-</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>↓</td>
<td>☓</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>↓</td>
<td>☓</td>
</tr>
<tr>
<td>8</td>
<td>++</td>
<td>-</td>
<td>0</td>
<td>o</td>
<td>+</td>
<td>0</td>
<td>+</td>
<td>↑</td>
<td>☓</td>
</tr>
<tr>
<td>9</td>
<td>++</td>
<td>-</td>
<td>--</td>
<td>++</td>
<td>++</td>
<td>0</td>
<td>+</td>
<td>↑</td>
<td>☓</td>
</tr>
<tr>
<td>10</td>
<td>--</td>
<td>+</td>
<td>++</td>
<td>-</td>
<td>o</td>
<td>--</td>
<td>--</td>
<td>↓</td>
<td>☓</td>
</tr>
</tbody>
</table>

negative -- - o + ++ positive

above average
average
below average

Implemented not implemented
Chapter 4

Fluidic Actuated Flow Control

Fluidic Actuated Flow Control (FAFC) is at a lower Technology Readiness Level (TRL) compared to HLFC. This requires a different approach for the KBE conceptual design. An approach more focused on the functions rather than the geometry. The actuators for FAFC are still in development; up till now only a few prototypes have been built.

This chapter will gather the knowledge available for FAFC starting with the working principle (4.1), followed by the requirements and operating conditions of the actuators in development (4.2). At the end a two actuators are selected to be incorporated into the product model (4.3).

4.1 Technology Principle

FAFC is a technology that exists in different forms; all apply flow control by injecting energy into the boundary layer. They are primarily aimed at the following targets [Jones, et al., 2003] [Kim, et al., 2006]:

- Increase $C_{L,max}$
- Delay stalling
- Delay buffeting\(^1\)

In Chapter 3 it was shown that a positive pressure gradient causes the boundary layer to become turbulent. Under high angles of attack this gradient is larger and drains the boundary layer of more energy, until there is not enough left for the flow to stay attached to the wing surface. Separation occurs and the lifting capabilities of the airfoil after the separation point decrease dramatically. Stalling occurs when the angle of attack increases beyond a certain point such that the total lift starts to decrease. The development of the boundary layer with separation is shown in Figure 4-1.

\(^1\) Buffeting is a high-frequency instability in the airflow caused by flow separation, among other factors. It creates unsteady loads on the wing which may cause fatigue problems.
Reenergising the boundary layer at the separation point stimulates it to stay attached and thus increases lift and decreases buffeting effects. When applied at the separation point for the critical angle of attack stalling is delayed.

Two main technologies for reenergising are identified:

- Continuous jets / pulsating jets (4.1.1)
- Synthetic jets (4.1.2)

Other methods are known, such as a suction slot [Stanewsky, et al., 2002], but these are not given priority by the Clean Sky JTI and therefore not discussed further in this thesis.

### 4.1.1 Continuous jets / pulsating jets

To reenergise the boundary layer this technology has a slot on the wing surface that ejects air at a high velocity. This air mixes with the air in the boundary layer and as a result increases the average speed in that layer. The method requires relatively high mass flow of air through the slots. To minimize this mass flow research has been and is still performed into pulsating jets. Instead of continuous blowing these jets blow at high frequency intervals. Due to the inertia of the airflow, a short amount time is needed for the flow to react to new conditions [Jones, et al., 2003]. If the pulses follow each other within this time the boundary layer cannot react and will thus remain attached. The result is a higher lift increase at the same mass-flow rate, shown in Figure 4-2, where the duty cycle refers to the slot opening time per pulse. This figure also shows the existence of a direct relation between mass-flow rate and the produced lift (4-1).

\[
\Delta C_L = f(\dot{m}) \tag{4-1}
\]
4.1.2 Synthetic jets

The working principle of a synthetic jet involves complex aerodynamics and is still a topic of research. A synthetic jet consists of an oscillating surface inside a cavity under the wing surface, shown in Figure 4-3. This oscillation causes suction and blowing alternately through an opening in the surface. As a result vortex pairs occur that mix with the boundary layer [Smith, et al., 2001]. When applied at the separation regions on the wing this is found to delay separation and with that increase lift, delay stalling and delay buffeting. Synthetic jets are also known to have a shape changing effect called ‘Virtual Aero Shaping’ [Kim, et al., 2006]. This effect could also delay separation and is shown in Figure 4-4.

Because this technology alternates suction with blowing it does not require a net mass-flow. This makes it easier to apply the technology at different places on the wing, as no pneumatic system is required.
4.2 Actuator requirements

Within the CleanSky JTI various actuators, both pulsating jets and synthetic jets, are under development. Their requirements for the supply system are therefore not precisely specified, however, rough figures for a few types of actuators are available and listed in Table 4-1. The mass flow and pressure for the pulsating jets could be driven by either a pump or with bleed-air from the engines.

Table 4-1 Fluidic actuator specifics [Schueller, et al., 2010]

<table>
<thead>
<tr>
<th>Actuator</th>
<th>Mass Flow</th>
<th>Pressure</th>
<th>Frequency</th>
<th>Space</th>
<th>Current / Voltage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONERA CTY-Valve pulsating jet</td>
<td>2 g/s</td>
<td>5 bar</td>
<td>Up to 1kHz</td>
<td>61.5 x 30 x 11.5mm</td>
<td>0.25A, 150V</td>
</tr>
<tr>
<td>ONERA LS-Valve pulsating jet</td>
<td>10 g/s</td>
<td>4 bar</td>
<td>Up to 1kHz</td>
<td>120 x 80 x 18.5mm</td>
<td>0.25A, 150V</td>
</tr>
<tr>
<td>INCAS synthetic jet</td>
<td>-</td>
<td>-</td>
<td>1-1.5 kHz</td>
<td>10 x 20 x 15mm</td>
<td>100V</td>
</tr>
<tr>
<td>ONERA synthetic jet</td>
<td>-</td>
<td>-</td>
<td>1kHz</td>
<td>20 x 20 x 10mm</td>
<td>150V</td>
</tr>
</tbody>
</table>

4.3 Actuator Selection

From the requirements in the table above it can be seen that pulsating jets require an internal air supply system. This system has, similar to the HLFC system, a large influence on the overall performance. Also shown, for both types of actuators, is the need for an electrical supply system.

The performance of the pneumatic system is in comparison to the electrical system more geometry related and is given a higher priority at this moment of the design process. Because time was limited for this thesis, it was therefore decided to only work
out the pneumatic system for pulsating jets. Provisions are made for other supply systems to be added at a later time, such as the electrical system.

The ONERA CTY-Valve and ONERA LS-Valve, shown in Figure 4-5, are both selected to be incorporated into the product model. These actuators have a fairly simple shape, with a number of holes on the top. Once fully developed they will be complete parts with fixed interfaces to the supply systems. These interfaces are at the moment still to be determined.

Figure 4-5 Drawings with dimensions for both actuators [Schueller, et al., 2010]
Chapter 5

Pneumatic system

Both HLFC and FAFC specify required flow conditions at their connection points to the pneumatic system. To deliver these flow conditions a pneumatic system is needed consisting of a set of pipes, junctions, pump(s) and an inlet (FAFC) and outlet (HLFC) slot. A typical layout for such a system is shown in Figure 5-1. Because the flow conditions are completely specified at the connection points, some control is required from the pneumatic system, either with active (control valves) or passive flow control (fixed geometry).

The knowledge gathered for the pneumatic system is described in this chapter. Starting with the pneumatic relations and assumptions that apply (5.1). Followed by the requirements for the system (5.2) and at last the various concepts sketches are described and selected for implementation (5.3).

Figure 5-1 Typical pneumatic system layout
5.1 Pneumatic principles

Extensive research to flows in pneumatic systems has been performed in the past and is still performed. This research has produced many relations that predict the impacts of geometry and surfaces on the flow. It has also led to the development of CAA tools that in most cases use relations or Computational Fluid Dynamics (CFD) to analyse the flows.

CFD computations are often more accurate, but they require extensive amounts of computation time. A problem that was encountered by Boonacker [2009], who had to use database of pre-evaluated pipe sections to reach acceptable computation times. Also CFD only analyses specified geometry, to apply passive control iterations are needed. This increases the computation times dramatically.

The accuracy of handbook rules and relations found with experimental data is generally lower compared to CFD and they limit the shapes available for design. Their computation times are however very short. Also their inverses can be used to determine geometries for passive flow control.

To evaluate multiple parametric concepts, with each having their own variable settings, fast computation times are preferred in the conceptual stage. Handbook rules and relations found with experiments are therefore chosen as the main source of knowledge for the pneumatic system. The relations concerning the parts encountered in the layout of Figure 5-1 are discussed in this section, starting with the general pneumatic relations (5.1.1), followed by those for straight (5.1.2) and bend pipe segments (5.1.3). Next the relations to predict the flows in junctions are addressed (5.1.4). After that the general relations for pumps are discussed (5.1.5) followed by the relations for the air inlet/outlet slot (5.1.6).

5.1.1 General pneumatic relations

When predicting the flow in a pneumatic system the equations that describe the state of the flow are the perfect gas law (5-1), the continuity equation (5-2), and the energy equation (5-3) [Lui, 2003].

\[ p = \rho RT \]  \hspace{1cm} (5-1)

\[ \rho u_1 S_1 = \rho u_2 S_2 \]  \hspace{1cm} (5-2)

\[ \frac{u_1^2}{2g} + \frac{p_1}{\rho g} + z_1 = \frac{u_2^2}{2g} + \frac{p_2}{\rho g} + z_2 + h_L - h_p \]  \hspace{1cm} (5-3)

Where subscript 1 and 2 refer to the upstream and downstream situation, \( S \) is the cross-sectional area and \( z \) the elevation. Because there is no significant elevation on an aircraft wing this can be ignored. \( h_L \) stands for the head loss along the pipe section, depending on the shape and material. \( h_p \) stands for the pump head, i.e. the energy per unit weight of fluid added by a pump between location 1 and 2.
For a pipe-section without a pump or elevation difference, equation (5-1) can be rewritten as equation (5-4) which shows the relations between the pressure drop ($\Delta p$) and the head loss.

\[
h_L = \frac{\Delta p}{\rho g} + \frac{u_1^2 - u_2^2}{2g}
\] (5-4)

Because the pipes of the pneumatic system are relatively short and the flow velocity inside is expected to remain below Mach 0.3 incompressible flow is assumed. This simplifies the relations and prevents iterative processes for pressure drop calculations.

5.1.2 Relations for straight pipe segments

In a straight pipe segment with a constant diameter the energy losses are mainly caused by friction with the inner surface of the pipe. If isentropic flow\(^1\) is assumed the pressure drop for smooth-walled pipes can be approximated by the Darcy-Weisbach equation [Barber, 1989].

\[
\Delta p = \frac{\rho f L u^2}{2D_h}
\] (5-5)

Where $f$ is the friction factor, $L$ is the length of the pipe and $D_h$ is the hydraulic diameter of the pipe, defined by $D_h = 4S/P$ where $P$ the wetted perimeter.

Due to the relatively short and often bend segments, the flow in a HLFC and FAFC system is primarily turbulent. The friction factor of a turbulent flow can be found using the Moody Chart shown in Figure 5-2 [Barber, 1989]. With the relative roughness being the ratio of the mean roughness height of the pipe to the pipe diameter or $\frac{\varepsilon}{D}$.

---

\(^1\) Isentropic flow assumes that no heat is transferred to the pipes or other surrounding objects. The Temperature of the flow will thus remain constant.
The friction factor for the turbulent region is approached by the Colebrook equation (5-6). This is an implicit equation that combines experimental studies of turbulent flow in smooth and rough pipes. Due to the implicit nature of the equation, solving requires iteration. The accuracy of this equation is ±2% [Darcy friction factor formulae, 2010].

\[
\frac{1}{\sqrt{f}} = -2 \log_{10} \left( \frac{e/D}{3.7} + \frac{2.51}{Re\sqrt{f}} \right)
\]  

(5-6)

### 5.1.3 Relations for bend segments

Bends in pneumatic systems cause directional changes to the flow, which lead to pressure losses. A directional change causes the flow to separate and experience adverse pressure gradients, as shown in Figure 5-3. This creates additional friction and pressure losses. These losses are often defined by an equivalent length \( L_e \) which substitutes \( L \) in equation (5-5) to determine the pressure loss in the bend.

According to the work of Crawford et al. [2003] the pressure drop through a 90° bend can be predicted by using the summation of equation (5-7), which predicts the effects of the adverse pressure gradient, and equation (5-8), which predicts the effects of separation.

\[
\left( \frac{L_e}{D} \right) = 19.833 \left( 0.003625 + 0.038 \left( Re \left( \frac{D}{2R_c} \right)^2 \right)^{-0.25} Re^{0.25} \left( \frac{2R_c}{D} \right)^{0.5} \right)
\]  

(5-7)

\[
\left( \frac{L_e}{D} \right) = 22.2126 \left( Re \left( \frac{D}{2R_c} \right)^2 \right)^{0.7888} \left( Re \left( \frac{2R_c}{D} \right)^{0.71438} \right)
\]  

(5-8)

With \( R_c \) as the radius of the bend, indicated on the right side of Figure 5-3. These relations where verified with experimental data with Re numbers up to \( Re = 3 \times 10^5 \) and showed a spread of +3 to -2 per cent. Preliminary CFD validation has also shown a good correlation for \( R_c/D \) values greater than 10.
To determine the pressure drop in non 90° bend the linear relation (5-9) is used.

\[
\left( \frac{L_e}{D} \right)_\beta = \beta \left( \frac{L_e}{D} \right)_{90^\circ}
\]  

(5-9)

Where \( \beta \) is the bend angle and \( \left( \frac{L_e}{D} \right)_{90^\circ} \) the pressure drop for the 90° bend.

Although other relations are known to provide a better prediction for losses in non 90° bends [Crane, 1988], it was still decided to use this linear relation. The main reason for this is the applicability for the parametric KBE model; this is explained in more detail in section 7.4.3.

### 5.1.4 Relations for Junctions

Pressure losses in junctions are mainly dependant on the directions of the flow relative to each other, the mass flow and the size of each branch. These losses can be predicted with the horizontal momentum equation [Basset, et al., 2001], which needs to be solved separately for junctions with joining and separating flows.

#### Joining flows

In their work Basset et al. [2001] derive the horizontal momentum equation for junctions where the cross-sectional area of the pipe A is equal to pipe C; shown in Figure 5-4. The dashed lines in this figure indicate the control volume used for the horizontal momentum equation. The relation they found for this type of junction was verified in their work against experimental data. During this validation it was found that an angle correction factor, where \( \theta^* = \frac{3}{4} \theta \), improves the predictions significantly. The theory behind this correction factor is that the inflow into the junction segment for the lateral leg would not be parallel to the leg itself. With this correction the results were considered: “surprisingly good given the simplicity of the analysis” [Basset, et al., 2001].

