

Design Automation of Air Intake Lips on an Aircraft

*How to implement design automation for air intake lips in a later
design concept phase*

Wilma Blixt
Hilda Schönning

Examiner: Johan Persson: *IEI, Linköping University*
Supervisor: Sanjay Nambiar: *IEI, Linköping University*

External Supervisors: Raghu Chaitanya Munjulury, Gustav Ehmke, Melker Nordqvist:
Saab Aeronautics

Abstract

Air intakes are complex components that are critical for the propulsion of the aircraft. The design has to consider requirements from several different departments, often contradictory. Additionally, the air intakes need to cooperate with other critical components. This makes testing of the models crucial, hence time-demanding. Design automation is a growing field which aims at minimizing repetitive work during product concept development. To follow the increasing digitalization, further investigations of design automation applied on air intakes are significant.

The application Imagine and Shape in 3D Experience CATIA handles subdivided surfaces. These surfaces are both flexible and provide a high order of continuity, which is often desired. While design automation in CATIA is well investigated, design automation in Imagine and Shape is not.

Knowledge based engineering techniques are often used to implement design automation. The methodology MOKA is frequently used when developing knowledge based engineering applications. This master thesis has followed MOKA in combination with Scrum.

The master thesis has resulted in a method to allow automation in Imagine and Shape by linking mesh nodes on subdivided surfaces to reference points that are parameterized. Further, a method for generating air intake configurations as well as the integration with a fuselage has been developed. The method includes wireframe models in Generative Shape Design, subdivided surfaces in Imagine and Shape, scripts in EKL as well as UserForm and scripts in VBA. Additionally, the order of continuity for an integration between air intakes and fuselage has been analyzed using tools in 3D Experience CATIA.

A conclusion drawn is that the method for generating air intakes cannot be completely automated. Instantiation and dimension of components can be automated, but manual work is required when using tools in Imagine and Shape during the integration between the components and the fuselage. Two methods for linking mesh nodes to reference points have been identified, one manual and one semi-automatic. The automatic method saves time and mouse clicks by utilizing VBA scripts. Further, the achieved order of continuity of an integration between subdivided surfaces depends on the individual components.

Acknowledgements

We would like to thank our supervisors Gustav Ehmke, Raghu Munjulury and Melker Nordqvist at Saab Aeronautics for their support and guidance throughout our project. Also, thank you for the opportunity to write our master thesis at Saab Aeronautics. We would also like to thank the employees at Saab who were helpful and participated in interviews as well as evaluation of our result.

At Linköping University, we would like to thank our supervisor Sanjay Nambiar for the academic inputs and help with our report. Also, a special thank you to our examiner Johan Persson for providing your opinion on our work.

Finally, we would like to thank our opponent Rebecka Thim for sharing your thoughts and feedback on improvements to our report.

*Wilma Blixt
Hilda Schönning
Linköping, June 2023*

Table of Contents

ABSTRACT	III
ACKNOWLEDGEMENTS	V
TABLE OF CONTENTS.....	VII
LIST OF FIGURES.....	XI
LIST OF TABLES	XIII
LIST OF ABBREVIATIONS	XV
1 INTRODUCTION	1
1.1 BACKGROUND	1
1.2 PURPOSE AND GOAL.....	2
1.3 RESEARCH QUESTIONS.....	3
1.4 TARGET USER.....	3
1.5 LIMITATIONS AND DELIMITATIONS.....	3
2 THEORETICAL FRAMEWORK	5
2.1 AIR INTAKE	5
2.1.1 <i>Subsonic and supersonic air intakes</i>	6
2.1.2 <i>Boundary layer</i>	6
2.2 AIRFOIL	7
2.3 DESIGN AUTOMATION	7
2.4 PRODUCT DEVELOPMENT PROCESS.....	8
2.5 MOKA	8
2.5.1 <i>Identify</i>	9
2.5.2 <i>Justify</i>	9
2.5.3 <i>Capture</i>	9
2.5.4 <i>Formalize</i>	9
2.5.5 <i>Package</i>	9
2.5.6 <i>Activate</i>	10
2.6 SCRUM.....	10
2.7 COMPATIBILITY MATRIX AND DESIGN STRUCTURE MATRIX.....	11
2.8 REQUIREMENT SPECIFICATION.....	12
2.9 CONTINUITY	12
2.10 3D EXPERIENCE CATIA	12
2.10.1 <i>Imagine and Shape</i>	13
2.10.2 <i>Visual Basic for Applications and Engineering Knowledge Language</i>	13

3	METHODOLOGY	15
3.1	IDENTIFY AND JUSTIFY	16
3.2	CAPTURE	16
3.2.1	<i>Literature study</i>	17
3.2.2	<i>Interviews</i>	17
3.3	FORMALIZE	17
3.4	PACKAGE AND SCRUM	18
3.4.1	<i>Design</i>	18
3.4.2	<i>Test</i>	18
3.4.3	<i>Develop</i>	18
3.4.4	<i>Integration</i>	19
3.5	ACTIVATE	19
4	IMPLEMENTATION	21
4.1	CAPTURE	21
4.1.1	<i>Interviews air intakes</i>	21
4.2	FORMALIZE	23
4.2.1	<i>Design Structure Matrix</i>	24
4.2.2	<i>Compatibility Matrix</i>	24
4.2.3	<i>Requirement specification</i>	25
4.3	PACKAGE	28
4.3.1	<i>Explore methods to adjust mesh nodes</i>	28
4.3.2	<i>Development of the air intake application</i>	31
4.3.3	<i>Integration between parts</i>	42
4.4	ACTIVATE	45
5	RESULT	47
5.1	EXPLORE METHODS TO ADJUST MESH NODES	47
5.2	MODELING AND AUTOMATION	48
5.3	CURVATURE ANALYSIS	54
5.4	EVALUATION OF IMAGINE AND SHAPE	55
6	DISCUSSION	59
6.1	METHOD DISCUSSION	59
6.2	EXPLORE METHODS TO ADJUST MESH NODES	60
6.3	MODELING AND AUTOMATION	60
6.3.1	<i>Requirement specification</i>	60
6.3.2	<i>Lips</i>	62
6.3.3	<i>Boundary layer diverter</i>	62
6.3.4	<i>KBE-application</i>	63
6.4	CURVATURE ANALYSIS	64
6.5	EVALUATION OF IMAGINE AND SHAPE	64
7	PERSPECTIVE	67
8	CONCLUSION	69
9	FUTURE STUDIES	71
10	REFERENCE	73

APPENDIX	77
APPENDIX 1 INFORMAL MODEL.....	77
APPENDIX 2 INTERVIEW QUESTIONS	80
APPENDIX 3 COMPATIBILITY MATRIX FOR AN AIRCRAFT.....	81
APPENDIX 4 ROTATION MATRIX	82

List of Figures

FIGURE 1. AN EXAMPLE OF AN AIR INTAKE ON THE SIDE OF A FUSELAGE.	2
FIGURE 2. SIMPLIFIED AIRCRAFT WITH HIGHLIGHTED COMPONENTS OF THE AIR INTAKE.	6
FIGURE 3. A SYMMETRIC AIRFOIL.	7
FIGURE 4. THE SIX STEPS OF THE PRODUCT DEVELOPMENT PROCESS ACCORDING TO ULRICH AND EPPINGER (2014).	8
FIGURE 5. THE SIX STEPS OF THE KBE CYCLE ACCORDING TO STOKES (2001).	9
FIGURE 6. THE SCRUM FRAMEWORK PRESENTED BY LAYTON (2015).	10
FIGURE 7. THE SPRINT PROCESS AS PROPOSED BY LAYTON (2015).	10
FIGURE 8. EXAMPLE OF A USERFORM WHICH ENABLES THE USER TO RUN SCRIPTS AFTER MAKING SEVERAL CHOICES REGARDING INPUT PARAMETERS.	14
FIGURE 9. IMPLEMENTATION OF MOKA WITH SCRUM.	16
FIGURE 10. THE CASE ON WHICH THE TWO LINKING METHODS ARE TESTED. THE ORANGE DOT REPRESENTS A MESH-NODE (N) ON THE SUBDIVIDED SURFACE, THE GREEN CIRCLE SHOWS A REFERENCE POINT (P_R) AND THE BLUE CIRCLE SHOWS A HELP POINT (P_H) CREATED WITH METHOD 2.	29
FIGURE 11. THE PROCESS OF METHOD 1. THE FIRST PICTURE SHOWS A CHOSEN MESH NODE (N). IN PICTURE 2 THE ROBOT IS REDEFINED TO THE LOCATION OF THE POINT (P_R). IN THE THIRD AND FORTH PICTURE THE ALIGNMENT PROCESS IS SHOWN. IN THE FIFTH PICTURE THE RESULT FOR ONE N IS SHOWN.	30
FIGURE 12. THE RESULT OF BOTH MESH NODE METHODS. A HIGHER RESOLUTION IS ACQUIRED THROUGH CREATION OF MORE P_R	31
FIGURE 13. THE SIMPLIFIED PLANE USED FOR MODELING OF THE LO-FI AIR INTAKES THE AIR INTAKE POSITIONS ARE MARKED WITH RED. ...	32
FIGURE 14. THE FRAME MODEL FOR THE RECTANGLE AND SEMICIRCLE.	33
FIGURE 15. THE FRAME MODEL FOR THE OVAL, CIRCLE AND BEAN. THE BEAN SHAPE CONTAINS THE MEASUREMENTS: WIDTH, HEIGHT1, HEIGHT2 AND HEIGHT3.	33
FIGURE 16. RECTANGLE LO-FI AIR INTAKE POSITIONED ON THE SIDE OF THE FUSELAGE. THE AIR INTAKE IS ROTATED AROUND THE X-, Y- AND Z AXIS.	34
FIGURE 17. SEMICIRCLE LO-FI AIR INTAKE POSITIONED ON THE TOP OF THE FUSELAGE. THE AIR INTAKE IS ROTATED AROUND THE Y-AXIS. ...	35
FIGURE 18. BEAN LO-FI AIR INTAKE POSITIONED ON THE BOTTOM OF THE FUSELAGE. THE AIR INTAKE IS NOT ROTATED IN ANY DIRECTION. ...	35
FIGURE 19. OVAL LO-FI AIR INTAKE POSITIONED UNDER THE WING. THE AIR INTAKE IS NOT ROTATED IN ANY DIRECTION.	36
FIGURE 20. THE POINTS AND CURVES THAT FORM THE AIRFOIL MODEL.	37
FIGURE 21. THE IMA SURFACE FOR THE RECTANGULAR AIR INTAKE LIP MODEL. FROM THE LEFT: PERSPECTIVE FRONT VIEW, FRONT VIEW, PERSPECTIVE BACK VIEW.	38
FIGURE 22. THE AIR INTAKE LIP MODEL FOR THE SEMICIRCULAR SHAPE.	38
FIGURE 23. THE AIR INTAKE LIP MODEL FOR THE CIRCULAR SHAPE. CAN BE MADE INTO AN OVAL BY CHANGING THE DIMENSIONS FOR THE HEIGHT AND WIDTH.	39
FIGURE 24. THE AIR INTAKE LIP MODEL FOR THE BEAN SHAPE.	39
FIGURE 25. THE GREY SWEEPED SURFACE, CUT BY THE DARK ORANGE DEFAULT SKETCH AND LIGHT ORANGE HEIGHT SKETCH. THE RESULT IS THE GREEN SPLIT SURFACES.	40
FIGURE 26. THE REFERENCE POINTS FOR THE SURFACE OF THE BLD. THE ORANGE DOTTED LINE DEFINES THE BACK OF THE SHAPE.	40
FIGURE 27. THE OFFSET CURVES AND THE FILL WHICH IS CREATED BETWEEN THEM.	41
FIGURE 28. TO THE LEFT: A BEAN BLD WHICH IS NOT ROTATED AROUND ITS X-AXIS. TO THE RIGHT: BOTH A RECTANGULAR AND BEAN BLD. THE BEAN BLD USES THE DEFAULT SKETCH AND THE RECTANGULAR USES THE USERSKETCH.	41
FIGURE 29. INTEGRATION BETWEEN BLD AND LIPS FOR A BEAN SHAPE.	42

FIGURE 30. THE CLOSED FUSELAGE WHICH SHOULD BE USED FOR INTEGRATION WITH THE AIR INTAKE. THE FUSELAGE IS MIRRORED AT THE BLACK EDGE.	43
FIGURE 31. AN INTEGRATION BETWEEN A RECTANGULAR AIR INTAKE AND THE PROVIDED FUSELAGE AT SIDE POSITION.	43
FIGURE 32. AN EXAMPLE OF A PORCUPINE ANALYSIS. THE RED COMB SHAPE REPRESENTS THE RADIUS AND THE GREEN BOUNDARY REPRESENTS THE CURVATURE.	44
FIGURE 33. THE POINT, PLANE AND INTERSECT USED FOR CURVATURE ANALYSIS IS SHOWN. ON THE LEFT THE AIR INTAKE IS ANALYZED AND ON THE RIGHT THE INTEGRATION BETWEEN THE AIR INTAKE AND FUSELAGE IS ANALYZED.	45
FIGURE 34. THE STEP-BY-STEP PROCESS FOR GENERATING AND INTEGRATING AIR INTAKES THROUGH DESIGN AUTOMATION.	48
FIGURE 35. THE START MODEL WITH THE INITIAL COMPONENTS NEEDED TO RUN THE AIR INTAKE APPLICATION.	49
FIGURE 36. THE USERFORM THAT POPS UP WHEN STARTING THE APPLICATION.	50
FIGURE 37. THE USERFORM DISPLAYS THE CHANGEABLE PARAMETER FOR THE AIR INTAKE LIPS. THE TWO TABS ARE MARKED WITH (r), WHICH MEANS THAT A RECTANGULAR OR SEMICIRCULAR AIR INTAKE IS INSTANTIATED.	51
FIGURE 38. THE USERFORM DISPLAYS THE CHANGEABLE PARAMETER FOR THE BLD.	51
FIGURE 39. HALF A BEAN SHAPED AIR INTAKE ON THE TOP OF A FUSELAGE, AND AN OPENING IN THE FUSELAGE WHERE THE AIR INTAKE WILL BE INTEGRATED.	52
FIGURE 40. THE IMA SURFACE FOR THE FUSELAGE MODIFIED TO MATCH THE BACK OF THE AIR INTAKE.	53
FIGURE 41. A FULLY INTEGRATED BEAN SHAPED AIR INTAKE.	53
FIGURE 42. RESULT OF CURVATURE ANALYSIS FOR THE INTEGRATION BETWEEN THE AIR INTAKE AND FUSELAGE. THE PORCUPINE ANALYSIS IS DISPLAYED ON THE LEFT AND THE CONNECT CHECKER ON THE RIGHT.	54
FIGURE 43. RESULT OF A CURVATURE ANALYSIS FOR A RECTANGULAR AIR INTAKE. THE PORCUPINE ANALYSIS IS DISPLAYED ON THE LEFT AND THE CONNECT CHECKER ON THE RIGHT.	54
FIGURE 44. RESULT OF A CURVATURE ANALYSIS FOR THE PROVIDED FUSELAGE. THE PORCUPINE ANALYSIS IS DISPLAYED ON THE LEFT AND THE CONNECT CHECKER ON THE RIGHT. THE DARK BLUE INDICATES THAT G^2 CONTINUITY IS NOT FULFILLED.	55
FIGURE 45. LIPS AND BLD ROTATED AROUND Y-AXIS. THE LINKING OF MESH NODES IN THE LEFT BLD IS NOT WORKING PROPERLY.	57

List of Tables

TABLE 1. COMPATIBILITY MATRIX, 0 INDICATES COMPATIBILITY AND 1 INDICATES INCOMPATIBILITY. 3 IS USED ALONG THE DIAGONAL AS AN ATTRIBUTE CANNOT BE COMPATIBLE WITH ITSELF.....	11
TABLE 2. DSM. AN “X” INDICATES A DEPENDENCY BETWEEN THE ATTRIBUTES. NO COUPLINGS BETWEEN ATTRIBUTES SINCE ALL “X” BELOW THE DIAGONAL.....	11
TABLE 3. KEYWORDS SPECIFIED IN SEARCH ENGINES.....	17
TABLE 4. OVERVIEW OF THE INTERVIEWEES’ EXPERTISE AND THE NUMBER OF MEETINGS HELD WITH EACH INTERVIEWEE.	21
TABLE 5. INTERESTING INPUT PARAMETERS TO INCLUDE IN THE APPLICATION, INFORMATION OBTAINED DURING INTERVIEWS.....	23
TABLE 6. DSM FOR THE DESIGN OF AN AIR INTAKE. A DEPENDENCY IS MARKED WITH AN “X”.	24
TABLE 7. COMPATIBILITY MATRIX FOR AN AIR INTAKE. 0 MARKS COMPATIBILITY AND 1 MARKS INCOMPATIBILITY, 3 IS USED ALONG THE DIAGONAL AS AN ATTRIBUTE CANNOT BE COMPATIBLE WITH ITSELF. USING THE TEMPLATE CREATED BY EKLUND AND KARNER (2017).	25
TABLE 8. REQUIREMENT SPECIFICATION FOR THE LO-FI MODEL. A RANK OF 1 INDICATES A HIGH PRIORITY, 2 IS MEDIUM PRIORITY AND 3 INDICATES A LOW PRIORITY.....	26
TABLE 9. REQUIREMENT SPECIFICATION OF THE HI-FI MODEL. A RANK OF 1 INDICATES A HIGH PRIORITY, 2 IS MEDIUM PRIORITY AND 3 INDICATES A LOW PRIORITY.....	27
TABLE 10. COMMON IMA ACTIONS.....	28
TABLE 11. RESULT OF EVALUATION OF MESH NODE METHODS.	47
TABLE 12. NUMBER OF MOUSE CLICKS NEEDED FOR EXECUTION OF THE TWO METHODS FOR DIFFERENT NUMBER OF MESH NODES.	48
TABLE 13. THE POWERCOPIES CREATED TO ENABLE DESIGN AUTOMATION OF AIR INTAKES.....	48
TABLE 14. A SUMMARY OF THE POSITIVE AND NEGATIVE ASPECTS OF USING IMA.....	56

List of Abbreviations

ABBREVIATION	MEANING
3DX	3D Experience
BLD	Boundary Layer Diverter
DA	Design Automation
DSM	Design Structure Matrix
EKL	Engineering Knowledge Language
Hi-fi	High-fidelity
IMA	Imagine and Shape
KBE	Knowledge Based Engineering
Lo-fi	Low-fidelity
MOKA	Methodology and software tools Oriented to Knowledge-based engineering Applications
UDF	User Defined Feature
VBA	Visual Basic Application

one

1 Introduction

The core values *trust*, *drive* and *expertise* strive Saab AB to maintain their vision about a safe society (SAAB AB n.d.a). The value *drive* is focused on innovation (SAAB AB n.d.a) and to be leading in their field, which can be seen in their wide range of constantly developing products (SAAB AB n.d.b). Their products are complex and made of multiple components that are connected and therefore dependent on each other. This makes every change time-demanding; when making changes to complex products, even the smallest can take a long time to implement. Therefore, it is essential that the digital models' efficiency and reliability keep up with the increasing digitalized society and industry.

Design automation (DA) is a growing field where knowledge is constantly increasing and demanded in the industry (Sandberg 2015). DA enables a more flexible design and allows a freer design process by facilitating changes both early and later in the process (Sandberg 2015). DA eliminates repetitive work and thus allows resources to be redistributed which makes it possible to work more effectively (Sandberg et al. 2016). DA also ensures that the company using the technology can be at the cutting edge (Sandberg et al. 2016). *Knowledge based engineering* (KBE) techniques are often used to implement DA to minimize routine tasks (Stokes 2001).

1.1 Background

Saab, founded in Sweden 1937, creates civil-, as well as military-, products and services with the aim of keeping society and people safe (SAAB AB n.d.a). Today the company acts on the global market in four different business areas, which are Aeronautics, Dynamics, Surveillance and Kockums (SAAB AB n.d.c; SAAB AB n.d.d). This thesis is conducted at the department of Airframe structural design and will thus focus on the business area of aeronautics, the area that acts in research, development and production of aircraft systems. At the department of Airframe structural design, parts of the airframe structure are developed. One of the parts which must be considered when designing an aircraft is the air intake. An air intake is a duct which provides the engine with the right amount of airflow to generate thrust (Hünecke 2005). There exist a wide variety of air intake configurations as both the shape and placement can vary. An example of an

air intake can be seen in Figure 1. The engine arrangement and number of engines also affect the air intake. Today these configurations are modeled and designed manually which is a time-consuming process.

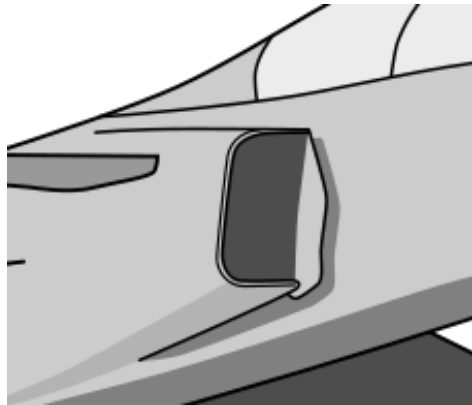


Figure 1. An example of an air intake on the side of a fuselage.

Saab aims to be leading in their field and therefore meets the customers' changing needs by constantly adapting and improving their products and services (SAAB AB n.d.c). Therefore, they want to explore how DA can be used to eliminate repetitive work when designing an aircraft, as well as to facilitate changes late in the design process. Usually when designing a product there is a lot of design freedom at the start of the project but little knowledge (Linköpings universitet n.d.). As the project proceeds and the acquired knowledge increases, the design freedom decreases, this is called the design paradox. One of the aims of DA is to increase the design freedom in the design process, which facilitate changes. Other advantages of DA include cost reduction, shorter lead time and more customized products.

Another way of staying at the cutting edge is to keep updated with new technology and be prepared to adopt new ways of working. Today the CAD-software CATIA V5 is used for concept development but an exploration of how the new software *3DEXperience* (3DX) can be used is desired. *Imagine and shape* (IMA) is a surface application in 3DX which ensures a high level of continuity between both lines and surfaces. Continuity describes the smoothness in transitions between curves or surfaces, where a high level of continuity is often desirable (Barsky & DeRose 1984). Therefore an evaluation of how IMA performs when used for concept development and DA is desired.

1.2 Purpose and goal

The purpose of the thesis is to facilitate the concept design phase by expanding the possibilities during concept development. This is done by creating an automated process for modeling of air intakes using IMA, 3DX and *Visual Basic for Application* (VBA). The automation includes the shape and placement of the air intakes, as well as their integration with the fuselage. This will also result in an evaluation of how IMA can be used for concept development in future studies and implementations.

The goal to develop a methodology for the automated process of the air intakes can be divided into following sub goals:

- Develop a method for the automation of the shape of the air intake.
- Develop a method for the automation of the placement of the air intake.
- Develop a method for an integration between the air intake and the fuselage.
- Explore methods to adjust mesh nodes.

