Development, Model Generation and Analysis of a Flying V Structure Concept

by

L.A. van der Schaft

MSc student, Faculty of Aerospace Engineering, Delft University of Technology, Delft, the Netherlands
and
Graduate Intern, Future Projects Office, Airbus GmbH, Hamburg, Germany

in partial fulfilment of the requirements for the degree of

Master of Science

in Aerospace Engineering

at the Delft University of Technology,

to be defended publicly on Wednesday the 14th of June, 2017 at 9:00 am.

Student number:	4090128	
Thesis registration ID:	129#17#MT#FPP	
Project duration:	1 June, 2016 – 14 June, 2017	
Daily supervisors:	R. Vos	Delft University of Technology, the Netherlands
	K. Bender	Airbus, Hamburg, Germany
Thesis committee:	P. Colonna	Delft University of Technology, the Netherlands
	R. Vos	Delft University of Technology, the Netherlands
	J.M.J.F. van Campen	Delft University of Technology, the Netherlands
	K. Bender	Airbus, Hamburg, Germany

Summary

The Flying V, a concept aircraft configuration, has been proposed by Airbus and TU Berlin. The initial indications of potential improvments in performance over conventional aircraft gave rise to a continuation of the research. For this, the mass characteristics of the configuration must be explored.

A primary structure concept is conceived based on expected load paths. The original dual-tube fuselage configuration is replaced with an oval cabin concept, in which various constant-curvature arcs are adjoined. Residual forces at their connection are carried by straight members. A parametrisation is set up, for which both aerodynamic and structure concept considerations are taken into account. A CAD model for the structure concept is generated in Para*Py* , a KBE framework with CAD integration and coupling to various aerodynamic and structure analysis tools.

A numerical sizing method is selected due to the complexity of the airframe geometry. A model generator is developed in Para*Py* , a KBE-framework with integration of a meshing algorithm and coupling to aerodynamic and structural analysis tools. A CAD model is obtained from the model generator using a mission parameter set similar to that of an Airbus A350. The model is meshed using the SALOME algorithm for a finite element analysis.

The mesh is exported to Patran for a Nastran analysis. Between Para*Py* and Patran, mesh element IDs are altered. This disables a feedback-enabled design loop. Analysis results cannot be interpreted in Para*Py* . Automated design is foregone.

Patran mesh verification highlights critical and non-critical mesh errors due to the chosen geometry generation procedure. The flawed elements are redefined manually. Point and distributed loads are applied. Aerodynamic loads are obtained from Airbus' LatticeBeta full-potential 3D panel method. Uniform material properties are assigned. Due to the thin-shelled nature of the model, out-of-plane element bending stiffness is low. Excessive deformations are observed under transverse load application and in-plane panel compression. A bending stiffness correction is applied.

The model is sized for material failure and local buckling by Airbus' in-house structure sizing tool ZORRO. A validation is set up using outboard wing cross-sections. These are sized using analytical relations given the numerical procedure loads. From this it is concluded individual regions can only be assessed qualitatively due to the bending stiffness correction applied in the numerical model. Wrongly designed structure regions and load path propagation can be interpreted. Quantitative results are not omitted from the result interpretation.

An unforeseen load path is identified in the wing transition between the outboard wing front spar and root plane trailing edge. The integration of tapered fuselage within the wing transition is feasible, although the fuselage attracts more loads than anticipated. The fuselage transition behaves as expected.

Integration of the oval cabin concept into a the Flying V configuration is shown to be feasible and beneficial to volume efficiency and planform design flexibility. However, the existing implementation of the cabin concept must be altered for better adaptation in existing regulations.

The assumed static stability as optimised for by Faggiano shows large discrepancy with respect to the determined static margin. Part of this is caused by the incorrect quantitative sizing results. However, it is expected that the aerodynamic profile may be significantly altered in a multi-disciplinary design procedure.

In order to obtain a correct mass estimation, a new model generation procedure is recommended in detail. This takes multiple facets into account. Geometric model continuity is ensured. A straightforward implementation of stiffener elements is enabled by introducing a foundation for this on the top-level fuselage model definition. This also adds more possibilities for designing the aft half of an aerodynamic profile for the fuselage section of the planform.

Feasible design automation consequently enables multidisciplinary optimisation. The main hurdles to overcome towards realising design automation are development of a complete model generator, further development of Para*Py*'s geometry generation and Nastran coupling functionalities and feedback between analysis results and model generation. Realising these aspects in future research will finally allow solid conclusions regarding the Flying V performance to be drawn.

Acknowledgements

The completion of this MSc thesis research project concludes a chapter in my book of life. Seven years ago, I walked into this chapter, not fully grasping what to expect. I have been on adventures, big and small, significant and insignificant. The common theme however has always been the studying the Aerospace Engineering course, culminating in this MSc thesis project. The weight of the studies, and that of conducting the thesis research in particular, has been relieved by the support of several parties.

Many thanks go to my daily supervisor Klaus Bender of Airbus' Future Projects Office for his continued support, insightful advice and critical yet constructive feedback. He offered me a unique chance of working with the most interesting office of a global player in aircraft design.

My academic committee members deserve recognition. Roelof Vos and Julien van Campen provided contrasting academic guidance and feedback throughout and towards the end of the project respectively. Communicating project details with Roelof Vos over a distance of 500km and a country border was not always convenient given the available methods. However, Roelof showed understanding for and willingness to work with the situation.

The Para*Py* platform has proven indispensable in automated model generation. I would like to thank the Para*Py* team led by Reinier van Dijk and Max Baan for their support in development of a Flying V structure model. The many hours Max has spent with me over web calls in order to track bugs and set up a necessary remodel of the code structure has been helpful and provided useful insight in the workings of CAD geometry generation. He stood by me with exceptional motivation and a strong belief in power and capabilities of design automation.

A special mention goes to Jörn Clausen of Airbus, for sharing his knowledge and skills of Patran, Nastran and ZORRO with me. Without him, I would not have been able to circumvent the mesh and analysis issues encountered in Nastran, nor would I have obtained structure analysis results. Insightful content has been obtained with the help of Jörn.

Throughout the Aerospace Engineering course, I have enjoyed the support, trust and guidance of family and friends in various scenarios and environments. Although studying has taken us separate ways at different paces, there has always been mutual reliance, exchange of advise and support, moments of relaxation and joint effort in extra-curricular activities. Thank all of you for being a part in this.

Being a student at Aerospace Engineering in Delft has taught me useful skills, provided me with a solid set of professional values and allowed me to embark on incredible adventures. On top of that, it has set me up for a fruitful career. From it, I am taking with me a heavy but powerful toolbox called engineering, the strive for excellency and a mind that values diversity. For this, I would like to direct my final words of appreciation to the Delft University of Technology itself.

L.A. van der Schaft

Delft, March 2017

Contents

List of Tables

List of Figures

Nomenclature

BWB	Blended-Wing Body aircraft
CoG	Centre of Gravity
CSM	Combined System Mass
CST	Class-function/Shape-function Transformation
FE	Finite Element
FEM	Finite Element Method
FW	Flying Wing
KBE	Knowledge-Based Engineering
LE	Leading Edge
MAC	Mean Aerodynamic Chord
MF	Fuel Mass
MS	Structure Mass
MTOM	Maximum Take-Off Mass
MTOW	Maximum Take-Off Weight
OEM	Operational Empty Mass
s.m.	Static margin
TaW	Tube-and-Wing
TE	Trailing Edge
UERF	Uncontained Engine Rotor Failure
Δp	Pressure differential
$^t/_c$	Airfoil thickness-to-chord ratio
$d_{itrsect}$	Intersectional span width
h_1	Crown height
h_2	Cabin height
h_3	Keel height
l_w	Cabin wall length
N_e	Number of engines
n_{ribs}	Number of ribs
N_t	Number of fuel tanks
r_1	Crown arc radius
r_2	Side arc radius
r_3	Keel arc radius
$s_{rib,max}$	Maximum allowed rib pitch

s_{rib}	Rib pitch
w_{avg}	Cabin average width
w_{cutout}	Front web cut-out width
w_c	Cabin ceiling width
w_f	Cabin floor width
$w_{t,i}$	Total fuselage width at *section i*
γ	Cabin interior angle
μ	Average cabin interior angle
ϕ	Cabin interior angle
$\psi_{interior}$	Mesh element interior angle
ρ_{fuel}	Fuel density
MPL	Payload Mass
W_{api}	Air-conditioning, pressurisation and de-icing system weight
W_{apu}	Auxiliary power unit weight
W_{bc}	Baggage/Cargo handling equipment weight
W_{els}	Electrical system weight
W_e	Engine weight
W_{fc}	Flight control system weight
W_{fs}	Fuel system weight
W_{fur}	Furnishings weight
W_F	Fuel weight
$W_{g,main}$	Main gear weight
$W_{g,nose}$	Nose gear weight
W_{hydr}	Hydraulic system weight
W_{iae}	Istrumentation, Avianoic and Electronic weight
W_{ox}	Oxygen system weight
W_{pt}	Paint weight
W_{pwr}	Powerplant weight
W_p	Propulsion system weight

1
Introduction

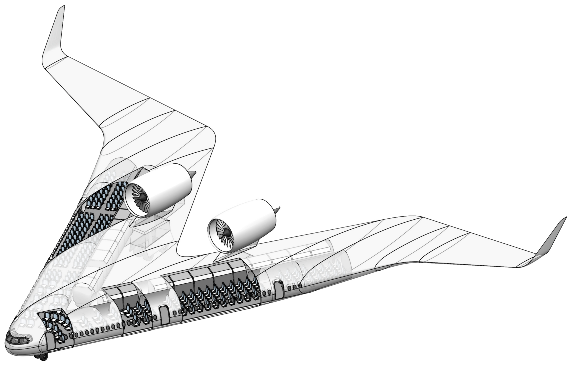
Since the 1950s, tube-and-wing (TaW) design aircraft have been the standard aircraft configuration. These aircraft, in which the near-cylindrical fuselage (the tube) and wing designs have been largely decoupled from each other, allowed for functionality-oriented design. Decades of tweaking the designs, partly enabled by the advent of powerful computers, have culminated into aircraft 50% more efficient in terms of fuel burn per passenger per kilometre, a primary figure of merit, compared to 50-year old aircraft [7, 8]. With the introduction of the Airbus A380, Airbus have inched closer to the limit of fuel efficiency these TaW designs can achieve, especially given the stricter operational and environmental constraints [9].