![Figure 5-4 Junction with equal cross-sectional areas for pipe A and C](image)

For this thesis the horizontal momentum equation has been extended such that the cross-sectional area may change at the junction. Such an area changing junction can provide a more constant flow through the pneumatic system. To do derive the horizontal momentum equation for this situation, the junction configuration is shown
in Figure 5-5. The horizontal momentum equation for the control volume bounded by the dashed lines is now given by

$$p_A S_A + p_B S_B \cos \theta - p_C S_C - R_w \sin \theta + R_x = -\dot{m}_A u_A - \dot{m}_B u_B \cos \theta + \dot{m}_C u_C$$  \hspace{1cm} (5-10)

Where $R_w$ and $R_x$ represent the wall forces. If the pressure is assumed vary linearly between the branches, then $R_w$ is defined by

$$R_w = \frac{p_B + p_C}{2} \frac{S_B}{\tan \theta}$$  \hspace{1cm} (5-11)

And $R_x$ by

$$R_x = \frac{p_A + p_B}{2} (S_C - S_A)$$  \hspace{1cm} (5-12)

Figure 5-5 Junction overview with unequal pipe diameters

With the assumption, used by Basset et al. [2001] and originally made by Benson et al. [1964], that the static pressure in two pipes that have flows towards a junction must be equal, equation (5-10) becomes

$$p_C \left( S_C + \frac{S_B}{2} \cos \theta \right) - p_A \left( S_C + \frac{S_B}{2} \cos \theta \right) = -\dot{m}_A u_A - \dot{m}_B u_B \cos \theta + \dot{m}_C u_C$$  \hspace{1cm} (5-13)

The correction factor $\theta^*$, discussed above, will be used for this junction configuration as well. Equation (5-13) gives a clear relation to predict the pressure drop in the unequal area junction which will be used in the pneumatic system during this thesis. Because this relation had not been verified, the results will be analysed in Chapter 9 by comparison with a CAA tool.

For T junctions where the flow exits through the lateral branch, the relations of Basset et al. gave very poor results (differences with experimental data of more than 50%). They were also only verified for a 90° angle. Fortunately this type of junction is rarely needed in the supply systems for both smart technologies. As the results were poor to start with, minimum effort is taken to extend the horizontal moment equation for these unequal area junctions. It has been restricted to a branch angle of 90°, shown in Figure 5-6. When the horizontal momentum equation is derived for the control volume in the direction of pipe B it becomes:
Because the pressure at the end of pipe A must be equal to pipe B, it is often necessary to lower the pressure in one of these pipes, e.g. by decreasing its diameter or with an obstruction as shown in Figure 5.7. For the pressure drop through this obstruction can be predicted with the Borda Carnot Equation (3.6). The method used for decreasing the diameter is described in more detail in Appendix E.

**Separating flows**

For separating flows Basset et al. [2001] use two control volumes and two horizontal momentum equation to predict the pressure drops in the junction: one for each pressure drop, shown in Figure 5.8. To solve both momentum equations the following assumptions were made in their research.

For both cases
- The mean pressure \( p_c^* \) on the dashed line from C to \( C_0 \) is assumed to be according to equation (5.15), where \( p_{0C} \) is the stagnation pressure.

\[
p_c^* = \frac{p_c + p_{0C}}{2} = p_c + \frac{1}{4} \rho u_c^2
\]  

(5.15)

For the pressure drop from C to A:
- The cross-sectional area of pipe A and pipe C are assumed to be equal.

For the pressure drop from C to B:
- The flow speed entering the control volume is equal to \( u_c \) and its direction deviates from the horizontal by \( \theta /4 \).
As the flow enters pipe B it is assumed to separate from the wall.

Figure 5-8 Control volumes for separating flow, the control volume in the right figure is the area between the points C-R-B-B-R-C₀.

The control volumes for a T shaped junction where the flow enters through pipe B are shown in Figure 5-9. For this junction the same assumptions were used by Basset et al. [2001] as for the pressure drop from pipe C to pipe B described above. In contrast to the T shaped junction with joining flows the relations for the separating flows were verified with much more success.

Figure 5-9 Control volume in a T shaped junction for the pressure loss from pipe B to pipe C

With the horizontal momentum equations for each control volume, the pressure drop can be rewritten into pressure loss coefficients (K factor). The definition of the K factor is given by (5-16).

\[
K = \frac{(p_{up} + \frac{1}{2} \rho u_{up}^2) - (p_{down} + \frac{1}{2} \rho u_{down}^2)}{\frac{1}{2} \rho u_{up}^2}
\]  

(5-16)

Where subscript \( up \) refers the pipe upstream, \( down \) to the pipe downstream. The pressure loss coefficient for the separating junctions are listed in, where \( q \) is the mass flow ratio \( (\dot{m}_{down}/\dot{m}_{up}) \) and \( \psi \) the area ratio \( (S_{down}/S_{up}) \). For example for \( K_1 \), \( q = \dot{m}_A/\dot{m}_C \) and \( \psi = S_A/S_C \).
Because the pressure loss coefficients $K_2, K_3$ and $K_4$ do not assume equal cross-sectional area’s these relations do not have to be adjusted to allow an un-equal area junction of the type shown in Figure 5-10. $K_1$ does assume that $S_C = S_A$ and therefore has to be adjusted. Assuming the mean pressure between points $C$ and $C_0$ is still according to equation (5-15), the momentum equation becomes

$$p_C q S_C - p_A S_A + \left( p_C + \frac{1}{4} \rho u_C^2 \right) (S_A - q S_C) = \dot{m}_A u_A - q \dot{m}_C u_C$$

(5-21)

Rewriting this to a pressure loss coefficient results in equation (5-22), which will replace equation (5-17) for the pressure loss predictions in this thesis work.

$$K_1 = \frac{q^2}{\psi^2} - \frac{3}{2} \frac{q}{\psi} + \frac{1}{2}$$

(5-22)

Again because these relations have not been verified, the outcomes will therefore be analysed in Chapter 9 by comparison with a CAA tool.

Although the pressure in pipe A at junction shown above does not have to be equal to that in pipe B it will not differ much. If a larger pressure difference is needed one of the pipe diameters should be decreased or an obstruction should be place, similar to joining junction.
5.1.5 Relations for Pumps
To maintain flow in the HLFC or FAFC system usually a pump is required. The head delivered by this pump must overcome out the head difference between inlet and exhaust plus the head lost in the system, equation (5-23):

\[ h_p = h_{\text{exhaust}} - h_{\text{inlet}} + h_L \]  

(5-23)

Common pump types to deliver this head to airflow are axial and radial pumps, shown in Figure 5-11. The performance of these pumps, and most other pump types, varies with the discharge rate (Q) of the system. The pump head delivered decreases with an increasing discharge rate, because the pump is unable to inject the same amount of energy per unit of air flow at higher speeds. The plot in Figure 5-12 shows the performance difference at different discharge rates.

Just as the pump head is related to the discharge rate, so is the head loss, which follows from equation (5-4). In Figure 5-13 the head loss and pump head are both plotted against the discharge rate. The operating point of the system is found at the intersection of both curves.

Figure 5-11 Schematic view of an axial pump (left) and a radial pump (right)

Figure 5-12 Typical pump performance graph
Because both HLFC and FAFC have specific inflow and outflow requirements, the discharge rate for both systems is no longer a variable. This means that they have a single specific operating point unrelated to the pump characteristics. Ideally this operating point is located exactly at the operating point in Figure 5-13, but more likely is the situation shown in Figure 5-14. In this case the pump is over dimensioned and delivers more head than needed. To create the correct flow conditions in the system this over-performance is corrected, e.g. by means of a valve. The losses introduced by this decreases the efficiency of the pump significantly.

The pump generally receives its power input via electricity. To prevent a large power surge on the electrical network during start-up of the pump, the ALTTA project suggests using bleed-air to start up the pump [ALTTA Synthesis Report, 2003]. Once the pump would operate at 83% of its nominal speed the electrical motor would take over. However with the new aircraft often being equipped with high capacity electrical networks, this might not be needed anymore.

### 5.1.6 Relations for the inlet/outlet slot
Where the pneumatic system for HLFC needs an outlet slot (Figure 5-15), the FAFC needs an inlet slot (Figure 5-16). The conditions at their locations influence the efficiency of the overall system. An outlet into a low pressure area requires less pump
power than a high pressure area and the opposite holds for an inlet slot. This follows from equation (5-23).

For an outlet slot equation (5-23) suggests that the head of the flow should be equal at the outlet as the head of the surface flow. This however does not have to be the case; the speed of the surface flow can also be used to speed up the flow from the outlet slot. This introduces extra drag on the aircraft, but is likely to be more efficient compared to using a more powerful pump.

The inlet slot uses all the head of the surface flow to minimize the required pump power. The slots increase the frontal area and disrupts the boundary layer, this to introduces extra drag to the aircraft.

Both the inlet and outlet slot thus have a negative effect on the aircraft aerodynamics. The first effect, related to the energy of the flow, can be calculated with the energy equation (5-3) and will be taken into account. The second effect, related to the disruption of the boundary layer, will be neglected during this thesis. Both HLFC and FAFC stretch out over a wide section of the wing, whereas the inlet/outlet slots have only a small width. Their secondary effects therefore occur local and less influential on the overall performance.

5.2 Requirements and Criteria

To supply the smart technologies with suction or blowing in an efficient way the following requirements and criteria are identified.

- Minimum internal losses. Energy losses due to the internal system will increase the power consumption of the system.
- Minimum pump over-performance.
- Flow regulation. The pneumatic system should be able to create to required flow conditions at the connection points to HLFC or FAFC.
- Minimum weight. Weight introduced by the system will decrease the maximum payload of the aircraft.
- Minimum cost. To make the technology affordable for new aircrafts the cost of the whole system should be kept to a minimum.
- Leave space for other devices. Because the nose and other parts of the wing are equipped with multiple systems, e.g. slats and de-icing devices, the pneumatic system must leave enough space to accommodate them and possibly route around them.
5.3 Pneumatic Concepts

For the technologies presented in Chapter 3 and Chapter 4 various types of layouts for the pneumatic system are sketched. These concepts together with a short description are presented in Table 5-2. These sketches are based upon a Dassault Falcon 7x aircraft equipped with an HLFC system. The objects in this layout are named in Figure 5-17, which shows a FAFC system as well. It can be seen that although the flow direction is in the opposite direction, the same layout can be used for both systems. The suction panels and outlet slot in the HLFC system are replaced with an actuator section and inlet slot for the FAFC system. Because a parametric model is created for these technologies, parameters such as the number of divisions and pumps can vary.

![HLFC and FAFC diagrams]

Figure 5-17 layout description with the difference between HLFC and FAFC

<table>
<thead>
<tr>
<th>Concept #</th>
<th>Sketch</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1.        | ![Sketch 1](image1.png) | Pump per section, root direction  
- Multiple pumps  
- Multiple short pipes  
- Multiple outlet slots on the wing  
- Pump positioned in root direction of the section |
| 2.        | ![Sketch 2](image2.png) | Pump behind section  
- Multiple pumps  
- Multiple short pipes  
- Multiple outlet slots on the wing  
- Pump positioned in the middle of a section. |
### 5.3 Pneumatic Concepts

<table>
<thead>
<tr>
<th>Concept #</th>
<th>Sketch</th>
<th>Description</th>
</tr>
</thead>
</table>
| 3.        | ![Sketch](image1.png) | Pump per section, wing tip direction  
- Multiple pumps  
- Multiple short pipes  
- Multiple outlet slots on the wing  
- Pump positioned in wing tip direction of the section |
| 4.        | ![Sketch](image2.png) | Pumps in series  
- Multiple pumps  
- Single outlet  
- Single pipe |
| 5.        | ![Sketch](image3.png) | Pumps parallel  
- Multiple pumps  
- Single or multiple outlet  
- Multiple pipes |
| 6.        | ![Sketch](image4.png) | Single pump, multiple pipes  
- Single pump  
- Single outlet  
- Multiple pipes connected with a diffuser to the pump |
| 7.        | ![Sketch](image5.png) | Single pump  
- Single pump  
- Single outlet  
- Single pipe |
5. Pneumatic system

From the architecture concepts of the smart technologies it becomes clear that concept 5 and 6 are only useful for HLFC because this technology requires different flow condition in chord wise direction.

Similar to Chapter 3 these concepts are graded on the criteria set out in section 5.2 and their ease to implement in the KBE system. With this their priority to be implemented in the product model is determined, the results are shown in Table 5-3.0

In general having more parts is considered to increase complexity. The number of pumps used has implications on multiple criteria. To prevent large over-performances of the pumps, each has to be specifically designed thus increasing the cost. From a safety point of view, multiple pumps provide more safety; one pump failure will not cause the complete system to malfunction. The internal losses also benefit from multiple pumps; with a single pump all flows need to be gathered together introducing more losses. From a KBE perspective concept 4 and 6 receive a lower score due to their different pipe to pump to in/outlet configuration.

The concepts selected are implemented into the product model which will allow a more detailed and fair trade-off. Concept 4 has not been implemented even though it receives a generally good score. This decision is based on its easy to implement in the KBE system, because it has a different layout relatively more work would have to be performed to model it.

On top of the selected concepts, during modelling it was found that combinations of the concepts could be created relatively easy. For example the pump(s) of concept 5 and 7 can also be placed in the middle or at the tip of the wing. Also concept 5 can use a separate pump for each section, such that it would use 9 pumps for the configuration shown.

Table 5-3 architecture concept priorities

<table>
<thead>
<tr>
<th>Concept #</th>
<th>Complexity</th>
<th>Internal losses</th>
<th>Space left</th>
<th>Weight</th>
<th>Safety</th>
<th>Cost</th>
<th>KBE final score</th>
<th>Implemented</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>+</td>
<td>✓</td>
</tr>
<tr>
<td>2</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>+</td>
<td>✓</td>
</tr>
<tr>
<td>3</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>+</td>
<td>✓</td>
</tr>
<tr>
<td>4</td>
<td>o</td>
<td>o</td>
<td>+</td>
<td>+</td>
<td>o</td>
<td>o</td>
<td>+</td>
<td>✓</td>
</tr>
<tr>
<td>5</td>
<td>o</td>
<td>+</td>
<td>-</td>
<td>o</td>
<td>+</td>
<td>-</td>
<td>+</td>
<td>✗</td>
</tr>
<tr>
<td>6</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>o</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>✗</td>
</tr>
<tr>
<td>7</td>
<td>++</td>
<td>-</td>
<td>-</td>
<td>o</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>✓</td>
</tr>
</tbody>
</table>

negative --    -    o    +    ++    positive

above average
average
below average

Implemented
not implemented
Besides these architectures various locations for the inlet/outlet slots have been identified, listed in Table 5-4. No trade-off is made between these concepts as they differ little in their functionality. Due to time-constraints the fuselage and tail inlet/outlet slots have however not been implement.

Table 5-4 Outlet and inlet slot concepts

<table>
<thead>
<tr>
<th>Concept #</th>
<th>Sketch</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td><img src="image1" alt="Wing outlet" /></td>
<td>Wing outlet</td>
</tr>
<tr>
<td>2.</td>
<td><img src="image2" alt="Fuselage outlet" /></td>
<td>Fuselage outlet</td>
</tr>
<tr>
<td>3.</td>
<td><img src="image3" alt="Tail outlet" /></td>
<td>Tail outlet</td>
</tr>
<tr>
<td>4.</td>
<td><img src="image4" alt="Wing inlet" /></td>
<td>Wing inlet</td>
</tr>
<tr>
<td>5.</td>
<td><img src="image5" alt="Fuselage inlet" /></td>
<td>Fuselage inlet</td>
</tr>
<tr>
<td>6.</td>
<td><img src="image6" alt="Bleed air inlet" /></td>
<td>Bleed air inlet</td>
</tr>
</tbody>
</table>
Chapter 6

Knowledge structure

The knowledge gathered in the previous chapters is captured in a knowledge structure. The structure used for this thesis will use a combination of all three types of classes identified in section 2.1.5: physical, functional and property. To realise a clear and intuitive structure, divisions are made based upon functions and individual components such that each class performs its own specific task. The properties of the created objects are assessed by property classes; these are easily used throughout the structure.