1.3 Research questions

In order to reach the goal presented in *1.2 Purpose and goal*, three research questions have been defined which will be answered in the thesis. The research questions are presented below.

RQ1: How can an automated framework be implemented with subdivided surfaces in a CAD program through KBE to eliminate repetitive work and facilitate changes during the concept development of air intakes?

RQ2: How can the position of mesh nodes on a subdivided surface in a CAD program be adjusted?

RQ3: What order of continuity can be achieved for an integration between an air intake and fuselage in a CAD program?

1.4 Target user

The target user of the thesis result is a designer in a later concept phase who distributes the generated model to departments for calculations and analysis. The intention is to be able to perform calculations on the model generated by the application in order to quickly evaluate different concepts that are developed.

1.5 Limitations and delimitations

There are several limitations and delimitations to the project that ensure completion of the work within the available 20 weeks and satisfaction of the internal requirements at Saab. The limitations are presented below.

- The software that will be used is limited to the application IMA in 3DX and the programming languages are VBA and *Engineering knowledge language* (EKL).
- The focus of the thesis will be investigating the outer geometry of the air intakes, research about implementing DA of the inner geometry or inner components will thus not be carried out.
- There will be no investigation of how to apply DA on hypersonic air intakes.
- Variable air intakes will not be investigated.
- The work will be limited to an investigation of 1-2 engines, which requires 1-2 intakes.
- The design will be following the current methods used at Saab and will be adapted to fit the design of the provided start model of an aircraft.

two

2 Theoretical Framework

In this chapter, a theoretical basis for the work will be presented. The chapter will present theory about the functionality of air intakes and different types that exist. This is followed by information about DA and KBE, a section about the product development process, the methodology used and finally information about the CAD program and the programming languages.

2.1 Air Intake

The air intake is a key component of the aircraft as it provides the engine with air of the right velocity (El-Sayed 2016). The engine requires air to generate thrust and the intake is therefore essential for efficient propulsion. The right amount of air is necessary for activities on the ground, during takeoff and during all phases of flight (Hünecke 2005). Another aspect which is critical is the velocity of the incoming air, thus the air intake also has to deaccelerate the airflow to meet the engine's limitations. The intake should also be well integrated along with the rest of the components to ensure aerodynamic efficiency. An example of an air intake is seen in Figure 2. The capture area is the opening area of the air intake, the lips form the front part of the air intake geometry, and the *Boundary Layer Diverter* (BLD) is further explained in 2.1.2 Boundary layer. As there exist a wide range of flight conditions there also exist different types of air intakes, these can be divided into the following categories:

- Variable or fixed
- Subsonic, supersonic or hypersonic

An air intake can have either a fixed geometry or a variable geometry (El-Sayed 2016). A fixed geometry is, as the name implies, fixed, and a variable geometry has a geometry which is flexible and can move to adapt to different air flows.

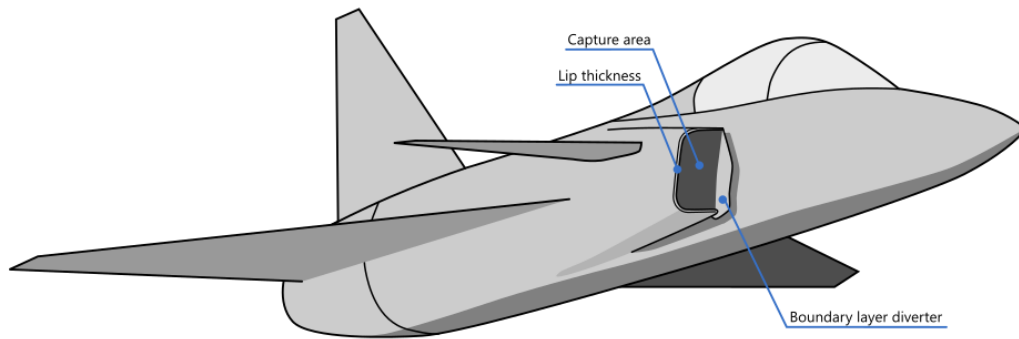


Figure 2. Simplified aircraft with highlighted components of the air intake.

2.1.1 Subsonic and supersonic air intakes

Subsonic air intakes are designed to operate at speeds which are lower than the speed of sound (Hünecke 2005). A measurement that is frequently used when talking about supersonic speed is the *Mach number*, which is the ratio between the velocity of a body and the speed of sound. Subsonic flight conditions have a Mach number lower than 1, while supersonic flight conditions have Mach numbers which are higher than 1. Flights at speed with Mach numbers higher than 5 classify as hypersonic. The task of deaccelerating the speed of the incoming air is even more critical when dealing with supersonic speed as the shock waves which are created have to be deaccelerated (Farokhi 2014). Compression shockwaves are created when the aircraft is moving faster than the speed of sound and large pressure gradients are created, often called *the sonic boom* (Hünecke 2005). Sound waves naturally arise in front of the aircraft as it moves at subsonic speeds, and the sonic boom occurs when a supersonic speed is reached as the sound waves spread behind the nose of the aircraft. The shock waves are slowed down by shocks induced at interaction with the air intake. Induced shocks can be both normal and oblique. The front of a normal shock is perpendicular to the air flow. This means that the flow will not change its direction but will decrease its speed, unlike flow which is influenced by oblique compression shock which changes its direction and keeps the supersonic speed. The lips of the supersonic intake should be thin, designed to avoid creating a normal shock wave in front of the air intake, as this can disturb the incoming airflow (El-Sayed 2016).

2.1.2 Boundary layer

A fluid mechanics phenomenon that needs to be considered when developing an aircraft – especially considering the air intakes – is the boundary layer. The boundary layer is a thin layer of fluid closest to a moving surface that occurs due to friction between the surface and the surrounding fluid (Schlichting & Gersten 2003). In this layer the velocity varies from zero – closest to the surface – to 99% (a frequent estimation) of the free stream velocity (Schlichting & Gersten 2003). Because of the low energy in the boundary layer, it is unwanted inside of the air intakes (Schlichting & Gersten 2003), and the performance of the air intake depends on how much of the boundary layer can be diverted from the intake (Farokhi 2014).

To prevent the flow of getting into the air intakes multiple solutions for controlling the boundary layer exist (Farokhi 2014). One of the solutions is a BLD, which averts the boundary layer from entering the air intake and allows the free stream to flow through the air intake (Svensson 2008). The functionality behind this is to create a gap between the fuselage and the air intake and divide the flow by placing a plate at the verge of the boundary layer (Svensson 2008). See an example of a BLD in Figure 2.

2.2 Airfoil

The cross section of a blade or wing profile is called an airfoil, which is often used when designing for aerodynamic purposes to create lift that is greater than the created drag (Torabi 2022). Usually airfoils are thin and they can be both asymmetric as well as symmetric. A symmetric airfoil and the basic terminology are shown in Figure 3. The front profile of the airfoil is called the *Leading edge* and comes into contact with the airflow first. At the other side of the airfoil a profile called *Trailing edge* is located. The *Root chord* connects the front and back profile of the airfoil and defines the length. For a symmetric airfoil the thickness from the root chord is symmetrical to the upper and bottom side. A commonly used set of airfoils is called NACA, where each airfoil is called NACA $XXYY$. The X 's and Y 's are numbers which define the characteristics of the airfoil. Symmetric airfoils are named NACA00 YY , where the Y 's denote the maximum thickness of the airfoil as a percentage of the root chord.



Figure 3. A symmetric airfoil.

2.3 Design automation

According to Tarkian (2012) DA aims at reducing manual and repetitive design tasks to increase the time spent on creative tasks, which results in a higher product quality. DA is typically a system which can be used to achieve a desired output by the use of a specified input (Amadori 2012). One type of DA is Geometric automation, which focuses on the reuse of already existing features, stored in CAD-templates, that can be instantiated to create new solutions (Tarkian 2012). Typically, these templates can be parametrically modified to create flexible solutions. *User Defined Feature* (UDF) and PowerCopies are frequently used CAD-templates which can be combined with scripting.

KBE is a method where knowledge is stored and reused through script and templates to reduce routine-like tasks (Stokes 2001). Because of this KBE can be seen as an enabler for DA, as the knowledge can be stored in terms of for example rules or templates (Amadori 2012). When applying DA two types of changes can be used. The first is topological changes, which are used

when adding instances and deciding their location in a system. The second is morphological changes, which is a parametric modification of the already instantiated instance's shape. These types of changes are usually combined to create High level CAD-templates.

2.4 Product development process

The product development process, which is presented by Ulrich and Eppinger (2014), consists of six steps. These steps describe the development of a product from planning and identification of customer needs to production ramp-up, see Figure 4. Concept development is the second step in the process and describes the process of generating and deciding what concepts should be further explored. According to Ulrich and Eppinger (2014) a concept development process should include the following tasks: identification of customer needs, establishment of target specifications, generation of concepts, selection of concept, testing of concepts, establishment of final requirements and planning for downstream development. According to Tarkian (2012) the concept development phase can be made more effective by application of KBE, as the repetitive work can be minimized.

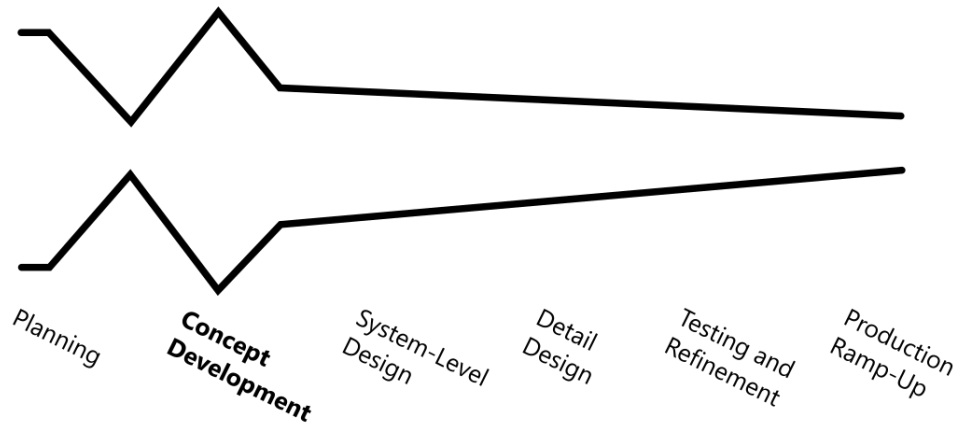


Figure 4. The six steps of the product development process according to Ulrich and Eppinger (2014).

2.5 MOKA

MOKA is an acronym for Methodology and software tools Oriented to Knowledge-based engineering Applications (Stokes 2001). The methodology aims at minimizing risks, time and costs associated with KBE through representing knowledge in the design phase. The methodology allows the engineer to be more productive as MOKA facilitates modularization of the work, and therefore enables reuse and modification of parts or models. The KBE lifecycle consists of six steps: Identify, Justify, Capture, Formalize, Package and Activate, see Figure 5. Capture and Formalize are the main focus of MOKA.

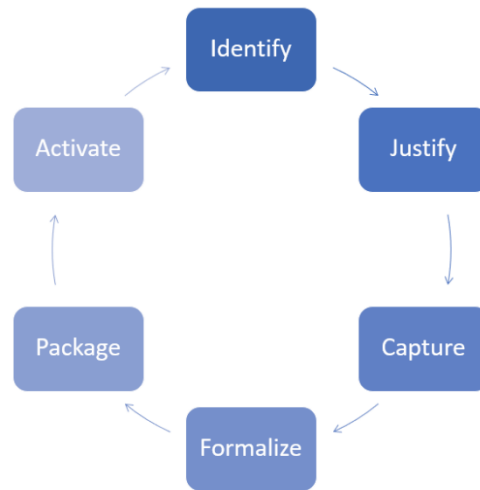


Figure 5. The six steps of the KBE cycle according to Stokes (2001).

2.5.1 Identify

As the name suggests this is the step where identification of opportunities, requirements and potential resources are conducted (Stokes 2001). Here the business needs and available knowledge resources are investigated. An evaluation of how KBE can be applied is also performed and the aim of the application is specified. This step should result in a technical requirement list.

2.5.2 Justify

In this step a project plan is formalized based on the identification gathered in the previous step, and a risk assessment is carried out (Stokes 2001). Here criteria for evaluation of the product are defined as well as an estimation of required resources.

2.5.3 Capture

Collection and structuring of information and knowledge are the main focus of Capture (Stokes 2001). This step uses the identified knowledge resources which are defined in Identify. The knowledge can be gathered from interviews, reports and documents. Lastly the information is structured in an informal model which can be used for representation of the acquired knowledge.

2.5.4 Formalize

A review of the project plan, time, cost and resources in greater detail allows for the second stage in the transition between raw knowledge and KBE (Stokes 2001). Here the informal model is developed into a formal model, which consists of two parts: the product model and the design process model. The design process model includes rules, activities and process flow, while the product model includes constraints and entities.

2.5.5 Package

A KBE system is constructed and a user interface is designed in the Package step (Stokes 2001). The formal model is the base of this construction.

2.5.6 Activate

The last step is Activate where the KBE is prepared for use (Stokes 2001). The preparations include distribution, introduction to new users and final use. This step also includes maintenance and evaluation of the application, which reactivates the life-cycle.

2.6 Scrum

Scrum is a framework which is frequently used during the Package stage of MOKA. For implementation of the scrum framework three different roles are needed, these being *the product owner*, *the development team*, and *the scrum master* (Layton 2015). The scrum master is responsible for a successful implementation of scrum. They should set up planning meetings for the scrum process, coach the sprint and evaluate the result from the sprint. The scrum framework consists of six steps, as shown in Figure 6, these are: Establishment of requirements, Product backlog, Sprint planning, Sprint, Sprint review and Sprint retrospective. The establishment of requirements result in the Product backlog which is a list of activities that should be completed during the sprint. After the product backlog has been decided a plan for the upcoming sprint is made in terms of duration, tasks, time estimation, and goals. The sprint is a 1–4 week period where development takes place. Usually, the sprint consists of a loop where the focus is: *designing*, *developing*, *integrating* and *testing*, as can be seen in Figure 7.



Figure 6. The scrum framework presented by Layton (2015).

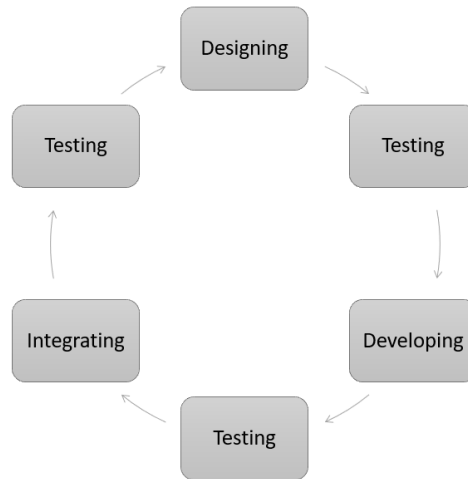


Figure 7. The sprint process as proposed by Layton (2015).

During the sprint there are daily scrum meetings where the members are informed about the activities of the day and the work is divided within the group (Layton 2015). At the end of the sprint a review of the previous sprint is conducted in terms of achievement of goals and

establishment of what tasks are left. Apart from this a meeting with the stakeholders is held to present the result of the sprint and the progress of the project. The last step is the sprint retrospective, where an assessment of how the activities can be improved before the next sprint is conducted. After this the loop continues.

2.7 Compatibility Matrix and Design Structure Matrix

During concept development some combinations of design features are not feasible due to constraints (Engler 2013). A Compatibility Matrix is a matrix which visually represents the possible design combinations for concept development, see a simplified example in Table 1. Aside from visualizing the compatible and incompatible combinations, the matrix can also provide information about which alternatives are dependent on each other. Each attribute in the Compatibility Matrix can have sub attributes which are compatible or incompatible with other sub attributes or features, which is not shown in the example in Table 1. However, for a quicker overview of the existing dependencies a *Design Structure Matrix* (DSM) can be conducted, see a simplified example in Table 2. A DSM can be used to structure the problem and represent which parameters require input from another parameter in order to generate an output (Persson & Ölvander 2021). The DSM provides a guide for the order of execution during concept development. To avoid coupled attributes reordering of the matrix is carried out, the aim is to get the “x” below the diagonal.

Table 1. Compatibility Matrix, 0 indicates compatibility and 1 indicates incompatibility. 3 is used along the diagonal as an attribute cannot be compatible with itself.

	Attribute 1	Attribute 2	Attribute 3	Attribute 4	Attribute 5
Attribute 1	3	0	0	1	0
Attribute 2	0	3	0	0	0
Attribute 3	0	0	3	0	0
Attribute 4	1	0	0	3	1
Attribute 5	0	0	0	1	3

Table 2. DSM. An “x” indicates a dependency between the attributes. No couplings between attributes since all “x” below the diagonal.

	Attribute 1	Attribute 2	Attribute 3	Attribute 4	Attribute 5
Attribute 1					
Attribute 2	x				
Attribute 3	x	x			
Attribute 4	x		x		
Attribute 5				x	

2.8 Requirement specification

A requirement specification should describe requirements and desired features for a concept (Liedholm 1999). Even though a requirement specification is dynamic and can change during the project when new information is discovered, it should be used both as a guide when developing a concept and a guideline when evaluating the result. An important aspect to consider when developing a concept is checking if the requirement specification aligns with the, by the customer, desired properties. The specification includes information about if the property is desired or required and the rank of the property from 1-3. The rank is used to determine which needs are the most important to consider.

2.9 Continuity

Continuity is a term used to describe transitions' smoothness between curves and surfaces (Barsky & DeRose 1984). There are both geometrical (G)- and parametric (C) continuity. To describe different grades of continuity, orders are used – here defined as the n^{th} order (Barsky & DeRose 1984). C continuity refers to continuity of the n^{th} order of derivative at the elements' meeting point, whereas G continuity is a less strict kind that, unlike C continuity, allows discontinuity as long as the curve or surface is geometrically smooth (Barsky & DeRose 1984).

The zeroth order (G^0) implies that curves are joined at an endpoint, but they do not take into account the curves tangent direction like higher continuity orders do (madhav_mohan 2022). G^1 continuity implies that the curves share tangent direction and G^2 continuity implies continuous curvature, thus the curves share curvature center (Comba 1993). The same principle applies for higher order of continuity G^n (Barsky & DeRose 1984).

2.103D Experience CATIA

3DX is a cloud-based platform that enables businesses to collaborate across their entire ecosystem (Dassault Systèmes n.d.a). The idea is to have one platform where all activities are gathered and where all people involved – people from different departments and disciplines, internal and external – can access data and work together (Dassault Systèmes n.d.a). This is made possible by combining various programs into 3DX, where one of the products is 3DX CATIA which is the next generation of the CAD program CATIA V5 (Dassault Systèmes n.d.b). 3DX CATIA supports a wide selection of applications connected to different roles made to meet the users' professional needs (Dassault Systèmes n.d.b; Dassault Systèmes n.d.c).

Generative Shape Design is an application in 3DX where basic operations such as points, lines, splines and planes are used to create models (Dassault Systèmes n.d.d). The application includes surface modeling tools that allow creation of surfaces and joining of surfaces.

2.10.1 Imagine and Shape

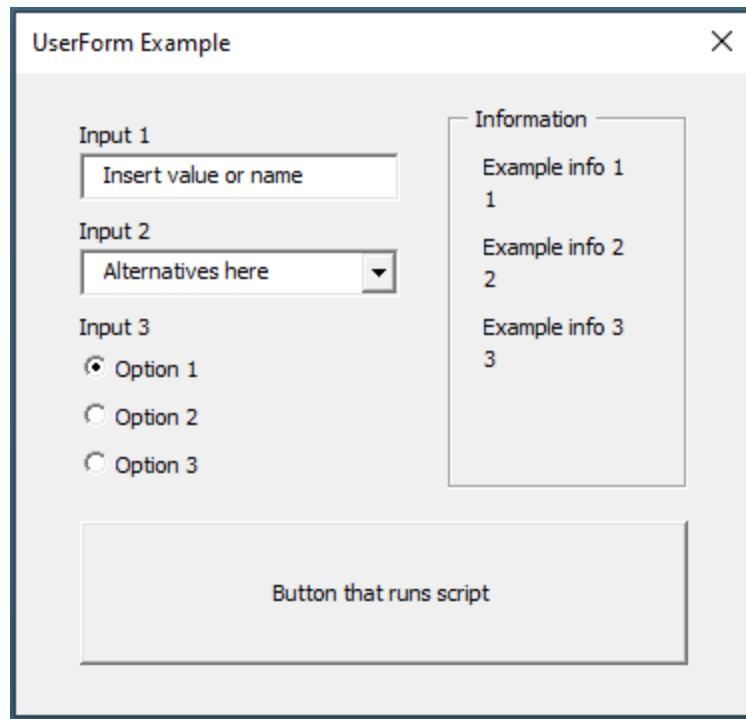
IMA is an application in 3DX CATIA that facilitates quick modeling and modification of subdivided surfaces (Dassault Systèmes n.d.e). The application allows the user to quickly realize ideas by modifying a mesh into desired shape, where the application automatically ensures high orders of continuity between curves and surfaces (Dassault Systèmes n.d.e). The surface can then be converted between different departments which facilitates the collaboration of the business (Dassault Systèmes n.d.e).

2.10.2 Visual Basic for Applications and Engineering Knowledge Language

VBA is a programming language created by and for Microsoft Office (Microsoft 2022a). It allows elimination of repetitive tasks and enables interaction with the user (Microsoft 2022a). The repetitive tasks can be reduced or eliminated by using macro scripts that automate computer processes, and when implemented into other programs such as CAD it can interact and act like an extension of the functions in that program (Kenton 2022). CATIA supports recording, editing and writing of macros which opens up the possibilities to automate processes in CATIA (DesignTech CAD Academy n.d.). However, the macro tools are limited and not supported in all CATIA applications, an example of this is the IMA application (Dassault Systèmes n.d.f).

CATIA has knowledgeware applications that enable the user to integrate knowledge in the design, thus allow control of the models through several tools including rules, checks and reactions (Lohith et al. 2013). This is provided through the use of EKL (Lohith et al. 2013). Another tool in the spectrum of the knowledgeware capabilities is Knowledge Pattern which can be used for instantiation of templates such as UDFs (Dassault Systèmes n.d.g).

A UserForm allows the user to navigate through the script by providing a user interface with options which control the output of the script (Microsoft 2022b). An example of a Userform is found in Figure 8. The inputs, which can be modified by the user, can be provided in textboxes which allow the inputs such as strings or more commonly used: numbers. The inputs can also be provided in drop-down lists where the user can choose an alternative. Another input can be clickable options, where the user can choose one alternative from a set of predefined options. A UserForm usually includes a button which executes the script upon click and information about the output of the script.