Martinéz-Val et al indicate three key performance indicators which dictate fuel efficiency, namely the engine's specific fuel consumption, the aerodynamic efficiency by increasing lift-to-drag ratio, and the structure efficiency [10]. The latter two of these are dependent on the aircraft design. Flying wing (FW) configurations improve the aerodynamic efficiency by eliminating fuselage-wing interference drag and structure efficiency by removing the need to carry a non-lifting body and accommodating inertial loads such that the internal bending moment is relieved. Liebeck, Bolsnunovsky as well as Martinéz-Val endorse the promising outlook of the Flying Wing (FW) concepts through aforementioned workings[8, 11, 12]. Their design studies, focussing on blended wing bodies (BWB) in particular, indicate the range of a FW can be increased to the order of 20%.

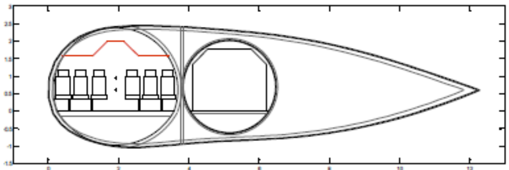
1.1. Problem Statement

The concept of FW has been applied to another configuration by Benad. In a collaboration between *Technische Universität Berlin* and the *Future Projects Office* of *Airbus GmbH* in Hamburg, Germany the Flying V concept was conceived in 2015 [3, 4] – see Figure 1.1a. Instead of the usual egg-shaped pressure cabin featured in other FW configurations, the Flying V features two highly-swept cylindrical pressure hulls constructing the *V*-shape. Around these an airfoil is constructed, as illustrated in Figure 1.1b. The cylindrical nature of the pressure cabins is beneficial to carrying the pressure differential loads during cruise flight [13]. The high sweep angle creates a highly elliptical cross-section streamwise, such that the airfoil constructed around it has a limited thickness ratio.

Benad's research comprised TaW handbook methods for estimating aerodynamic characteristics and the structural weight. From this, an aerodynamic improvement of 10% and a structure weight reduction of 2% given a mission similar to that of an Airbus A350-900 were obtained. However, the accuracy of the handbook methods applied to an unconventional configuration remains unknown.



(a) Illustration of Flying V concept.



(b) Dual-tube configuration in LE-perpendicular cut.

Figure 1.1: Flying V concept illustrations [3, 4].

1.2. Research Objective

This thesis project is positioned within a larger research. Its specific objective is

to design and analyse a structure concept for the Flying V configuration and estimate its mass characteristics.

In order to fulfil this objective, answers to the following main question and subquestions are sought:

What are the mass characteristics of a Flying V configuration?

> What are the main structure components of the Flying V primary load-carrying structure?
>
> How does the integration of aerodynamic and structure design translate into a Flying V parametrisation?
>
> In what way must a geometry model be generated?
>
> In what way can the mass of a novel configuration be estimated?
>
> What are the mass and load propagation characteristics of the Flying V configuration structure concept?

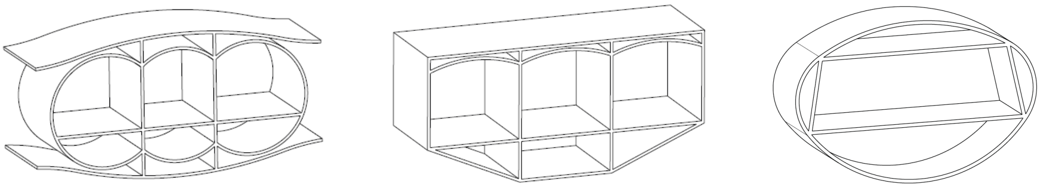
Note that the full design optimisation is excluded from the scope of this project. The coupling between design parameters, aerodynamic analysis and structure analysis is purely feed-forward. Aerodynamic design and overall design are not affected by results of structure sizing.

A structure concept is generated and presented in Chapter 2. Generation of a structure model, to be used for analysis, of this concept is discussed in detail in Chapter 3. The employed sizing procedure is outlined in Chapter 4. Implementation hereof is verified in Chapter 5. Results from the sizing procedure are presented in Chapter 6. Based on these results, recommendations for future research are elaborated upon in detail in Chapter 7. Lastly, the research is concluded in Chapter 8.

1.3. Flying Wing Fuselage Structures

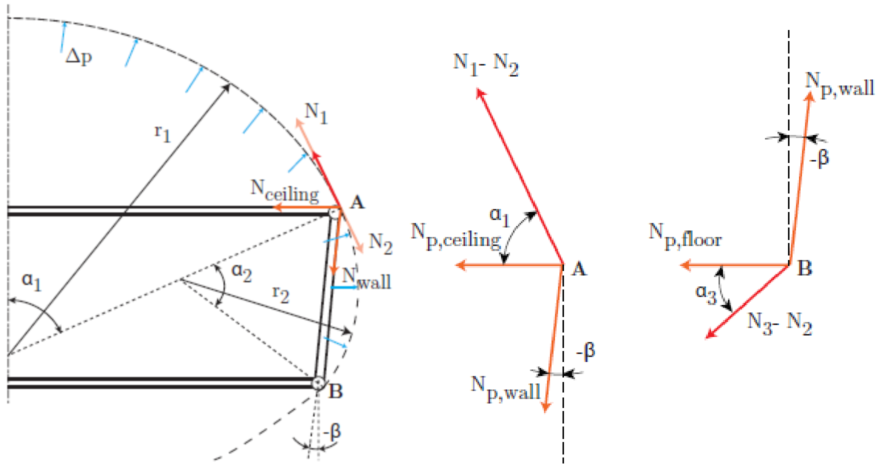
Several FW studies have already been conducted by Liebeck, Martinéz-Val, Denisov, Li as well as Mukhopadhay [8, 12, 14–16]. These authors have all considered various integrations of the pressure cabin into the outer skin. Two recurring options are identified, both illustrated in Figure 1.2:

- segregated multi-bubble-like, in which several radial cross-sections are adjoined, as seen in Figure 1.2a.



(a) Multi-bubble configuration. (b) Integrated skin shell configuration. (c) Oval fuselage configuration.

Figure 1.2: Various integral structures of wide-body cabin concepts (figures from Roelofs [5]).



(a) Residual forces in top cabin member vertex due to pressurisation. (b) Forces in top and bottom cabin member vertices.

Figure 1.3: Force balance upon pressurisation of the oval cabin concept.

– stiffened-skin, in which the skin is stiffened to carry additional pressure loads, as seen in Figure 1.2b.

There is no consensus between the authors on a preferred design philosophy. For each option, there are advantages and disadvantages. The stiffened-skin approach removes the need for an inter-shell-structure and requires only a single cut-out for the egress options. The multi-bubble structure on the other hand can be better optimised to carry pressure loads independently. Both these approaches result in a closed-cabin concept due to the need for multiple mid-cabin, vertical walls.

Additionally, the authors provide some indication on the internal structure they have modelled or assumed. As their FWs do not feature a semi-span fuselage tube but rather a pressure egg, this is of limited use. The outer wing structure is generally similar to TaW designs. Often, a third main structure element is introduced spanning to the aft root section [8]. This will not be needed in the Flying V due to absence of lifting surface here. The frame and rib positioning of Liebeck's structure in particular is of relevance, for its change in orientation of the fuselage frames in the nose region, where spars attach under high sweep angles. This bears resemblance to the Flying V ideology, in which the two fuselage tubes under high sweep also act as spar members.

Lastly, a different wide-body fuselage concept, the *oval fuselage*, has been conceived by Vos, Geuskens and Hoogreef [17–19]. This concept uses different constant-curvature arcs to carry the pressure loads. These arcs are joined tangentially. At the arc connections, structure members are introduced to carry tensile and compressive resultant forces as a result of different internal bending moments between the arcs upon pressurisation of the cabin. A detail of this is shown in Figure 1.3. In addition, the walls carry the shear forces introduced by the fuselage self-inertia and payload mass. In contrast with the vaulted and multi-bubble concepts presented before, this concept features an uninterrupted cabin layout.

A cross-section of the oval fuselage can be defined by only four parameters, namely cabin height, crown

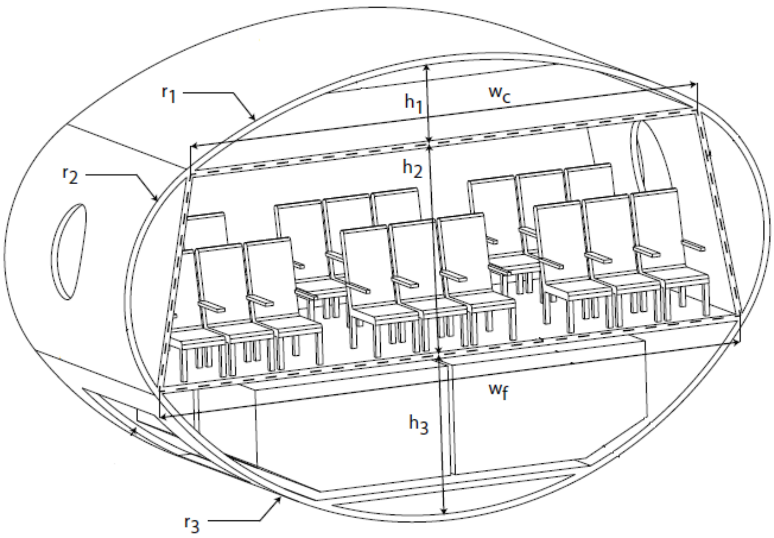


Figure 1.4: Oval fuselage cross-section definition. [6]

height (between the ceiling and top arc), keel height (between the floor and bottom arc) and cabin width, making it ideal for optimisation purposes and initial concept designs. See Figure 1.4 for a visualisation hereof.