By creating all these different divisions the product model will allow the knowledge inside to be easily extended and improved during later stages of the design. Additional property classes, e.g. safety & reliability, can be added to the tree to incorporate additional disciplines. Also if more knowledge becomes available for an existing property class, the knowledge has to stored only once for the whole structure to benefit from it.

This chapter aims to give an explanation of the structures and the relations between the classes inside. The classes themselves are detailed in Chapter 7. First a global overview of the structure is given with its position in complete aircraft structure (6.1). Next the structures of the HLFC system (6.2) and FAFC system (6.3) are worked out in more depth. Then the pipe-assembly structure is discussed (6.4) and at last the air inlet/outlet assembly (6.5).
6.1 Global overview

As is described in the introduction the DARwing is used to generate the outer surface of the wing. Both HLFC and FAFC are children of the multi-element wing placed next to the already existing components. Because they can be applied to the wing separately from each other they are standalone classes. An overview of the structure is shown in Figure 6-1.

In Chapter 5 it was identified that both systems show a similarity in their need for a supply system. Therefore, their structures are generally similar and for some parts will use the same classes. Both are split up into four sub-assemblies, based upon the concept considerations presented in Chapter 3 to 5. The first class selects the area on the wing for each technology. The second contains the technology specific parts, the third the pipe-assembly to the pump and the last the pump together with the inlet or outlet assembly.

Right below the *multi-element-wing*, the *Cp-distribution* is placed. This class has a functional geometry; a surface over the entire wing that represents the pressure coefficient distribution. With this surface the $C_p$ value can be determined at any place on the wing. This surface is at this moment constructed in a similar way as the wing outer-surface where the airfoil(s) are replaced by $C_p$-curves. Other methods using e.g. cloud points from aerodynamic analysis data could also be used in the future.
6. Knowledge structure

6.2 HLFC structure

The structure used for the HLFC assembly is shown in Figure 6-2. It can be seen that, besides the children shown above, an extra functional class is added: skin-chamber-distribution. This class creates a skin-chamber layout that matches the suction profile specified. For external aerodynamic analysis the obtained suction distribution is also determined.

The suction assembly is split into multiple classes based upon the different parts of the assembly. Each class creates its own surfaces and openings. Because flow regulation is the main function of these parts, the flow conditions are worked out in these classes as well. Each class has a property list: flow-data which is used to store the flow conditions at that location.

Also shown in the structure is the flow-regulation-through-orifice class, which determines the size of the orifice needed to create the required flow condition in its parent. To do this it also needs the pressure inside the next component the flow passes through, determined by the Suction-assembly class.

The manifold class determines the location, vector and required flow-conditions for the pipes of the pipe-assembly. The property classes Head-loss, cost and weight are placed as children of various components determining the properties of their parents.

Figure 6-2 HLFC structure
6.3 FAFC structure

The FAFC structure, shown in Figure 6-3, is less complex. The actuator-area class creates the top surface for every actuator. These surfaces are used in the actuator-section class that places an actuator underneath. The actuators are a complete class that creates the shape and determines the interfaces to other systems.

Each created actuator object determines a location, vector and the flow-conditions for the pipe-assembly to connect to. Again the cost and weight components are present in the structure. It was decided not to give the actuator-top-surface a cost and weight, but to incorporate this into the actuator class.

![Figure 6-3 FAFC structure]

6.4 Pipe assembly structure

The structure used for the pipe-assembly of the pneumatic system is shown in Figure 6-4. In both cases the pipe assembly receives a set of pipe-connection-points with vectors and flow-data property lists. The input specifications for the layout, given by the user, determine which pipes will be connected to which pump. Before the paths of these pipes are created, first a check is performed by the pump-restrictions class. This is a functional class that recognises if a pump will be overloaded, e.g. on capacity. When this occurs, the pipe assembly is split and an extra pump is used.

When the pipe-connection-points are assigned to a pump, the pipe paths are created: consisting of a main-duct path, a path inside the wing in span-wise direction, and connector paths, a path connecting the main-duct path and the pipe connection points.

With the pipe-paths defined the pipes and junction themselves can be created. First a junction-list is created with the functional class create-junction-list. This list holds information on which pipe segment connects to which pipe segment(s). Next the following steps are taken for every pipe segment starting at the pipe segment through which all the air flow comes.

- First the surface of the pipe segment is created and, if this pipe connects to two other pipes, a junction is created.
The flow conditions at the ends of the pipe and the junction are evaluated with pipe-flow-evaluation and junction-flow-evaluation. Usually the flow-data at end points of the complete pipe-assembly are available as input. At other places in the assembly the flow-data is found in the connecting pipe segments.

If the pressure at the downstream or upstream location of a pipe segment should be adjusted to prevent back-flow an obstruction is placed and a reduced diameter is suggested.

Next the connecting pipe segment(s) is/are created using the same steps. As shown they are children of the current segment. This process continues until the ends of the pipe-assembly are reached.

Similar to the previous sections the physical classes have a cost and weight class. The head-loss class is also present as a child for classes where this occurs.

6.5 Air inlet/outlet assembly structure

The Air inlet/outlet assembly structure, shown in Figure 6-5, has four children: an inlet/outlet-slot, a pump, a path-between-points-with-vectors and a pipe-route.

The pump class creates a pump of the type specified by the user and is placed at the start of the pipe-assembly. The pump class specifies the location and vector for the pipe providing the in/outflow. It also calculates the performance of the pump (e.g. power-consumption, head-delivered)

The path-between-points-with-vectors class creates a smooth path from the pump to the inlet/outlet slot. This path is created into a pipe by the pipe-route class, which is the same as the class shown in Figure 6-4.
The *inlet/outlet-slot* creates the slots surfaces and computes the flow conditions at the wing surface.

*Figure 6-5 air inlet/outlet assembly structure*
Chapter 7

Classes and routines

The structures presented in chapter 6 contain the individual classes described in this chapter, not all classes are discussed as that would result in a very long list. Only the most interesting and influential are addressed. They are presented in groups ordered by structure trees.

First the common storage structure for flow conditions is discussed (7.1). Next the HLFC classes (7.2) and the FAFC classes (7.3) are described. Followed by the pipe-assembly (7.4) and the Air-inlet/outlet-slot-assembly classes (7.5). At last the common classes \textit{Cp-value, weight and cost} are described (7.6).

7.1 Common storage structure for flow conditions

Pneumatic relations are an important part of many classes. Therefore a standard for the flow conditions was developed in the form of a property list: \textit{flow-data}. This list at minimum holds the information listed in Table 7-1.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description [units]</th>
</tr>
</thead>
<tbody>
<tr>
<td>:p</td>
<td>Pressure [Pa]</td>
</tr>
<tr>
<td>:rho</td>
<td>Density [kg/m$^3$]</td>
</tr>
<tr>
<td>:T</td>
<td>Temperature [K]</td>
</tr>
<tr>
<td>:u</td>
<td>Flow-rate [m$^3$/s]</td>
</tr>
<tr>
<td>:mdot</td>
<td>Mass-flow [kg/m$^3$]</td>
</tr>
</tbody>
</table>

7.2 HLFC classes

The main classes of the HLFC product model are discussed in this section. For each class the most important inputs are mentioned, the complete list of inputs can be found in
Appendix A. Most classes of the HLFC structure tree are explained in this section, with an exception to the sub-manifold class because this class is very similar to the skin-chamber class. The complete geometry of the HLFC section with its pneumatic system is shown in Figure 7-1.

Figure 7-1 Complete HLFC section, with its pneumatic system

7.2.1 Skin-chamber distribution
In this research two methods were developed to create a skin-chamber distribution, both make a discretisation of the suction distribution. The main inputs needed for this class are the following:

- the suction distribution
- the skin-chamber distribution method
- the skin-chamber width

The suction distribution can be provided\(^1\) as a file containing a list of graph corner points or \(x/\bar{c}\)-vs.-\(y\) points shown in Figure 7-2. Where \(x/\bar{c}\) describes the location on the wing in the chord direction, with \(\bar{c}\) as the mean chord length and \(x\) the distance from the LE. A negative value of \(x/\bar{c}\) indicates the bottom side of the wing and a positive value the top.

The figure shows some precision being lost with the \(x/\bar{c}\)-vs.-\(y\) points; this can be improved by using more or better distributed points. The width of the skin-chamber is specified in \(dx/\bar{c}\) and thus influences the number of skin-chamber used in chord direction.

As said, both distribution methods use a discretisation. The first uses a simple discretisation that creates chambers that each have a similar width. The suction coefficient \((C_Q)\) assigned to each chamber is obtained by taking an average. The second method uses a smart discretisation. It uses the same approach as the first for the areas where \(C_Q\) is not constant. At the regions where \(C_Q\) is constant it creates only one chamber. The results of both methods are shown Figure 7-3.

The \(C_Q\) values found for the skin-chambers are stored in a list, e.g. \((-0.001 \ -0.0015\ -0.0025 \ -0.0008\) and the \(x/\bar{c}\) values found for the skin-chambers are translated into \(u\) values and stored in a list, e.g. \((0.4\ 0.45\ 0.49\ \ldots\ 0.56\ 0.65)\).

\(^1\) The suction distribution is a result of external Aerodynamic analysis in this case performed by DLR; their preferred suction distribution is provided as list of graph corner points.
7.2.2 Perforated surface
A perforated surface is created for each skin-chamber in the HLFC system. The main inputs are the following:

- The wing-surface
- The \( u \) and \( v \) boundaries\(^1\) for the section
- The \( C_Q \) value

With the \( u \) and \( v \) boundaries a section of the wing is selected. Next the flow conditions outside the surface are determined using the \textit{Cp-value} class (7.6.1). After that the pressure drop through the surface, corresponding to the suction specified by \( C_Q \), is computed using equations (3-1) and (3-2). At last the flow conditions on the inside of the perforated surface are derived and stored in a \textit{flow-data} property list.

7.2.3 Skin-chamber
The \textit{skin-chamber} class creates the inner surface and side walls of the cavity. The main inputs are the following:

- The perforated surface with its \textit{flow-data}
- Thickness of the skin-chamber

---

\(^1\) \( U \) and \( v \) are parameters that indicate a location on a surface. A figure that shows the directions of these parameters is shown in the manual in Appendix I.
The inner surface is created by making an offset of the perforated surface at a distance equal to the thickness of the skin-chamber. Next, using the edges of both surfaces, the sides of the skin-chamber are created. The suction-assembly class specifies which sides will be created, this prevents two connecting skin-chambers to create the same wall twice. The construction overview and the result form GDL are shown in Figure 7-4.

The area of the inner-surface is slightly smaller than the perforated surface, therefore the flow conditions change. Because the difference is very small the pressure and density are assumed to remain constant. The mass-flow remains constant as well, such that the velocity of the flow has to change: \( \rho S_1 u_1 = \rho S_2 u_2 \). The adjusted flow conditions are as usual stored in a flow-data property list.

7.2.4 Suction-assembly

The suction-assembly connects the classes described in this sub-chapter, thus has access to all the information of its children. By requesting the flow-conditions in all skin-chambers it determines which chamber needs the lowest pressure. Next the maximum size of the orifice to this chamber is determined. This size of this orifice is used to determine the pressure in the manifold (or sub-manifold) by applying the Borda Carnot equation \((3-6)\) and \((3-7)\). With equation \((3-9)\) and \((3-10)\) the rest of the flow-conditions are computed. These are not stored in a flow-data list but given as input to the (sub-)manifold. If sub-manifolds are used this process is performed twice, once for the skin-chambers to the sub-manifolds and once for the sub-manifolds to the manifold.

7.2.5 Flow-regulation-through-orifice

Flow-regulation-through-orifice is a class that is used by both the skin-chamber class and the sub-manifold class. The main inputs are the following:

- The surface in which the orifice is to be made
- The flow conditions upstream
- The pressure downstream

The pressure drop needed is controlled by the size of the orifice. To determine this size the Borda Carnot equation must be solved. This is a complex relation that is not easily rewritten; therefore the get-value-from-formula function is used. This function is explained in detail in Appendix B. When solved the orifice is created by subtracting a cylinder from the surface given as input. The results of this operation for a complete section of skin-chambers are shown in Figure 7-5. Visible is how the sizes of the orifice can be different for each skin-chamber.
7.2.6 Manifold

The last component of the suction-assembly is the *manifold*. This class creates the cavity behind the skin-chambers or sub-manifolds (depending on the architecture concept used). In this explanation it is assumed that sub-manifolds are not used. The main inputs are the following:

- The combined skin-chamber inner-surface
- The shape
- The thickness
- The flow-conditions
- The locations and sizes of the connecting pipes

To create the cavity the *manifold* class uses the same approach as the *skin-chamber* class. Using an offset surface on airfoils with tight corners sometimes produces impossible surfaces, shown in Figure 7-6a. To avoid this two other approaches to create the inner-surface of the manifold where developed: scaled (Figure 7-6b) and diagonal (Figure 7-6c). In the figure the diagonal shaped manifold takes up to complete nose section of the wing, but if suction is only used on the top or bottom side of the wing, it will create a more acceptable manifold.

The diagonal shape uses a relatively simple approach by making straight surfaces between the edges of the input surface. The scaled shape uses a more complex method by downsizing the surface and placing it at an average distances. More detail on this approach is given in Appendix C.

At last the holes for the pipes are subtracted from the created inner surface. For every hole a *flow-data* property list is created. This property list holds, besides the usual flow conditions, extra information: the diameter of the hole created and the surface of the area in which the hole lies. If more holes are present this area is divided by the amount of holes and so is the mass flow. The pressure losses introduced by the entrance to the pipe are computed by the *pipe-route* class.
7.3 FAFC classes

7.3.1 Actuator-area
The actuator-area class creates the complete area of an actuator section and divides it up into smaller segments that become the top surface of each individual actuator. The main inputs are the following:

- Chord position of the jets on the wing
- Span region on the wing
- Actuator geometry specifications: length, width, jet-position

Where the actuator geometry specification are found in the actuator class. These inputs are illustrated in Figure 7-7. The figure also shows a small part of the specified span-region that remains unused. Actuator-area determines the maximum number of actuators that can be fitted in the span-region and creates top-surfaces for each. Any region left is too small for an actuator and is therefore not used.

7.3.2 Actuator
The actuator class performs three tasks: It creates a body of the actuator, defines the connection points for the pipe and specifies the supply requirements. The main inputs are the following:

- The top surface
- The type of actuator
- The required $\Delta C_L$ (optional)

The geometry of every type of actuator is different, the specifications for the supply system are, however, in the same form. Therefore it was decided to use a mixin structure (Figure 7-8). Meaning that for every type of actuator a separate class is written. They have, however, the same actuator mixin. This mixin is a class which holds
the common knowledge of the actuators. It uses the actuator characteristics of the actuator specific class.

The actuator specific class creates the shape and also provides the length, width and jet position needed for the actuator-area class. Beside the geometry it also holds information regarding the supply requirements, such as air pressure and mass-flow. In Chapter 4 it was identified that the lift increase, created by the pulsating jet, stands in relation to the mass-flow. Therefore, instead of providing only a fixed mass-flow the actuator specific class can also hold a delta-cl-vs-mdot-relation, similar to equation (4-1).

The actuator mixin class gathers the flow data from the actuator specific object and stores it in a flow-data property list. If the mass-flow is not fixed the relation and ΔC_L are used to determine it.