The image shows a Windows-style dialog box titled "UserForm Example" with a close button (X) in the top right corner. The dialog is divided into two main sections. On the left, there are three input fields: "Input 1" with a text box containing "Insert value or name", "Input 2" with a dropdown menu showing "Alternatives here", and "Input 3" with three radio button options: "Option 1" (selected), "Option 2", and "Option 3". On the right, there is a section titled "Information" containing three lines of text: "Example info 1", "Example info 2", and "Example info 3". At the bottom of the dialog, there is a large rectangular button labeled "Button that runs script".

Figure 8. Example of a UserForm which enables the user to run scripts after making several choices regarding input parameters.

three

3 Methodology

This chapter describes the application of the described methodologies with focus on the implementation of MOKA alongside Scrum. Scrum is applied at the implementation step, called Package, of the KBE-lifecycle and is adapted to fit the project.

The main method used in the project is MOKA from which the KBE-lifecycle is applied. MOKA allows the creation of fast and flexible concepts of air intakes by implementing KBE. The lifecycle is implemented as a linear method consisting of the steps Identify, Justify, Capture, Formalize, Package and Activate. The method Scrum is used as an additional method during the Package step to ensure a successful development of the application. Scrum is used as a continuous loop which is iterated until a final application is created. An overview of the used method can be seen in Figure 9.

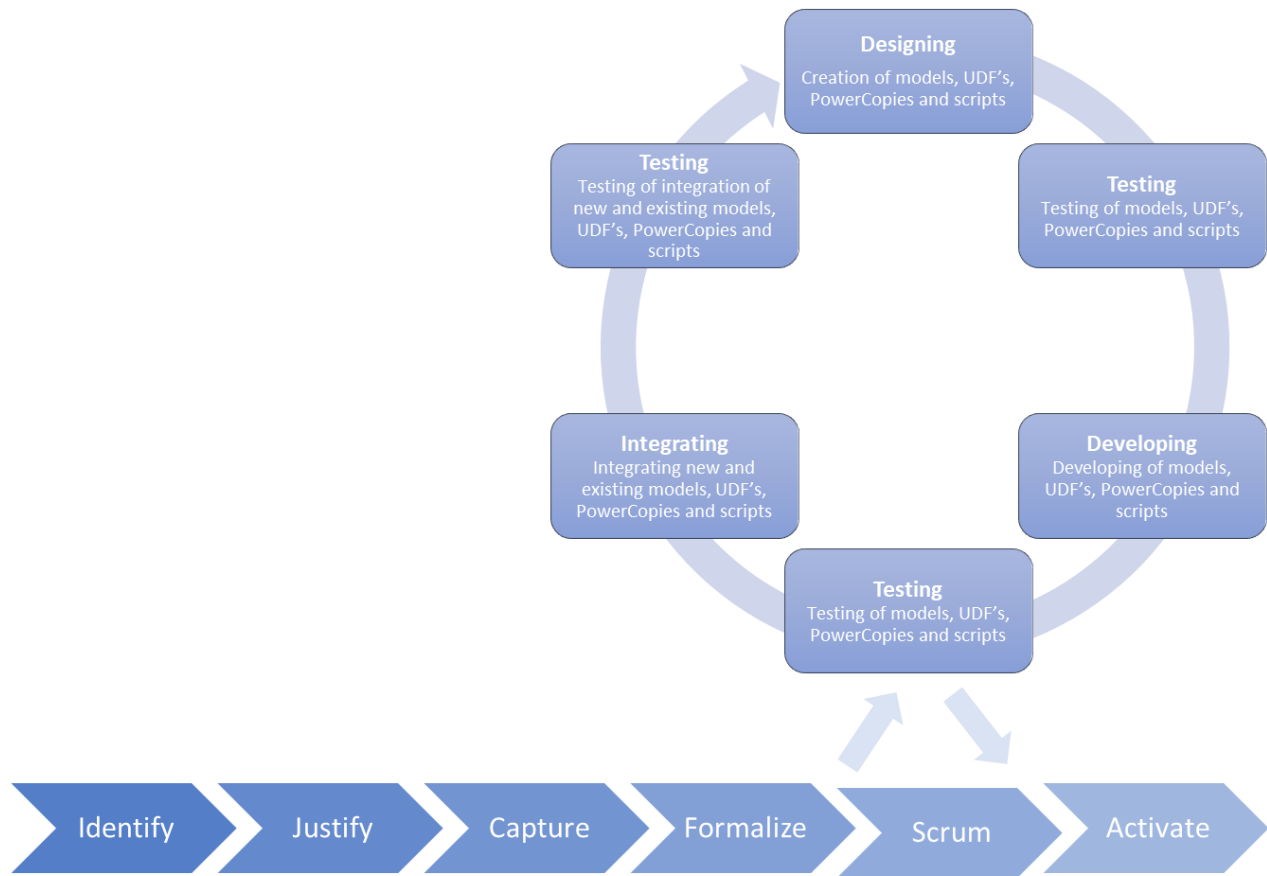


Figure 9. Implementation of MOKA with Scrum.

3.1 Identify and justify

The Identify and Justify steps are implemented during the first part of the project where the scope of the project is not yet clearly defined. An investigation of the aim of the project, the project goals and limitations as well as delimitations is conducted alongside the stakeholders of the project. The investigation includes meetings with stakeholders from Saab and Linköping University and a brief literature study for an increased understanding of the scope of the project. A project plan is established based on the information that is captured during the investigation; the main parts of the project plan can be found in 1 Introduction. These steps also include identification of available knowledge resources, both in terms of human resources, available methods and relevant literature.

3.2 Capture

In MOKA, Capture focuses on building an informal model through knowledge capture. The informal model should provide the necessary information for a successful implementation of KBE when building a DA model. The base of the knowledge capture is the identified knowledge resources from the previous steps of MOKA, although new knowledge resources are identified during the implementation of Capture. The collection of information includes both a literature study and interviews with experts.

3.2.1 Literature study

The literature study is conducted to increase the understanding of air intakes, their purpose and the configuration of existing air intakes. Other areas of investigation are how design automation could be applied to similar projects and potential methods. Initially basic literature resources such as introductory books to aircraft development and previous master theses about the use of KBE for similar applications are studied. Some literature is provided by an expert interviewee at Saab. Additional literature is found by searching the database UniSearch and using the available resources at the Linköping University library. Some of the keywords used can be found in Table 3. The acquired information is compiled in 2 Theoretical Framework.

Table 3. Keywords specified in search engines.

Search engine	Keywords
UniSearch	Air intake, air inlet, aircraft design, engine propulsion, supersonic speed, design automation, knowledge based engineering, MOKA, sonic boom, airfoil
Google Scholar	Air intake, boundary layer diverter, boundary layer, interview structure, geometric continuity, continuity order

3.2.2 Interviews

Semi structured interviews are conducted to complement the information collected by the literature study and to further explore the potential future user of the final result of the thesis. A few questions are prepared leading up to the interviews as semi structured interviews combine the aspects of a structured interview with an unstructured (Leavy 2014). Apart from these questions the interviews do not follow a strict template of predetermined questions, which increases the freedom for follow-up questions while still exploiting the benefits of a structured interview.

Notes are taken during the interviews which are later used for an analysis of the result. The gathered knowledge from the interviews is used with the information from the literature study to build the informal model.

The result from each interview is structured into categories which are themed after areas identified before or during the interviews. The result of all the interviews is compiled and compared according to these categories. This provides an overview of the important aspects to consider when designing an air intake.

3.3 Formalize

The translation of the informal model to a formal model is the main focus of the Formalize step and is executed in several steps. First, a DSM is created, followed by a Compatibility Matrix, and lastly a requirement specification is built based on the information captured in previous steps of MOKA. The DSM is designed to include the dependencies of the various input parameters which could be used in a UserForm. The Compatibility Matrix includes the configurations of possible air intakes while considering the overall integration with a complete aircraft as presented by Eklund and Karner (2017). Two requirement specifications are created, one for the low-fidelity (Lo-fi) prototype, and one for the high-fidelity (Hi-fi). The entities of the requirement specifications are

categorized as desired or required and are ranked from priority 1-3. The rank 1 indicates a high priority while 3 indicates a low priority.

3.4 Package and Scrum

Designing of the KBE interface is implemented during the Package step in MOKA. To enable the Package step the framework Scrum is used as a guideline. The implementation of Scrum is done by weekly sprints. At the start of the week the requirement specification of the application is revised and updated to match the scope of the project. After updating the requirement specification, the activities of the upcoming sprint are planned along with a plan of the execution of the planned activities. Each sprint is implemented through the loop: *design*, *test*, *develop*, *test*, *integrate*, *test* and back to *design*, which can be seen in Figure 9. After each sprint the result of the sprint is evaluated with the stakeholders of the thesis. During this step improvement areas are identified which are implemented during the next sprint.

In the following text a short description of the activities included in each step of the sprint is presented.

3.4.1 Design

The *design* step of Scrum focuses on the creation of the KBE system. This step includes the creation of parametric CAD-models, knowledge templates such as UDF's and PowerCopies along with scripts for automation. The updated requirement specification and the planning of the sprint is used as a reference while executing the *design* step of the sprint. The *design* step also includes designing of a UserForm.

3.4.2 Test

Between the steps *design* and *develop*, as well as between *develop* and *integrate* testing sessions are implemented. These sessions focus on ensuring the fidelity of the created models and the working integration with existing models. Testing is executed by identification of errors, potential future errors and improvement areas.

Another form of testing is user testing where users evaluate prototypes to ensure the prototypes' fidelity (Hertzum 2022). The testing follows the workflow presented by Hertzum (2022) and includes planning of the tests, execution and analysis of the result. The planning of the tests includes choice of users as well as preparation of exercises and questions. The execution of the tests includes time taking, observation of the users and note taking, as well as follow up questions after the tests. After the tests the result is analyzed and summarized.

3.4.3 Develop

The *develop* step is performed to further develop the models created in the *design* step based on the result of the *test* step. Improvements are implemented in the current model and errors are resolved. When a functioning model of the attribute, which is designed during the sprint, has been established another round of testing is done before the sprint moves on to the *integration* step.

3.4.4 Integration

After the potential errors have been resolved and a following test has been executed the integration of the current attribute or script is done with the rest of the models and scripts. After this step the integration is tested and errors upon integration are resolved. Improvement areas are also identified and the sprint loop restarts.

3.5 Activate

The Activate step of the KBE-lifecycle focuses on the hand over of the final result of the thesis to Saab. This step includes a thorough test session where the attributes of the application are tested, both on their own and their integration with the whole application. Potential weaknesses of the application are identified and resolved before a verification of the fidelity is established. For the hand over to Saab a brochure is created which includes information about how the application works, potential future areas of use and information about potential maintenance.

four

4 Implementation

In this chapter the result of the steps Capture, Formalize and the workflow during the step Package and Activate is presented; the chapter contains the obtained information from interviews, the formalization of the informal model, methods investigated for the controlling of the mesh nodes and the modeling and automation of the air intakes.

4.1 Capture

The Capture step of MOKA collects knowledge into an informal model. The informal model is based on the conducted interviews and literature study. The result of the interviews is presented in 4.1.1 Interviews air intakes, and the complete informal model can be found in Appendix 1.

4.1.1 Interviews air intakes

Five interviews are held with experts from the areas *concept development*, *aircraft propulsion* and *structural design*. Table 4 summarizes the conducted interviews. Potential interviewees are identified either during the Identify and Justify steps or during previously held interviews. The interview questions follow a predefined template but are adapted before each interview depending on the expertise of the interviewee, the template of open questions is found in Appendix 2.

Table 4. Overview of the interviewees' expertise and the number of meetings held with each interviewee.

Interviewee	Number of meetings	Expertise
P1	2	Aircraft propulsion
P2	1	Structural design
P3	1	Structural design
P4	1	Concept development

The information provided during the interviews resulted in the following categories:

- Subsonic and supersonic traveling speeds
- Different types of air intakes
- Different shapes of air intakes
- Different positions of air intakes
- Potential input parameters for a UserForm
- Potential future users and their different desires

There are a wide variety of air intake types, considering both subsonic and supersonic traveling speeds. The simplest being pitot intake which only consists of an air duct to the engine. Other types include ramp intake, bump intake, caret intake and cone intake, all developed to handle supersonic speed. Most of the names indicate the component that regulates the airflow and enables the aircraft to travel supersonic. Examples of this are that ramp intake has ramps installed inside the air intake that handles the shock waves created in supersonic speeds, and bump intake has a bump in the air intake front opening that diverts the boundary layer but also acts to diminish the radar signature.

Concerning the position and the shape of the air intakes there are several possibilities. The relevant placements that are brought up are on the sides of-, on top of- or under the fuselage and over-, under- or integrated in the wings. Several of the interviewees mention the disadvantages of the placements on top of- and under the fuselage. The placement on top of the fuselage is problematic when handling steep angles of attack – the angle between the airflow and the body's centerline – since the air intake is not getting enough air. Likewise the placement under the fuselage has disadvantages when on the ground since dirt can easily travel into the air intake.

The shapes of air intakes mentioned are rectangle, circle or oval, semicircle and bean shaped. It is common that the shape is affected by the placement of the air intakes and the type of air intake. The bean shape is usually developed in combination with a bump. The number of engines defines the possible number of air intakes, which in turn defines where the air intakes can be positioned. For example, one engine can have both one and two air intakes while two engines require at least two. Further, one air intake will not be placed on one side of the fuselage or on one wing since this creates imbalance at propulsion. The obtained information regarding what input parameters the users of the application would want is presented in Table 5.

Table 5. Interesting input parameters to include in the application, information obtained during interviews.

Input Parameter	Description
Capture area	The opening area of the air duct.
Throat area	The smaller area further in from the opening.
Lip thickness	Thickness of the lips, usually thicker for civil and thinner for military aircrafts. The thickness can vary on different lips, for example on rectangle shaped air intakes the lips don't need to have the same thickness on all sides.
Lip profile	For example the roundness of the lips, which can vary if it is the outside- or inside lip radius.
Cross section	The shape of the air intake seen from the front.
Dimensions of cross section	The width and height of the cross section.
Length of the air intake	The length of the air intake seen from the side.
Position along the x-axis	How far ahead the air intakes should be placed seen from the side, this placement can affect the pilot's view.
BLD	If a BLD should be included or not, the BLD affects the radar signature negatively.
Distance BLD	The distance from the fuselage to the BLD, make sure the boundary layer does not get into the air intakes.
Angles of opening	The angles of the air intake opening towards the airflow.
Angles of BLD	The angles of the BLD seen from the side.

Depending on the future users' field of expertise and -responsibility they would have different requirements and expectations on the application. For example:

- Departments focused on the radar signature would want to make sure to reduce the aircraft's visibility on radars as much as possible, thus care about the shape, the lips and the placement of the air intake.
- Departments focused on solid mechanics would care about the construction and what the air intakes are made of.
- The concept development department would be interested in the possible shapes and placements of the air intakes.

When designing the application it is important to define who the users are and what the users want to be able to adjust in order for the application to be useful.

4.2 Formalize

In the step Formalize a formal model is created through a translation of the informal model to formal requirements. It is later used as a reference when designing the KBE-application and when modeling the air intakes. It consists of a DSM, a Compatibility Matrix and a requirement specification. The result of the formal model is presented below.

4.2.1 Design Structure Matrix

First the DSM is presented in Table 6, which visualizes which input variables depend on each other. It can be seen that the number of intakes depends on the number of engines and that the presence of a BLD depends on if the intakes are integrated in the fuselage. The position depends on the number of intakes and the capture area depends on both the cross section of the air intake and if there is a BLD.


Table 6. DSM for the design of an air intake. A dependency is marked with an “x”.

	Number of engines	Number of intakes	Integrated in fuselage	Boundary layer diverter	Cross section	Position	Capture area
Number of engines							
Number of intakes	x						
Integrated in fuselage							
Boundary layer diverter			x				
Cross section							
Position		x					
Capture area				x	x		

4.2.2 Compatibility Matrix

The Compatibility Matrix for an air intake can be found in Table 7 and the Compatibility Matrix for an entire aircraft is found in Appendix 3. The Compatibility Matrix represents a more detailed explanation of the dependencies between the attributes of each input variable. Incompatibility between two attributes is marked with a 1 whereas compatibility is marked with 0. The dependency between the number of intakes and number of engines, as shown in Table 6, exists because there cannot exist one (1) air intake for two (2) engines. This implies that if there exist two (2) engines the position of the air intake cannot be on top of the fuselage or under the fuselage. Similarly, if the number of air intakes is one (1) the available positions are limited. The reason for this is that there cannot exist only one air intake if the position is selected as integrated in wing, on wing, under wing or on the sides of the fuselage.

Table 7. Compatibility matrix for an air intake. 0 marks compatibility and 1 marks incompatibility, 3 is used along the diagonal as an attribute cannot be compatible with itself. Using the template created by Eklund and Karner (2017).

<div> <div>Compatibility Matrix</div>  </div>			Attributes																
			Number of engines		Number of intakes		Integrated in fuselage		Boundary layer diverter		Cross section				Position				
			1	2	1	2	Yes	No	Yes	No	Rectangle	Semicircle	Half circle	Bean	Top	Bottom	Sides	Over wing	Under wing
Attributes	Number of engines		1	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			2	1	3	1	0	0	0	0	0	0	0	0	1	1	0	0	0
	Number of intakes		1	0	1	3	1	0	0	0	0	0	0	0	0	0	1	1	1
			2	0	0	1	3	0	0	0	0	0	0	0	0	1	1	0	0
	Integrated in fuselage		Yes	0	0	0	0	3	1	1	0	0	0	0	0	0	0	1	1
			No	0	0	0	0	1	3	0	1	0	0	0	0	0	0	0	0
	Boundary layer diverter		Yes	0	0	0	0	1	0	3	1	0	0	0	1	0	0	0	1
			No	0	0	0	0	0	1	3	0	0	0	0	0	0	0	0	0
	Cross section		Rectangle	0	0	0	0	0	0	0	3	1	1	1	0	0	0	0	0
			Semicircle	0	0	0	0	0	0	0	1	3	1	1	0	0	0	0	0
			Half circle	0	0	0	0	0	0	0	1	1	3	1	0	0	0	0	0
			Bean	0	0	0	0	0	0	0	1	1	1	3	0	0	0	0	0
	Position		Top	0	1	0	1	0	0	1	0	0	0	0	3	1	1	1	1
			Bottom	0	1	0	1	0	0	0	0	0	0	0	1	3	1	1	1
			Sides	0	0	1	0	0	0	0	0	0	0	0	1	1	3	1	1
			Over wing	0	0	1	0	1	0	0	0	0	0	0	1	1	1	3	1
			Under wing	0	0	1	0	1	0	0	0	0	0	0	1	1	1	1	3
			Integrated in wing	0	0	1	0	1	0	1	0	0	0	0	1	1	1	1	3

4.2.3 Requirement specification

Requirements are formulated to specify the functionality and fidelity of the prototypes. Both a Lo-fi prototype and a Hi-fi prototype of the different air intakes are developed. The Lo-fi prototypes are less detailed and focus on the functionality and basic structure of the air intake and the KBE-application. The main focus of the Lo-fi prototypes is to gather knowledge for how the Hi-fi prototypes should be structured so that potential errors can be avoided. Another aspect that the Lo-fi prototype does not consider is the integration with the fuselage, because of this the basic structure of the airplane is simplified, as can be seen in 4.3.2.1 Low-fidelity modeling and automation. The formal model includes a requirement specification for the Lo-fi prototype, which can be found in Table 8. This specification includes the need, a description of the entity, if the entity is desired or required and a rank from 1-3. It includes information about both the air intake model and the KBE-application.

Table 8. Requirement specification for the Lo-fi model. A rank of 1 indicates a high priority, 2 is medium priority and 3 indicates a low priority.

Lo-fi model and application

#	Need	Formal - Entity	R/D	Rank
1	Various cross section options	Enable four (4) different cross sections (rectangle, oval, semicircle, bean)	R	1
2	Various position options	Enable five (5) different positions (top, under, sides, over-, under wing)	R	1
3	Parametric model (dimensions, placement xyz, capture area, angles)	Enable parametrization of dimensions, placement xyz, capture area and angles through code	R	1
4	Automatic linking of mesh nodes	Through code reach desired mesh node, coincide mesh node and point and create an IMA-link	D	3
5	Boundary layer option	Enable instantiation of BLD with required references for air intake UDF	R	1
6	Structured and minimized code	Use comments, loops and functions to structure the code	D	2
7	Minimized required inputs for UDF	Use points and planes as UDF-inputs	D	2
8	Add air intakes automatically	Add air intakes through code	R	1
9	Remove air intakes automatically	Remove air intakes through code	R	1
10	Assure requirements from Compatibility Matrix	Inform about possible combinations	R	1
11	Update the model fast	Minimize number of part updates and references	D	2
12	Choose number of engines and air intakes	Choose number of engines and air intakes in UserForm	R	1
13	Warning message when outside limits	Warning message pop-up in UserForm when outside limits	D	3
14	Default max/min value when outside limits	Automatically restriction of input values when outside limits	D	2

When the modeling of the Lo-fi prototype is finished the requirement specification for the Hi-fi prototype is updated. The modeling of the Hi-fi prototypes uses the requirements identified during interviews and the literature study, while using the gathered knowledge from designing of the Lo-fi models. As the Hi-fi prototypes are the ones which will be distributed to Saab these prototypes include details and consider the integration with the fuselage. The requirement specification for the Hi-fi model is found in Table 9.

Table 9. Requirement specification of the Hi-fi model. A rank of 1 indicates a high priority, 2 is medium priority and 3 indicates a low priority.

Hi-fi model and application

#	Need	Formal - Entity	R/D	Rank
1	Various cross section options	Enable four (4) different cross sections (rectangle, oval, semicircle, bean)	R	1
2	Various position options	Enable three (3) different positions (top, bottom, sides of fuselage)	R	1
3	Parametric model (dimensions, placement xyz, angles xyz, lip thickness, shear around x-axis for rectangle)	Enable parametrization of dimensions, placement xyz, angles xyz, lip thickness and shear around x-axis for the rectangle shape through code	R	1
4	Automatic linking of mesh nodes	Through code reach desired mesh node, coincide mesh node and point and create an IMA-link	D	3
5	Boundary layer option	Enable implementation and instantiation of BLD through creating a distance from the fuselage and adding a BLD	R	1
6	Structured and minimized code	Use comments and loops to structure the code	R	1
7	Minimized required inputs for UDF	Use points, planes, curves and axis system as UDF-inputs	D	2
8	Add air intakes automatically	Add air intakes through code	R	1
9	Remove air intakes automatically	Remove air intakes through code	R	1
10	Assure requirements from compatibility matrix	Inform about possible combinations in UserForm	R	1
11	Update the model fast	Minimize number of part updates and references	D	2
12	Choose number of engines and air intakes	Choose number of engines and air intakes in UserForm	R	1
13	Integration with fuselage surface and investigate the continuity	Integration with fuselage surface and investigate the continuity	R	1
14	Warning message when outside limits	Warning message pop-up in UserForm when outside limits	D	2
15	Default max/min value when outside limits	Automatic restriction of input values when outside limits	D	2
16	Intuitive UserForm	Remove non-selectable options when designing the UserForm	D	3
17	Naming convention	Naming convention of objects and parts	D	2
18	Inform about capture area	Approximate capture area in UserForm before automation and calculate actual capture area and inform user through output parameter	R	1
19	Model air intake lips correctly	Model air intake lips according to given guidelines	R	1

4.3 Package

The Package step of MOKA includes the execution of the SCRUM-loop where the KBE-application is developed. The application used is the surface tool IMA, Table 10 describes a few commonly used actions in IMA.