Hoogreef, the primary researcher of the oval cabin concept, described the geometrical and force member relations in great detail. [18] By setting arc tangency conditions in the connecting nodes, the arc radii and centres of curvature can be determined. Moreover, the floor and ceiling width as well as the wall height and cant angle can be calculated. From these relations, the internal forces are determined.

A major potential threat to the oval fuselage concept is the non-uniform deformation between the various arcs, resulting in loss of form upon pressurisation. In addition, cut-outs in the cabin structure members are not investigated in the previously performed sizing research for the oval fuselage. This means that the volume between residual force members and arcs cannot yet be used for cargo. In addition, a cut-out weight penalty for ingress and egress must be considered for both the oval and cabin structure members.

1.4. Structure Weight Estimation Methods

In order to come to a feasible design solution, the airframe mass must be determined iteratively. Several methods towards this have been identified and their advantages and disadvantages have been identified by Ardema, each described below [20]. These methods can be classified as empirical, analytical and numerical estimation.

Empirical estimation requires a database for similar structures, from which relations between structure characteristics such as geometry and masses are derived at low fidelity. Roskam's method provides estimations for airframe and generic system masses [21]. Howe has expanded on this to determine a weight penalty for blended-wing body aircraft (BWB) [22].

Analytical weight estimation methods rely on appropriate simplification of the structure in order to attain a high fidelity at reasonable computational effort. This means important structure characteristics must be captured, and less important variations omitted. Should the structure employ several recurring structure configurations, an analytical tailored to each specifically must be set up. Instead of analysing the entire structure at once as suggested by Ardema, Eustace suggests breaking the structure in recurring and independent parts [23]. Analytical sizing procedures are successfully employed for wing and fuselage structures in separated fashion [5, 6, 24].

Numerical sizing has been applied in several design studies, each of which reported a major drawback was the complexity of model creation for various configuration [16, 25, 26]. By applying a knowledge-based engineering (KBE) modelling approach, the generation of a finite element model and meshing thereof can be automated [27–29]. La Rocca outlines clearly how the automated meshing of a higher-level wing primitive (HLP) can be incorporated [29]. However, such guidelines do not exist for fuselage structures specifically, nor for FW concepts in general.

1.5. Model Generation and Analysis Platform

In order to quickly conceive a parametric model for various design parameter sets and to allow coupling with various analysis tools, a knowledge-based engineering (KBE) approach is applied in the Para*Py* software package (version 1.0.7) [30]. Para*Py* is a Python-based (object-oriented) KBE environment developed from a TU Delft spin-off featuring integration of the OpenCascade 3D modelling kernel, the SALOME meshing platform and various aerodynamic and structure analysis tools.

2
Structure Concept

The Flying V structure concept is based on existing structures of other FW aircraft and existing aircraft coupled to the expected load introductions. The final primary structure concept is shown in Figure 2.1a. A wide pressure cabin indicates a deviation from Benad's dual-tube configuration. Instead, an oval fuselage configuration is chosen, as elaborated upon in Section 2.1. The choice of oval fuselage implies two transition structures must be conceived – one of the fuselage along the root plane and one in the connection between fuselage and pure wing structure. These are described in Sections 2.3 and 2.4. Positioning of the gears and engine, as well as incorporating a carrying structure for these, is described in Section 2.5. Lastly, definition of fuel tank volumes follow from the generated structure. More detail on this is provided in Section 2.6.

2.1. Oval Fuselage

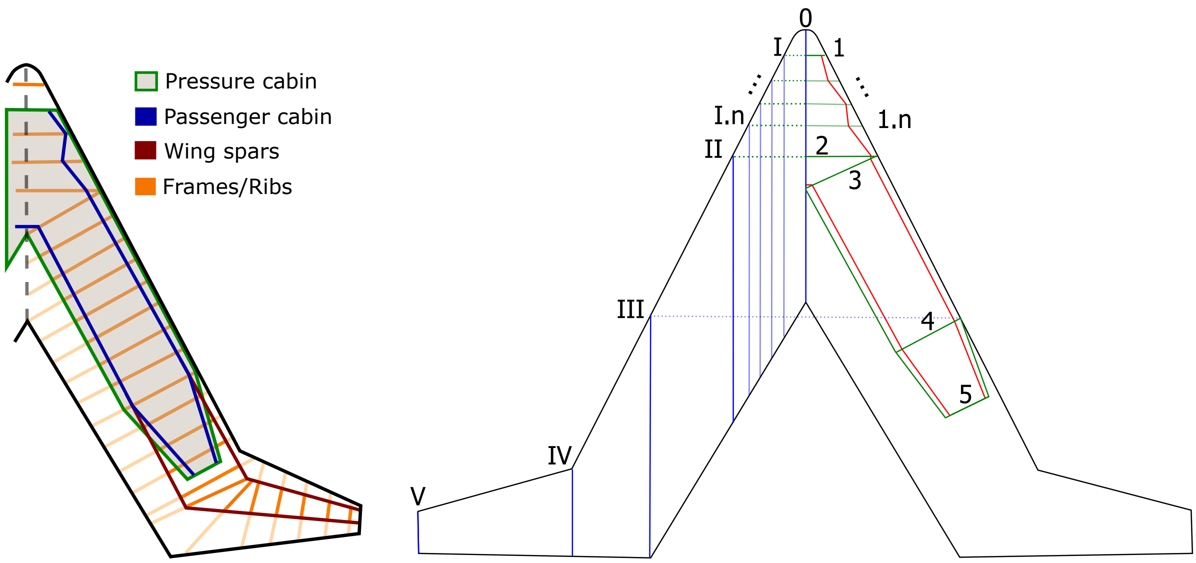
The oval fuselage concept is introduced to reinforce the design philosophy of Benad concerning the reasoning behind a high-sweep angle for a tandem circular fuselage configuration. The choice of this concept hinges on three main ideas:

1) By sweeping the fuselage cylinder, a streamwise cut results in a smaller leading edge radius and a better fit in conventional airfoils.
2) The airfoil is fitted around the fuselage perpendicular to the leading edge (LE). As such streamwise cut will feature a longer chord length but no increase in thickness, hence a lower thickness ratio is obtained. This again provides a better fit in conventional airfoils.
3) Lastly, the second cargo cylinder is fitted into the airfoil by featuring a shorter radius, as required by the cargo format selected by Benad. This fills volume in the airfoil aft of the main fuselage and enhances the used volume ratio of the airfoil, a driving factor in BWB design.

Items 1 and 2 build on the importance of tuning the airfoil design to transonic flow conditions, necessitated by the transport aircraft mission profile. These ideas are visualised in Figure 2.2a. The airfoil leading edge and thickness-to-chord ratio can still only be altered using a single fuselage parameter, providing little control over the resulting airfoil shape for the forward chord half. The oval fuselage can be tailored to airfoil requirements by changing the crown, cabin and keel heights as well as floor width – see Table 2.1.

The third idea can be clearly understood with help of Figure 2.2b. The dual tube solution is contrasted against the oval fuselage for a roughly similar airfoil. It is readily apparent that a larger volume percentage of the airfoil is utilised.

In the main fuselage, the oval fuselage cross-sections are oriented perpendicular to the leading edge. In the nose section however, the sections are oriented spanwise. By doing this, symmetry conditions

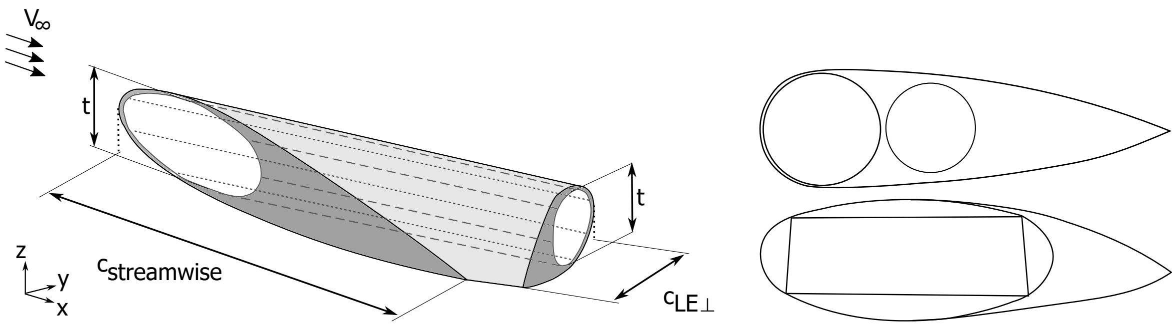


(a) Final primary structure concept. (b) Parametrised planform sections from which global design can be derived.

Figure 2.1: Structure concept and top-level parametrisation thereof.

Table 2.1: Relation between oval fuselage parameters and airfoil fitting.

Adjust	by changing
$^{t}/_{c}$	h_1, h_2 and/or h_3
leading edge radius	h_2 and/or w_f
camber line	h_1, h_2 and h_3



(a) Effect of sweeping fuselage cylinder on airfoil conformity. (b) Comparison of airfoil volume utilisation

Figure 2.2: Driving ideas behind oval fuselage concept selection (based on Benad). [3]

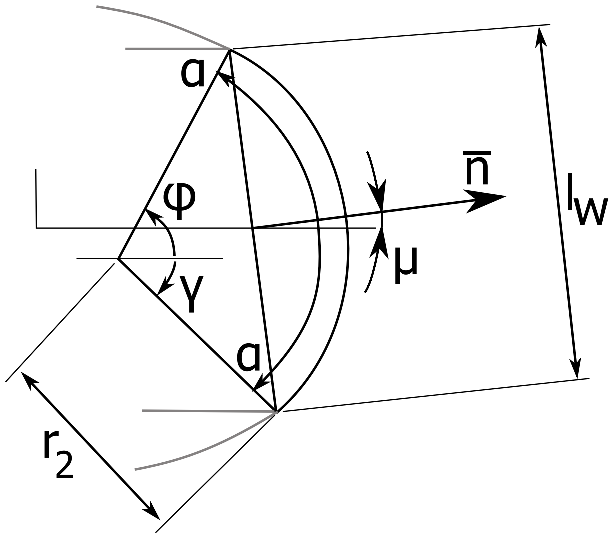


Figure 2.3: Parameters of interest for checking if node force balance is retained.

remove overly complex structures in which root-plane structure kinks are expected and in which primary load-carrying members are intersected often.