![Figure 7-8 Actuator mixin structure](image)

### 7.4 Pipe-assembly classes

#### 7.4.1 Pipe-path

The pipe-path class creates an F shaped pipe network, shown in Figure 7-9. This is constructed out of two classes: the main-duct-path and the connection-path. The main inputs are the following:

- The points with vectors to connect to
- The chord-location of the main-duct-path
- The type of main-duct-path
- The angle at which the connection paths will meet the main-duct-path

The main-duct path can be created using 2 approaches: on-chord and at-offset. With on-chord a path is created in middle of the wing along the provided chord location. With at-offset a path is created at a constant distance from the wing surface along the chord-location. With both approaches this path is cut into smaller sections, so that a separate path is created between each connection path. The on-chord method is not yet available for wings with dihedral, on these wings the at-offset method should be used instead.
The connection path is a b-spline-curve created from 2 end-points and 2 control points. How the locations of these points are determined is best shown in Figure 7-10. The intersection-point also serves as the location to cut the main-duct-path.

The connection-path for the last point is slightly different. It connects tangentially to the main-duct-path as is shown in Figure 7-9. Therefore the first control for this path is placed at the location of the minimum distance.

![Figure 7-9 F-shaped pipe-layout](image)

At last the paths and flow-data (if present) are placed in a property list: the pipe-data-list. This list serves as input for the pipe-route class. Every entry of this list houses the information available for a specific pipe segment. At minimum this is the :path, optional are :diameter, :location, :p, :rho, etc. Where :location determines where in the pipe the flow conditions are specified, upstream or downstream.

7.4.2 Pipe-route

The pipe-route class creates the pipes and junction based on the paths provided by pipe-path. It creates the whole pipe-network segment by segment. After each created segment it creates new pipe-route object for each connecting segment. The main inputs are the following:

- The pipe-data-list
- The junction-list
For more detail on the pipe-data-list and the junction-list see Appendix D.

In this description it is assumed that the *pipe-route* object will have a junction and thus two *pipe-route* objects as its children.

The *pipe-route* object will first check the data available for its pipe-segment. Any information, that is not available in the *pipe-data-list*, is determined with the information of its children or parent. For example the diameter of the pipe can be calculated using the sum of the cross-sectional areas of the children: \( S_1 = S_2 + S_3 \).

To prevent unnecessary losses the path of the pipe is adjusted and to create space for the junction the pipes are shortened. The operations performed are schematically shown in Figure 7-11. The results are shown in Figure 7-12, as shown the outer surface of the junctions have not been created up till now.

\[
d_1 = \frac{D_C - D_A}{2}
\]

**Figure 7-11 pipe manipulations**

**Figure 7-12 Pipe network**

Next the flow conditions are determined for both the upstream-location and downstream location of the pipe using *pipe-evaluation* and *junction-evaluation*. The flow-conditions are stored into flow-data lists as usual, this time for the upstream location and downstream location.

### 7.4.3 Pipe-evaluation

*Pipe-evaluation* handles the pressure losses introduced by a single pipe segments. The main inputs are the following:

- Flow-specifications up- or downstream
- The pipe
- The pipe-roughness

With these inputs *pipe-evaluation* performs three basic operations.
1. Determine equivalent length
2. Determine pressure loss
3. Determine diameter of obstruction and the optimal diameter for a following model generation.

To determine the pressure loss in straight segments as well as bend segments, *pipe-evaluation* first determines the equivalent length of the pipe segment. Because a single pipe segment can have straight and bend parts, *pipe-evaluation* separates the segment into multiple smaller sections. For each section the curvature is measured. This is used to determine the equivalent length with equation (5-7) and (5-8). Because these equations are based upon the 90° bends, the corner angle ($\beta$) is also measured and used with the linear relation (5-9). Finally all a summation of all sections is made to determine the complete equivalent section.

The reason why the linear relation is used now becomes clear. The results obtained with the linear relation do not affect the equivalent length for a single bend that is split up into more sections. The sum of the equivalent lengths of the first four sections in Figure 7-13 will result in the same equivalent length found when the whole bend is assessed at once.

![Figure 7-13 A pipe segment split up into multiple sections](image)

With the equivalent length determined, the pressure loss can be determined using (5-5) and (5-6). On top of this comes the pressure loss introduced by the pipe-entry, if this is specified in the pipe-data-list. The drop due to the contraction is also included using equation (3-11).

If the flow conditions at both ends of the pipe are already known an extra pressure drop is introduced using an obstruction or a smaller pipe-diameter. For more detail on this see Appendix E.

### 7.4.4 Junction-evaluation

The *junction-evaluation* handles the pressure losses inside the junction. The main inputs are the following:

- Flow-specifications up- or downstream
- The three connecting pipes

With these inputs the *junction-evaluation* performs three tasks

1. Determine the connecting angle of the branch
2. Determine the pressure losses
3. Determine pressure changes needed in connecting pipes.
The connecting angle \( \theta \) is determined by comparing the tangents of all pipes at the intersection. This angle is needed to perform the pressure loss calculations. In case of suction the horizontal momentum equation (5-13) is used to determine the pressure losses, in case of blowing the K-factors from Table 5-1.

To guarantee the required flow through all pipes the junction often concludes that the pressure in one of the connecting pipes needs to be reduced. In this case the flow conditions for both ends of that pipe are determined. *Pipe-evaluation*, described above, will thus have to introduce an obstruction and calculate the optimum diameter.

### 7.5 Air-inlet/outlet-slot-assembly classes

#### 7.5.1 Inlet/outlet slot
The inlet/outlet slot is at this moment a simple class. It creates a small cavity and an opening in the wing surface. The main inputs are the following:

- The span-locations
- The chord-locations
- The pipe-diameter

With the chord and span-locations a section of the wing is selected. A triangle shaped slot is created under this section to which a pipe can connect. For HLFC the pipe connects at the LE side, for FAFC at the TE side. A schematic overview together with the result for the outlet slot is shown in Figure 7-14.

Beside the surface this class also gives the location and vector for the connecting pipe and the flow-conditions at this location. These are determined using the relations described in section 5.1.6.

![Figure 7-14 inlet and outlet slot](image)

#### 7.5.2 Path-between-points-with-vectors
The *path-between-points-with-vectors* creates a single path that connects the pump and the outlet slot. The main inputs are the following:

- The default-radius,
- The locations and vectors at both ends.

The class creates two control points at the default-radius distance from the input-points, positioned along the input-vectors. From these two control points and the input-points a b-spline-curve is made. An overview of this is shown in Figure 7-15.
7.5.3 Pump

The *pump* class performs three tasks: It creates a body of the pump, defines the connection points for the pipes and it calculates the power requirements. The main inputs are the following:

- The pump-type
- The location
- The flow-conditions on both sides of the pump

The shape, pipe-locations, head-function and power-function for every type of pump is different, while the performance relations are always similar. Therefore, similar to the actuators, a mixin structure is used (Figure 7-16). For every specific pump a separate class has to be written, but they all use the same mixin. The mixin class holds the relations needed to determine the pump performance under the found flow conditions. It uses the pump characteristics of the *pump specific* class.

The relations in the mixin use the following steps to determine the performance of the pump.

1. The flow conditions at the entry and exit of the pump are obtained from the connecting pipes.
2. The flow-rate is determined from the upstream flow conditions.
3. The flow-rate is used in the head-function and power-function to determine the head delivered \( (h_{p,in}) \) and the power requirement \( (P_{in}) \) of the pump.
4. The head increase of the flow is determined \( (h_{p,out}) \) using relation (5-3)
5. The difference between \( h_{p,\text{out}} \) and \( h_{p,\text{in}} \) determines the over-performance of the pump.
6. The efficiency of the pump is computed using \( \eta = \frac{P_{\text{out}}}{P_{\text{in}}} \), where \( P_{\text{out}} = mgh_{p,\text{out}} \).

7.6 Common classes

7.6.1 Cp-value
The \textit{Cp-value} class is of the functional type. It determines the pressure coefficient at a point or range on the wing. The inputs this class needs are the following:

- The location \((x/c)\) or a region on the wing, span \((v_1\text{ to }v_2)\) and chord \((x_1/c\text{ to }x_2/c)\)
- The \textit{Cp-surface}
- The accuracy (only needed if a region is used), this determines the intervals used to assess the average Cp-value.

To determine the pressure coefficient it uses the following steps (a visualisation is shown in Figure 7.17):

1. Vertical lines are created at the location or region on the wing
2. Intersection-points between the Cp-curve and the intersection-lines are obtained.
3. The intersection points are filtered for the top and bottom sides of the wing, for this it is assumed that the pressure on the top side is always lower than the bottom side.
4. The z-value of the points are requested, these are the \( C_p \) values
5. When a range is used the average of values found is taken.

![Figure 7-17 Cp-value](image)
7.6.2 Weight & Cost

Cost and weight are both property classes, they assess their parent on weight and cost. Because these classes are used throughout the whole product model, they accept many different inputs:

- A surface
- A list with multiple surfaces
- A body
- A list with multiple body’s
- Subtracted surface(s)
- Subtracted body(s)

In the cases where surfaces are used the parent is requested for the sheet-thickness. With these inputs the volume and areas are determined. Beside the geometry inputs both class also need to know the material type, which is also requested from the parent.

Currently the weight class purely uses the volume and density to determine the weight. In the future this could be extended with weights for example introduced by connections between two surfaces, i.e. welds or bolts.

The cost class in the case of solid bodies uses the weight and price-per-kg to determine the price. When surface are used, a sheet-price-per-m3-formula is used, where the price is dependent on the volume of the sheet. This class also has the potential to be extended for additional costs such as production and assembly.

The density, price-per-kg and sheet-price-per-m3-formula are properties of the material used. These properties are stored in the material database and requested when needed by the cost or weight class. Many classes in the product model will request properties of the same material. To shorten search times, once accessed the material properties are stored in a temporary database. This is schematically shown in Figure 7-18. With this approach the large material database only has to be searched when the material type is used for the first times. In all other cases a search in the much smaller temporary database is sufficient.

The material database is currently a property list where each entry has the properties :name, :density, :price-per-kg and :sheet-price-per-m3-formula. A data-base with three entries is shown in Table 7-2. The values are derived from the prices stated on www.onlinemetals.com. The :sheet-price-per-m3-formulae as stated in the table have to translated into a form that is accepted by GDL. For the third entry this would be 

```
#'(lambda (volume) (polynomial volume -20670487.96 513966.39 20)).
```

At last both classes also accept a value of the cost and weight as input. This is useful for classes with a fixed price such as pumps and actuators.
Figure 7-18 Cost and Weight database structure

Table 7-2 Materials database

<table>
<thead>
<tr>
<th>:name</th>
<th>:density</th>
<th>:price-per-kg</th>
<th>:sheet-price-per-m3-formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>aluminium 7075</td>
<td>2730</td>
<td>$29.43</td>
<td>$15.317 + volume * 62241</td>
</tr>
<tr>
<td>aluminium 6061</td>
<td>2700</td>
<td>$15.30</td>
<td>2.3199 + volume * 37533</td>
</tr>
<tr>
<td>titanium grade 2</td>
<td>4511.824</td>
<td>$128.44</td>
<td>-20670487.96 + volume * volume * 2 + 513966.39 + volume + 20</td>
</tr>
</tbody>
</table>
Chapter 8

External aerodynamic analysis

To demonstrate the potential of the KBE design approach external aerodynamic analysis for HLFC has been connected to the product model. This analysis has not been incorporated into the GDL environment but is accessed through an interface to Xfoil-suc analysis software. Xfoil-suc was selected because an interface to Xfoil, which it is an extension to, already existed.

This chapter will first give a short introduction of the software package (8.1) next the interface to analyse an airfoil with and without suction is discussed (8.2), followed by the interface to determine the suction needed (8.3).

8.1 Introduction to Xfoil-suc

Xfoil-suc is an extension, created by L.W.M. Boermans, to Xfoil from Mark Drela, MIT Area & Astro. Xfoil is a 2D design and analysis program for subsonic, isolated airfoils [Drela, et al., 2001]. It uses a high-order panel method with fully-coupled viscous/inviscid interaction. Among others the program analyses the boundary layer for viscous flow conditions. The results interesting for this thesis are mainly the pressure distribution, N-factor distribution, and lift and drag predictions. The results are provided into .dat files, where the pressure distribution and N-factor are given as a list of points vs. x/c. The inputs necessary for this analysis are:

- Airfoil, provided as a list of x/c vs. z points in a .dat file
- Reynolds number
- Mach number
- Angle of Attack or Lift Coefficient
- Number of panels. More panels give a higher precision but require longer computation times.
- Max iterations, the viscous analysis of Xfoil uses an iterative process. If conversion is not reached within the max iteration, solutions are to increase this number or increase the number of panels.
The Xfoil-suc adds the effects of BLS to the analysis; this requires only one additional input:

- Suction profile. Proved as a list of x/c vs. $C_Q$ points in a .dat file

Besides analysis of the effects of BLS, Xfoil-suc can also determine the suction needed to prevent/delay boundary layer transition from a laminar to turbulent. To do this one extra input is needed:

- The start and end panel that suction can be applied on, shown in Figure 8-1

This gives the suction profile as a result, stored as a list of x/c vs. $C_Q$ points in a .dat file

![Figure 8-1 Start and end indication for suction on an airfoil divided in panels](image)

### 8.2 Xfoil-suc interface to analyse suction

Both Xfoil and Xfoil-suc are operated using a command window under Microsoft Windows. All commands are given with the keyboard using short statements followed by an {ENTER}. During the analysis various times a window pops up with a graphical representation of the airfoil and analysis.

An interface between GDL and Xfoil has already been developed in the past, for Xfoil-suc an interface was created during this thesis work. In the Xfoil interface all commands are directly given to the command windows. Unfortunately this was not possible for Xfoil-suc. Xfoil allowed all commands to be given in one large string, Xfoil-suc however requires input given in two strings (due to a warning at the start of the program). An extra step through Visual Basic was therefore necessary. A flow diagram of this is shown in Figure 8-2.

To perform analysis over the entire wing, the analysis routines are executed for multiple sections along the wing-span. The number of sections is dependent on the input of the user and the locations of the suction areas. At the start and end of each suction area extra sections are added, as shown in Figure 8-3. These extra sections are aimed to incorporate changes in the results caused by the sudden presence or lack of suction.

![Figure 8-2 Xfoil-suc interface flow-diagram](image)
For every section a .dat file with x/c vs. z points representing the airfoil is created using routines available from the Xfoil interface. Besides the airfoils, the realised suction profiles undergo a similar treatment. When suction is present at the airfoil section the realised profile is translated to a list of x/c vs. $C_Q$ points in a .dat file. When there is no suction, again a .dat file is written, only this time $C_Q$ is set 0 for every point such that there is no suction.

For every section analysis is performed by Xfoil-suc, so that in the end all $C_p$ curves can be used to create the $C_p$-surface, which is used to determine the internal pressures in the suction system. A work flow diagram of the steps described above is shown in Figure 8-4.

### 8.3 Xfoil-suc interface to determine suction

With the ability to determine the suction required Xfoil-suc completes the external analysis for a BLS design. As discussed in 8.1 the only extra input required to determine the suction profile is an indication of which panels are available for suction.

This region is usually specified with x/c locations and will thus have to be translated in the panel numbers. To perform this translation the airfoil is first divided into panels. This is performed during the first steps of the Xfoil-suc. Once the panels are created the airfoil is stored again in a .dat file, where every point marks the start of the next panel.