Table 10. Common IMA actions.

Action	Description
<i>IMA – Modification</i>	Modifies the subdivided surface by different operations such as modifying placement of mesh nodes on the subdivided surface.
<i>IMA – Face cutting</i>	Allows the user to increase the number of subdivisions on a surface.
<i>IMA – Merge</i>	Merges two subdivided surfaces where the two edges to be merged contain the same number of subdivisions.
<i>IMA – Cut by plane</i>	Cuts the subdivided feature with an intersecting plane.
<i>IMA – Cut by edges</i>	Cuts the subdivided feature with its own edges
<i>IMA – Mesh tools</i>	Includes operations such as merging mesh nodes, removing faces and adding faces.
<i>IMA – Link</i>	Links mesh nodes on the subdivided surface to points or lines

4.3.1 Explore methods to adjust mesh nodes

One of the research questions in the project is to determine the best method for controlling mesh nodes on a subdivided surface in IMA. As IMA is not directly compatible with VBA the modification has to be completed through manual work or a new method has to be explored. Operations done to the mesh in other applications, for example the operation *Translate* in Generative Shape Design, locks the surface from further modification in IMA. This limits the work to the available functions in IMA and basic point/spline operations. Two methods are explored, one manual and one method which implements some automation. The reasoning behind the methods is that if a mesh node on the subdivided surface is linked to a reference point, the surface can be parameterized. Both methods assume that a subdivided surface with the right number of cuts is already created. Reference points and planes are instantiated on splines in the same file as the surface. The methods are both tested on simplified cases such as the one in Figure 10. They are evaluated depending on the required time to fit a subdivided surface to the available reference points, as well as the number of errors. The methods explored are presented in 4.3.1.1 Method 1 and 4.3.1.2 Method 2.

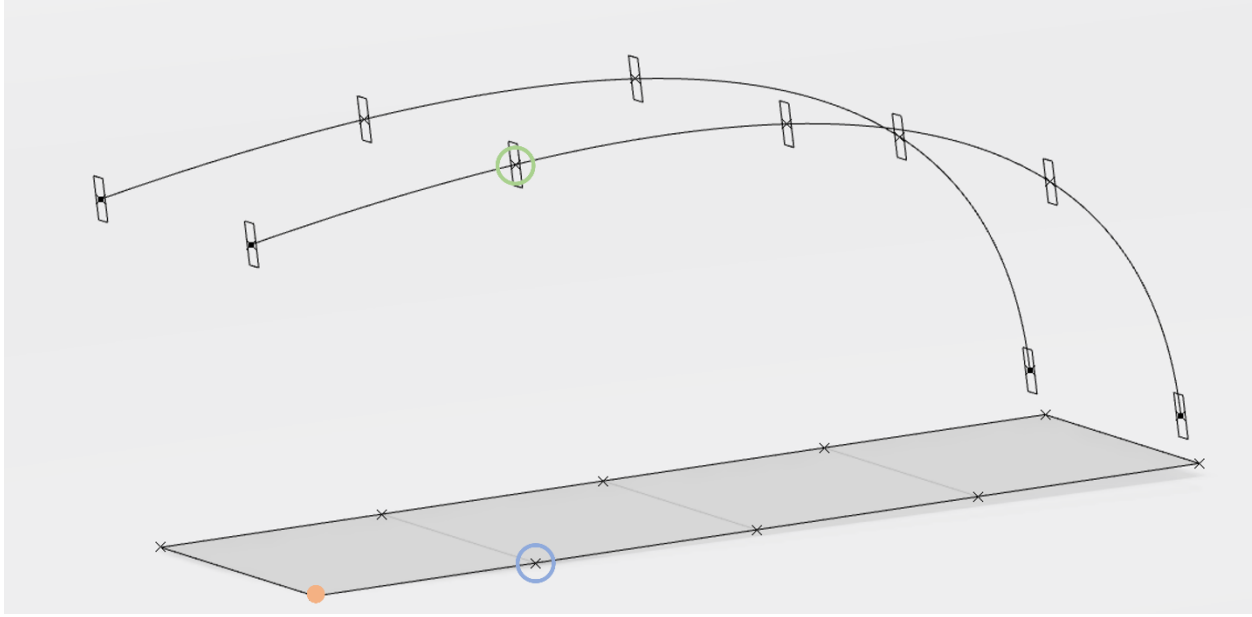


Figure 10. The case on which the two linking methods are tested. The orange dot represents a mesh-node (n) on the subdivided surface, the green circle shows a reference point (p_r) and the blue circle shows a help point (p_h) created with Method 2.

4.3.1.1 Method 1

The first method requires the user to use the tool *IMA – Modification* to manually adjust the mesh nodes (n) to the same position as the reference point (p_r). First, the robot (a local coordinate system) has to be redefined to the location of p_r by using the action *Robot definition*. This is an action included in *IMA – Modification* that opens the robot-mode. In this step the robot action *Pick* is used to define p_r as the target location. With the *IMA – Modification* action *Alignment* the user can use the x-, y-, and z-axis to coincide n and p_r . This first part of the method is shown in Figure 11. After n and p_r are coincided the action *IMA – Link* is used for linking n and p_r so that modifications of p_r adjusts the subdivided surface. When this process is repeated for all nodes available on the surface, it is modifiable by adjustments of the curves, the result can be seen in Figure 12.

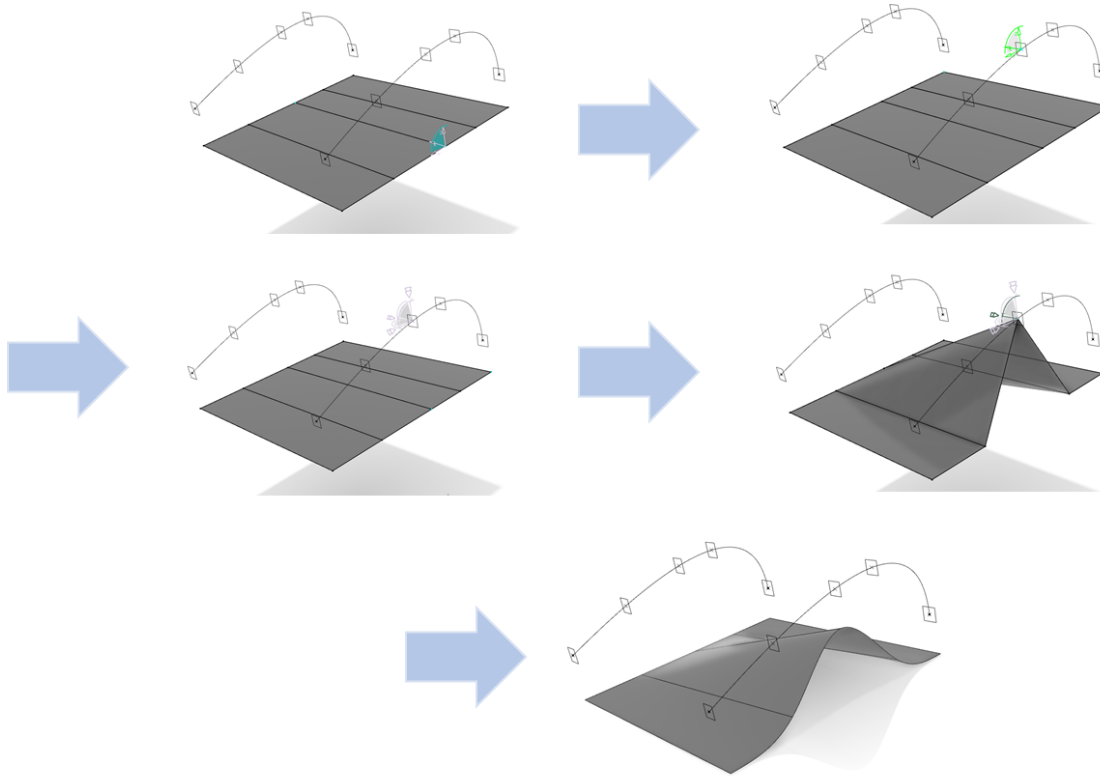


Figure 11. The process of Method 1. The first picture shows a chosen mesh node (n). In picture 2 the robot is redefined to the location of the point (p_r). In the third and forth picture the alignment process is shown. In the fifth picture the result for one n is shown.

4.3.1.2 Method 2

The second method combines manual work with automation. The subdivided surface has to be positioned almost parallel to the spline, on which the p_r are created. The method consists of the user executing a script which creates help points (p_h) on the nodes of the subdivided surface with a reference to p_r . The user is asked to select the desired surface. The script starts by retrieving the coordinates of all p_r and storing these coordinates in a list. Then all nodes on the surface are automatically selected and the script goes through a loop investigating the coordinates for each n . In the loop, n 's coordinates are retrieved and a calculation of minimized distance determines which p_r n should be connected to. When this is determined a p_h is created at the location of n , referencing p_r using the previously calculated coordinates of n and p_r . All the nodes go through the same loop and p_h are created for all n .

The script finishes and the user is asked to manually use *IMA – Link* to connect every p_h to n . Since *IMA* is not compatible with VBA the linking with *IMA – Link* needs to be done manually. When this is completed, the next script can be started, which moves all p_h to the location of p_r , the result can be seen in Figure 12. As n is linked to p_h , and p_h is connected to p_r the surface is modifiable by alterations of the position of p_r .

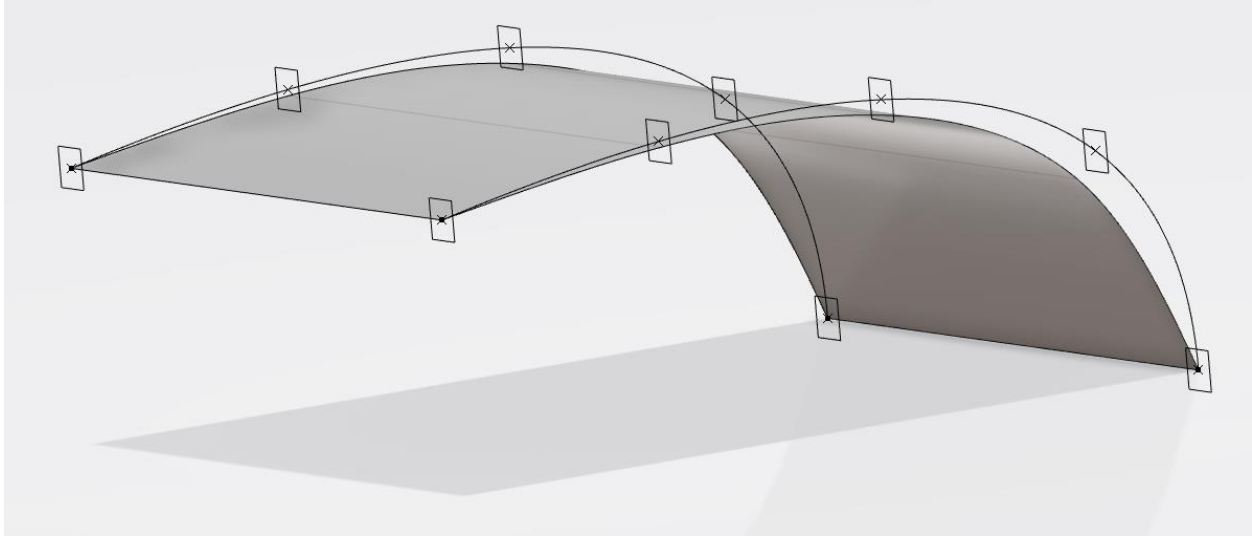


Figure 12. The result of both mesh node methods. A higher resolution is acquired through creation of more p_r .

4.3.1.3 Evaluation of the methods

An evaluation of the mesh node methods is performed to validate which one is the best in terms of usability, time consumption, total number of mouse clicks and number of errors. The evaluation is performed through user testing with four users with different experiences of IMA. The users are asked to perform Method 1 as well as Method 2 for a surface with 10 mesh nodes. The users are provided with a quick explanation of the necessary tools in IMA. During the user testing time consumption and number of errors are noted, and after the test the users are asked to answer the following questions:

- Which method did you find best to complete the task?
- Which method do you think would be best suited for a user with no previous experience of IMA?

4.3.2 Development of the air intake application

The modeling and automation of the air intakes is based on the formal model. The Lo-fi models have less detail and mostly focus on the functionality of the application – the shapes and the positions of the air intakes – and the Hi-fi models have a higher resolution and include the integration with the fuselage.

4.3.2.1 Low-fidelity modeling and automation

For the creation of Lo-fi models, a simplified test model of a fuselage and a wing is created. The fuselage is represented by three roughly made curves, and the wing is represented by a flat surface, the test model can be seen in Figure 13. Position points are created on the three curves and on the wing to mark the alternative positions for air intakes, red markings in Figure 13.

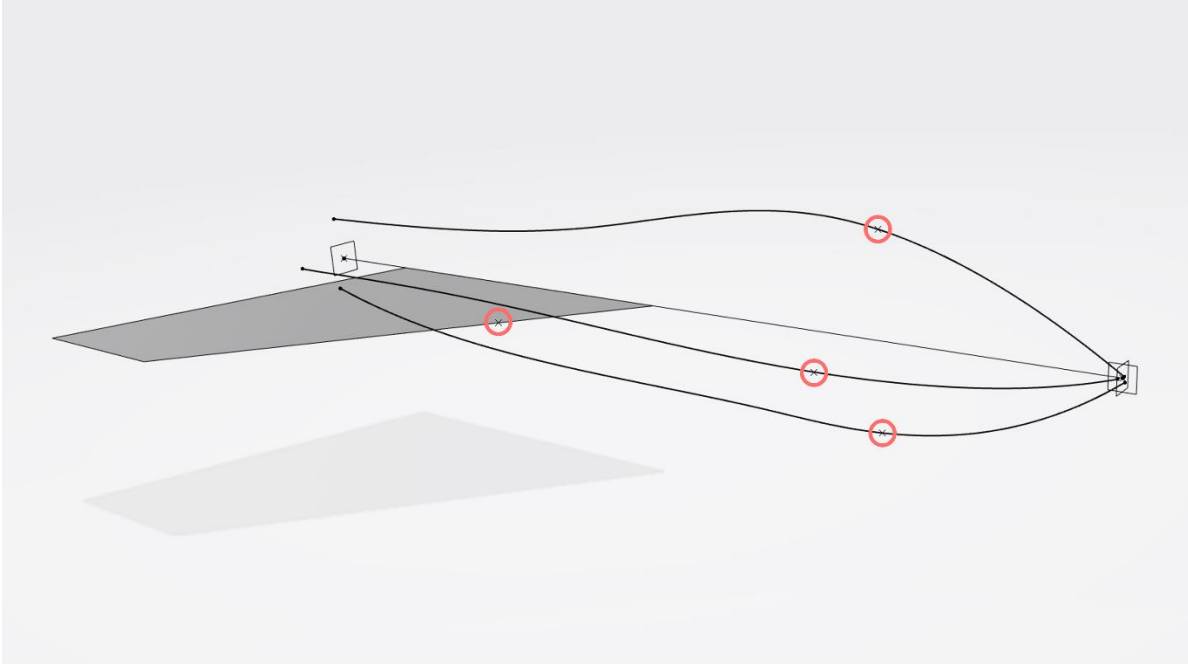


Figure 13. The simplified plane used for modeling of the Lo-fi air intakes The air intake positions are marked with red.

Before the modeling of the air intakes starts, the modeling process is broken down and structured into a step-by-step plan. The plan contains which objects and features to use, in which order the instantiations should be executed and how the surface of the air intake should be created and connected. In order to attain the requirement #7 *Minimized required inputs for UDF* all the shapes include an airduct center point which controls the rest of the geometry, as well as a locally created axis system. The axis system enables rotations around the shapes' rotation center. Rotation around the x-, y- and z-axis are allowed in order to attain parts of the requirement #3 *Parametric model*. As the local axis system is included in the UDF, the shape inherits the direction and is correctly rotated according to the user's desire. The rotation of the axis system is decided by a rotation matrix which references the three parameters: αX , αY and αZ . The rotation matrix used can be found in Appendix 4. The rectangle shape also includes an additional parameter, which shears the shape around the x-axis.

An integration of as many shapes as possible into the same frame is conducted to facilitate the integration with the fuselage. The rectangle and the semicircle are integrated in the same wireframe model. The basic shape of these wireframes is modeled using points, planes, lines and *Conic* curves, see Figure 14. Conic curves are used to facilitate the change from the sharp edge of a rectangle to the smooth radius of a semicircle. The shapes' dimensions can be adjusted through parameters, including height, width and corner radiuses, in order to attain parts of the requirement #3 *Parametric model*.

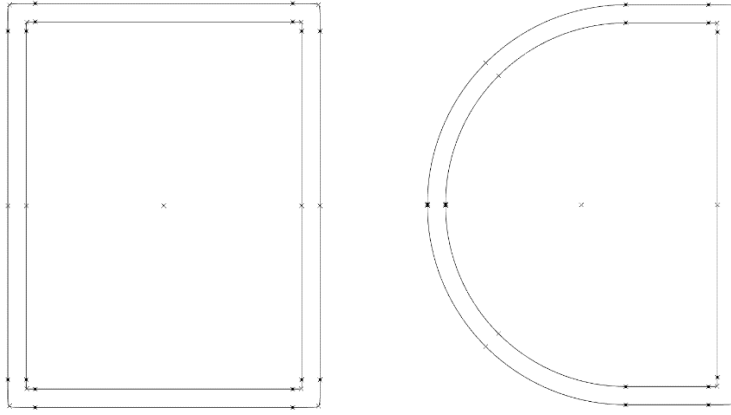


Figure 14. The frame model for the rectangle and semicircle.

The circle, oval and bean are integrated in the same frame model. They are made with points, a spline and five parameters. These parameters are *width*, *height1*, *height2*, *height3* and *innerlength*. The parameters adjust the points' position to form the chosen shape. The parameter *width* sets the diameter of the circle, and the width of the other shapes. The parameter *height1* sets the vertical radius for the oval and semicircle and acts as one of the height measurements for the bean. *Height2* sets another height measurement for the bean shape and *height3* decides the length of the bean bumps. The different shapes are found in Figure 15, where the parameters are visualized in the bean shape.

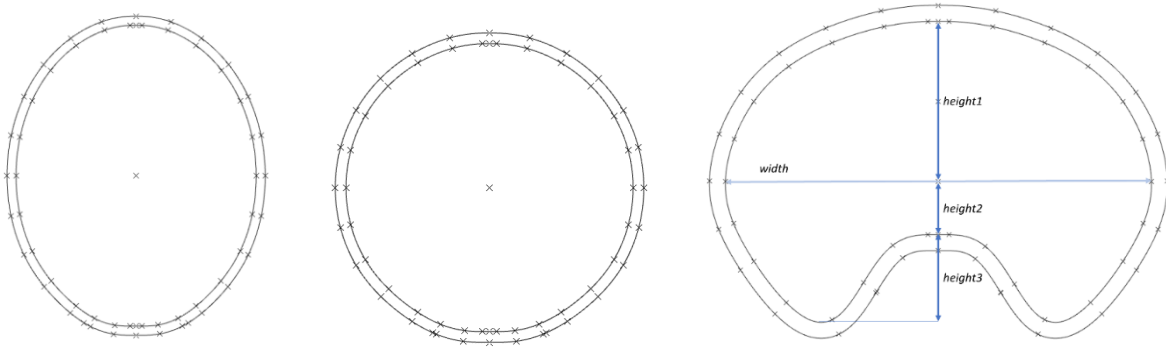


Figure 15. The frame model for the oval, circle and bean. The bean shape contains the measurements: *width*, *height1*, *height2* and *height3*.

To be able to instantiate the frame-UDFs along the curves of the aircraft their positions need to be defined. This is done by manually splitting the fuselage-curve using the position point and a point further down the curve with the air intake length distance from the position point, and then instantiating a UDF with points and planes along the split curve using knowledge pattern. The number of points and planes is defined by a parameter and can therefore be adjusted according to the fidelity the user wants. In the Lo-fi model the number of points is set to 4, and the points and planes are placed with a calculated ratio along the split curve.

The wireframe-UDFs can then be instantiated using the points and planes. The instantiation is made with a knowledge pattern that defines which frame-UDF is placed at which point and sets the parameters for the frames. The UDF with both inner and outer frame is instantiated at the two points closest to the front of the aircraft, and the UDF with the outer frame is instantiated at the remainder of points. Lastly a surface in IMA is created, subdivided and modified to match the shape of the frames. Figure 16 presents the rectangle air intake at the side position, Figure 17 presents the semicircle air intake at the top position, Figure 18 presents the bean intake at the bottom position and Figure 19 presents the oval air intake positioned under the wing. The linking is made according to the method described in 4.3.1.1 Method 1. The linked surface then follows the frame's position, for example when changing the air intake position or the length of the air intake.

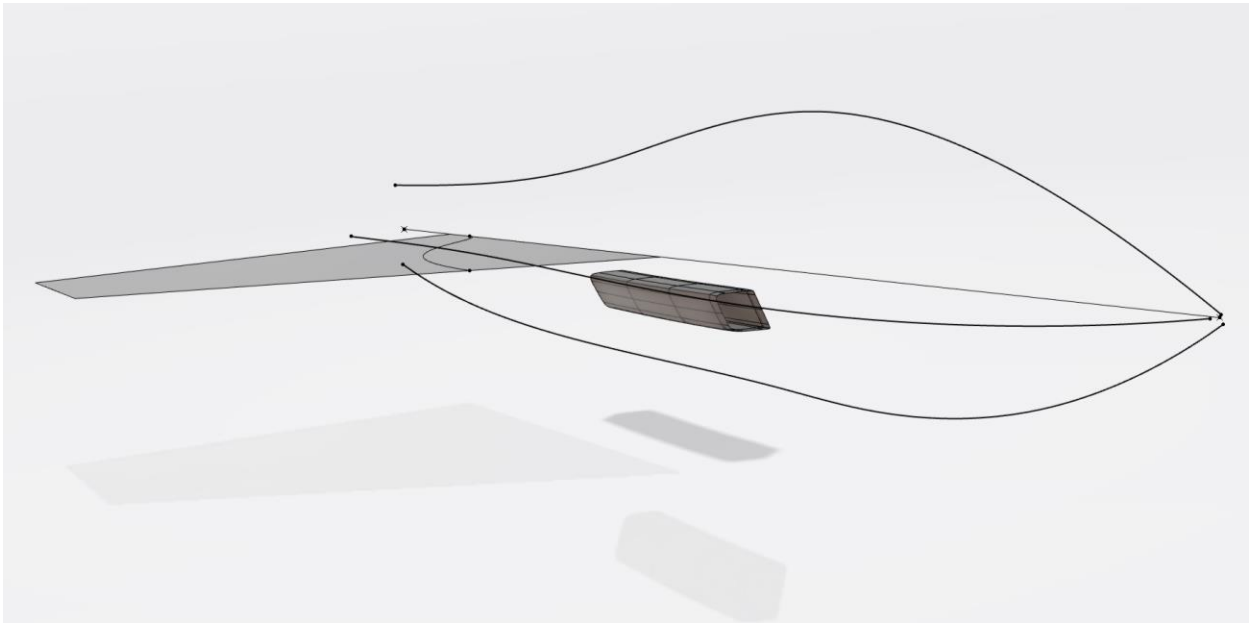


Figure 16. Rectangle Lo-fi air intake positioned on the side of the fuselage. The air intake is rotated around the x-, y- and z axis.