2.2. Load-carrying Structure of the Fuselage Members

As the fuselage is an integral part of the primary structure, an expected function of all components is described – this expectation can be verified by analysis of load path propagation. This is based on operational as well as manufacturing limitations. The oval arcs' constant radii ensures most efficient carrying of the pressurisation load.

In an analytical sizing by Roelofs, in which a conventional fuselage in a TaW aircraft was replaced by the oval fuselage, the walls were superimposed on the arcs [5]. This assumption corresponded well with a finite element analysis (FEA). Interchanging the function of carrying shear loads between arc and wall is allowed. To maximise enclosed area to carry torsion loading as effective as possible, the oval arcs should carry all flight loads.

On the inboard half of the oval fuselage – not defined for the nose section due to root plane symmetry – the airfoil ribs attach. For manufacturing reasons, it is beneficial to attach these to a flat surface. This eliminates the replaces the inboard arc with the inboard wall as main shear web. It is then of interest to see whether a stiffened wall can also carry the pressure loading instead of the arc. A look is shed upon the dependence of residual forces in the oval nodes cancelling out.

The resultant forces in horizontal and vertical direction for a pressure differential Δp applied to the wall with length l_w under an angle $\mu = \frac{\phi - \gamma}{2}$, as seen in Figure 2.3, is calculated using Equation 2.1. The resultant forces for the curved arc subject to the same pressure differential is given by Equation 2.2.

$$\begin{aligned} P_{f,x} &= \Delta p l_w \cos(\mu) \\ P_{f,y} &= \Delta p l_w \sin(\mu) \end{aligned} \tag{2.1}$$

$$
\begin{aligned}
P_c &= \int_{-\gamma}^{\phi} \Delta p r_2 \cos(\alpha)\,\mathrm{d}\alpha \\
&= -2\Delta p r_2 \sin(\frac{\phi+\gamma}{2}) \\
P_{c,x} &= P_c \cos(\mu) \\
P_{c,y} &= P_c \sin(\mu)
\end{aligned} \tag{2.2}
$$

Upon equating the horizontal components, a ratio $\frac{l_w}{r_2}$ is found.

$$
\frac{l_w}{r_2} = -2\sin(\frac{\phi+\gamma}{2}) \tag{2.3}
$$

From geometrical relations, it can be shown that

$$
l_w = \sqrt{(\mathrm{d}y)^2 + (\mathrm{d}x)^2} = r_2\sqrt{(\sin(\gamma)+\sin(\phi))^2 + (\cos(\gamma)-\cos(\phi))^2} \tag{2.4}
$$

Equations 2.3 and 2.4 are only equal if $\phi = -\gamma$, implying a vertical wall. Otherwise, the fuselage cross-section is out of balance and will deform under pressurisation until this balance is restored, warping the shape. The severity of this warping was not investigated. The function of carrying pressurisation loads of the oval arc components cannot be transferred to the cabin walls.

2.3. Fuselage Sweep Transition

The fuselage sweep transition is defined as the fuselage loft between *section 2* and *section 3* (see Figure 2.1). Shells are created between the oval arc and cabin member pairs. These function as a closing of the pressure cabin and continuation of the "front spar" respectively. The front edge of the outboard wall is projected normally onto the root and a shell is created in between, creating the front web. Similar root plane connections are made for the top and bottom oval shells, ceiling, floor and inboard wall. This establishes a box-like structure connection through the root plane which is considered the equivalent of a conventional wing carry-through structure.

This structure is presented in Figure 2.4. The front web features a cut-out in order to maintain an open cabin concept without structure members limiting the space utilisation. The definition of this cut-out is given in Figure 3.1.

2.4. Fuselage-to-Wing Transition

The purpose of the fuselage-to-wing transition is to connect the primary structures of the fuselage and wing sections and transfer loads between these. The front and rear wing spar must be linked to the fuselage front oval arc and rear cabin wall respectively. On top of that, the fuselage crown and keel arcs must connect to the wing box skin surfaces. This is effectuated by taking the profiles from the start of the fuselage tapered section.

From here, the wing shape transforms into that of a pure wing and as such cannot continue to accommodate the full oval fuselage. By reducing the cross-section dimensions, a fit is ensured. Over this span of the fuselage, ribs with cut-outs connect the cabin to the wing box of the transition primary structure. See Figure 2.5 for a detail. In order to prevent too short rib webs close to the fuselage tapering start, additional frames may be introduced to improve stiffness. These are not implemented as it is expected the connection between passenger cabin and outer shape, introduced by the rib web, ensures sufficient structural rigidity.

The rear wing spar is extended inboard and the fuselage "aft spar" is extended rearward until they intersect. The same is done for the forward spar solution, except that the outboard cabin wall profile is used instead of the outboard oval arc. The top and bottom shells of the wing box are constructed using the skin surface between these spars.

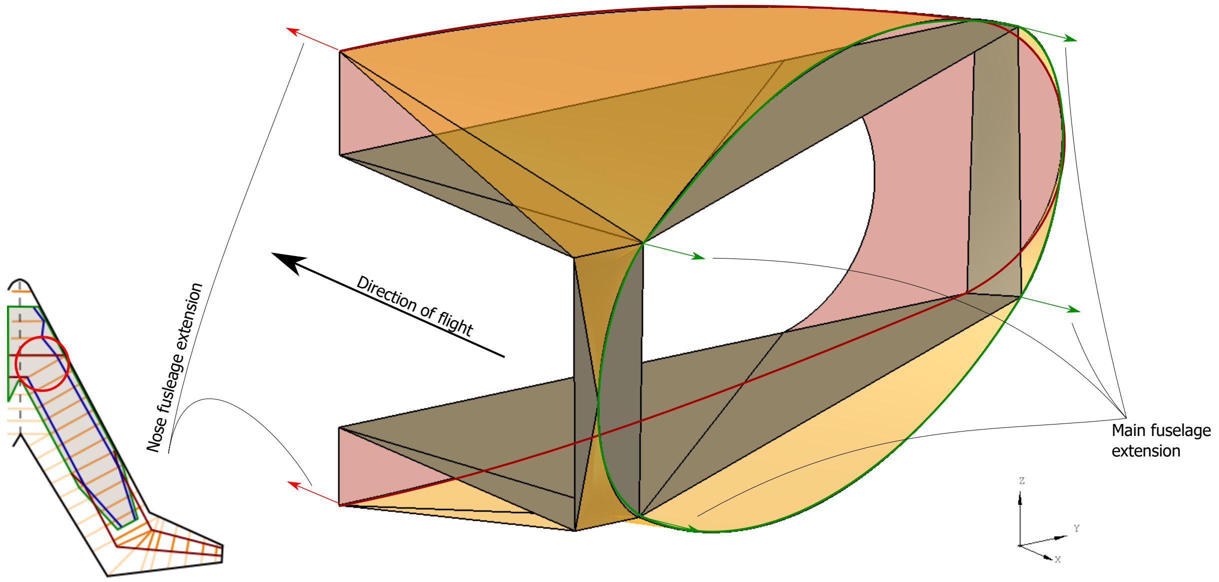


Figure 2.4: Detail of fuselage sweep transition structure.

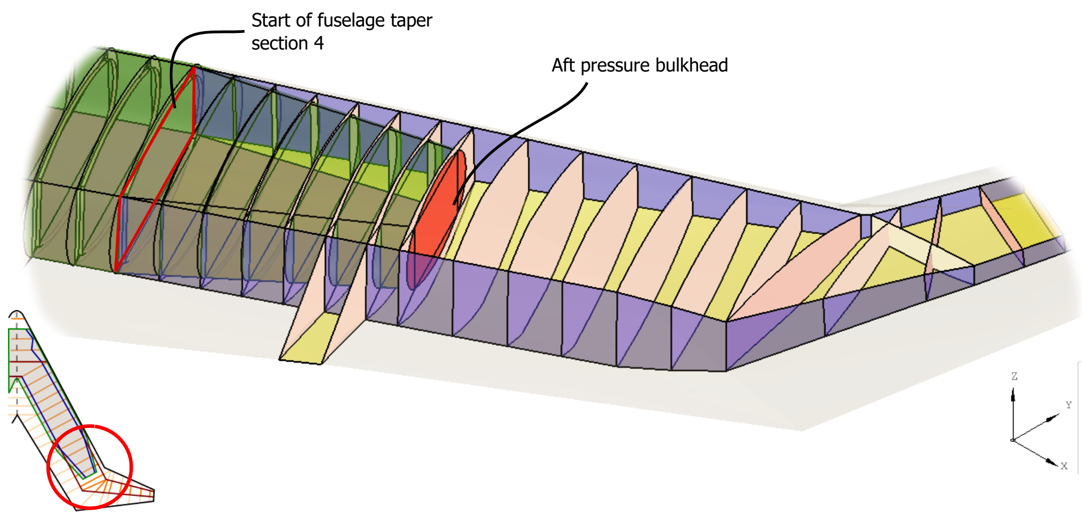


Figure 2.5: Concept fuselage-to-wing transition.

Wing rib placement is varied through the transition. Starting from the end of the pressure cabin, the ribs are position equidistant until the kink rib. In the analysed model, the initial maximum rib spacing was set to $0.5m$. The kink rib connects the forward and aft spar kinks. Outboard of the kink rib, positioning of the ribs is such that the forward and aft spars are divided into equal-span sections. Outer shape definition in the model generator created a kink in the skin – this is supported by an additional rib between the kink and wing ribs.

2.5. Engine and Gear Positioning

The engine is positioned aft of the passenger cabin and close to the fuselage aft spar. This position will be varied during the sizing process in order to change the structure centre of gravity (CoG) such that the CoG assumed by Faggiano is approximated. Uncontained engine rotor failure (UERF) is not taken into account in this study, but must be considered either in engine placement or nacelle design in future studies.