Next, using the newly created file, the x/c locations specified for suction are translated into panel numbers, e.g. panel 81 and panel 92. Now Xfoil-suc is executed again; this time to determine the needed suction with as result a suction-profile.

Finally, after a skin-chamber distribution has been created and the realised suction distribution is determined, the results of the BLS is analysed again using the Xfoil-suc interface to analyse suction. A complete overview of the workflow described above is given in Figure 8-5.
Figure 8-4 Xfoil-suc interface to analyse suction
Figure 8-5 Xfoil-suc interface to determine suction
8.4 Conclusions: Xfoil-suc

The interface is created successfully and works in an automated loop. Errors with convergence sometimes occur, but are detected by GDL and automatically the number of panels is increased until convergence is reached.

Because the interface uses Visual Basic care must be taken when operated. The user is urged to leave all input devices untouched until the simulation is completed. It was found that the link via Visual Basic behaves differently depending on computer type. Inputs are sometimes not correctly received by Xfoil-suc, which leads to no or less results. GDL detects when the results are ready. If not all results are generated within a specified time\(^1\), a warning is given to the user and a new attempt is made with the same inputs. This often resolves the problem.

Xfoil-suc is however not intended for the flight conditions of HLFC. Because the program only analyses 2D effects, the instabilities due to cross-flow are neglected. Also the Xfoil-suc program is designed for speeds below Mach 0.5, while HLFC is intended for higher Mach numbers. The validity of the Xfoil-suc interface to determine and analyse suction for HLFC is therefore unknown.

\(^1\) Currently set to: 1min for Xfoil-suc-analyse 5min for Xfoil-suc-determine
Chapter 9

Validation

To prove the validity of the analysis modules, described in the previous chapters, verification is needed. This can be done by either comparing the results with experiments, with data available from previous experiments or with results from established CAA tools.

Verification is the most important for the internal flow analysis. The weight and cost analysis use simple relations, these have been verified using hand-made calculations. The external flow analysis is performed by a CAA tool and can therefore rely on the validity of the tool. Keeping in mind the boundaries with respect to the Mach number and 2D flow.

In this chapter the internal flow analysis verification will be performed on various parts of the system. Starting with the suction assembly of the HLFC system (9.1) followed by the pipe-flow analysis (9.2) and the junction flow analysis (9.3). Next a small network of pipes is evaluated (9.4) and at last conclusions will be presented (9.5).

9.1 HLFC suction assembly

A CAA tool that can analyse the internal flow of a suction system similar to the one used for HLFC has not been found. Experimental data on these flows was not available either. The last option, performing experiments, would take up to much time and was therefore not chosen.

However, the relations used in the suction model are themselves based upon experiments performed in the past. Both the Goldstein equation (3-1) and the Borda Carnot (3-6) have been derived from experimental data. Their accuracy is therefore not further verified in this thesis.

The use of these relations in the product model and the effects of the size and shapes of the skin-chambers and manifolds(s) have been discussed with Arne Seitz, an aerodynamicist from DLR, who has gained experience with HLFC suction systems from,
among others, the ALTTA project. In his opinion the methods used are correct for the purpose of the product model.

9.2 Pipe-flow-analysis

Pipe flow analysis is an important topic for many industries; this has led to the development of many different CAA tools. The analysis software AFT Arrow has been chosen to verify the pipe-flow-evaluation object in the product model. AFT Arrow uses compressible flow relations and tables to predict the flow conditions throughout a pipe system. It has been used for applications ranging from chemical, to pharmaceutical, to aerospace and more [AFT Arrow, 2010]. The program uses pipe systems consisting of straight pipe segments with components in between. These components can be anything a flow can pass through, e.g. bends, orifices, valves, junctions, pumps but also user specified objects. The losses introduced by these components are in all cases calculated using pressure loss coefficient (K-factors).

The first verification done is the pressure loss due to wall friction in a straight pipe. For this a pipe of 1 meter with an internal diameter (ID) of 50mm was used. Both the flow velocity and pipe-roughness where varied during the tests. The setups used in AFT Arrow and created by the product model (GDL) are shown Figure 9-1. The assigned pressure was set to 25,000 Pa at 228 K, which is of the same order as expected for the HLFC system.

The pressure difference at the start and end of the pipes for the different settings are shown in Figure 9-2a and Figure 9-2b. Looking at the flow velocities, it can be seen that both analyses show similar trends. The results for flow-velocities below 60 m/s are within 3% of each other, but for higher velocities they start to deviate more. This can be explained by difference between incompressible relations used in GDL and the compressible relations used in AFT Arrow. The effects of the pipe roughness at a flow velocity of 10 m/s show a near perfect match (0.25% difference).

Next the pressure losses through bend segments were verified. Again a pipe with an ID of 50mm and a flow speed of 10m/s was used. AFT Arrow requires two straight segments to be added to the setup, their length was set to 1mm and their impacts are therefore neglected. The setups used are show in Figure 9-3. The assigned pressure was set to 25,000 Pa at 228 K with an absolute pipe-roughness of 0.06mm.

The pressure losses were evaluated for different curvatures and different bend angles, their results are shown in Figure 9-4a and Figure 9-4b respectively. Visible is the good correlation of the curvature effects on the pressure loss, the difference between the two analyses are below 15%. For different bend angles the results become less accurate when moving further away from the 90° bend angle. This is as expected due to the linear relation that is used (discussed in sections 5.3.1 and 7.4.3). However for angles between 45 and 180 degrees the results are within 20% accurate and between 65 and 135 within 10% accurate.
9. Validation

Gathering Knowledge

Figure 9-1 Setups used for verification of the straight segment

Figure 9-2 Pressure loss in a straight pipe segment (length = 1m, ID = 50mm)

Figure 9-3 Setups used for the verification of the bend segment

Figure 9-4 Pressure losses through a bend segment (ID = 50mm, u = 10 m/s)
9.3 Junction flow analysis

In this section verification is performed for unequal area junctions where the common flow passes through one of the straight legs. Unequal area junctions where the common flows pass through the lateral branch are not available in AFT Arrow and could therefore not be verified.

9.3.1 Joining flows

In the works of Basset et al. [2001] verification for joining flows in equal area junctions \( (S_{\text{run}} = S_{\text{com}}) \) has been performed using experimental data. As discussed in section 5.1.4 it was found that the relations over-predicted the pressure losses, an angle correction factor was therefore introduced to counteract this. The relations for un-equal area changing junctions \( (S_{\text{run}} \neq S_{\text{com}}) \) have not been verified before and are therefore verified in this section.

The only un-equal area junction available in AFT Arrow are junctions where \( S_{\text{run}} + S_{\text{branch}} = S_{\text{com}} \). The branch angle in this junction can vary between 15 and 165 degrees. The setup used by AFT Arrow and GDL are shown in Figure 9-5. The pipe segments used where all 1m long with an ID of 50mm and the assigned pressure was set to 25,000 Pa and 228K.

The total pressure drop in this small system was predicted by the GDL product model and AFT Arrow at different flow velocities and branch angles; the results are shown in Figure 9-6a and Figure 9-6b respectively. It can be seen that for the flow velocities the pressure drop predicted by the GDL product model is around 25% lower than what is predicted by AFT Arrow. For the changing branch angles the GDL model under-predicts the losses also around 25% with respect to AFT Arrow. Up till a branch angle of 60° the trend of both predictions is similar. Above 60° AFT Arrow shows an abrupt drop in the pressure loss, which from that point on remains constant. This behaviour is unexpected as the directional changes the flow experiences keep increasing, which usually leads to larger pressure losses. An explanation for this sudden change could lie in the tables used by AFT Arrow.

![Figure 9-5 Setup used for verification of a junction with suction.](image)
9.3.2 Separating flows

The same evaluation was performed for blowing conditions. The setup used for both models is shown in Figure 9-7. The pressure and temperature conditions are set to the conditions found in the FAFC flight phase, 100,000 Pa and 287 K. The inner diameter of the pipes is set to 50 mm. For separating flows the pressure losses from the common pipe to the run and the branch are calculated separately. The pressure losses where therefore measured in the junction itself. The length of the pipes is thus irrelevant.

The pressure losses predicted by both models at different flow velocities and branch angles are shown in Figure 9-8a and Figure 9-8b respectively. In both cases the results of both models follow similar trends, with an exception for branch angles larger than 90°. At these angles AFT Arrow predicts the pressure loss to the branch to remain constant. This is again unexpected as the directional changes to the flow increase with the angle.

The predictions of both models for the pressure loss to the branch differ around 35% from each other. To losses to the run are in both cases predicted to be zero (except for the larger branch angles).
9.4 Small network

The demo version of AFT Arrow, which is used for the verification in this chapter, has a few limitations: saving and copying data has been disabled and the number pipes in a network is limited to five. It is therefore not possible to verify a large network. Thus a small network consisting of five straight pipes, two bends and one junction is created and evaluated with joining flow conditions. These parts are typically encountered by HLFC when using pneumatic architecture concept 1 (see Table 5-2). The setups used are shown in Figure 9-9.

The pressure at the start of the pipes was set to 25,000 Pa and the Temperature to 228 K. The inner diameters of the before the junction pipes was set to 50mm and after the junction to 70,71mm. The absolute pipe roughness was set to 0.06mm. The pressure losses through this small network where analysed at different velocities, the results are shown in Figure 9-10. It was found that for velocities under 20m/s the difference between both predictions was below 25%, for higher speeds the difference gradually increases (35% at 40m/s and 47% at 60m/s).
### 9. Validation

The verifications of the internal flow analysis have shown that the GDL analysis predicts the flow for straight and 90° bend pipe segments very close to the predictions of AFT Arrow (within 3% for speeds below 60 m/s). For bends with an angle other than 90° the results vary much more, due to the linear relation used (within 20% for angles between 45° to 180°). For un-equal area junctions the results show differences up to 35%.

Applying this in a small network has shown similar differences for the total pressure loss, only at velocities higher than 40m/s these differences start to increase.

AFT Arrow has shown unexpected results for junctions with large branch angles. This may be caused by the tables AFT Arrow uses to compute the pressure losses. Similar trends were found for equal area junctions. The experimental data of Basset et al. [2001], however, does not show this trend (see Appendix F for more detail).
Chapter 10

Solution for an example case

The SFWA project is at the moment of writing still in its early stages. As a consequence many inputs needed for the product model of both smart technologies are still unknown. It is therefore not yet possible to generate realistic outcomes for both smart technologies and a trade-off between the internal system concepts cannot be made. To show the results that could be expected from the product model this chapter will work out one example of internal system configuration. It must be stressed that these results are only for indication.

The most influential missing inputs are the aircraft concept and external aerodynamic analysis data. The last, however, can for HLFC be performed using the Xfoil-suc interface. This chapter will therefore only present results for one particular HLFC configuration. First the inputs selected for this example configuration are discussed (10.1), followed by the results obtained (10.2). The last section will show a couple of geometry examples for other configuration (10.3).

10.1 Aircraft and settings

To create results first the settings for the product model must be chosen. As indicated in Chapter 1, the Dassault Falcon 7X has been suggest as a possible candidate for the smart technologies. This aircraft has a total wing span of 26.21 m and a mean wing chord of 3.35, the aspect ratio of the wing is 9.7 [Dassault Falcon 7X, 2009]. The wing has been created with the DARwing product model and is shown in Figure 10-1. Because HLFC uses a NLF airfoil the standard airfoil of the 7X (which was also unavailable) has been replaced by an airfoil provided by DLR. This airfoil is intended for use with HLFC at high Mach numbers. The created wing is thus not a representation the real wing of the 7X, only the outline is similar.

Typical long range cruise conditions for the 7X are the following:

- Speed: Mach 0.8
- Altitude: 10,000 m
To determine the region where BLS must be applied, the airfoil is first analysed by Xfoil-suc. This analysis is performed at an Angle of Attack (AOA) of 2°, N-factor curve resulting from this analysis is shown in Figure 10-2. It shows that boundary layer transition occurs at $\approx 0.33x/c$, but the N-factor already starts building up from $0.05x/c$. The suction region is assumed to be bounded by the front spar, therefore it is set to 0.05 to 0.25 $x/c$. With this setting the bottom of the wing remains available for a high-lift device such as a Krueger flap.

45% of the leading edge is selected to apply BLS as is shown in Figure 10-3. The internal systems layout was set to concept 1 (base) for HLFC and concept 1 (per-section) for the pneumatic system. The skin-chamber width was set to $0.025x/c$, such that there would be eight skin-chambers in chord-wise direction. The other inputs used can be found in Appendix A.

Currently there is only one pump specified in the product model, this pump is based on the pump that was designed for the ALTTA project, its weight is set to 24,7 kg and its costs to $ 5000,-. Ideally more pumps should be specified, such that a selection can be made based on the flow conditions in the pneumatic system.
10.2 Results of the example solution

The geometry created by the parametric model, with the inputs specified above, is shown in Figure 10-4. A routine has gathered all the data stored inside the model and provided it in .htm format which is also readable by Microsoft Excel. A screenshot of the results is shown in Figure 10-5.

Xfoil-suc analysis with the suction, obtained from the skin-chamber configuration, has shown a shift of the transition point to \( \approx 0.45x/\bar{c} \). A summary of the results is shown in Table 10-1. To calculate the power consumption of the pumps it is assumed that they operate with an efficiency of 60% (\( \eta = 0.6 \)).

<table>
<thead>
<tr>
<th>Table 10-1 Product model results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total weight</td>
</tr>
<tr>
<td>Total cost</td>
</tr>
<tr>
<td>Power added to the internal air flow by the pumps</td>
</tr>
<tr>
<td>Power consumption of the pumps (( \eta = 0.6 ))</td>
</tr>
<tr>
<td>Drag introduced at the exhaust</td>
</tr>
<tr>
<td>Drag coefficient without suction</td>
</tr>
<tr>
<td>Drag coefficient with suction</td>
</tr>
</tbody>
</table>

The power consumption of the pumps and the drag introduced at the exhaust can be expressed in a drag coefficient with equation (10-1), where \( P \) is the power and \( S \) the wing area.

\[
C_{d,q} = \frac{P}{0.5 \rho V^3 S} = 3.937 \cdot 10^{-6}
\]  

(10-1)

The profile drag reduction of the wing can be calculated with equation (10-2).
\[ \frac{C_{d_{\text{suction}}} + C_{d_{\text{d}}}}{C_{d_{\text{clean}}}} = \frac{8.175 \cdot 10^{-3} + 3.937 \cdot 10^{-6}}{8.810 \cdot 10^{-3}} = 0.928 \]  

(10.2)

For this test-case it can thus be concluded that BLS, applied over 45% of the leading edge of the created NLF wing at an AOA of 2°, reduces the profile drag by 7%. The weight of the system is subtracted from the maximum fuel weight, which becomes 14,324 kg (99.1 % of the original maximum fuel weight) [Dassault Falcon 7X, 2009].

Because other aerodynamic data of the Dassault Falcon 7X is unknown it is not possible to estimate the fuel savings. It must also be stressed that the analysis of Xfoil-suc is not intended for swept wings and speeds of Mach 0.8. Yet it was the best available analysis at this moment.
10.3 **Geometry Results**

For indication purposes only, some geometry results for different concepts settings are shown in Figure 10-6 to 10-10.