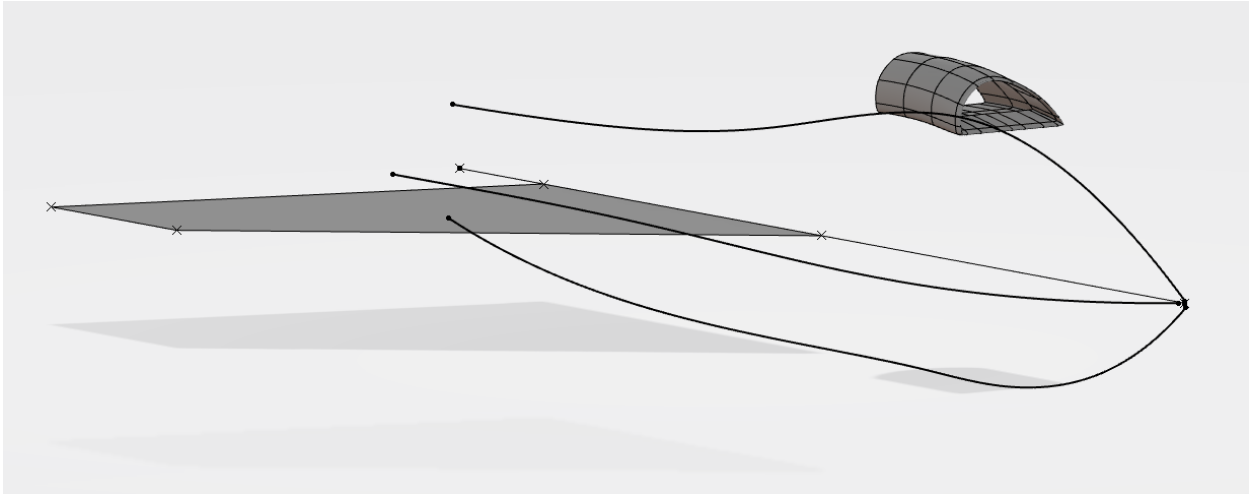


Figure 17. Semicircle Lo-fi air intake positioned on the top of the fuselage. The air intake is rotated around the y-axis.

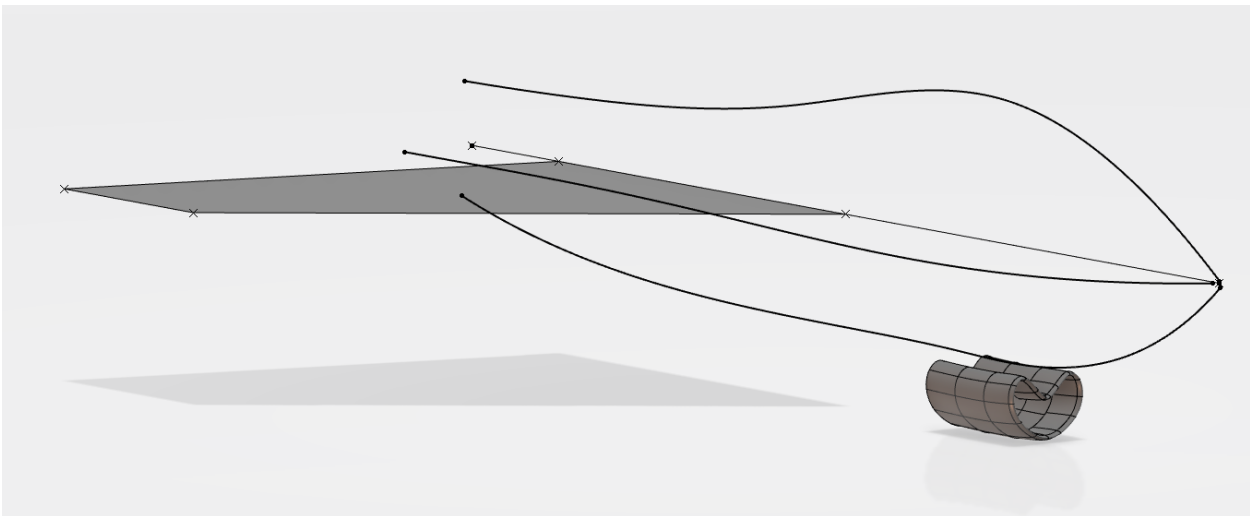


Figure 18. Bean Lo-fi air intake positioned on the bottom of the fuselage. The air intake is not rotated in any direction.

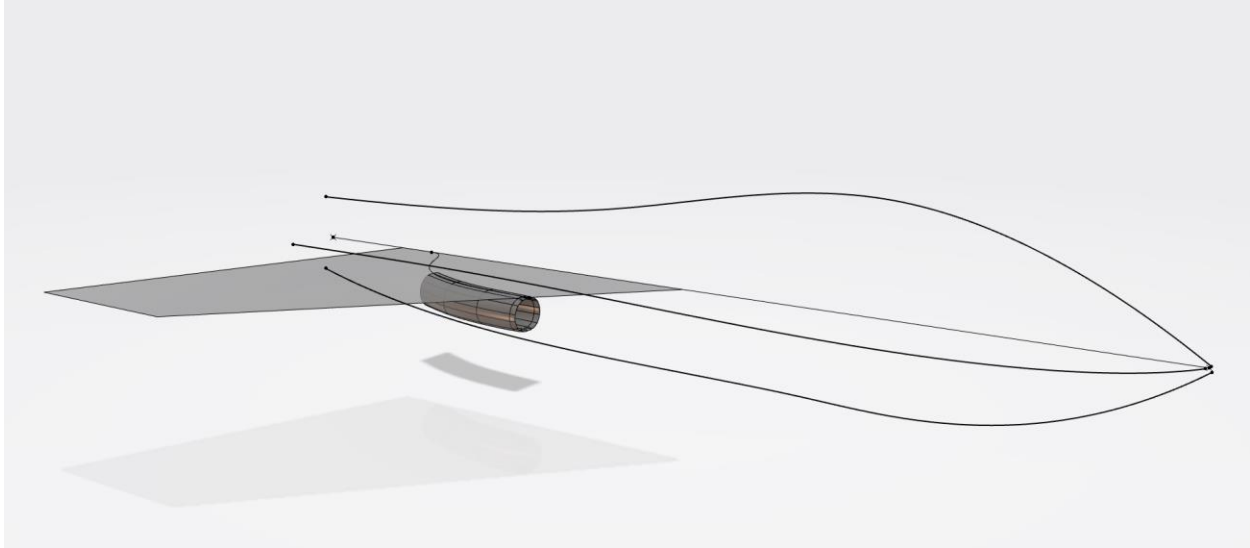


Figure 19. Oval Lo-fi air intake positioned under the wing. The air intake is not rotated in any direction.

The capture area is calculated by projecting the first frame onto a yz plane placed by the aircraft's nose. The shape's area is then determined in a rule which uses EKL. Rules informing the user about possible combinations upon choosing number of engines, position and number of intakes have also been implemented.

4.3.2.2 High fidelity modeling and automation

The main modeling and automation of the Hi-Fi prototypes is divided into two parts: one part focusing on the lips of the air intake and the other focusing on the BLD. A description of the methodology for the creation of these parts is provided in this subchapter.

Lips

As described in 4.3.1 Explore methods to adjust mesh nodes, the subdivided surfaces in IMA require reference points to be able to link and control the surfaces without the need to modify each mesh node. A wireframe model that consists of frames, points, airfoils, lines and curves is therefore created as a basis for the lips' subdivided surface. The lips form the front part of the air intake and are based on a frame placed at the front. Two lip models are created using the same frames mentioned in 4.3.2.1 Low-fidelity modeling and automation – one for the rectangle/semicircle and one for the circle/oval/bean. The front frame acts as the capture area of the airflow and allows rotation around the x-, y- and z-axis, the same functionality that was presented in the Lo-Fi models.

The next step is to use airfoils for the outer geometry of the lips' wireframe. The standard airfoil NACA0012 is used with given coordinates for 13 points, see Figure 20. The coordinates are imported into parameters in CATIA with a rule, and to keep the proportions the parameters are connected to each other with formulas. The airfoil is then created by connecting these points with lines and *Connect Curves*, presented in Figure 20. For the lips only half of the airfoil is used – refers to point 7 to 13. The outer geometry of the wireframe is built by placing UDFs with airfoils along the front frame at certain chosen points. The number of airfoils placed varies from the rectangle/semicircle and the circle/oval/bean since their frames are created in different ways with different numbers of points.

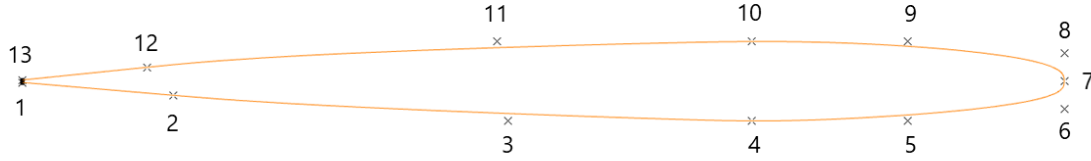


Figure 20. The points and curves that form the airfoil model.

To be able to link to the subdivided surface later, five points are positioned on the airfoils between the 7th and the 10th airfoil point. These five points are also included in the UDF for the airfoil. The airfoil UDF needs two inputs: a local axis system and a line tangent to the curve at the 7th point. Another UDF including the inputs needed for the airfoil UDF is used to place all the airfoils at the chosen points. The local axis system allows rotation around the x-, y- and z-axis, and since the airfoil is connected to the local axis system, it follows the rotation of the axis system.

To create the inner geometry of the wireframe, firstly the 10th point of every airfoil is translated inwards in the tangent line's direction. Then the 10th point of each airfoil and the newly translated points are projected onto a plane at the end of the lip model. The end plane is parallel with the global yz plane, thus not rotating with the rest of the model. For the inner part of the lips, a UDF with a curve created with the function *Curve Blend* is used. *Curve Blend* is a tool where the user can choose the desired order of continuity between curves and surfaces. These curves match the curvature of the airfoil curve (at the 7th point) and form the inner half of the lip profile. Similar to the airfoil, five points positioned along these curves are also included in the UDF for the linking with the subdivided surface.

With the use of the translated and projected points, the five points on the airfoils and on the curves, inner and outer frames are created to get the whole wireframe structure for the lips. The last step is to create the subdivided surface in IMA, aligning and linking it to the points on the wireframe model. Figure 21 presents the IMA surface for the rectangular air intake lip model. In Figure 22 the rectangle is modified into a semicircular shape.

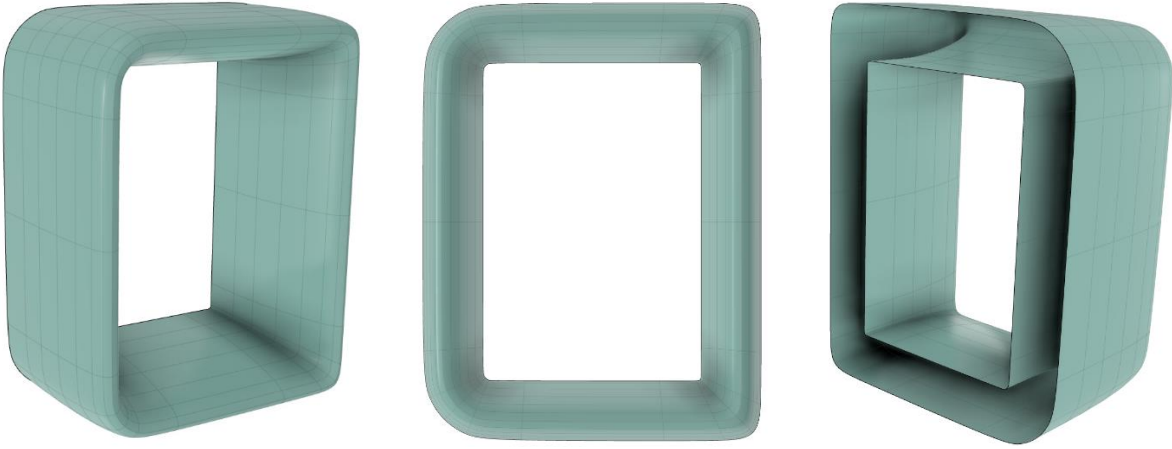


Figure 21. The IMA surface for the rectangular air intake lip model. From the left: Perspective front view, front view, perspective back view.

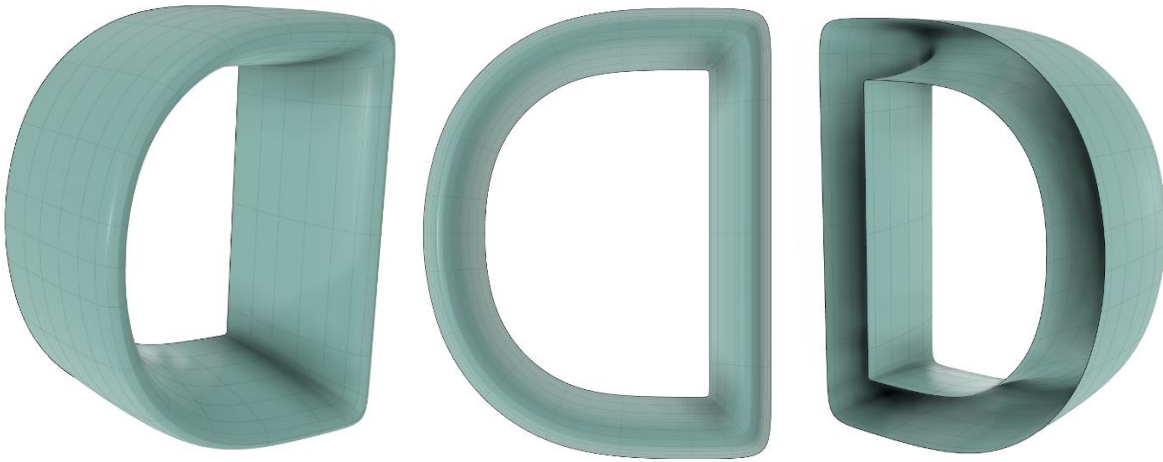


Figure 22. The air intake lip model for the semicircular shape.

The lip model for the circle, oval and bean is made similar as the rectangle and the semicircle, see Figure 23 and Figure 24. To get the right shape, this model requires more airfoils to be placed along the frame.



Figure 23. The air intake lip model for the circular shape. Can be made into an oval by changing the dimensions for the height and width.



Figure 24. The air intake lip model for the bean shape.

Knowledge patterns are used for measuring dimensions in the lips' geometry. These measurements are used to position components in the geometry of the lips, and to position and dimension the BLD for the integration between parts. The knowledge patterns are automatically executed when the whole part is updated. The priority order of the knowledge patterns is manually set to allow each script to run without the risk of colliding with another script.

Boundary layer diverter

Two BLD's are created, one for the rectangle/semicircle shape, and one for the circle/oval/bean. Both are created using the same methodology and using the previously created frame models, presented in 4.3.2.1 Low-fidelity modeling and automation. The first steps are implemented by using tools in Generative Shape Design. The double frame is swept along a spline, creating an extruded air intake of the chosen shape. Two sketches are created on local xy- and zx planes, one where the user can choose the profile of the BLD and one which ensures that geometry other than the chosen profile is removed. The shape of the profile-sketch is either decided by a default sketch which results in a parabolic shaped diverter, or by a *usersketch*. The default sketch is modifiable by alterations of a width and depth parameter and the *usersketch* allows the user to manually modify an existing sketch by using control points or replacing the sketch altogether. The swept shape is split two times with extrudes created from the sketches, as seen in Figure 25. A rule decides which part of the split should be kept, which enables robustness when modifying the profile of the BLD.

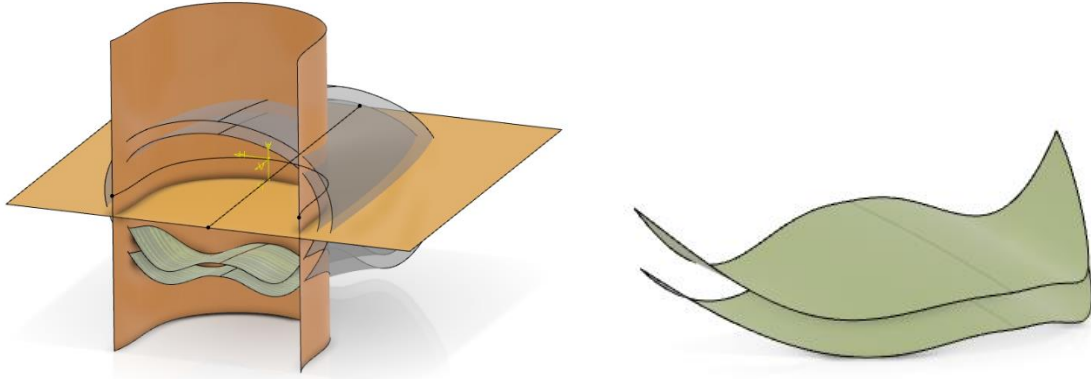


Figure 25. The grey swept surface, cut by the dark orange default sketch and light orange height sketch. The result is the green split surfaces.

The edges of the two split surfaces are extracted using an action called boundary. A split operation of the boundaries creates a front, as well as a back profile for the top and bottom surface. The back profile is represented by the orange line in Figure 26. A UDF creates points on the front and back profile of the top boundary, placed on the top surface. Lines are created between corresponding points on the top front and back boundaries, in total 11 lines are created to get the desired resolution, these can be seen in Figure 26. A rule using EKL calculates the length of each line. The lines are used to instantiate five points per line on the top surface on a distance which is decided by the calculated length. The points created on the top surface are translated to the bottom surface. Two subdivided surfaces, one for the top and one for the bottom, are created from the created reference points. The result is a subdivided surface, fitted to the swept and split surfaces created in Generative Shape Design.

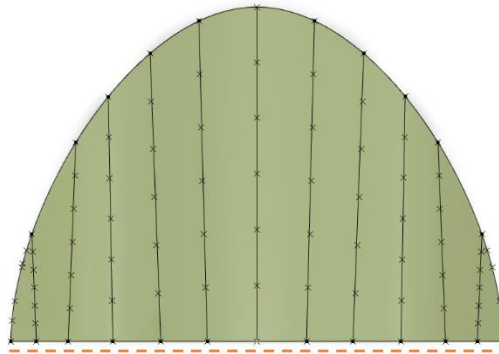


Figure 26. The reference points for the surface of the BLD. The orange dotted line defines the back of the shape.

Lips are created to fill the surface between the top and bottom surface. To create the lips the front boundary of the top and bottom surface is offset using a 3DOffset. The offset uses a parameter with the same value as the front length of the airfoils. A fill is created between the offset curves, as can be seen in Figure 27. Points are created along the top offset using a UDF. In each point a

line which is normal to the fill is created with another UDF. The normal line is used as an input for the airfoil reference UDF for the BLD and sets the x-direction of the reference coordinate system. The airfoil reference UDF for the BLD also needs a position point, a fill as well as the z-axis of the reference frame. Since the airfoil references are created in this way the shape is able to rotate around its own x-axis, enabling different positions and rotations. This also ensures that the airfoil is always rotated exactly 90 degrees from the fill.



Figure 27. The offset curves and the fill which is created between them.

A symmetric airfoil is used for the BLD and is instantiated using the previously mentioned references. A subdivided surface is created, referencing the points on the airfoil, creating lips for the BLD. This subdivided surface is merged with the top and bottom subdivided surfaces using *IMA - Merge*, followed by linking of the mesh nodes using *IMA - Link*. An example of a bean BLD is shown in Figure 28.

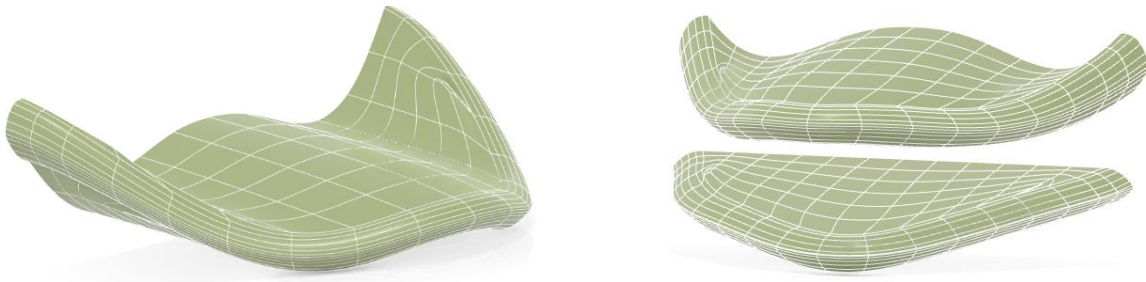


Figure 28. To the left: a bean BLD which is not rotated around its x-axis. To the right: both a rectangular and bean BLD. The bean BLD uses the default sketch and the rectangular uses the usersketch.

4.3.3 Integration between parts

The parts need to be integrated with each other as well as with a provided fuselage. The following sections explain how this can be implemented as well as how an analysis of the attained continuity can be carried out.

4.3.3.1 Integration between lips and boundary layer diverter

The integration between the lips and BLD is implemented through several manual steps. The first step includes calculation of the new parameter values for the BLD, as the lips and BLD are created using different approaches. As mentioned in 4.3.2.2 High fidelity modeling and automation, the BLD uses both the inner and outer frame of the shape whereas the lips only use the inner frame, which results in the BLD being bigger compared to the lips. If the lips are rotated around the y- or z-axis a new placement has to be calculated for the BLD. The reason for this is that the BLD is only rotatable around its x-axis and therefore has to be moved to match up with the placement of the lips. After the BLD has been placed correctly it is cut by a plane, using *IMA – Cut by Plane*. The plane is included in the lip PowerCopy, placed at the front of the lip shape.