Positioning of the aft gear must be done such that existing regulations on aircraft performance are satisfied. The major considerations to be taken into account for aft gear placement are derived for take-off, landing and airport ramp scenarios:

- Upon take-off, the relatively small aft lifting surface must be able to rotate the craft. This provides a lower limit on the arm of this surface to the rotation point (the gear).
- At any attitude throughout the relevant flight envelope, the wing tips may not strike the ground. A limit can be derived for the aft-and-outboard gear positioning.
- The maximum gear track is defined for airport operations, introducing a maximum limit on the spanwise position. A minimum limit can be derived from required ground stability during sideways acceleration in turns.

The expected required angle of attack for take-off and landing is not considered to change considerably compared to conventional aircraft. The reason for this is that the entire wing features conventional profiles which provide lift over the entire chord. As such, the gear characteristics can be kept comparable to those of conventional TaW aircraft.

2.6. Fuel Tank Placement

The Flying V concept provides many options for fuel tank placements. Aside from the conventional location in the wing, as seen in TaW designs and ideal for providing bending relief, significant fuel volumes can be found in the following locations. These are also visualised in Figure 2.8.

- The transition wing box – ideal as it features a high vertical space due to proximity of the fuselage.
- Aft and inboard of the fuselage – the absence of movable wing surfaces allows for storage towards the trailing edge.
- In the fuselage keel – a low height but long span provides a long arm to the aerodynamic centre, meaning effectiveness as a trim tank is increased. However, this may interfere with certification requirements for gear-up landings.

The wing transition tank pictured features a section flush with the pressure bulkhead. The tank section flush with the fuselage is invalidated due to conflicting spatial position of the aft pressure bulkhead under pressurisation.

2.6.1. Transitional Element between Wing Transition and Fuselage

Between the wall profile at *section 4* (see Figure 2.1) and the outboard oval arc one station upstream, an extra shell is introduced to transfer loads directly to the fuselage equivalent front spar. See Figure 2.6 for a detail. In the analytical model, this was not required as loads were transferred one-to-one between the booms.

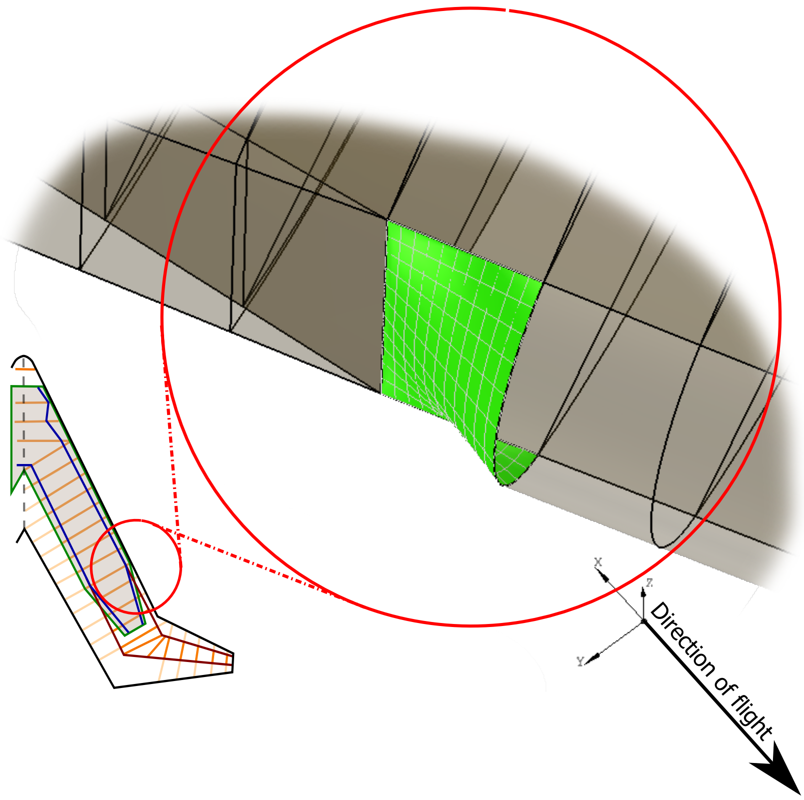


Figure 2.6: Extra element between fuselage and wing transition spars included for FEA.

subsectionRib Extension Boxes for Engine and Gear Support The loads are no longer mapped to an equivalent beam. Additional structure members must be introduced to support the engine and aft gear. These are not positioned within an existing structure member, but aft of the fuselage and transitional wing box. The procedure for generating these extra structures, the so-called rib extension boxes, is detailed in Figure B.3 in Appendix B. The closest pair of ribs are extended aft, as shown in Figure 2.7. Close to the wing root, the ribs do not end in the trailing edge but rather along the chord of the root profile. In this case, the aft edges of the box are rewired and a *PlanarFace* lid is created to be sewn in.

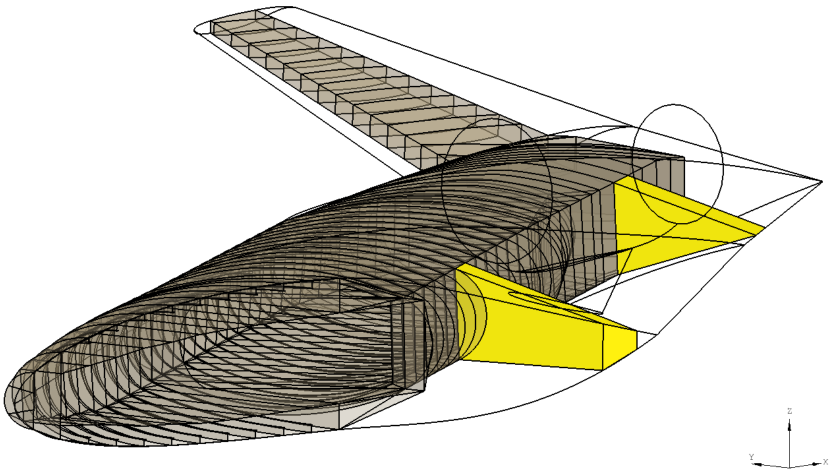


Figure 2.7: Extension boxes added for engine and aft gear support, highlighted in yellow.

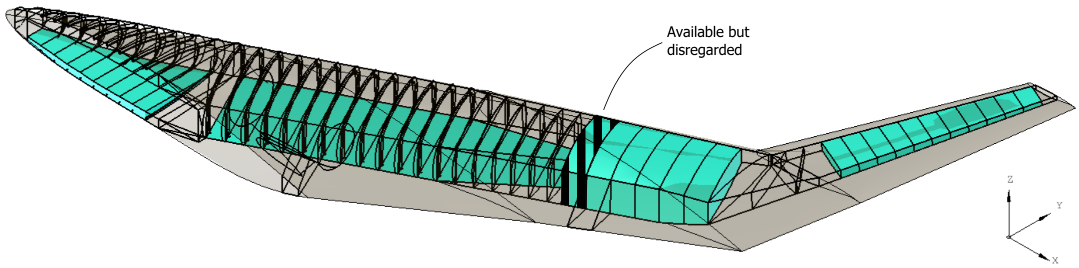


Figure 2.8: Potential fuel volumes in the Flying V planform.

3

Model Generation

A CAD-model is generated from the structure concept. a knowledge-based engineering (KBE) approach is applied in the Para*Py* software package (version 1.0.7) [30]. With this, a parametric model for various design parameter sets is quickly conceived and coupling with various analysis tools is enabled. Para*Py* is a Python-based (thus object-oriented) KBE environment developed from a TU Delft spin-off featuring integration of the OpenCascade 3D modelling kernel, the SALOME meshing platform and various aerodynamic and structure analysis tools. A parametrisation of the Flying V is set up based on aerodynamic and structure design considerations. By combining these parameters with design rules described in Section 3.3, a structure is generated.

3.1. Global Parametrisation

The Flying V global design, i.e. the wing shape, is closely tied to aerodynamic requirements and structure limitations. Several sections are identified for structure and aerodynamic parametrisation, at which varying sets of parameters must be defined. The sections are visualised in Figure 2.1b, the relevant parameters are listed in Table 3.1. For all entries where a fitting is referenced, the fitting is established using error minimisation with SciPy's SLSQP minimisation algorithm. For the parameter fittings, a lower bound is set to $0.1m$ for the nose floor widths and floor width in order to prevent non-feasible cross-sections. The outboard wing aerodynamic sections are defined using a *class-function/shape-function transformation* (CST) method [31].

The root and LE profiles are Bezier-curves with tangency conditions at the root plane and connection to *section 2*. Additional control points may be introduced by the user. Faggiano has used a difference parametrisation. As such a mismatch between aerodynamic force introductions and the new geometry occurs between $0m$ and $5m$ from the nose tip. As can be seen in Figure 4.2a in Chapter 4, the aerodynamic loads in this region are average. The effect of the non-conforming structures is therefore assumed minor.

In order to maintain the constant radii along the arcs of the Flying V, several restrictions are identified by Hoogreef and Schmidt [6, 18]:

- The cabin midpoints must lie on the same out-of-plane axis.
- The oval sections must share a lateral plane of symmetry.
- The oval section definition planes must be parallel.

These provide a strong dependence for positioning between defining oval sections with respect to each other.

Table 3.1: Description of planform sections.

Structure section	h_1	h_2	h_3	w_f	x
1-1.n	root profile	user-defined	root profile	LE profile	user-defined
2		user-defined		fit to $w_{t,3}$	touching LE
3	user-defined				touching LE
					touching LE
4	user-defined				upstream of TE kink
5	user-defined				fit into wing shape
Aerodynamic section	**Method**				x
I-I.n	Fit to streamwise cuts of cabin				*sections 1-1.n*
II	Fit to projection of *section 3*				*section 2*
III	Fit to projection of *section 3*				*section 3*
IV	CST from aerodynamic optimisation				On LE kink
V	CST from aerodynamic optimisation				On wing tip

3.2. Structure Parametrisation

The outer wing shape has been shaped by a planform parametrisation with joint aerodynamic and structure considerations. In addition to this, the internal structure is parametrised such that flexibility in the primary structure is incorporated. The number of variables is limited by using global parameters instead of one for each component. The introduced parameters are:

- Wing rib pitch.
- Fuselage frames:
  - Pitch
  - Web dimension
  - Flange width
- Wing spar locations

Wing rib pitch and fuselage frame pitch specify the maximum allowed spacing between the individual frames or ribs. It is defined for each fuselage and wing section and recalculated for each of these sections, as described in Section 3.3. In the current model, a linear rib distribution is implemented.