---

**Figure 10-6 Divisions, per section configuration**

---

**Figure 10-7 Divisions, per division configuration**

---

**Figure 10-8 Double layer, combined configuration**
Figure 10-9 FAFC configuration with ONERA-CTY and Pump per section

Figure 10-10 FAFC configuration with ONERA-LS and bleed-air
Chapter 11

Conclusions and Recommendations

This chapter presents the conclusions that can be made from this thesis report (11.1) and evaluates the goals set for the product model (11.2). At last recommendations are made for further research (11.3).

11.1 Conclusions

The research goals for this thesis were:

1. **Support the system architecture design for HLFC and FAFC on the Smart Fixed Wing Aircraft using Knowledge Based Engineering.**
2. **Demonstrate the capabilities and benefits of Knowledge Based Engineering in a multi-disciplinary conceptual design by creating an autonomous KBE application.**

To reach these goals a parametric model was created in the KBE system Genworks GDL. This model is able to create the geometry for an HLFC and FAFC system with a pneumatic supply system. Multiple system architecture concepts have been identified and a selection of these has been made available in the parametric model. The model is able to construct these system architectures on an aircraft created by the DARwing Multi Model Generator.

The parametric model uses an object oriented structure in which divisions are made between component and functions. Such that each class in the structure performs its own specific task. The structure has, in the development of this thesis, proven to easily allow extensions and improvements to the model.

Various analysis disciplines have also been incorporated into the GDL environment, which allows close coupling between the analysis and geometry construction. They analyse the geometry created by the parametric model, but also advise geometry improvements. The three disciplines that have been incorporated are internal aerodynamics, weight and cost. The predictions of the internal aerodynamic analysis for the pneumatic system have been compared to the compressible pipe-flow analysis...
program AFT Arrow. For straight segments with flow speeds below 60 m/s a difference below 3% was found\(^1\). For 90° bend segment the results were found to be within 15% of each other. For the non-90° case the results varied more, around 20% for bend-angles between 45° and 180°. For the junctions the results varied more, up to 35%. To conclude: the results for the internal aerodynamic analysis show quite some variation, especially at the junctions, but could still be used to perform a detailed trade-off between the various architecture concepts.

External aerodynamic analysis is also made available to the model through interfaces to Xfoil and Xfoil-suc. The interface to Xfoil already existed, but it robustness has been improved. The interface to Xfoil-suc has been developed to perform suction analysis and to determine suction needed. The interface allows the product model to predict the total performance benefits of an HLFC conceptual design. Although Xfoil and Xfoil-suc are not intended for transonic speeds, it does complete the product model to operate autonomously.

The example solution has shown the results that can be expected from the model. A real trade-off between the various concepts could not be performed due to the lack of available information, e.g. aircraft type.

It can be concluded that both goals set have been partially reached.

1. A product model has been created that constructs and evaluates system architecture for both Smart Technologies, even though the aircraft is still unknown. The system architecture is, however, still limited to pneumatics.

2. The created product model can analyse the concepts on weight, cost and internal aerodynamics. With the interface to Xfoil-suc a more autonomous KBE application is created that can design and evaluate the HLFC system on a given wing.

\section*{11.2 Product model conclusions}

The requirements set out for the product model in section 2.2.3 have mostly been reached.

The requirements with respect to the documentation have been satisfied. A manual has been written and included in Appendix I. Most objects have been commented using the YADD console; some screenshots of this are shown in Appendix G. Complex steps inside the classes are in most cases described by a few extra comments, but can be difficult to understand for other programmers.

With respect to the user friendliness, most requirements have been reached as well. Incorrect inputs are spotted and automatically corrected. If errors still occur they are mostly detailed and often give an explanation how to resolve. The input can be given in .dat files and outputs are given in .htm format that is also accepted by Microsoft Excel.

\footnote{Typical flow speeds in the pneumatic system where found to be below 60 m/s.}
A routine has also been incorporated to allow multiple input sets to be evaluated consecutively.

11.3 Recommendations

At the moment of writing the Clean Sky JTI is still in its early stages. Many decisions are yet to be made and technologies are to be further developed. This has led to many unknowns and limited the results from the product model. When more information does come available from other partners in the project, the model can demonstrate its potential by evaluating the different technologies and suggesting appropriate supply systems.

In the first chapter a priority list was made of disciplines that could be incorporated, five of those have been incorporated. To support the system architecture design of the SFWA better it is recommended to incorporate more disciplines. There is also still room for improvement of the current disciplines.

The following recommendations are made with respect to other disciplines:

- **Safety and Reliability**, currently a safety and reliability KBE tool is developed at the Delft University of Technology. To incorporate this tool a similar approach as done for cost and weight could be used.
- **Electrical supply system**. Fokker Elmo has shown interest in the parametric model to start up the electrical design layout. For this the pumps and actuators are to be extended such that they provide information on their electrical supply requirements.
- **De-contamination**. Contamination on the wing can decrease the performance of HLFC significantly. Various types of de-contamination systems have been suggested and researched in the past [Thiede, 2000]. To review the impact of a de-contamination system, it should be incorporated in the model. This will most likely require a significant amount of work, but will decrease some of the concerns for HLFC.
- **Space**. Once the aircraft type is identified the available space should be taken into account, already routines could be added that route the pneumatic system around other systems.
- **Control**. Especially the FAFC system needs a control system to monitor and adapt its workings. This requires a set of sensors and controllers to be incorporated to the model.
- **Simulate flight manoeuvres**. To evaluate the performance benefits of the systems during operation, complete flight manoeuvres should be simulated. Models to simulate this are available in MATLAB and could possibly be accessed through an interface.
- **Bleed-air impacts**. Incorporate more detailed analysis to the performance decrease of the engines when bleed-air is used.
The following recommendations are made with respect to the current disciplines:

- **Cost.** Extend the cost with more production costs. Currently costs are purely based upon the price of (sheet)material, while in reality production and assembly will also contribute to the total price. The architecture concept with sub-manifolds will loosen tolerances and possibly decrease costs, to accurately evaluate these this should be incorporated in the cost analysis.

- **Internal Aerodynamics in bends.** Current non 90° bend sections are evaluated using a linear relation. Other more accurate methods to predict the losses in non 90° bends are available. To incorporate such a relation a method must be developed that identifies the total length of a bend and can handle bends with a non-constant radius.

- **Internal Aerodynamics in T-sections.** The pressure drop inside a T-section with common flow passing through the lateral branch is currently unverified. If this type of junction is used in future models its relations should be reconsidered and verified.

- **Pumps.** Currently only one type of pump is created, ideally this should be extended small database of pumps. Routines could be included that automatically select an appropriate pump.

- **Suction distribution realised.** Currently the product model assumes a constant pressure above each skin-chamber. In reality this varies, the suction velocity will therefore vary along the chord of skin-chamber. To create a more detailed result of the realised suction distribution, this varying pressure should be incorporated into the routine.

- **Different flight conditions.** An evaluation routine should be created to determine the obtained suction distribution under different flight conditions. An approach for this is described in Appendix F.

Besides the recommendations for the disciplines also a few recommendations can be made with respect to the GDL-code

- The *f-shaped-pipe-path* occasionally suffers from trimming errors, mainly occurring with the :center pump-location and small connection-angles. This needs to be resolved to improve the robustness and user-friendliness of the model.

- **Concepts.** The priority analysis of the concepts in Chapter 3 and 5 have shown a few concepts with a reasonably good score that have not been implemented, mainly due to time restrictions. To give these a fair change it is recommended to incorporate some of these concepts as well.

- **Inlets and outlets.** Create inlet and outlet slots on the fuselage.

- **Junctions.** Create the outer surface of the junctions in the pipe-assembly.
References


References


Appendix A

Inputs used for the example solution

Table A-1 Default inputs for HLFC test-results

Written at 12:38:17 on 7-28-2010

<table>
<thead>
<tr>
<th>Name-of-the-run</th>
<th>suction-assembly-per-section</th>
</tr>
</thead>
<tbody>
<tr>
<td>HLFC version</td>
<td>6.0 27-07-2010</td>
</tr>
<tr>
<td>airfoil-data</td>
<td>D:\GDL\SFWA\DATA-INPUT\airfoils\DLR-airfoil-1.dat</td>
</tr>
<tr>
<td>c_p-source</td>
<td>xfoilsuc</td>
</tr>
<tr>
<td>C_q-source</td>
<td>Xfoilsuc</td>
</tr>
<tr>
<td>suction-file</td>
<td>D:\GDL\HLFC\DATA-INPUT\DLR-suction-1.dat</td>
</tr>
</tbody>
</table>

flight conditions:

<table>
<thead>
<tr>
<th>p_inf</th>
<th>26190</th>
</tr>
</thead>
<tbody>
<tr>
<td>rho_inf</td>
<td>0,409687951</td>
</tr>
<tr>
<td>T_inf</td>
<td>228</td>
</tr>
<tr>
<td>H</td>
<td>10000</td>
</tr>
<tr>
<td>Mach</td>
<td>0,80</td>
</tr>
<tr>
<td>R</td>
<td>287,053</td>
</tr>
<tr>
<td>alfa-in</td>
<td>2</td>
</tr>
<tr>
<td>C_l-in</td>
<td></td>
</tr>
</tbody>
</table>

KEYWORDS

<table>
<thead>
<tr>
<th>suction-manifold-layout</th>
<th>single-layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>no-of-submanifolds</td>
<td>3</td>
</tr>
<tr>
<td>no-of-holes-per-submanifold</td>
<td>2 2 2</td>
</tr>
<tr>
<td>no-of-manifold-divisions</td>
<td>1 1 1</td>
</tr>
<tr>
<td>pipe-assembly-key</td>
<td>per-section</td>
</tr>
<tr>
<td>pipe-assembly-start-location-key</td>
<td>root-direction</td>
</tr>
<tr>
<td>pump-types</td>
<td>type-1 type-1 type-1</td>
</tr>
<tr>
<td>suction-chamber-detail</td>
<td>0,025</td>
</tr>
<tr>
<td>Suction-chamber-distribution-method</td>
<td>Smart</td>
</tr>
</tbody>
</table>

PARAMETERS per section

<table>
<thead>
<tr>
<th>span-distribution-list-input</th>
<th>(0.1 0.25) (0.3 0.45) (0.6 0.75)</th>
</tr>
</thead>
<tbody>
<tr>
<td>suction-manifold-hole-location-chord</td>
<td>(0.5 0.5) (0.5 0.5) (0.5 0.5)</td>
</tr>
<tr>
<td>suction-manifold-hole-location-span</td>
<td>(0.2 0.8) (0.2 0.8) (0.2 0.8)</td>
</tr>
<tr>
<td>connector-pipe-initial-diameters</td>
<td>(50 50) (50 50) (50 50)</td>
</tr>
<tr>
<td>main-pipe-chord-locations</td>
<td>0,25 0,25 0,25</td>
</tr>
<tr>
<td>main-pipe-maximum-diameters</td>
<td>100 100 100</td>
</tr>
<tr>
<td>outlet-slot-ranges</td>
<td>(0.3 0.35) (0.3 0.35) (0.3 0.35)</td>
</tr>
<tr>
<td>Suction-chord-locations</td>
<td>(0.05 0.25) (0.05 0.25) (0.05 0.25)</td>
</tr>
<tr>
<td>Parameter</td>
<td>Value</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>perforated-surface-thickness</td>
<td>0.6</td>
</tr>
<tr>
<td>perforated-surface-hole-ratio</td>
<td>0.0078</td>
</tr>
<tr>
<td>perforated-surface-material</td>
<td>titanium grade 2</td>
</tr>
<tr>
<td>skin-chamber-thickness</td>
<td>10</td>
</tr>
<tr>
<td>skin-chamber-sheet-thickness</td>
<td>0.6</td>
</tr>
<tr>
<td>skin-chamber-sheet-material</td>
<td>titanium grade 2</td>
</tr>
<tr>
<td>submanifold-thickness</td>
<td>10</td>
</tr>
<tr>
<td>submanifold-material</td>
<td>titanium grade 2</td>
</tr>
<tr>
<td>submanifold-sheet-thickness</td>
<td>0.6</td>
</tr>
<tr>
<td>suction-manifold-thickness</td>
<td>30</td>
</tr>
<tr>
<td>suction-manifold-sheet-thickness</td>
<td>0.6</td>
</tr>
<tr>
<td>suction-manifold-sheet-material</td>
<td>titanium grade 2</td>
</tr>
<tr>
<td>suction-manifold-shape</td>
<td>scaled</td>
</tr>
<tr>
<td>pipe-thickness</td>
<td>1</td>
</tr>
<tr>
<td>pipe-material</td>
<td>aluminium 6061</td>
</tr>
<tr>
<td>pipe-roughness</td>
<td>0.06</td>
</tr>
<tr>
<td>outlet-slot-material</td>
<td>titanium grade 2</td>
</tr>
<tr>
<td>outlet-slot-sheet-thickness</td>
<td>0.6</td>
</tr>
</tbody>
</table>
Appendix B

Get-value-from-formula

The get-value-from-formula class is created to solve equations that are not easily rewritten such as the Borda-Carnot equation. To do this it uses the geometry capabilities of the GDL environment. In short it plots the results of the formula in a graph together with a horizontal line and looks for the intersection-point. This is shown in Figure B-1.

![Figure B-1 get-value-from-formula curves](image)

The inputs needed for this object are described in Table B-1, with these the object will try to solve the formula using the steps shown in Figure B-2. A short description to these steps will be given in the following paragraphs.

The first step for the object is to create the plot. It divides the evaluation region (x-min to x-max) in a list of x-values using the step input. For every x-value it evaluates the formula and creates a point at (x, f(x), 0). Connecting these points with a curve gives the formula curve. In the same workspace a horizontal line is created at the result input.

Next intersection points between the two lines are searched. If none are found the evaluation region is shifted to the direction where the results where closest. Then the object is reinitiated again until at least one intersection point is found (more is also possible).
Because the formula curve is created from little steps, the x-value of an intersection point may not be precisely accurate. Therefore the formula is evaluated at this value for every intersection point found. If none of the results are accurate, the number of steps is increased by a tenfold and the object is reinitiated until at least one result is accurate enough.

At last if there are still multiple values that give accurate results the first is selected.

Table B-1 get-value-from-formula inputs

<table>
<thead>
<tr>
<th>Formula</th>
<th>The formula to be solved, dependent on one variable: ( y = f(x) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>result</td>
<td>The result of the formula required</td>
</tr>
<tr>
<td>x-min</td>
<td>The lower boundary for the first attempt</td>
</tr>
<tr>
<td>x-max</td>
<td>The upper boundary for the first attempt</td>
</tr>
<tr>
<td>accuracy</td>
<td>The % accurate the result using the found value should give.</td>
</tr>
<tr>
<td>Steps</td>
<td>The amount of points used to create the plot</td>
</tr>
</tbody>
</table>

Figure B-2 get-value-from-formula work-flow
Appendix C

Scaled-offset-surface

This class is created to avoid errors occurring in tight corners when creating an offset-surface. The offset-surface class, provided standard in GDL, uses the normal at every point as the direction to offset. To problems that can occur with this approach are clearly shown in the 2D example of Figure C-1.

![Figure C-1 Offset-surface problem]

To avoid this problem the scaled-offset-surface class uses one direction, rather than multiple normal vectors, to create the offset. It uses the following steps (also shown in Figure C-2).

1. Create planar sections
2. Scale the surface in the plane specified, for a wing this is typically the vertical direction
3. Place the surface at mean distance
4. Remove any access material using minimal-distance
5. Loft over resulting curves.