When the BLD has been cut by the plane, the front of the lip model is cut using *IMA – Erasing* or *IMA – Cut by Edges* to allow merging with the BLD. As the lip shape contains fewer subdivisions than the BLD shape it has to be subdivided so the number of subdivisions at the merging edges match up, using the *IMA - Face Cutting*. If modifications are necessary to match the surfaces the tool *IMA – Modification* is used, after this the surfaces are ready to be merged. *IMA – Merge* is used to merge the two surfaces, the first surface selected keeps its links to the mesh nodes, while the second surface is unlinked. Because of this the more complex lip shape should be chosen first to eliminate the repetitive work. An example of an integration between the lips and BLD is shown in Figure 29.

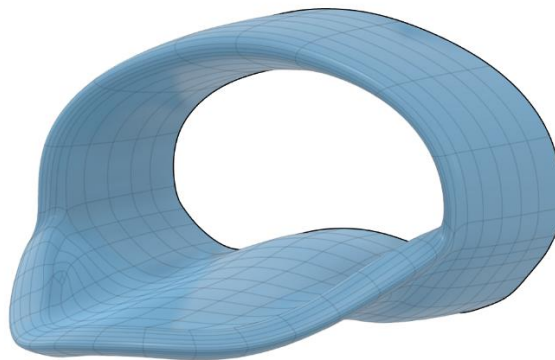


Figure 29. Integration between BLD and lips for a bean shape.

4.3.3.2 Integration between air intake and fuselage

The integration between the air intake, more specifically the lip shape, and the provided fuselage has to be completed following a similar methodology as the integration between the BLD and lips. The provided fuselage has an existing air intake and therefore a closed fuselage has been created using *IMA – Modification*. The fuselage model, seen in Figure 30, consists of half a fuselage subdivided surface mirrored by the *zx*-plane. This implies that after the air intake has been positioned and shaped according to the users desire the fuselage has to be cut by using *IMA – Erasing* or *IMA – Cut by Edges* to create an opening where the intake can be merged.

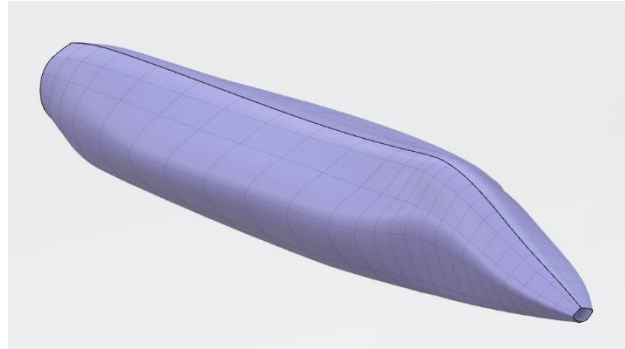


Figure 30. The closed fuselage which should be used for integration with the air intake. The fuselage is mirrored at the black edge.

When the fuselage has been cut the opening of the surface has to be shaped using *IMA – Modification* to ensure that the cut edges of the fuselage follow roughly the same shape as the free edges of the air intake. After this the action *IMA – Face Cutting* is used to subdivide the cut edges of the fuselage to match the number of subdivisions on the air intake. Then the two surfaces can be merged using *IMA – Merge*, it is important that the air intake shape is chosen before the fuselage shape when applying the changes to keep the linking of mesh nodes for the air intake. An example of what the integration can look like is shown in Figure 31.

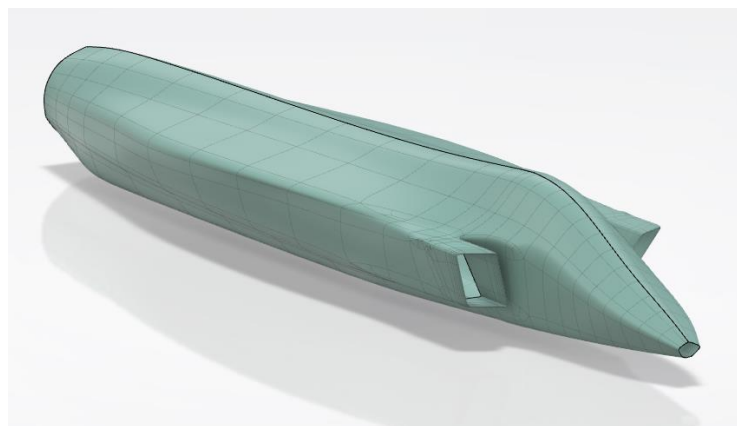


Figure 31. An integration between a rectangular air intake and the provided fuselage at side position.

4.3.3.3 Curvature analysis

A curvature analysis can be made with the two functions *Porcupine Curvature Analysis* and *Connect Checker* (Dassault Systèmes n.d.h, Dassault Systèmes n.d.i). *Porcupine Curvature Analysis* can be used both when analyzing curves and surfaces (Dassault Systèmes n.d.h). It displays the continuity through a comb-like shape which represents the radius as well as a green boundary which indicates the curvature (Dassault Systèmes n.d.h). An example of a porcupine analysis can be seen in Figure 32. *Connect Checker* returns what level of continuity is met by a surface, curve or connection between two surfaces or curves (Dassault Systèmes n.d.i). *Connect Checker* can also detect overlapping between surfaces or curves.

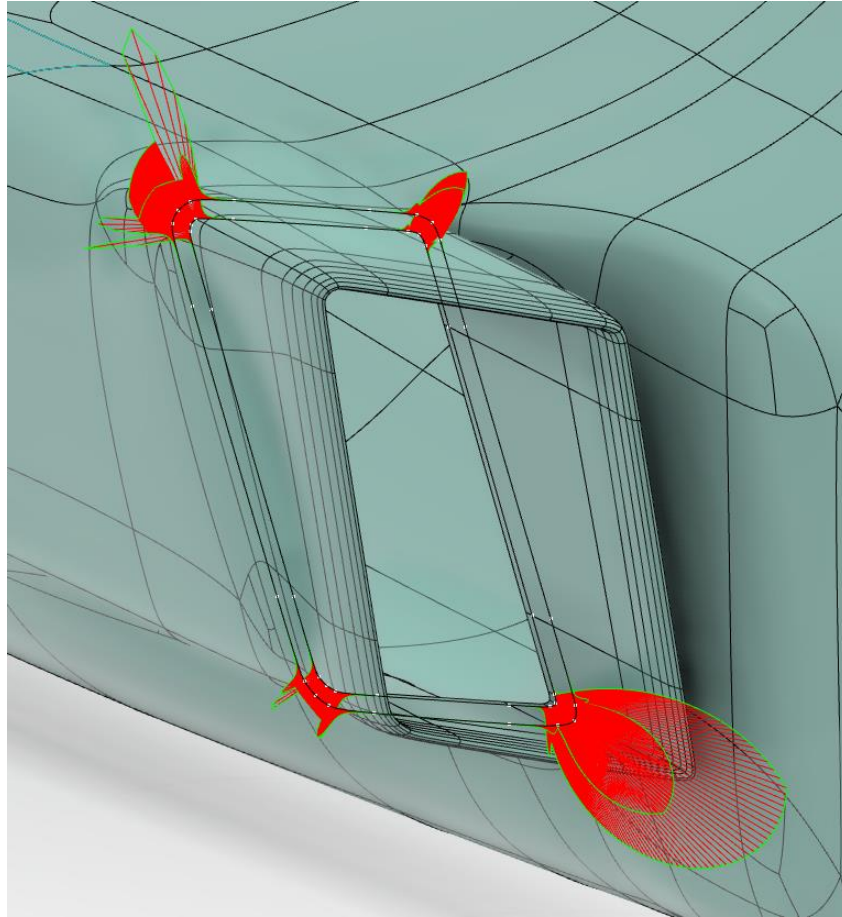


Figure 32. An example of a porcupine analysis. The red comb shape represents the radius and the green boundary represents the curvature.

The curvature analysis is implemented through the creation of a point which can move along the x-axis, or length of the fuselage. In this point a plane is created, which moves when the point is redefined. The plane is used to create an intersecting curve of the surface, a cross section, which can be analyzed using the two previously mentioned curvature methods. The intersect can be moved along with the point which means that cross sections along the whole geometry can be analyzed. Examples of how this is implemented are shown in Figure 33.

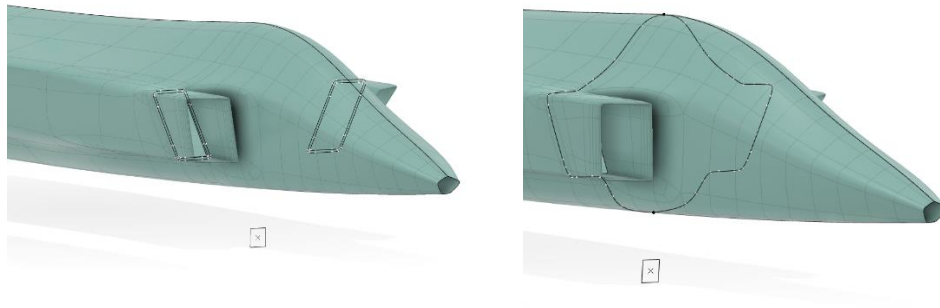


Figure 33. The point, plane and intersect used for curvature analysis is shown. On the left the air intake is analyzed and on the right the integration between the air intake and fuselage is analyzed.

4.4 Activate

The Activate step of MOKA includes testing of the KBE-application and creation of materials for the hand over to Saab. The testing includes trying out the different combinations and configurations as well as change of the available parameters in the UserForm. Limitations of parameter values are also established when possible. The hand over material is the KBE-application, the CAD-files, VBA as well as EKL scripts and videos presenting the functionality and workflow of the application.

five

5 Result

In this chapter the result of the Package step is presented. First the result from the evaluation of the mesh node methods is presented, second the result of the hi-fi modeling and automation is summarized, followed by the result of the curvature analysis. Last an evaluation of IMA is presented.

5.1 Explore methods to adjust mesh nodes

The evaluation of the two mesh node methods using user tests is presented in Table 11. It can be seen that both the number of errors and time consumption is minimized when the users applied Method 2. The average time saved when using Method 2 is 167 seconds per implementation for a surface containing 10 mesh nodes.

Table 11. Result of evaluation of mesh node methods.

User	Previous experience in IMA	Number of errors Method 1	Time consumption Method 1	Number of errors Method 2	Time consumption Method 2
P1	Experienced	1	3 min 28 s	0	1 min 5 s
P2	Experienced	0	2 min 10 s	0	51 s
P3	No experience	3	5 min 11 s	0	1 min 58 s
P4	No experience	3	7 min 55 s	1	3 min 39 s

The result of the number of mouse clicks needed to align and link different number of mesh nodes using the two methods is presented in Table 12. The number of saved mouse clicks using Method 2 increases with the number of mesh nodes on the surface.

Table 12. Number of mouse clicks needed for execution of the two methods for different number of mesh nodes.

Number of mesh nodes	Mouse clicks Method 1	Mouse clicks Method 2	Number of saved clicks using Method 2
1	13	11	2
10	112	38	74
100	1102	308	794

The result of the user test questions shows that the users agreed that Method 2 was best suited for completion of the task as well as for a user with no previous experience of IMA. Method 2 also performs better regarding time consumption, mouse clicks as well as minimization of errors. When taking the previously mentioned evaluations into account it can be established that Method 2 performs better than Method 1 for this purpose.

5.2 Modeling and automation

Result of the hi-fi modeling and automation, and the integration with the fuselage.

The methodology created to generate air intakes through design automation is divided into five steps, seen in Figure 34. The steps include the three sub goals regarding the development of methods set in 1 Introduction. The methodology will explain how the user goes from no air intake to an integrated fuselage-air intake model and includes both automated as well as manual steps. The PowerCopies needed for the application are presented in Table 13.

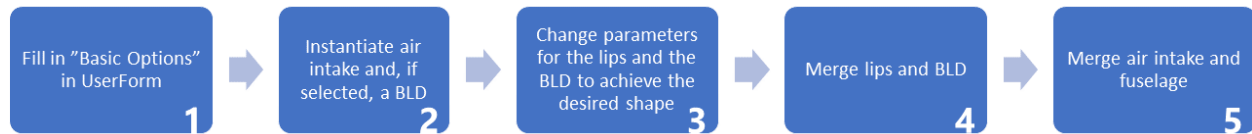


Figure 34. The step-by-step process for generating and integrating air intakes through design automation.

Table 13. The PowerCopies created to enable design automation of air intakes.

PowerCopy	Inputs
Points and planes	<ul style="list-style-type: none"> • A spline that defines the fuselage • A global yz plane placed at the nose of the aircraft
Air intake lips	<ul style="list-style-type: none"> • A point along the fuselage spline that specifies the positioning of the air intake lips • A plane at the same place as the point • A global yz plane • A global axis system
Boundary layer diverter	<ul style="list-style-type: none"> • A spline that defines the fuselage • Three points along the fuselage spline

From the PowerCopies' inputs, some of the initial requirements needed for the application are specified, which are:

- Splines that define the fuselage outer shape, positioned where the air intakes should be placed
- A global axis system and a global yz plane at the aircraft's nose

For the integration of the lips and the BLD surfaces a closed fuselage subdivided surface needs to be included in the start model. Some parameters are shared between the air intake lips and the BLD. To enable these components' parameters to be connected, thus change simultaneously, global parameters should be created in the file. The document with these initial components, see Figure 35, has to be open when the application is run.

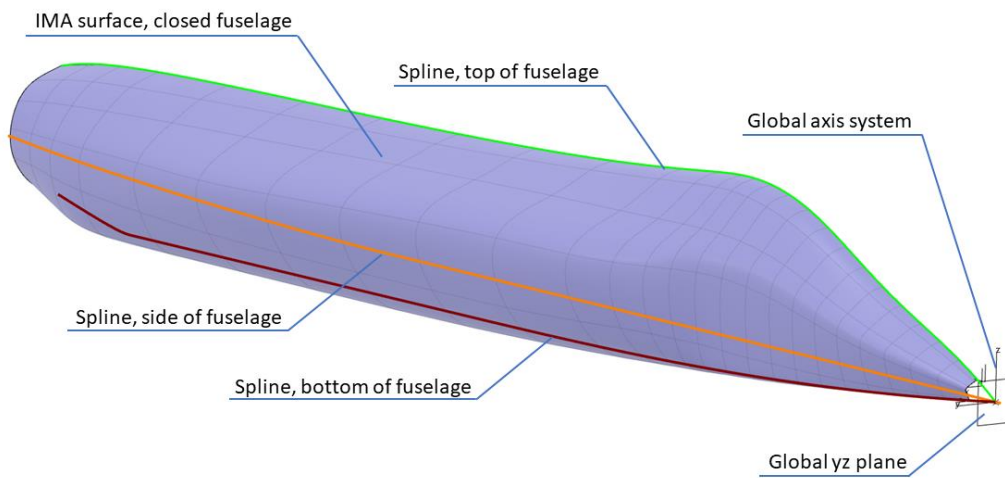


Figure 35. The start model with the initial components needed to run the air intake application.

When starting the application, a UserForm pops up that gives the user three options: generate a new air intake, change parameters for an existing air intake or delete an old air intake, see Figure 36. For generating a new air intake, firstly (step 1) the user fills in the “Basic Options” in the UserForm which consist of six selections: number of engines, number of air intakes, cross section, position, boundary layer diverter, and if the air intake should be integrated in fuselage. To meet the requirements from the compatibility matrix, the number of alternatives of the inputs changes depending on the input selections the user makes. For example, two engines will require two air intakes whereas one engine can have both one and two air intakes, when selecting two engines the option one air intake will be removed.

In step 2, after the basic selections are made, the user can press the button *Instantiate air intake* which will first instantiate points and planes along a spline at the selected position and then generate a PowerCopy with the air intake lips using the newly instantiated points and planes. If a BLD is selected, the lower button in Figure 36, *Instantiate BLD*, appears which will instantiate a

BLD into the model when pressed. At the instantiation of the air intake and the BLD, the dimensions of the components will initially be set to default values.

Figure 36. The UserForm that pops up when starting the application.

In step 3, the user gets to change the dimensions of the components through changing their parameters. Depending on the selected cross section, different tabs with boxes where the user can change parameters will appear in the UserForm, see Figure 37 and Figure 38. An information frame called *Dimensions* appears on the right side in the UserForm to give the user information about the current parameter values of the model. If the user would press the button *I want to change parameters to my existing air intake* when starting the application, the UserForm would change into this parameter view.

Figure 37. The UserForm displays the changeable parameter for the air intake lips. The two tabs are marked with (r), which means that a rectangular or semicircular air intake is instantiated.

Figure 38. The UserForm displays the changeable parameter for the BLD.

Since there are two models for the cross sections as mentioned in 4.3.2 Development of the air intake application, the parameters for the rectangle and the semicircle can be adjusted in the same tabs (marked as (r)) and the same applies for the circle, the oval and the bean (marked as (c)). Figure 37 presents the changeable parameters for rectangular lips and Figure 38 presents the changeable parameters for a rectangular BLD. The user can change parameters both in the

UserForm and in the CATIA parameter tree, except for the angle parameters which have to be changed in the UserForm. For the profile of the BLD the user can choose between *BLDDefault* and *BLDUserSketch*, the parameter *BLD Type* in Figure 38. If *BLDUserSketch* is selected, the user is asked to manually adjust the sketch in CATIA.

In step 4 (Merge lips and BLD) and 5 (Merge air intake and fuselage) the components will be integrated into each other and the fuselage. These steps must be done manually by the user using IMA tools. The sub-steps for steps 4 and 5 follow the same structure, therefore picture examples will only be given for the integration between fuselage and lips. The integration process between BLD and lips is further explained in 4.3.3.1 Integration between lips and boundary layer diverter. If the air intake is placed at the top or the bottom of the fuselage, the air intake surface needs to be cut by a global zx plane, using *IMA – Cut by plane*, since the start model is half a fuselage. Further, the target surface has to be cut open where the other surface should be integrated by using the tool *IMA – Erasing* or *IMA – Cut by Edges*. In step 4, the front of the lips needs to be cut and in step 5 the fuselage needs to be cut. In Figure 39, a bean shaped air intake is placed at the top of the fuselage and divided by a zx plane, and the fuselage is cut where the air intake will be merged.

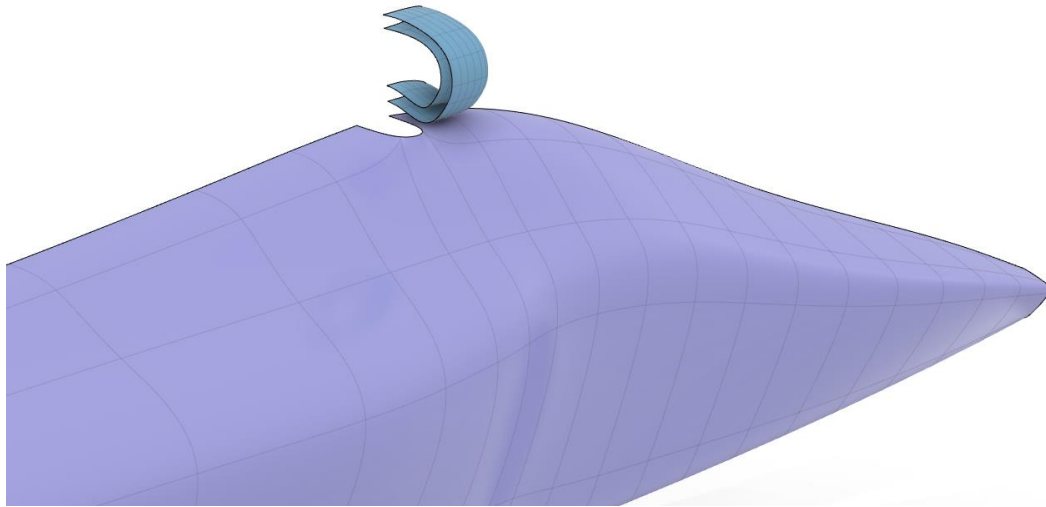


Figure 39. Half a bean shaped air intake on the top of a fuselage, and an opening in the fuselage where the air intake will be integrated.

In step 5, the fuselage surface then needs to be adjusted with *IMA – Modification* to follow the shape of the air intake and to make the open edges meet the back of the air intake before merging. In step 4, this sub-step is not necessary since the BLD is placed at the right position during instantiation. To be able to merge two surfaces together the number of subdivisions at the edges needs to match. The last sub-step before merging the surfaces is to subdivide the surface with the least number of subdivisions at the open edges with the tool *IMA – Face cutting*. In Figure 40 the fuselage surface is modified and subdivided to fit the outer geometry of the air intake surface. The last sub-step is to merge the surfaces with the tool *IMA – Merge*. The surface that should keep its linking should be selected first, and the other surface second. The merging is then done by selecting

each edge to merge, may need to be done in sections. Figure 41 presents bean shaped lips merged with a fuselage.

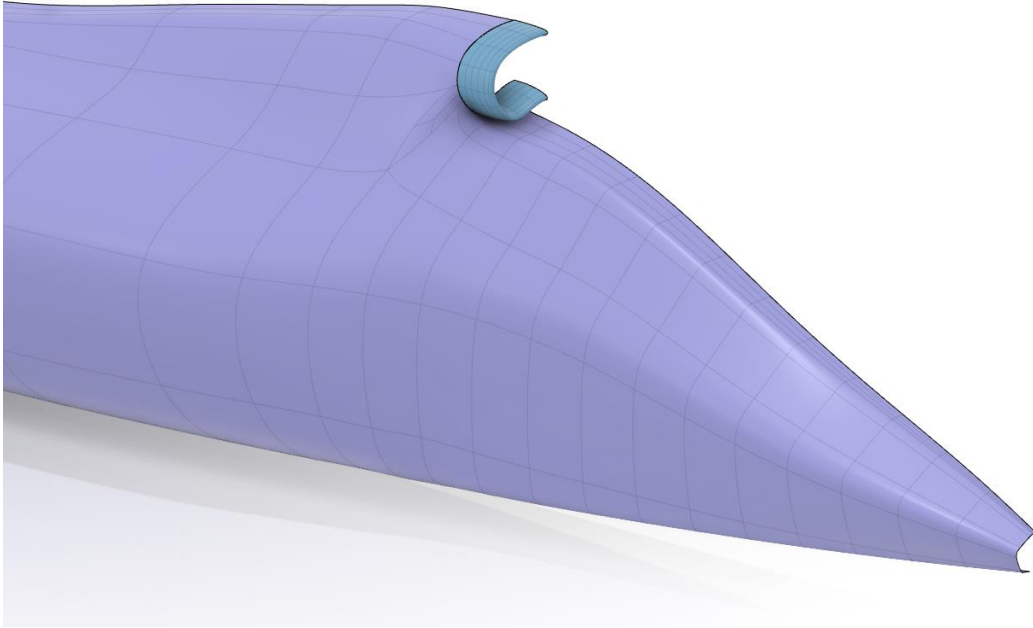


Figure 40. The IMA surface for the fuselage modified to match the back of the air intake.