The fuselage frame dimensions are determined for the webs and flanges as two groups. The web dimension is varied with the outer web wire circumference; this is scaled by a user-defined factor. For this model, it is scaled to 90%. The flange width is specified at $0.1m$ for all flanges. Lastly, wing spar locations are set at a percentage of the local chord length.

3.3. Design Rules

Modelling of the Flying V was finalised using multiple design rules. The ones with most significant impact are described here.

3.3.1. Front Web Cut-Out Width

In order to maintain the open cabin concept, a cut-out is introduced. The geometry of this cut-out is dictated by the cut-out width w_{cutout}, ceiling width w_c, floor width w_f, cabin height h_2 and side arc radius r_2, see Figure 3.1. The cut-out width w_{co} is determined by user input. The constraint in Equation

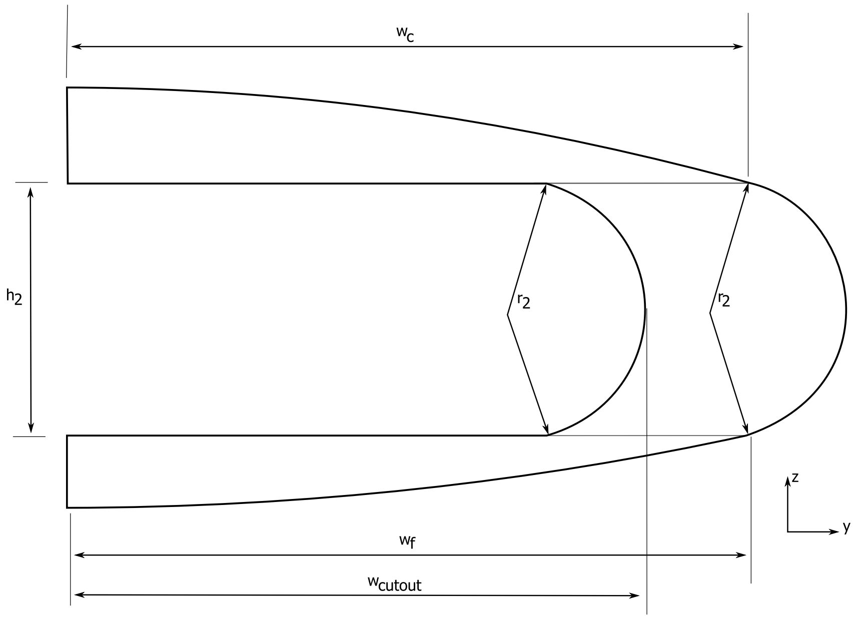


Figure 3.1: Front web geometry.

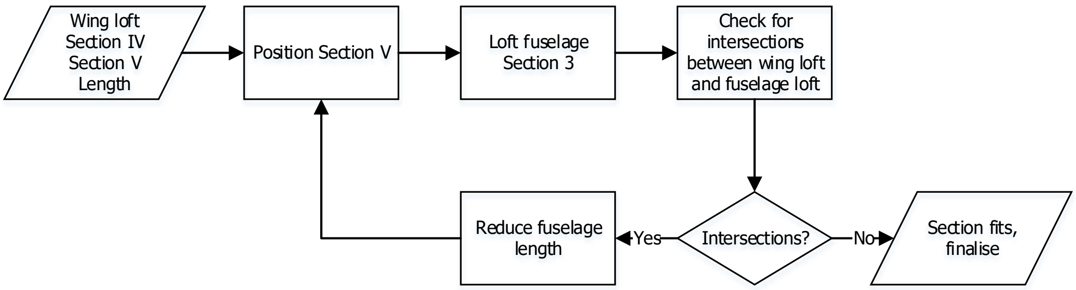


Figure 3.2: Scheme for fitting *section 5*.

3.1 applies.

$$\frac{h_2}{2} - r_2 \leq w_{cutout} \leq \frac{w_c + w_f}{2} \tag{3.1}$$

The expected bending load is around the y-axis. The top and bottom edges of the cut-out are supported by the ceiling and floor structure respectively. Therefore, it is not expected that the cut-out requires additional flanges unlike the frames.

3.3.2. Fitting of Fuselage *Section 5*

Fuselage section 5 is, as per Table 3.1, fully user defined. In order to fit it into the transition wing box, the position along the main fuselage's longitudinal axis is varied. This is done according to the scheme in Figure 3.2. The check for fit is made geometrically. The section is considered internal if no intersections exist between the outer shape and *section 5*, and the maximum z-coordinate on the section wire is smaller than that of the airfoil. The implemented reduction scheme is based on a 5% reduction of the current section length in order to reduce process time.

3.3.3. Frame and Rib Spacing

Frame and rib spacing are determined per inter-sectional part of the structure. A maximum rib pitch is defined by the user. The number of ribs is calculated from the inter-sectional span and maximum rib pitch and subsequently rounded to the next integer. From this number of ribs, a new required rib pitch is calculated.

$$s_{rib} = \frac{d_{itrsect}}{n_{ribs}} \text{ with } n_{ribs} = ceil\left(\frac{d_{itrsect}}{s_{rib,max}}\right) \tag{3.2}$$

3.3.4. Transition Ribs

The transition consists of multiple, differently oriented sections. There is a span in which the tapered fuselage is contained – rib position is defined by fuselage frame spacing. Next comes the span to the kink rib. The kink rib is a rib oriented such that it connects the two spar kinks, created by extending and intersecting the fuselage spars and wing spars with each other. The distance used to determine rib pitch between *section 5* and the kink rib is equal to the shortest of the distances between *section 5* and the projections of the front and rear spar kink onto the fuselage axis.

Multiple ribs are defined between the kink rib and *section* IV, spanning between the front and rear spar sections between the kink rib and first wing rib. User-defined fractional input determines where these ribs are placed along the spar sections. The tilt is based on the same fractions, varying linearly between the kink rib tilt and the first wing rib tilt. Note that the first wing rib is not placed along *section* IV, but rather further outboard as explained next.

3.3.5. Wing Ribs

An imaginary axis is constructed halfway between the wing spar centrelines in order to determine the positioning of the wing ribs. The inboard and outboard leading and trailing edge points are projected onto this axis such that the distance between point and projected point is minimised. The shortest distance between these inboard and outboard projected points is taken as the wing span for which the rib pitch is determined. This is done according to the same method outlined for the transition ribs. An additional tip rib is placed at the outboard streamwise section. No streamwise wing root rib is implemented, given the different orientation of the surrounding ribs.

3.4. Structure Generation

An overview of the logical flow of the structure generation process with its interdependencies is shown in Figure 3.3. Detailed generation flowcharts can be found in Appendix Section B. The *Compound* operation implies the structures are collected logically, but not connected geometrically.

The *General Geometry Generation* activity, shown in Figure 3.3, concerns itself with the hybrid model generation in which the sections as in Figure 2.1b are generated. It is important to note that in this activity group, the aerodynamic model and structure model are already decoupled. Faggiano made the design choice to use the oval fuselage cross-sections, extract sample points from the edges and fit a new curve through these.

The fuselage and wing structure activity groups, of which details are provided in Figures B.1 and B.2, are connected through the exchange of information on the rib planes and the outer shell geometry. Frame generation is detailed in Figure 3.4. After the separate structures are compounded, the fuel tanks are sized based on faces making up fuel volumes between the ribs.

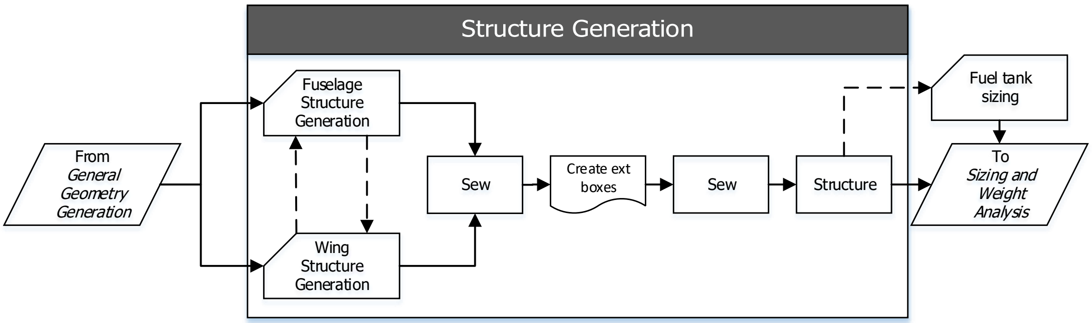


Figure 3.3: Overview of structure generation process.

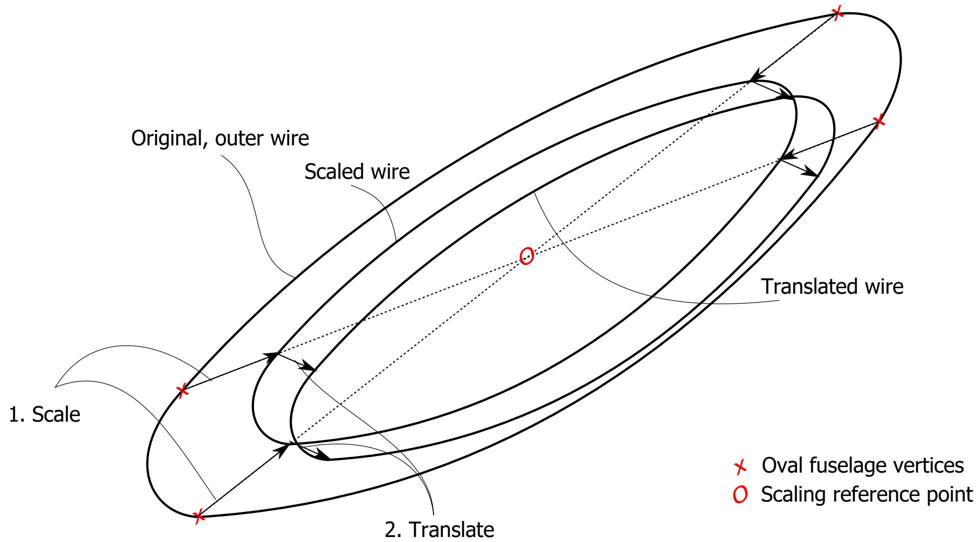


Figure 3.4: Visual representation of frame generation procedure.