Although the scaled-offset-surface can also be applied to surfaces without sharp corners, it is advised to be only used when necessary. Mainly because the scaled-offset-surface will never be at the exact offset distance on all locations. For the nose of a wing, however, the results are fairly good.
Figure C-2 scaled-offset-surface steps

- Original surface section
- Scaled-offset-surface section
- End-points
- Minimum-distance points
Appendix D

Pipe-route

This appendix aims to give a more detailed explanation for the structure of the *pipe-route* object; the object that creates the pipe-segments and computes the flow through them.

To create a complete pipe network, the *pipe-route* object needs the following lists as inputs:

- Pipe-data-list, which is a list of property-lists. The characteristics of each individual pipe-segment are stored in these property-lists.
- Junction-list, which houses the information to which pipe from the pipe-data-list is connected to which other pipe(s)

The structure of the *pipe-route* object is dependent on this junction-list and will thus differ for each pipe network. How this junction-list is translated into a structure will be discussed first (D.1) followed by the principles used to compute the flow-conditions in the complete system (D.2).

### D.1 Junction-list

The junction-list is a list with the indexes of the pipe-data-list, each number represents the nth entry of the pipe-data-list. The junction-list has multiple levels, it contains lists within lists. Every level has one, two or three values: the first is the index of the current pipe-segment. The second and third are the junction-lists for the *pipe-route* objects of the next level. An example of the junction-list and how it is used through the levels is shown in Table D-1, the pipe layout for this junction list is shown in Figure D-1.
Table D-1 Junction-list split out

<table>
<thead>
<tr>
<th>Level in the structure</th>
<th>Pipe</th>
<th>1st child</th>
<th>2nd child</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st level</td>
<td></td>
<td>(0 3 (1 (5 4) (2 6)))</td>
<td></td>
</tr>
<tr>
<td>2nd level</td>
<td></td>
<td>3</td>
<td>(1 (5 4) (2 6))</td>
</tr>
<tr>
<td>3rd level</td>
<td></td>
<td>(5 4)</td>
<td>(2 6)</td>
</tr>
<tr>
<td>4th level</td>
<td>4</td>
<td></td>
<td>6</td>
</tr>
</tbody>
</table>

Figure D-1 Pipe layout matching the junction list

D.2 Flow-conditions

Every property-list in the pipe-data-list must contain a path for the pipe-segment. Besides this it may also hold information regarding the flow-conditions, the diameter and the inlet situation. A typical example of a pipe-data-list is shown below.

```
(list :path pipe-path0)
  (list :path pipe-path1 :diameter 20 :location :downstream :p 19000 :u 120.29 :rho 0.23 :T 228)
  (list :path pipe-paths2 :diameter 40 :location :downstream :p 19000 :u 120.29 :rho 0.23 :T 228 :S-before-entry 0.21543)
```

If the flow-conditions for a pipe-segment are not specified in the pipe-data-list, they are found using the flow-conditions at the parents and children. For the pressure and the diameter this is shown in Figure D-2 (suction) and Figure D-3 (blowing).
Suction

Junction-list = (0 (1 2 (3 4)))

Figure D-2 Flow-data example suction
Blowing

Junction-list = {0 1 2 (3 4)}

- Pipe flow evaluation
- Junction flow evaluation

Pipe 0

\[ p_{up}(0) \]

\[ s(0) = s(1) \]

\[ p_{down}(0) = p_{up}(1) \]

Pipe 1

\[ p_{up}(1) + p_{down}(0) \]

\[ s(1) = s(2) + s(3) \]

\[ p_{down}(1,0) = p_{up}(2) \]

Pipe 2

Flow-data from parent

\[ s(2) = s(3) + s(4) \]

Flow-data from pipe-data-list

Pipe 3

Flow-data from children

Flow-data from parent

\[ p_{down}(2) = p_{down}(0,0) \]

\[ p_{up}(2) = p_{down}(1,0) \]

Pipe 4

Flow-data from parent

Flow-data from pipe-data-list

\[ s(4) = s(3) + s(4) \]

\[ p_{down}(4) = p_{down}(3) \]

\[ p_{down}(4) = p_{down}(0,0) \]

Flow data from pipe-data-list

Figure D-3 Flow-data example for blowing
Appendix E

Obstruction and diameter reduction

When the junction in a pipe network is evaluated it is often found that the pressure in one of the connecting pipes needs to be reduced. This is can be done using two different approaches: either by introducing an obstruction into the pipe (E.1) or by decreasing the pipe-diameter (E.2).

E.1 Obstruction

Using an obstruction in a pipe-segment is a clean and effective method. A plate with an orifice is placed in the middle of the segment. The size of the orifice is calculated with the Borda-Carnot equation (3-6), solved by get-value-from-formula.

How junction evaluation works together with pipes and obstructions for both the suction and the blowing case is shown in Figure E-2 and Figure E-3. The setup used is shown in Figure E-1.
E.1 Obstruction

![Diagram for Obstruction use with suction](image)

**Figure E-2 Obstruction use with suction**

![Diagram for Obstruction use with blowing](image)

**Figure E-3 Obstruction use with blowing**
E.2 Reducing the pipe diameter

Although an obstruction is easy to use and implement, it is not the most optimal solution. It uses extra material and oversized pipes. Using the pipe-diameter to create the extra pressure drop gives a weight and probably cost decrease. Unfortunately it is not as easy to use. Changing the diameter has effect on many different places.

1. The junction evaluation changes, because the area of at least one pipe changes.
2. The equivalent length can change, due to the changing bend ratio $r/D$.
3. The absolute length changes, because the size of the junction changes.
4. The friction in the pipe-segment changes.
5. The flow-speed changes.
6. The friction at the pipe-inlet, if present, changes.

Because of this complexity this problem has to be solved by iterations of the DEE, shown in Figure E-4. The size of the new-diameter used for the next iteration is calculated using only the last three impacts. These are largely influenced by the pipe-diameter and do not require geometry manipulation. Still to determine the new-diameter three relations have an effect on each other need to be solved. To do this they are combined into one function that gives the relation between the pressure drop and the diameter: $\Delta p = f(D)$. Each individual function is stacked in a list ordered from upstream to downstream. Together with the upstream flow conditions this is converted into one single relation using the diagram in Figure E-5. With this relation the required diameter is solved using \textit{get-value-from-formula} and stored into a list to be used for the next iteration of the product model. The flow chart for this is shown Figure E-6. Where root_deciding refers to the last pipe of that tree section that did not require an extra pressure drop.

![Flow chart](image-url)
E.2 Reducing the pipe diameter

Diagram E-5: Combine-flow-relations diagram

Diagram E-6: Determine new diameters
Appendix F

Comparison between AFT Arrow and the horizontal momentum equations for equal area junctions.

Although the horizontal momentum equation for equal area junctions have already been verified with experimental data [Basset, et al., 2001], it is still interesting to compare them with the methods used by AFT Arrow. Especially for large branch angles, where AFT Arrow showed an unexpected trend. The setups used are similar to those used in Chapter 9.

The results for the total pressure drop over the complete section with joining flows at different branch angles are set out in Figure F-1. The flow conditions were set to match conditions typically found for HLFC (p=25,000 Pa, T=228K, u=10m/s, ID=50mm, absolute-roughness=0.06mm). It can be seen that the GDL model under-predicts the pressure loss with maximum 20% in comparison to AFT Arrow. Also shown is that the pressure loss for AFT Arrow remains constant at branch angles larger than 80°. This trend is not confirmed with by experimental data of Basset, et al. [2001], which could be a reason to question the relations used by AFT Arrow for large branch angles.

Using the same setup for separating flows gives the results shown in Figure F-2. The prediction of GDL and AFT Arrow show a very good match for branch angles between 30° and 90° (<15% difference). The pressure drops to the run (the straight leg) in both cases remain zero. Again AFT Arrow predicts questionable results for angles above 90 degrees. The pressure loss is predicted to remain constant, in contrast to values found in experimental data.

To conclude: the pressure losses predicted by the horizontal momentum equations and AFT Arrow show similar trends for branch angles up to 90° and their differences are within 20%.
Pressure losses in an equal area junction with suction

Pressure losses in an equal area junction with blowing
Appendix G

YADD console

This appendix gives a shows a few screenshots of the documentation written for the YADD console. This is a standardized method of documentation for GDL. The documentation is written in the classes and read out by the YADD console, which displays it in a structured view. Figure G-1 shows a part of the list of described classes and functions in the HLFC package. Figure G-2 shows a part of the description of the HLFC class and Figure G-3 of the get-value-from-formula class. Figure G-4 shows the description of the Goldstein-function.
Figure G-1 part of the list of described HLFC classes.
Object: HLFC (The HLFC Package)

Author
Marijn van den Berg

m.a.b.vandenberg@student.tudelft.nl

VERSION 6.0
27-07-2010

Description

This is the main assembly file for the Hybrid Laminar Flow Control design. It has a large input section, which is used to build up the wing and suction system.

This object creates the following parts:
HLF-distribution - using the HLF-distribution file
This method translates the DLR-suction input to a list with chord and suction values. As this input is translated from a graph to a discrete list, the detail of the chambers is given by the suction-chamber-detail, this gives the approximated chamber width for discretized sections.
clean-wing - using the HLF-wing file
This object creates a simple wing surface, with taper ratio and airfoil data input.
HLF-section - using the wing-splitter file
This object creates a section of the clean-wing using the boundary's if the chord-distribution and span-distribution.

HLF-system - using the HLF-assembly file
This object, creates the skin-chambers and suction manifold. It also determines the locations of the holes, used for the pipe-assembly routine.

Pipe-assembly - uses pipe-assembly
This method creates a path's for the pipe system and uses the pipe object to generate pipes along it. The pipe assembly consists of many small pipe sections. Sections stop at a split. And two new sections begin after the split. This way the pressure relations and pipe-sizes can be determined.

Text is
These objects are for developer purposes. It shows the user which direction is u and which is v.

Input Slots (optional)

alpha-in number
the angle of attack [degree]. either alpha-in or Cl-in must be nil

Cl-in number
the Cl for which the xfoil analysis is done. either alpha-in or Cl-in must be nil

clean-wing surface
the clean-wing-surface

Figure G-2 Part of the description of the HLFC class
**Object: get-value-from-formula (The HLFC Package)**

**Mixins:** base-object

## Author

Merijn van den Berg  
t: 0616416599  
e: m.a.b.vandenberg@student.tudelft.nl

**VERSION 2.1**  
22-03-2010

## Description

This method is designed to retrieve intersection points in a formula.  
It has no graphical purpose.

The method draws a line using a formula from xmin to xmax with a given amount of steps  
Afterwards this method can be used to obtain intersection points on this formula.

At intersection point x a check is performed to whether there are no one or more intersection points.  
If no intersection points are found the xmin and xmax are shifted to the left or right to find y  
(this will not work for parabolic formulas)

The intersection point is checked to whether it is accurate enough with the accuracy given (0.01 standard)  
If it appears to be not accurate enough a warning is given and the method is repeated with  
xmin and xmax around the earlier found intersection point and with 10x as much steps.

This makes it possible to solve complex equations, without having to rewrite the equation.

### Input Slots (optional)

**formula function**  
the formula to be evaluated

**formula-shape keyword**  
the shape of the formula. options are  
:linear  
:exponential (xmin must be positive)  
default is :exponential

**y number**  
the y value where x is searched for

### Input Slots (optional, settable)

**accuracy number**  
the accuracy of the answer. Default is 0.99

Figure G-3 the Get-value-from-formula-object
**Function: Hf::Delta_P_Goldstein-Function**

`delta_p_Goldstein-function number`

This function gives the `delta_p` according to the Goldstein function. Typically used for flow through a porous sheet.

**arguments:**
- `mu_s-over-mu_ref number`
  - Viscosity at the skin over the viscosity in the free-stream
- `rho_s number`
  - The density at the skin
- `w_s number`
  - The velocity of the flow through the skin

**optional arguments:**
- `a number`, Default Value: `a`
  - The constant A
- `b number`, Default Value: `b`
  - The constant B
- `rho_ref number`, Default Value: `rho_ref`
  - The density of the free-stream air

---

**Figure G-4 the description of the Goldstein-function**
Appendix H

Suction analysis at different flight conditions

The HLFC system uses a rigid structure to regulate the suction, which is optimized for cruise conditions. If the aircraft is found to fly under different conditions, a different suction distribution will be obtained. The created product model is currently unable to evaluate this distribution under non-cruise conditions. To do this extra analysis routines are needed. The relations for the internal aerodynamics will not change, but a different approach to use them is needed. It is therefore recommended to look at this analysis as an extra discipline that can added to the DEE.

Because the suction-distribution is unknown the flow conditions will have to be calculated backwards; starting at the outlet slot working back to the suction panels. Because the flow-rate and density are related to the suction-distribution an iterative process has to be used.

A recommended flow-diagram of the analysis is shown in Figure H-1. First a design solution is created using the cruise conditions. Once completed the suction-analysis-at-different-flight-conditions is started. First a guess has to be made regarding the flow-rate and density, this guess could be guided by the conditions during cruise. Next the discipline gathers all the useful information from the components: the pressure drop formula, cross-sectional area etc. With this the flow-conditions in the components under the current flight conditions are worked out. The result is a new obtained suction-distribution, flow-rate and density. If these differ too much from the guessed values a new iteration is performed, until the density and flow-rate are converged. The final result is the suction-distribution that is achieved under the current flight conditions.
Figure H-1 Function flow to determine suction distribution at different flight conditions

Cruise Conditions \(\rightarrow\) Product Model \(\rightarrow\) Design solution \(\rightarrow\) Performance

Current flight Conditions \(\rightarrow\) Suction-analysis-at-different-flight-conditions \(\rightarrow\) Guess \(p\) and \(Q\) \(\rightarrow\) Convergence? \(\rightarrow\) yes / no

Determine \(p\) and \(Q\)

\[\Delta p_1 = f(p, u)\]
\[\Delta p_2 = f(p, u)\]
\[\Delta p_n = f(p, u)\]

Input Operation Component Data Decision Output
Appendix I

Smart Fixed Wing Manual

This manual aims to give a short users guide to the Smart Fixed Wing product model created by the writer. More information to the structure and workings of the product model can be found in the thesis report of the writer: M.A.B. van den Berg, “Smart Fixed Wing Design using Knowledge Based Engineering”, Delft University of Technology, 2010.

First a small overview of the packages used in the product model is given and how these should be loaded. Next the inputs and the form of the inputs will be discussed. Third the outputs and output methods are addressed and at last some attention points are made.

I.1 Packages

The SFWA product model has multiple packages:

1. Cost&weight
2. Pipe-object (uses 1)
3. Xfoil
4. FAFC (uses 1 and 2)
5. HLFC (uses 1, 2 and 3)
6. SFWA (uses 1, 2, 3, 4 and 5)

For full functionality thus all packages will have to be loaded. If one wishes to use a package separately often other packages are also needed, as indicated in the list. In fact the HLFC and FAFC packages are designed to run by themselves. The SFWA package can be seen as the DEE engine around them. It provides user-friendliness and an optimization routine.

To load all execute:
I.2 Inputs

Most of the objects inside the SFWA product model have been created such that they can be operated individually. To do this all input fields in these objects have been given default values. The purpose is to make it easier for other users to inspect and possible improve specific objects. Inputs in this documented are always indicated by a : and italic script, e.g. :clean-wing-input.