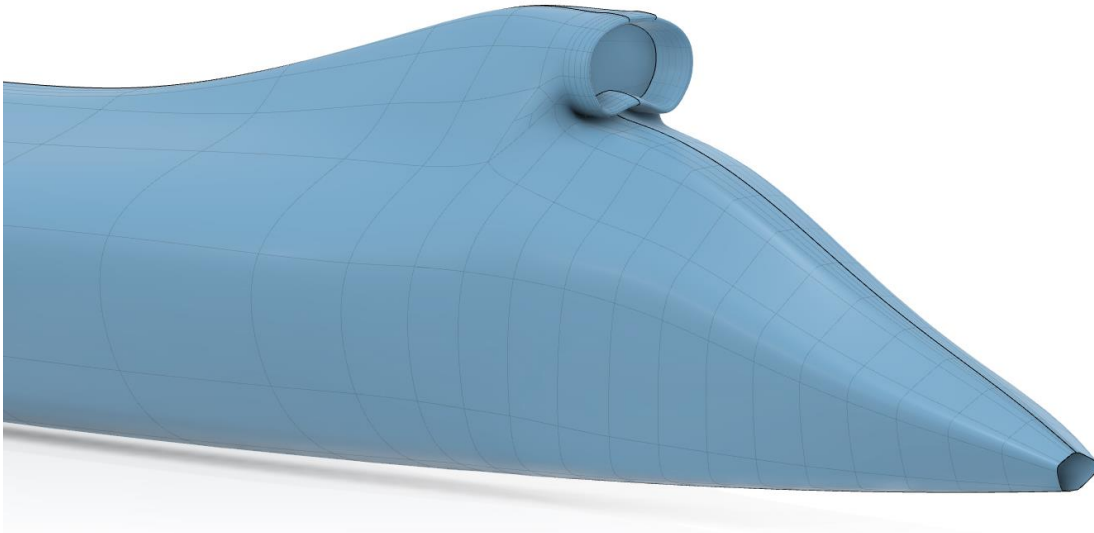


Figure 41. A fully integrated bean shaped air intake.

5.3 Curvature analysis

The curvature analysis for an integration between a rectangular air intake and the provided fuselage is presented in Figure 42. The figure presents both the result from the porcupine analysis and the connect checker analysis. In the figure a red comb-shape can be seen, it represents the radius of the curvature. A green boundary around the red comb indicates the continuity of the cross section. For the connect checker the result is shown as a percentage of the deviation from G^3 and G^2 continuity. In the pictures in this subchapter the light blue indicates a deviation in G^3 continuity and dark blue indicates a deviation in G^2 continuity. The overall integration follows G^2 continuity, though the integration follows G^3 continuity at parts. The parts following G^3 are not highlighted in any color in the connect checker. The biggest deviation in continuity for the integration is found in the corners of the air intake.

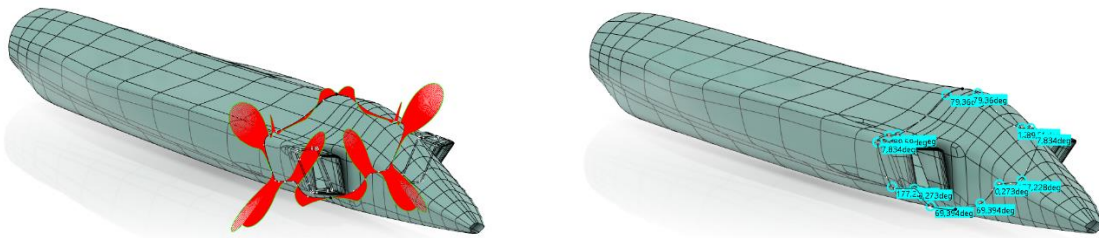


Figure 42. Result of curvature analysis for the integration between the air intake and fuselage. The porcupine analysis is displayed on the left and the connect checker on the right.

The result of a continuity analysis for the rectangular air intake and the fuselage is presented in Figure 43 and Figure 44 respectively. The figures show that both the fuselage and the air intake follow G^2 continuity and even G^3 at parts. The fuselage also contains parts where the surfaces only follow G^1 continuity, represented by dark blue in the connect checker.

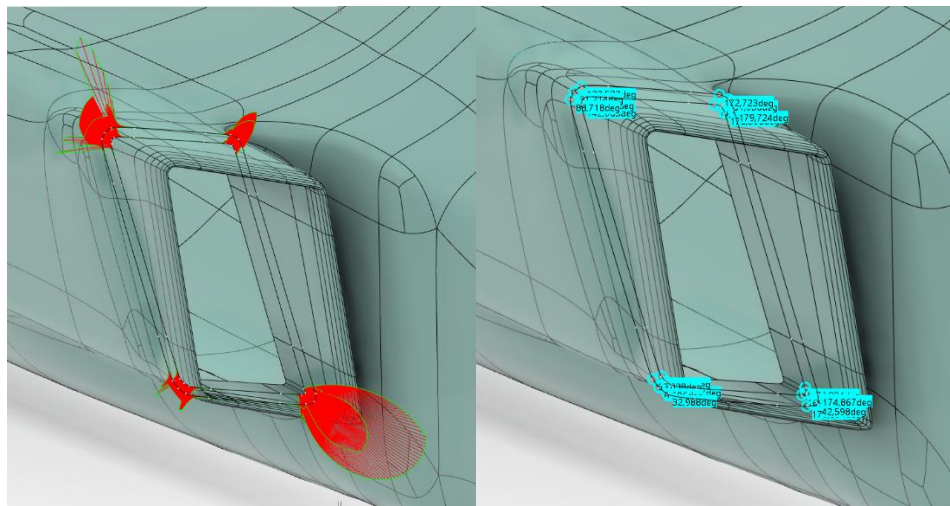


Figure 43. Result of a curvature analysis for a rectangular air intake. The porcupine analysis is displayed on the left and the connect checker on the right.

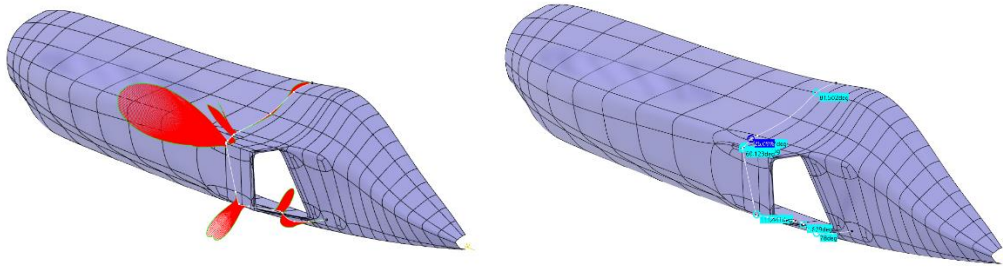


Figure 44. Result of a curvature analysis for the provided fuselage. The porcupine analysis is displayed on the left and the connect checker on the right. The dark blue indicates that G^2 continuity is not fulfilled.

5.4 Evaluation of Imagine and Shape

The positive as well as negative aspects of using IMA for automation in a later concept development stage is summarized in Table 14. It can be seen that subdivided surfaces enable flexible and fast concept modeling with a high level of continuity, but that the actions in IMA are limited and not compatible with design automation without manual work.

Table 14. A summary of the positive and negative aspects of using IMA.

Positive aspects	Negative aspects
Subdivided surfaces ensure surfaces with a high level of continuity(G^2), even when merging two surfaces.	Contains few, limited actions.
Enables fast and flexible modeling of non-exact concepts. Similar to clay modeling.	Subdivided surfaces can't be integrated with surfaces from other applications while still keeping their characteristics.
Easy to use for modeling of a concept from reference photos.	A lot of manual work is needed: <ul style="list-style-type: none"> • Parametrization of IMA models is possible but requires alignment and linking of mesh nodes to points. This is time consuming for complex models, even when using Method 2. • IMA is not directly compatible with VBA.
Easy to learn for a user who has worked with other applications in 3DX CATIA.	Unreliable: <ul style="list-style-type: none"> • Linking of mesh nodes to points is not always exact. In some cases, the mesh nodes move away from the point they are linked to. An example of this can be seen in Figure 45. • The linking function does not always work. At times the nodes can look like they are linked even though they are not. Thus, they have to be linked several times.
It is possible to parameterize surfaces through linking of mesh nodes to points or lines and the action <i>IMA – Control of dimension</i> .	When using <i>IMA – Merge</i> one of the surfaces to be merged loses its link to the reference points.
It is possible to draw a profile in IMA by using for example <i>IMA – Mesh drawing</i> .	To merge two subdivided surfaces, the edges to be merged need to have the same number of subdivisions, which can be hard when integrating two models which are both linked.
	When using <i>IMA – Link</i> , models quickly become complex, and changes therefore takes a lot of time to load and the application often crashes.
	Carries no history; it is hard to go back to a previous version of a model.
	The subdivided surface does not indicate if the surface is deformed, for example if a mesh node is misplaced which creates a fold in the surface. This has to be analyzed with the tool Connect checker.

As can be seen in Table 14 the action *IMA – Link* is not always reliable. This can be seen in Figure 45 where the mesh nodes remain linked but have moved from the point which they are linked to.

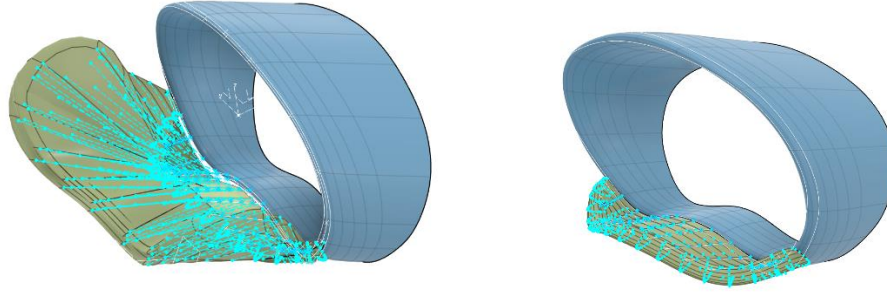


Figure 45. Lips and BLD rotated around y-axis. The linking of mesh nodes in the left BLD is not working properly.

six

6 Discussion

In this chapter the methods used in the thesis, the result and potential improvements are discussed.

6.1 Method discussion

The main methodology used in the project is MOKA in combination with SCRUM. MOKA is created for development of KBE-applications. The implementation of MOKA supported the initial parts of the project where the main focus was planning and understanding the scope of the project. The formal model divided the gathered knowledge into a requirement specification, a DSM-matrix and a compatibility matrix. This was of great importance for the execution of the project. The requirement specification was used during designing, evaluating and testing the application to ensure that necessary requirements were met.

The sprints in Package have not been implemented using the same time frame. While some sprints have been completed during one week, other sprints have taken a longer time to complete. By extending some sprints the result could be of higher quality as more time could be spent on details. Each week the work for the upcoming week has been organized using planning tools. In addition, the GANTT-chart was updated regularly and weekly meetings were held with stakeholders to ensure that the work was heading in the right direction. While the time frame for the sprints has not been the same for all parts of the project the overall time frame for the project has been held. The detailed planning for the weekly work as well as for a longer time frame has helped to delimit the work and estimate time consumption for specific tasks.

Early in the project interviews were held to determine desired parameters for the application. The interviews also aimed at increasing the understanding of what the design process looks like today, as well as increasing the understanding of air intakes. Holding the interviews in an early project stage was beneficial for building a basic understanding before the development of the KBE-application. Since the interviewees were from different departments, a broad range of knowledge and desires was acquired. The semi structured layout of the interviews allowed follow-up questions and increased the freedom during the interview. This was advantageous as while preparations

before the interviews increased the basic understanding in known areas, follow-up questions during the interviews deepened the knowledge of unknown areas.

The utilization of Lo-Fi models before the creation of Hi-Fi models was beneficial for the result of the thesis. This step allowed fast iterations of new concepts after feedback from supervisors as the concepts were not detailed. A thorough exploration of the design space and possible concepts could therefore be implemented. The Hi-Fi modeling used the frame models created during the Lo-Fi modeling and thus more time could be dedicated to modeling details and creating correct geometries. Though the time spent on Lo-Fi modeling could probably have been less to allow creation of more detailed geometries.

6.2 Explore methods to adjust mesh nodes

The time spent on exploring methods to control mesh nodes was limited as it was a sub goal of the project. Because of this a few brainstorming sessions were conducted where Method 1 and 2 were chosen for further investigation. If more time had been spent on exploration of potential methods, another method, more effective in handling complex geometries, might have been discovered. Method 2 requires the mesh node to be closest to the reference point it should be linked to, as it exploits distances to calculate which mesh node should be linked to which point. For complex geometries several mesh nodes may have to be moved manually before the script is run. This is to ensure that the alignment is implemented correctly. This implies that the number of mouse clicks, or manual work, required by the user increases, but the total amount of manual work is less than in Method 1.

The methods were evaluated using 4 users with different levels of experience of IMA. Further tests could be executed after the end of the project with more users with different experience of IMA. This should be done to ensure that Method 2 performs the best. More tests should also be performed for a more complex geometry to test how well the methods perform in such cases. This to ensure that the fidelity of the obtained result. Method 2 could potentially be used for other purposes when automating subdivided surfaces, as well as when exploitation of distances is needed.

6.3 Modeling and automation

In this subchapter the result of the modeling and automation of the air intake application is discussed.

6.3.1 Requirement specification

Most of the requirements from the specification were met. Some examples are *#1 Various cross section options*, *#2 Various position options*, *#Parametric model*, *#5 Boundary layer option*, *#7 Minimized required inputs for UDF* and *#10 Assure requirements from compatibility matrix*. Some requirements were not fulfilled. While many of the nonfulfilled requirements are of a high rank, the application still fills its main purpose by eliminating repetitive work, thus considered feasible.

One requirement that is not fulfilled is *#11 Update the model fast*. Each update of the model has to be loaded for a long time. This is due to the complex geometry of linked surfaces in IMA, even small changes in parameters can take minutes to load. Another aspect affected by the loading time

is defining the file with active bodies in CATIA when running a script for changing parameters. There might be an issue when the script tries to access the defined part. 3DX is a cloud-based program and as no files are saved locally on the hard drive, this may affect the required time for running the script as the cloud has to be explored.

Another requirement was to *#8 Add air intakes automatically* and *#9 Remove air intakes automatically*. The adding of air intakes is possible but the new air intake is not integrated with the fuselage and thus has to be modified, merged and relinked manually. The reason is that IMA actions are not compatible with automation. Any automation implemented exploits the benefit that mesh nodes can be linked to points. The built-in actions of IMA have to be handled manually. Surface actions in Generative Shape Design remove the special characteristics of a subdivided surface and implies that these actions can't be used. To automate the air intakes as much as possible the wireframe models are created through UDF's and instantiated through knowledge patterns. Here desired parameters are included and enables modification of the shape according to the users liking. An instantiated air intake is thus parameterized and flexible as the subdivided surfaces are linked to the reference points created in the wireframe model.

Removing air intakes is not completely possible in the created KBE-application. When using the button *Delete air intake* in the UserForm the instantiated geometry, parameters and formulas are automatically removed. But formulas that are created in the code, connecting the global parameters to the local parameters for the BLD and lips are not removed since the global parameters are not deleted. The instantiated relation-folders as well as instantiated knowledge patterns are not removed. This could possibly be solved through a search for the latest created formulas, knowledge patterns and folders in the global relation folder.

For the Lo-Fi requirement specification, the air intake positions *on wing*, *under wing* and *integrated in wing* were included – *#2 Various position options*. These positions were removed from the Hi-Fi requirement specification before the Hi-Fi modeling started as the scope of the project was too large. The wing was not created with subdivided surfaces which would have complicated the integration between the air intake and the wing. A further exploration of how these positions can be implemented is recommended for future studies.

Other aspects which are not completely fulfilled from the Hi-Fi requirement specification are *#14 Warning messages when outside limits*, *#15 Default max/min value when outside limits* and *#17 Naming convention*. The reason that no limits or maximum and minimum values exist is that the model is complex and a lot of parameters depend on each other. To compensate for the fact that these requirements are not fulfilled, starting values have been set in the instantiated model and the user is informed about these values in the UserForm. This makes it easier for the user to approximate what values the parameters can be set to and possible combinations. The reason that the naming convention is not fully implemented is that some UDF's that were created early in the project, for example during Lo-fi modeling, didn't follow this convention. When these UDF's are used later on their names are as they were created. These three requirements were desired, not required.

6.3.2 Lips

As mentioned in 4.3.2 Development of the air intake application, the front frame of the lips rotates with its local axis system. This functionality was based on the requirement to enable rotation around the x-, y- and z-axis. The rotation of the axis system follows a rotation matrix whose formulas are put into the coordinates of the axis system; the sequence of the rotations will always be the same. This also causes the rotations to happen around its own axis. Since the rotation is local, if the frame is rotated around both y and z the frame will be sheared. To get a rotation without shearing the frame, the rotation could be a combination of using an axis system and formulas for the frame's points. The rotation around x could follow an axis system and the rotation around y and z could be made possible through creating formulas for the points that build the frame, with the axis system as a reference. Although this would require equations for the coordinates of each point which would be troublesome to maintain.

Knowledge patterns are used for measuring dimensions in the lips' geometry and are automatically executed when the part is updated. Even though the priority order is set to different orders, due to unclear reasons the knowledge patterns are not always correctly executed every time the part is updated. Usually some of the knowledge patterns are executed correctly the first time, and some are executed the second time a parameter, and the whole part, is updated. Since these knowledge patterns are essential for both the modelling as well as the integration between the components, the execution failure of the knowledge patterns causes problems when instantiating the components.

In the current methodology, a standard airfoil NACA0012 is used. Since the coordinates for the airfoil points are imported into parameters in CATIA with a rule, its coordinates could easily be interchanged to other standard airfoils. Although when connecting the parameters to each other with formulas, the interchangeability becomes more challenging since the proportions will not be included without manual work. Additionally, when including the airfoil and its parameters in a UDF the interchange of airfoil would not be updated in already instantiated UDFs, and will require an update of the UDF followed by a re-instantiation.

6.3.3 Boundary layer diverter

The BLD has to be modeled by using actions in Generative Shape design to create suitable references for a subdivided surface. Thus, a regular CAD surface first has to be created, on which reference points are created. Extra steps have to be implemented to create the same surface in IMA. The same applies for the lips of the BLD, which use airfoils. In Generative Shape Design the lips could have been created using the action *Sweep*. In IMA the airfoils have to be placed along an offset, to create reference points for mesh nodes. This means that modeling in IMA is more time-demanding than in Generative Shape Design for complex and exact geometries.

The main references for the subdivided BLD are a split and an offset of the top curve. Splits can cause problems when combined with automation, as it creates two surfaces and the side to be kept cannot be controlled and is not robust. Because of this, scripts deciding which side of the split surface to keep have to be created in EKL. Other problems have also been identified, where most problems could be solved using scripts, while others could not. For example the BLD cannot rotate around the y and z- axis and therefore a calculation of the placement of the BLD is necessary when

the lips are rotated around these axes. The reason that the BLD is not possible to rotate around these axes is because the splits and actions from Generative Shape Design, used for reference modeling, sometimes fail. But the calculation of the new placement of the BLD compensates for the restricted rotations as the BLD uses *IMA – Cut by Plane* to achieve the desired shape.

Another problem when rotating the lips is that when a rotation around both the y-axis and the z-axis is implemented the shape is in a way sheared around the x-axis. Thus, the BLD is misplaced in the z-direction. A compensation for this has been implemented by using the parameter which allows the user to reposition the shape in both y- and z-direction. This parameter has to be modified manually by the user to attain the desired position. For the rectangular/semicircular shape this can also be compensated by utilization of the parameter which shears the shape around the x-axis, which is possible for the BLD as well.

At first it was desired to enable shaping of the BLD in more ways than later turned out to be possible. It was desired to create a profile for the height of the BLD, which was not possible because this created a boundary which was not continuous and could therefore not be used as a reference later on. For a similar reason the UserSketch has to be continuous which limits what the profile can look like. But the UserSketch is still flexible and enables the user to create its own profile, while the BLD's placement is automatically adjusted.

6.3.4 KBE-application

The creation of the KBE-application could be improved in some areas which have not been mentioned in the requirement specification. The lips and BLD are created differently, where the lips use the inner frame as a middle frame while the BLD uses both frames. This difference has to be considered upon integration between the two modules as calculations for alignment is needed and is implemented through a knowledge pattern. As knowledge patterns are not always updated when input values are, the BLD does not always get the right input, which can lead to errors. Ideally the BLD and lips would have the same input to avoid this problem.

Designing a subdivided surface which follows a reference surface or spline requires many reference points. This minimizes the calculation process in IMA to attain a surface of high continuity. When airfoils are used during the surface modeling a lot of airfoils are needed to create a surface, following the wireframe model. The models quickly attain a high complexity and are hard to maintain as well as take a long time to update changes.

Another aspect which should have been considered is that the different basic shapes, rectangle/semicircle and oval/circle/bean, should have been created with the same number of points on the wireframe. The reason is that the subdivided surfaces are created with the same number of subdivisions as the number of reference points. The number of subdivisions is important later, when the BLD and lips are to be merged as they have to consist of the same number of subdivisions to be merged completely. When the lips are subdivided to fit the BLD before merging, new mesh nodes are created that need new reference points. This should have been considered in an earlier phase but there was also a tradeoff between resolution and time required for the linking process when modeling. The action *IMA – Merge* had not been explored when the wireframe models had been created either, therefore the problem was unknown.

The creation of the wireframe models for the lips and BLD could be more automated and it might be possible to automate the complete modeling process in the future to save time. As the wireframe models are created mostly in the application Generative Shape Design in 3DX, which is compatible with automation, the creation of the wireframe models can be more automated. Since a methodology for creation of the modules was explored, automation was difficult to implement for creation of the wireframes. Future studies may include an exploration of what parts of the modeling process can be completely automated.

More tests should have been executed earlier to minimize the number of errors found and corrected during final testing. The final testing of the application should have been more thorough to ensure robustness. As too much time was spent on modeling and automation the final testing only contained necessary tests to ensure fidelity of the application and to determine limitations. Tests should have been made together with a potential user to get feedback on the UserForm and required inputs.

When an instantiated air intake has been merged with the fuselage only the air intake remains linked to its mesh nodes. This implies that the user manually has to relink the fuselage nodes, and potentially align the nodes with the points using one of the mesh node methods. What also needs to be considered is that reference geometry for modification of the fuselage surface to match the air intake are not included in the PowerCopy. The fuselage shape is thus not updated when air intake parameters are. A solution to this could be to include references for the fuselage in the air intake PowerCopy. More frames or splines could be used as additional references.