4
Sizing Method

The various sizing methods have been outlined in Section 1.4 from which a preferred method, or combination thereof, must be selected. The objective is to obtain a mass estimate for the primary structure. The various methods are assessed on applicability, achievability, accuracy, flexibility and resource-intensiveness in Section 4.1. A sizing procedure is discussed in Section 4.2.

4.1. Weight Estimation Method Selection

The empirical method is not fully applicable to the Flying V configuration as no physical demonstrator exists. As noted in Section 1.4, existing relations for non-structure system masses hold true and can be used to successful extent. The method is achievable, consistent with the desired order of mass estimation accuracy and flexible for different system types as no variations are considered. The simple nature of the relations mean few computational resources are needed.

The analytical sizing can be applied in varying degrees of accuracy depending on the made assumptions and simplifications. Accuracy of the results fully depends on the made simplifications. However, flexibility is low because models are based on the assumptions tailored for each cross-section. Addition of a different cross-section requires an addition in every analysis module for that specific cross-section. Required computational power varies with the level of detail. Schmidt and Vos have shown the flexibility of analytical fuselage sizing which Roelofs has consecutively applied to the oval fuselage concept [5, 32]. Elham has developed a similar boom-sizing method for conventional wings, which can be implemented in similar fashion [24].

The numerical sizing method is often not chosen due to the time-intensive model and mesh creation for every configuration update, limiting flexibility of the method. Nevertheless, La Rocca and Van Tooren have demonstrated the required effort can be severely reduced by applying a knowledge-based engineering (KBE) approach. This enables automation of model and mesh generation, which can be modelled in Para*Py* [29]. This leaves more time for the user to implement different configurations and even allows for a fully-automated design procedure.

The Flying V structure presented in the previous chapter features multiple unique cross-sections. The fuselage frames extending into ribs differ between one another. The cabin parameters are out of the validated range of Roelofs' method. Setting up an analytical sizing for this structure approaches a FEA-based sizing procedure. Therefore, an FEA-based is set up using available software.

4.2. Sizing Procedure

The ideal sizing procedure features a feed-back loop between analysis results and design model generation, as presented in Chapter 7. However, Para*Py* functionality does not feature sufficient integration with Nastran analyses:

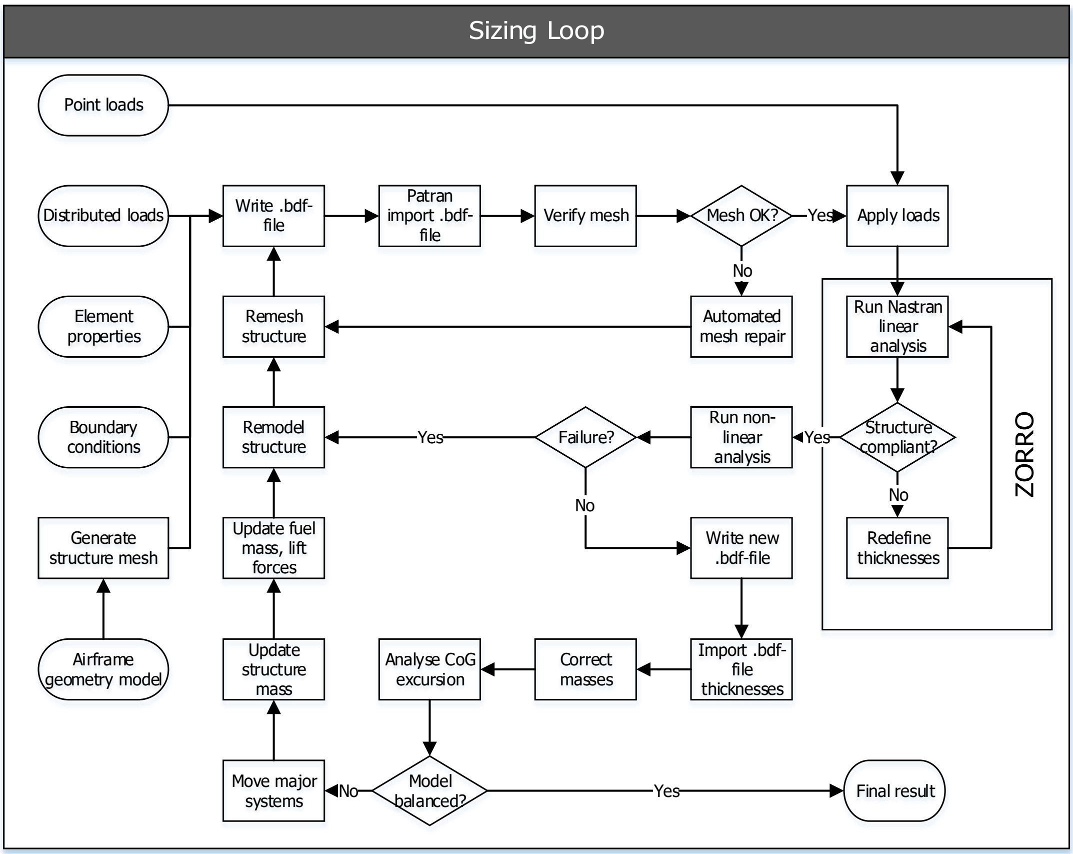


Figure 4.1: Applied single-iteration sizing procedure.

- Upon importing the .bdf-file into Patran pre-processing, material references were limited to a single material (a second material was included for testing purposes). Additionally, element shell properties were also limited to a single property object. Load cases, face groups and boundary conditions were not imported.
- Mesh IDs were changed between Para*Py* and Patran.

Because of these two limitations, *it is not possible to read back a ZORRO-generated .bdf-file of the same structure into ParaPy and match the new element thicknesses to the old elements*.

A limited sizing procedure is set up for a single iteration, using one design parameter set. This procedure is presented in Figure 7.1. Use is made of Para*Py* 's limited coupling to Nastran for structure analysis. Sizing of the primary structure and residual load minimisation is handled by ZORRO, Airbus' in-house preliminary structure sizing software.

4.2.1. Loads

The introduced loads are categorised as point loads and distributed loads. Point loads are provided to the Patran pre-processor as a list of force vectors at a spatial coordinate. ZORRO connects these forces to a selected section of the structure mesh using rigid bar elements. Distributed loads are applied directly to mesh elements.

Table 4.1: Load factor flight conditions.

Load factor	**2.5 G**	**-1 G**
Mass $[kg]$	250000	250000
Angle of attack $[°]$	10	-7
Mach	0.45	0.38

Point Loads

The point loads comprise aerodynamic loads as well as both distributed and discrete system loads. For analytical sizing, planar AVL results were used. These cannot be applied to the FEM model. The planar results contain the solutions for the upper and lower pressure distributions. Connecting these to the outer shells of the model will not correctly apply the local dynamic pressure loading.

Faggiano has provided a surface pressure distribution for cruise conditions. Obtaining the pressure force direction and projecting this onto the structure as per the procedure described in Appendix D.4 proved resource-intensive due to the large amount of geometric operations performed. Memory issues were encountered. An alternative was sought and found in Airbus' *LatticeBeta*. *LatticeBeta* is a full-potential 3D panel method, calculating a load distribution for given streamwise profiles of the outer wing shape and a Mach number.

For both load case load factors, an aerodynamic load distribution is determined. These are presented in Figure 4.2. The flight conditions for these loads are determined from a basic manoeuvre envelope and CS-25's definition of manoeuvre speed. These are listed in Table 4.1. The mass was assumed $2.5 \cdot 10^5 kg$, similar to an A350 maximum take-off mass. Gust loads were not taken into account, as they are usually not predominant for the manoeuvre speed [1]. The angle of attack α was taken such that local stall almost occurs.

Note that for both load distributions, an increase in the loads occur on both the top and bottom surface of the outboard wing trailing edge. This contradicts expectations and is likely the result of a long cusp trailing edge profile. The magnitude of the net force balance along the trailing edge is as expected. As the spatial points for these pressure forces are almost identical, application to the mesh model will be done by connecting the pressure force pairs to identical nodes, except for one or two (out of ten). Therefore, cumulative bending load effects are not expected to be observed in the final ZORRO results.

The discrete system masses estimated using Roskam's method comprise the front and aft gears, the engine and the auxiliary power unit. A position is determined by means of intersecting a box of approximate dimensions with the wing shape. The CoG is geometrically determined, implying the assumption of isotropic mass distribution throughout the system volume. The engine position is determined from the geometric CoG of the cylindrical engine model by Faggiano. Each of these systems' line of action is shown in Figure 4.3

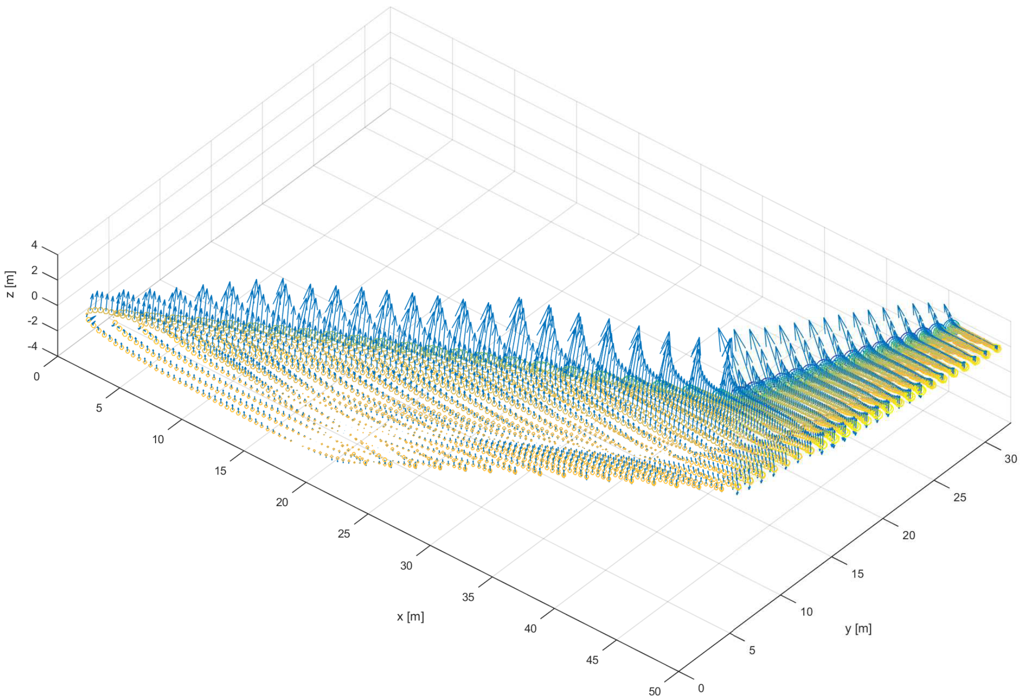
Distributed Loads

Distributed loads are a collection of the pressurisation load, the cabin floor loading and the fuel tank bottom loading. Each distributed load is then collected in a mesh face group, further discussed in Section 4.2.5.