I.2.1 Wing and cp-profile

The two most fundamental inputs to the SFWA product model are the :clean-wing-input and the :clean-wing-cp. The :clean-wing-input uses per default a representation of the Dassault Falcon 7X. The :clean-wing-cp default depends on the :cp-source input (a complete list of the inputs can be found in tables starting at page 132).

- :airfoil-data. The default :clean-wing-cp is the cp-profile of the Dassault Falcon 7X created using the cp-profile inside the airfoil file. To create a correct cp-distribution multiple airfoil files should be specified.
- :xfoil. The default :clean-wing-cp is evaluated using the Xfoil program, with the :clean-wing-input as input.
- :Xfoilsuc. The default :clean-wing-cp is evaluated using the Xfoilsuc program. This is only possible of HLFC is also used.

The :clean-wing-input will have to satisfy the following standards:

- It must be a single-lofted-surface
- :u and :v parameters should be as indicated in Figure I-1 (u may also be in the reverse direction, this will be automatically corrected.) if u and v are swapped use the :swapped-uv-surface child of your surface.
  e.g. (the clean-wing-fokker swapped-uv-surface)
- x, y and z axis must be as shown in Figure I-1.
The :clean-wing-cp only requirement is to lie on over the wing and that the z-value is the cp-value. The z-value must be multiplied by 1/dimension-factor (explained later in this document).

### I.2.2 Fixed parameters

Some fixed parameters are stored in the META_OBJECTS folder of each package. Typical parameters are the following:

- Flight-conditions, descriptions can be found inside the file itself or the yadd-console. If the SFWA package is loaded the flight-conditions of HLFC and FAFC are automatically updated.
- The dimension. The dimension is a keyword that indicates which units are used for the model (options are :mm :cm :dm :m :km :inch :foot and yard). It is important that the clean-wing is given in the same units. If the SFWA package is loaded the dimension of cost&weight, pipe-object, HLFC and FAFC is automatically updated.
  Warning: All packages must use the same unit in their inputs!
- The accuracy, the accuracy of get-value-from-formula when used.
- The file-containers, the locations of the input and output locations.

### I.2.3 SFWA inputs

Because the SFWA object serves as the DEE it uses some very basic inputs, all are commented in the yadd console. Besides the :clean-wing and :clean-wing-c_p below is a list as well

- :show-FAFC & :show-HLFC, set to t to show the FAFC system or HLFC system. Once set the tree has to be created for the tasty console, therefore large parts of the model have to be computed.
- :using-input-file. When set to t the system will use input-files to create the FAFC or HLFC system.
- FAFC-input-file-name & :HLFC-input-file-name the file-names of the input files. If these are not given the system will look for a file named input-#.dat where # is the next data-set-number (more on this in the output section.
- :c_p-source the source of the :clean-wing-c_p when not specified.
- :alpha-in or :cl-in the angle of attack or the lift coefficient used by Xfoil or Xfoil-suc.
1.2 Inputs

The inputs in the input-files are listed below

**HLFC**

<table>
<thead>
<tr>
<th>Input</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:Name-of-the-run</td>
<td>The name under which the output is stored</td>
</tr>
<tr>
<td>:C_q-source</td>
<td>The source of the suction distribution</td>
</tr>
<tr>
<td>:Suction-file-name</td>
<td>Name of the .dat file with the set of corner points</td>
</tr>
<tr>
<td>:suction-chord-locations</td>
<td>List with the x/c locations where suction is allowed for every span-section</td>
</tr>
<tr>
<td>:Span-distribution-list-input</td>
<td>% regions on the wing e.g. ( {(0.2, 0.3), (0.5, 0.6)} )</td>
</tr>
<tr>
<td>:Skin-chamber-detail</td>
<td>The width of the skin-chambers e.g. ( 0.06 \bar{x}/c )</td>
</tr>
<tr>
<td>:Skin-chamber-distribution-method</td>
<td>:discrete, :smart</td>
</tr>
<tr>
<td>:Suction-manifold-Layout</td>
<td>:single-layer, :double layer</td>
</tr>
<tr>
<td>:no-of-submanifolds</td>
<td>Number of divisions in the middle layer</td>
</tr>
<tr>
<td>:no-of-holes-per-submanifold</td>
<td>List with the number of holes for each sub-manifold</td>
</tr>
<tr>
<td>:no-of-manifold-divisions</td>
<td>List with the number of manifolds after each skin-section</td>
</tr>
<tr>
<td>:Pipe-assembly-key</td>
<td>The layout of the duct system.</td>
</tr>
<tr>
<td>:Main-path-type</td>
<td>The method to create the main-duct-path</td>
</tr>
<tr>
<td>:Pipe-assembly-start-location-key</td>
<td>The location of the pump for each duct section</td>
</tr>
<tr>
<td>:Pump-types</td>
<td>List with the names of the pumps as keywords</td>
</tr>
<tr>
<td>:connection-angle</td>
<td>The angle in radians at which the branches come to the main-duct</td>
</tr>
<tr>
<td>:accuracy-c_p</td>
<td>The accuracy with which ( c_p ) is determined over a region ( 0.25 ) means that 4 points are taken</td>
</tr>
<tr>
<td>:accuracy-delta</td>
<td>The accuracy of the results of get-value-from-formula</td>
</tr>
<tr>
<td>:steps-delta</td>
<td>The initial amount of steps taken by get-value-from-formula</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>pipe-accuracy-delta</code></td>
<td>The accuracy of the results of get-value-from-formula for the pipe-sections</td>
</tr>
<tr>
<td><code>pipe-steps-delta</code></td>
<td>The initial amount of steps taken by get-value-from-formula for the pipe-sections</td>
</tr>
<tr>
<td><code>Number-of-sections</code></td>
<td>Number of sections for Xfoil</td>
</tr>
<tr>
<td><code>No-panels</code></td>
<td>No-panels for xfoil</td>
</tr>
<tr>
<td><code>Suction-manifold-hole-location-chord</code></td>
<td>List with % points for every manifold</td>
</tr>
<tr>
<td><code>Suction-manifold-hole-location-span</code></td>
<td>List with % points for every manifold</td>
</tr>
<tr>
<td><code>connector-pipe-initial-diameter</code></td>
<td>List with the diameters for the holes in every manifold.</td>
</tr>
<tr>
<td><code>Main-pipe-chord-locations</code></td>
<td>List with x/c locations for every main pipe</td>
</tr>
<tr>
<td><code>Main-pipe-maximum-diameters</code></td>
<td>List with the maximum diameter for every main pipe</td>
</tr>
<tr>
<td><code>Main-pipe-path-offsets</code></td>
<td>The offset from the airfoil surface for every main-pipe</td>
</tr>
<tr>
<td><code>outlet-slot-ranges</code></td>
<td>List with the x/c ranges for the outlet slot</td>
</tr>
<tr>
<td><code>perforated-surface-thickness</code></td>
<td>The thickness of the perforated surface</td>
</tr>
<tr>
<td><code>perforated-surface-material</code></td>
<td>The material of the perforated surface</td>
</tr>
<tr>
<td><code>skin-chamber-thickness</code></td>
<td>The thickness of the skin-chambers</td>
</tr>
<tr>
<td><code>skin-chamber-sheet-thickness</code></td>
<td>The thickness of the sheet-material of the skin-chambers</td>
</tr>
<tr>
<td><code>skin-chamber-sheet-material</code></td>
<td>The material of the sheets of the skin-chambers</td>
</tr>
<tr>
<td><code>submanifold-thickness</code></td>
<td>The thickness of the submanifolds</td>
</tr>
<tr>
<td><code>submanifold-material</code></td>
<td>The thickness of the sheet-material of the sub-manifolds</td>
</tr>
<tr>
<td><code>submanifold-sheet-thickness</code></td>
<td>The material of the sheets of the sub-manifolds</td>
</tr>
<tr>
<td><code>submanifold-shape</code></td>
<td>The approach taken for the sub-manifolds</td>
</tr>
<tr>
<td><code>scaled:following</code></td>
<td></td>
</tr>
</tbody>
</table>
## 1.2 Inputs

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:suction-manifold-thickness</td>
<td>The thickness of the manifolds</td>
</tr>
<tr>
<td>:suction-manifold-sheet-thickness</td>
<td>The thickness of the sheet-material of the manifolds</td>
</tr>
<tr>
<td>:suction-manifold-sheet-material</td>
<td>The material of the sheets of the manifolds</td>
</tr>
<tr>
<td>:suction-manifold-shape</td>
<td>The approach taken for the manifolds</td>
</tr>
<tr>
<td>:scaled :following</td>
<td></td>
</tr>
<tr>
<td>:pipe-thickness</td>
<td>The wall thickness of the pipes</td>
</tr>
<tr>
<td>:pipe-material</td>
<td>The material of the pipes</td>
</tr>
<tr>
<td>:outlet-slot-material</td>
<td>The material of the outlet-slot-sheets</td>
</tr>
<tr>
<td>:outlet-slot-sheet-thickness</td>
<td>The thickness of the outlet-slot-sheets</td>
</tr>
</tbody>
</table>

**FAFC**

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:Name-of-the-run</td>
<td>The name under which the output is stored</td>
</tr>
<tr>
<td>:Chord-locations</td>
<td>$x/c$ locations of the jets on the wing surface</td>
</tr>
<tr>
<td>:Span-locations</td>
<td>Regions for the actuators on the wing surface, e.g. <code>(0.2 0.3) </code>(0.5 0.6)`</td>
</tr>
<tr>
<td>:Delta-cl-input</td>
<td>The lift increase to be realized by the jets</td>
</tr>
<tr>
<td>:Actuator-type</td>
<td>The type of actuator to be used</td>
</tr>
<tr>
<td>:Pipe-assembly-key</td>
<td>The layout of the duct system.</td>
</tr>
<tr>
<td></td>
<td>:per section, :per division, :combined</td>
</tr>
<tr>
<td>:air-source-location-key</td>
<td>The type of source for the air</td>
</tr>
<tr>
<td></td>
<td>:bottom-wing-slot, :bleed-air</td>
</tr>
<tr>
<td>Engine-location-key</td>
<td>The location of the engine (currently only one option)</td>
</tr>
<tr>
<td></td>
<td>:rear-fuselage</td>
</tr>
<tr>
<td>:Main-path-type</td>
<td>The method to create the main-duct path</td>
</tr>
<tr>
<td></td>
<td>:on-chord, :at-offset</td>
</tr>
<tr>
<td>:Pipe-assembly-start-location-key</td>
<td>The location of the pump for each duct section</td>
</tr>
<tr>
<td>:center, :root-direction, :tip-direction, :fuselage</td>
<td></td>
</tr>
<tr>
<td>:Pump-types</td>
<td>List with the names of the pumps as keywords</td>
</tr>
<tr>
<td>:connection-angle</td>
<td>The angle in radians at which the branches come to the main-duct</td>
</tr>
<tr>
<td>:inlet-slot-ranges</td>
<td>List with $x/c$ locations for the inlet slot</td>
</tr>
<tr>
<td>:pipe-thickness</td>
<td>The wall thickness of the pipes</td>
</tr>
<tr>
<td>:pipe-material</td>
<td>The material of the pipes</td>
</tr>
<tr>
<td>:inlet-slot-material</td>
<td>The material of the inlet-slot-sheets</td>
</tr>
<tr>
<td>:inlet-slot-sheet-thickness</td>
<td>The thickness of the inlet-slot-sheets</td>
</tr>
</tbody>
</table>
1.3 Outputs

The SFWA package can crease output data without the needing to create a visualisation in the ta2 or tasty console. This is done by inspecting the following entries

- SFWA HLFC-data-output CREATE-OUTPUT
- SFWA FAFC-data-output CREATE-OUTPUT

When done so many data is gathered from the current configuration and stored in .htm format that is also readable by Microsoft Excel. If no specific input-file-name has been given but input-#.dat is used, the routine will check if there is an input with input-(#+1).dat is available. If it is, a new configuration is created with this new file and output is generated again. With this method a whole set of configurations can be evaluated. In this process the log.dat file is very important.

Log.dat

The first entry in the log.dat file is a number; this is the data-set number. Next time the product model is opened it will look for this number and increase it by 1 to get the next input file. If one wishes to start a new set of configuration this log.dat file is best to be moved/removed. As are the output files created, because two output files with similar names will unfortunately cause an error.

1.4 Pumps

Pumps are a special case in the product model. If an extra type is to be added to the model an extra object must be written. The pump-files are located in the PHYSICAL_OBJECTS folder of HLFC and FAFC. This object must hold the following entries

| :Pump | a brep of the pump |
| :pump-restriction. | The restrictions of the pump such as the maximum :flow-rate |
| :Pump-head-in | The head-delivered by the pump, could be given as a function of the :flow-rate |
| :Pump-power-in | The power needed by the pump, could be given as a function of the :flow-rate |
| :pipe-downstream-location-on-pump | the location of the downstream-pipe with respect to the center of the pump brep |
| :pipe-downstream-diameter | the diameter of the downstream-pipe |
| :pipe-downstream-vector | the diameter of the downstream-pipe |
| :cost-input | The price of the pump |
| :weight-input | The weight of the pump |
If an extra pump is added, two additional changes are needed

1. in air-outlet-assembly-wing-slot.lisp for the pump :type, the ecase form needs to be extended with the new entry
2. In HLFC.lisp or FAFC.lisp at the input-slot for pump-restriction the ecase form needs to be extended with the new entry

1.5 Actuators

The actuators are a special case as well. If an extra type is to be added to the model an extra object must be written. The actuator-files are located in the PHYSICAL_OBJECTS folder of FAFC. This object must hold the following entries.

<table>
<thead>
<tr>
<th>:body</th>
<th>a brep of the actuator</th>
</tr>
</thead>
<tbody>
<tr>
<td>:actuator-width</td>
<td>The width of the actuator</td>
</tr>
<tr>
<td>:actuator-length</td>
<td>The length of the actuator</td>
</tr>
<tr>
<td>:actuator-height</td>
<td>The height of the actuator</td>
</tr>
<tr>
<td>:jet-position-chord-%</td>
<td>The location of the jet position on top of the actuator</td>
</tr>
<tr>
<td>:p</td>
<td>The required pressure</td>
</tr>
<tr>
<td>:mdot</td>
<td>The mass-flow required, could be a function of the delta_cl needed</td>
</tr>
<tr>
<td>Pipe-location-on-actuator</td>
<td>Measured from the bottom front right corner</td>
</tr>
<tr>
<td>:cost-input</td>
<td>The price of the pump</td>
</tr>
<tr>
<td>:weight-input</td>
<td>The weight of the pump</td>
</tr>
</tbody>
</table>

If an extra actuator is added, two additional changes are needed

1. in FAFC-section.lisp for the actuator :type, the ecase form needs to be extended with the new entry
2. In FAFC.lisp for the actuator :type, the ecase form needs to be extended with the new entry.

1.6 Final Remarks

All though many incorrect inputs are filtered out, some errors might still occur. A few remarks for better operation of the product model are given below:

- When a set of configuration is used for output an out-of-memory error often occurs after around 5 to 10 configurations. The best action is to restart GDL and decrease the data-set-# in the log-file by one and inspect CREATE-OUTPUT again.
- Error: trimming u2 is smaller than u1, this error is found to sometimes occur when the main-duct path is split in separate segments. When the pipe-assembly-start-key is set to :center this is often solved by setting the
connection angle more towards $\pi/2$.
Also when using manifold-divisions make sure that no two pipes are on the same span-location.

For further questions feel free to email me

Merijn.vandenberg@gmail.com