6.4 Curvature analysis

The result from the curvature analysis for an integrated air intake is dependent on the air intake model and how well the user has modified the air intake to match the fuselage. Thus, every integration looks different and the continuity of the surface may deviate. The example shown in 5.3 Curvature analysis is of a rectangular air intake and therefore the integration of a bean intake may look different. The provided fuselage does not follow G^3 continuity everywhere, and even break G^2 continuity at certain parts. Therefore the integration between the fuselage and air intake will at most follow G^2 continuity if no further modification is implemented. Furthermore, only G -continuity can be measured in 3DX and the stricter C -continuity will not be discussed further.

When designing the wireframe for the lips an action called *Curve Blend* was used to ensure a high continuity between curves. Even though reference points were created on these curves the air intake did not attain a continuity of this level after the subdivided surface had been linked. An action, similar to *Curve Blend*, handling surfaces is *Patch from curves*. This action allows selection of the desired level of continuity with the disadvantage that the result is not a subdivided surface. Thus, other applications, surfaces and actions can be exploited if designing for G^3 continuity.

6.5 Evaluation of Imagine and Shape

IMA contains few, limited actions allowing modifications of a subdivided surface. The subdivided surface is not compatible with changes made in other applications than IMA. The subdivided surface thus becomes a regular CAD surface when applying changes using other applications and is no longer compatible with the actions available in IMA. This limits what geometries can be

modeled and implies that in order to build certain geometries complex references have to be created. An example of this is the BLD which first had to be modeled using surfaces and actions in Generative Shape Design to create reference points which a subdivided surface could later reference. IMA is created to enable fast sketching of concepts during an early concept development phase and is thus not meant to be used for exact modeling. Because of this the time consumption needed when working with the application IMA is high as the limitations of the application have to be worked around. The same applies to design automation in IMA as the application does not allow automation of its actions. Thus, the subdivided surface has to be parameterized by utilizing a wireframe model, linked to the mesh nodes of the subdivided surface. This is time consuming and if changes are made to the wireframe model, such as adding or removing reference points, the surface has to be relinked.

As the subdivided surface does not always keep its link to reference points it is not robust and is unpredictable and therefore not suitable for automation of complicated geometries. While IMA works well for early concept development, and even automation of less complicated geometries, it is not suitable in a later concept development stage. There is a tradeoff between robustness, time consumption and the order of continuity offered by subdivided surfaces.

seven

7 Perspective

Design automation is a growing field which along with other types of automation aims at minimizing manual work. This increases the possibility to spend time on other tasks and can let the engineer be more creative. Design automation also allows for fast iterations of new concepts with a mouse click. It means that cost savings can be made as a computer executes the manual work. Although design automation should also be applied with carefulness as it also can affect our society. Some jobs may be replaced by a computer while others' tasks can change.

Effective air intakes are essential for the propulsion of an aircraft. Because of this it is beneficial to be able to create fast iterations of new, flexible concepts for calculations and tests. In this context design automation may lead to more time being spent on running tests or calculations of different models, rather than developing concepts. This can result in more effective propulsion of the aircraft, thus affecting the environment in a positive way.

eight

8 Conclusion

In this chapter the conclusions drawn for each research question are presented.

RQ1: How can an automated framework be implemented with subdivided surfaces in a CAD program through KBE to eliminate repetitive work and facilitate changes during the concept development of air intakes?

The framework cannot be fully automated in the current version of 3DX. The instantiation and dimensions of the air intake components can be automated and parameterized. Although the steps for integration of the components require manual work. The automation has been implemented through utilization of PowerCopies, UDF's, VBA, rules and Knowledge Patterns.

RQ2: How can the position of mesh nodes on a subdivided surface in a CAD program be adjusted?

Mesh nodes can be controlled using two methods: one manual and one semi-automatic. The manual method requires utilization of IMA tools. The automatic method replaces most of the manual steps using VBA scripts – saving time and mouse clicks.

RQ3: What order of continuity can be achieved for an integration between an air intake and fuselage in a CAD program?

The geometric continuity for an integration between subdivided surfaces can be analyzed using integrated analysis tools in the 3DX. The highest order of continuity that can be achieved is the order of continuity of the air intake and fuselage components.

nine

9 Future studies

The research performed in this master thesis has created a basis for implementing design automation in the concept development process. To further expand the possibilities of design automation of air intakes and the integrations between surfaces, the following areas of future research have been identified:

- Investigate how the air intakes can be placed on the positions: on wing, under wing and integrated in wing
- Conduct research for how a similar framework can be developed for variable air intakes
- Develop a methodology for creation of the inner geometry of air intakes
- Explore if there exists a mesh node linking method that can better handle complex geometries
- Investigate what parts of the creation of the components lips and BLD can be automated
- Explore other tools for creating models of a high order of continuity and tools for analyzing the continuity of these models

ten

10 Reference

Amadori, Kristian. 2012. *Geometry Based Design Automation: Applied to Aircraft Modelling and Optimization*. Linköping University.

<http://liu.diva-portal.org/smash/record.jsf?pid=diva2%3A466519&dswid=-2599>

Barsky, Brian A. and DeRose, Tony D. 1984. *Geometric Continuity of Parametric Curves*. (Publication No. UCB/CSD 84/205). University of California.

<https://digitalassets.lib.berkeley.edu/techreports/ucb/text/CSD-84-205.pdf>

Comba, Joao L. 1993. *Continuity Aspects of Spline Curves* [Handout]. Stanford University, Stanford, California.

Dassault Systèmes. n.d.a. *The 3DEXperience platform*. Available at:

<https://www.3ds.com/3dexperience> [Accessed 2023-02-14]

Dassault Systèmes. n.d.b. *CATIA / Shape the world we live in*. Available at:

<https://www.3ds.com/products-services/catia> [Accessed 2023-02-17]

Dassault Systèmes n.d.c. *Beyond 3D CAD design products*. Available at:

<https://www.3ds.com/products-services/catia/products/> [Accessed 2023-02-27]

Dassault Systèmes. n.d.d. *CATIA V5-6R Drives higher design excellence - CATIA - Generative Shape Design 2 (GSD)*. Available at:

https://www.3ds.com/products-services/catia/products/v5/portfolio/domain/Shape_Design_Styling/product/GSD/#overview
[Accessed 2023-05-17]

Dassault Systèmes. n.d.e. *CATIA V5-6R Drives higher design excellence - CATIA - Imagine & Shape 2 (IMA)*. Available at:

<https://www.3ds.com/products->

services/catia/products/v5/portfolio/domain/Shape_Design_Styling/product/IMA/ [Accessed 2023-02-27]

Dassault Systèmes. n.d.f. *Working with the Imagine and Shape Product*. Available at: http://catiadoc.free.fr/online/CATIAfr_C2/imaugCATIAfrs.htm [Accessed 2023-03-01]

Dassault Systèmes. n.d.g. *Instantiating a User Feature Using the Knowledge Pattern*. Available at: http://catiadoc.free.fr/online/cfyugpkt_C2/cfyugpktudf0012.htm [Accessed 2023-03-01]

Dassault Systèmes. n.d.h. *Performing a Curvature Analysis*. Available at: http://catiadoc.free.fr/online/cfyugfss_C2/cfyugfssut0304.htm [Accessed 2023-05-02]

Dassault Systèmes. n.d.i. *Connect Checker*. Available at: http://catiadoc.free.fr/online/cfyugfss_C2/cfyugfssut_implicitmode_0311.htm [Accessed 2023-05-02]

DesignTech CAD Academy. n.d. *CATIA Macros*. Available at: <https://www.designtechcadacademy.com/knowledge-base/catia-macros> [Accessed 2023-03-01]

Eklund, Adam and Karner, Jesper. 2017. *Development of a Framework for Concept Selection and Design Automation: Utilizing hybrid modeling for indirect parametric control of subdivision surfaces*. [Master thesis] Linköping university
<http://liu.diva-portal.org/smash/record.jsf?pid=diva2%3A1231551&dswid=-1632>

El-Sayed, Ahmed F. 2016. *Fundamentals of Aircraft and Rocket propulsion*. Springer-Verlag London Ltd.

Engler, William O. 2013. *A Methodology for Creating Expert-Based Quantitative Models for Early Phase Design*. Georgia Institute of Technology
https://smartech.gatech.edu/bitstream/handle/1853/47670/engler_william_o_201305_phd.pdf?sequence=1&isAllowed=y

Eppinger, Steven D and Ulrich, Karl T. 2014. *Produktutveckling: konstruktion och design*. 1st edition. Lund : Studentlitteratur

Farokhi, Saeed. 2014. *Aircraft Propulsion*. 2nd edition. John Wiley & Sons Ltd.

Hallin, Anette and Karrbom Gustavsson, Tina. 2019. *Projektledning*. 3rd edition. Liber AB

Hertzum, Morten. 2022. *Usability Testing: A Practitioner's Guide to Evaluating the User Experience*. 1st edition. Springer Nature Switzerland AG.

Hünecke, Klaus. 2005. *Jet engines: Fundamentals of theory, design and operation*. The Crowood Press Ltd.

Kenton, Will. 2022. *Visual Basic for Applications (VBA): Definition, Uses, Examples*. Available at: <https://www.investopedia.com/terms/v/visual-basic-for-applications-vba.asp> [Accessed 2023-03-01]

Layton. Mark C. 2015. *Scrum for dummies*. John Wiley & Sons

Leavy, Patricia. 2014. *The Oxford Handbook of Qualitative Research*. Oxford University Press, Incorporated

Liedholm, Ulf. 1999. *Systematisk konceptutveckling*. Linköping.

Linköpings universitet. n.d. *Design Automation Lab*.

Available at: <https://liu.se/en/research/design-automation-lab> [Accessed 2023-01-24]

Lohith, M. L.; Prasanna, L. and Vaderahobli, D. H. 2013. *Translating MOKA based Knowledge models into a Generative CAD model in CATIA V5 using Knowledgware*. International Conference on Modeling, Simulation and Visualization Methods (MSV). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

madhav_mohan. 2022. *Parametric & Geometric Continuity of Curves in Computer Graphics*.

Available at: <https://www.geeksforgeeks.org/parametric-geometric-continuity-of-curves-in-computer-graphics/> [Accessed 2023-03-01]

Microsoft. 2022a. *Getting started with VBA in Office*. Available at:

<https://learn.microsoft.com/en-us/office/vba/library-reference/concepts/getting-started-with-vba-in-office> [Accessed 2023-03-01]

Microsoft. 2022b. UserForm object. Available at:

<https://learn.microsoft.com/en-us/office/vba/language/reference/user-interface-help/userform-object> [Accessed 2023-03-03]

Persson, Johan and Ölvander, Johan. 2021. *An Introduction to Engineering Optimization: Theory and Practical Guidelines*. Linköping university. [Internal document]

SAAB AB. n.d.a. *Purpose and values*. Available at: <https://www.saab.com/about/company-in-brief/purpose-and-values> [Accessed 2023-01-23]

SAAB AB. n.d.b. *Products*. Available at: <https://www.saab.com/products> [Accessed 2023-01-23]

SAAB AB. n.d.c. *Company in brief*. Available at: <https://www.saab.com/about/company-in-brief> [Accessed 2023-01-23]

SAAB AB. n.d.d. *Organisation*. Available at: <https://www.saab.com/about/company-in-brief/organisation> [Accessed 2023-01-23]

Sandberg, Marcus. 2015. *Towards a Knowledge-Based Engineering Methodology for Construction*. 1-8. DOI: 10.1061/9780784479377.001.

Sandberg, Marcus; Gerth, Robert.; Lu, Weizhuo; Jansson, Gustav; Mikkavaara, Jani and Olofsson, Thomas. 2016. Design automation in construction : An overview. *Proceedings of the 33rd CIB W78 Conference 2016, Oct. 31st – Nov. 2nd 2016, Brisbane, Australia, 2016*.

<http://urn.kb.se/resolve?urn=urn:nbn:se:ltu:diva-60207>

Schlichting, Hermann and Gersten, Klaus. 2003. *Boundary-Layer Theory*. 8th edition. Springer Berlin Heidelberg.

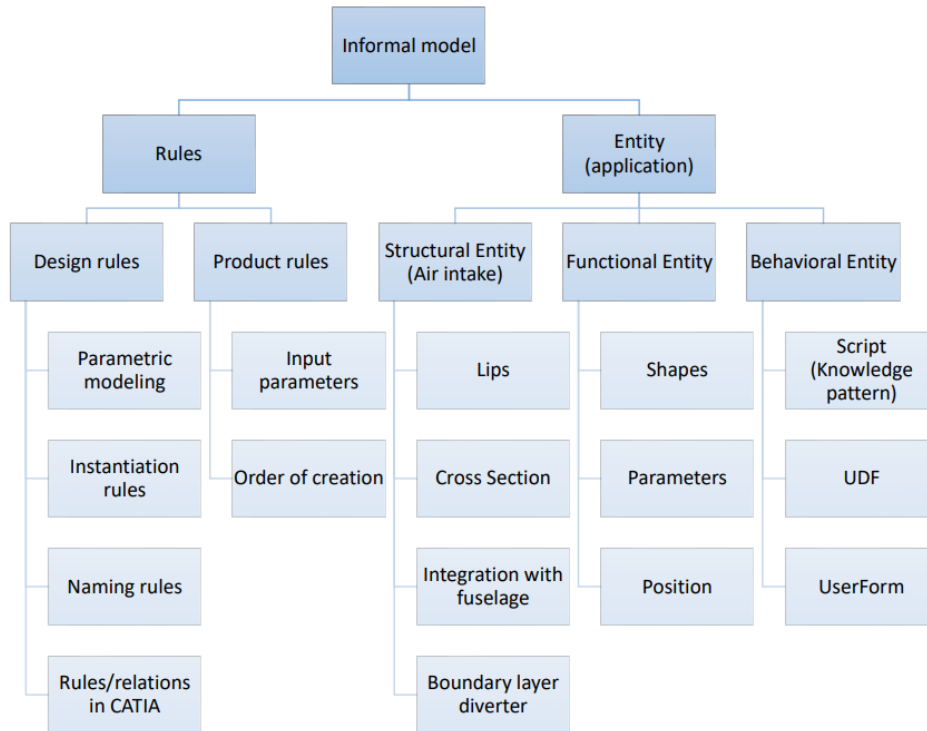
Stokes, Melody. 2001. *MOKA: Methodology for Knowledge Based Engineering Applications*. Professional Engineering Publishing Limited

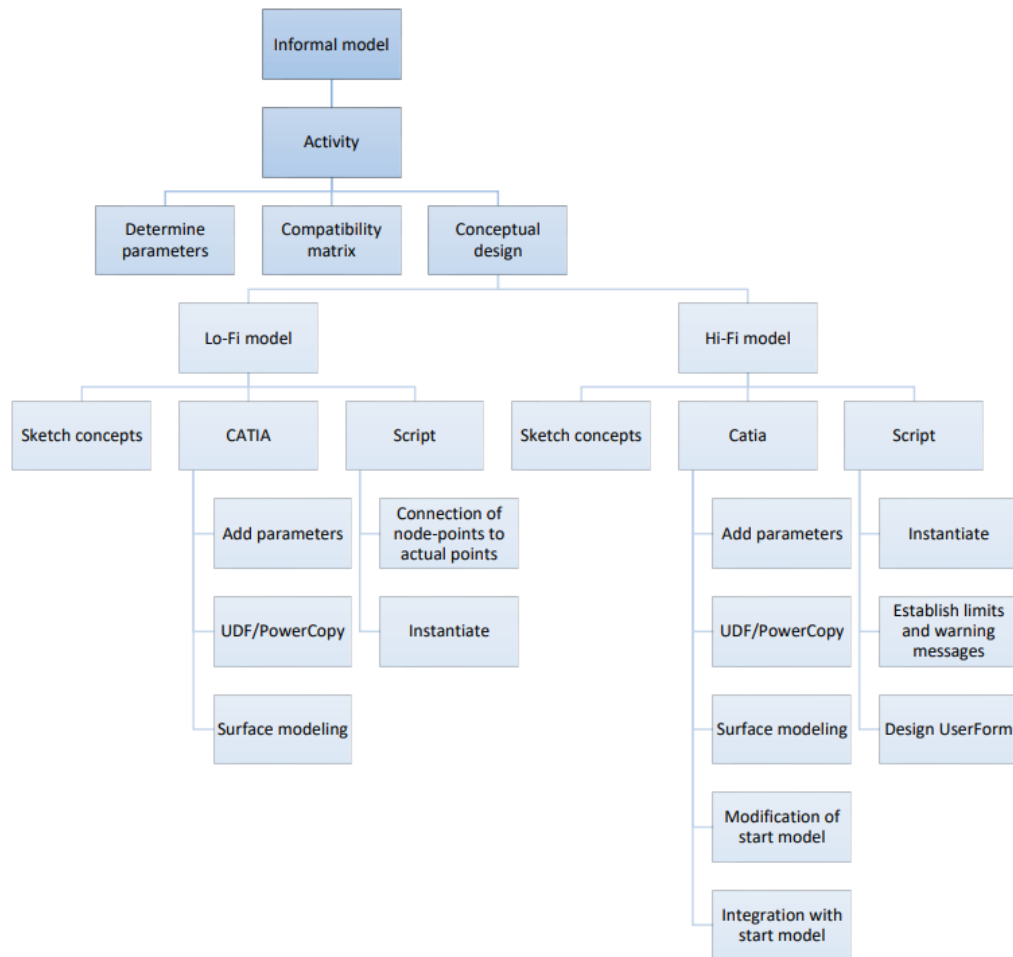
Svensson, Marlene. 2008. *A CFD Investigation of a Generic Bump and its Application to a Diverterless Supersonic Inlet*. Linköping University, Linköping.
<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-12490>

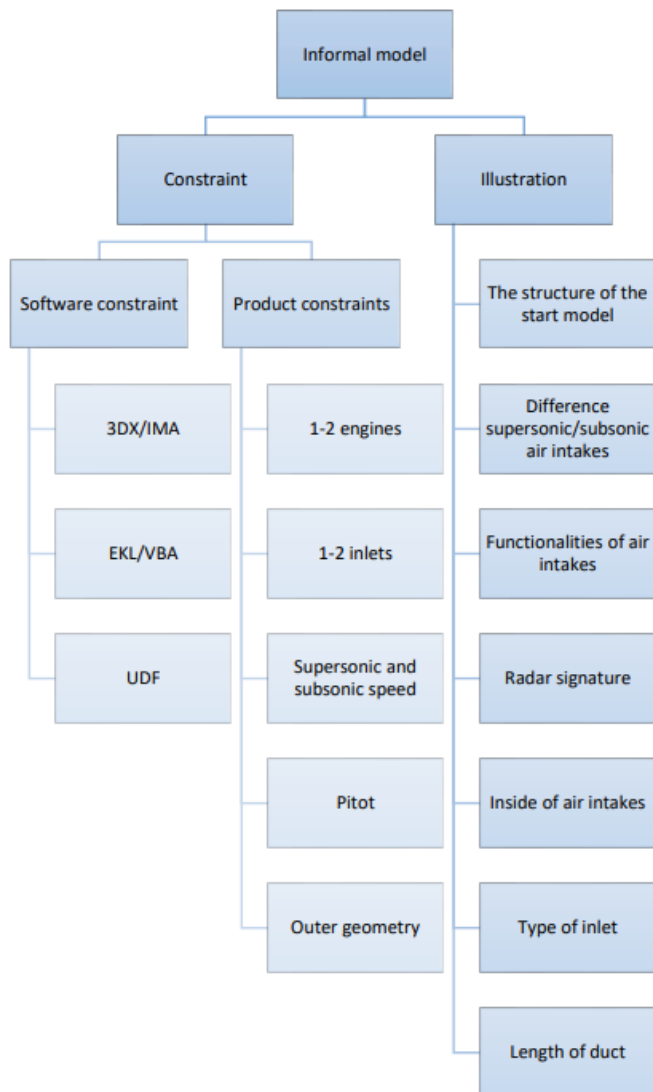
Tarkian, Mehdi. 2012. *Design Automation for Multidisciplinary Optimization: A High Level CAD Template Approach*. Linköping university.
<http://liu.diva-portal.org/smash/get/diva2:556208/FULLTEXT03>

Torabi, Farschad. 2022. *Chapter 3 - Basics of aerodynamics. In Wind Energy Engineering, Fundamentals of Wind Farm Aerodynamic Layout Design*. Academic Press. DOI:
10.1016/B978-0-12-823016-9.00009-7

Appendix 1 Informal model







Appendix 2 Interview questions

Examples of questions which were asked during the interviews, presented as they were asked.

- Vill du förklara i detalj hur luftintag fungerar, vad de fyller för syfte?
- Vilka typer av luftintag finns?
- Hur skiljer sig supersonic och subsonic hastigheter från varandra?
- Hur måste luftintag (främst yttre konstruktion) konstrueras för att de ska klara av hastigheter över ljudets hastighet?
- Vart kan luftintag placeras? Var är det vanligast att luftintagen är placerade?
- Vilka former brukar luftintag ha? Vilka är vanligast? Hur skiljer sig det beroende på placering (uppe/nere/sidor/vinge)?
- Hur många luftintag finns det vanligtvis?
- Vilka typer av motorer finns det, och hur påverkar de luftintagen?
- Vilka parametrar tror du är relevanta att ha med i ett tidigt stadie av konceptgenerering?
- Vilka parametrar styrde luftintag på draken, vingen och gripen?
- Hur jobbar ni på konceptavdelningen? Hur ser er process ut?
- Hur har ni jobbat med automation tidigare/nu?

The complete Compatibility Matrix for an aircraft, combined with the matrix created by Eklund and Karner (2017).

81

Appendix 4 Rotation matrix

$$\begin{aligned}
 R_x R_y R_z &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha_x) & -\sin(\alpha_x) \\ 0 & \sin(\alpha_x) & \cos(\alpha_x) \end{pmatrix} \begin{pmatrix} \cos(\alpha_y) & 0 & \sin(\alpha_y) \\ 0 & 1 & 0 \\ -\sin(\alpha_y) & 0 & \cos(\alpha_y) \end{pmatrix} \begin{pmatrix} \cos(\alpha_z) & -\sin(\alpha_z) & 0 \\ \sin(\alpha_z) & \cos(\alpha_z) & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
 &= \begin{pmatrix} \cos(\alpha_y)\cos(\alpha_z) & -\cos(\alpha_y)\sin(\alpha_z) & \sin(\alpha_y) \\ \cos(\alpha_x)\sin(\alpha_z) + \sin(\alpha_x)\sin(\alpha_y)\cos(\alpha_z) & \cos(\alpha_x)\cos(\alpha_z) - \sin(\alpha_x)\sin(\alpha_y)\sin(\alpha_z) & -\sin(\alpha_x)\cos(\alpha_y) \\ \sin(\alpha_x)\sin(\alpha_z) - \cos(\alpha_x)\sin(\alpha_y)\cos(\alpha_z) & \sin(\alpha_x)\cos(\alpha_z) + \cos(\alpha_x)\sin(\alpha_y)\sin(\alpha_z) & \cos(\alpha_x)\cos(\alpha_y) \end{pmatrix}
 \end{aligned}$$