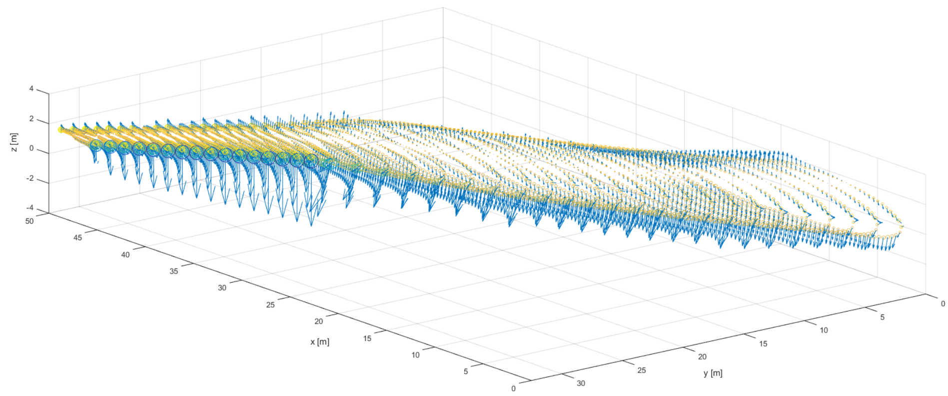
The pressurisation load is determined from the operation ceiling altitude pressure $p_{ceiling}$ and the cabin altitude pressure p_{cabin}. Values for these are provided in Table 4.2. The cabin floor loading is set at $21.5^{kN}/_{m^2}$, based on an empirical value from Niu [33]. The fuel loading varies with the volume of each tank section (which is determined per fuel tank) and the fuel density. Fuel density is assumed at $\rho_{fuel} = 800^{kg}/_{m^3}$.

Miscellaneous System Masses

Several system masses are distributed over part or all of the FEM grid elements. This is done by the division in Table 4.3. The *Fuselage*-component in this table comprises passenger cabin and pressure



(a) 2.5G load case.



(b) -1G load case.

Figure 4.2: Isometric view of applied aerodynamic load distribution.

Table 4.2: Design pressure altitudes.

	Altitude $[ft]$	**Pressure** Pa
ceiling	41300	17687.2
cabin	10000	68684.6
	Δp	50997.4

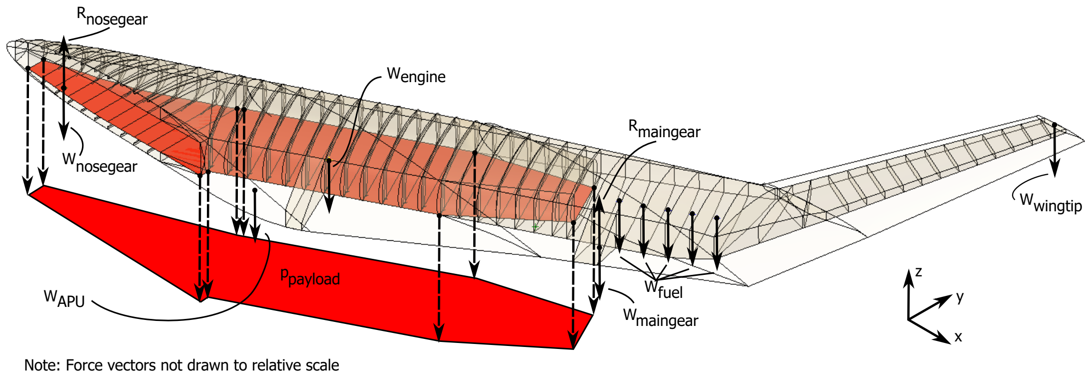


Figure 4.3: Lines of action of introduced inertia forces, indicated for positive load factor.

Table 4.3: Division of system mass distribution.

System	Full	Fuselage
Instrumentation, avionics, electronics	x	
Paint	x	
Electrical Systems		x
Onboard Baggage/Cargo Handling		x
Hydraulics		x
Oxygen system		x
Flight Control System	x	
Climate/De-icing	x	
Furnishing		x

shell elements.

4.2.2. Load Cases

Load cases are defined to obtain the correct point and distributed loads. The selected cases are listed in Table 4.4. The considered load cases are all symmetrical. This corresponds with Faggiano's disregard of asymmetric flight conditions. A reference load distribution at given Mach number is scaled to provide the correct amount of total lift force. The free-stream dynamic pressure is not taken into account as capabilities to determine this are not available in this stage of the design.

The reference load and pure pressurisation load cases 1 and 2 are ran first to check for any major flaws in the structure model. The structure is then sized according to the remaining load cases. In all te flight cases, a pressurisation differential is applied. For all these load cases, a pressurisation differential is applied. CS-25 stipulates the structure must withstand all simultaneous maximum expected loads [34, 25.571.b.5.i]. The cabin is not pressurised for take-off and braking cases.

Load case 2 introduces a 100% increase of pressurisation loads. This is stipulated by CS-25 for regular expected flight conditions [34, 25.571.b.5.ii]. The structure must be able to withstand an additional 33% of pressure differential, combined with a safety factor of 50%.

The 2.5G and -1G load cases stem from manoeuvre capability requirements in CS-25 regulations [34]. No gust load conditions are considered – discrete gust velocity changes are negligible with respect to the flight velocity, dynamic gusts are not considered in a static analysis. In a TaW aircraft analysis, the maximum wing loads are obtained by imposing maximum lift generation at minimum bending relief, i.e. maximum payload, maximum load factor and no fuel load. This is not as straightforward for a FW, especially considering that the payload is distributed over a significant section of the span. Bending relief and required lift are varied by adding and removing payload and fuel mass. The structure is sized for various modes of bending relief at both 2.5G and -1G load factors.

[1]Mass value valid for every iteration. Initial masses provided in Appendix A

Table 4.4: Identified load cases to be used for sizing the structure.[1]

No.	Name	Load factor	Mass	Gear forces?	Pressurisation
1.	Fatigue	1G	MTOM - 1/2FM	No	$1.33\Delta p$
2.	Pure Δp	0G	MTOM	No	$2\Delta p$
3.	2.5Full	2.5G	MTOM	No	$1.33\Delta p$
4.	2.5Empty	2.5G	OEM	No	$1.33\Delta p$
5.	2.5NoPL	2.5G	OEM + FM	No	$1.33\Delta p$
6.	2.5NoFuel	2.5G	OEM + PL	No	$1.33\Delta p$
7.	-1Full	-1G	MTOM	No	$1.33\Delta p$
8.	-1Empty	-1G	OEM	No	$1.33\Delta p$
9.	-1NoPax	-1G	OEM + FM	No	$1.33\Delta p$
10.	-1NoFuel	-1G	OEM + PL	No	$1.33\Delta p$
11.	Take-off	1.7	MTOM	Yes	0
12.	Braking	1.9	MTOM	Yes, additional $N_x = -0.8$	0

Table 4.5: Alcoa aluminium AL2024T6 material property [1, 2].

Property	Value	
Yield stress	380	MPa
Tensile fatigue yield stress ($5 \cdot 10^8$ cycles)	100	MPa
Young's modulus	$73 \cdot 10^9$	Pa
Poisson's ratio	0.33	
Density	800	$^{kg}/_{m^3}$
Minimum thickness	1.2	mm

No landing load case has been included. CS-25 states take-off roll and braking load cases which are considered decisive as they feature a higher mass at relatively similar load factor. The load factor for landing load cases is hard to determine as no details of the gear shock absorption system are known. Load case 12 introduces a longitudinal load factor on top of the vertical load factor. Individual gear forces for these cases are determined by the FEA through displacement boundary conditions.

4.2.3. Element Properties

Each mesh element is attributed a certain material from the materials deck. The Flying V mass estimation assumes a single-material design. The selected material is AL2024T6, a metal common in the aviation industry. Its properties are provided in Table 4.5. The isotropic nature of metal ensures mesh element orientation analyses must be performed.

Material thickness is set to minimum gage thickness of $1.2mm$ for each element, as used in the TU Delft Aircraft Initiator [35]. Nastran takes into account the self-mass and distribution in the analyses, ZORRO is responsible for determining a shell thickness distribution.

4.2.4. Boundary Conditions

In principle, the complete finite element (FE) model is in balance if there is no difference between calculated and assumed structural mass CoG. The main systems are repositioned based on checks of CoG positions during the FEM sizing procedure in order to obtain this balance.

The assumed CoG is only approximated and hence a resultant force and moment will always be present. The FEA can only be run if all forces are in balance. Therefore the following boundary conditions for forces along the x-axis, y-axis and z-axis are imposed on the following nodes:

- F_x: imposed on the nodes of the first rib which lie in the symmetry plane, as well as at the symmetry plane nodes of the carry-through aft wall surface.

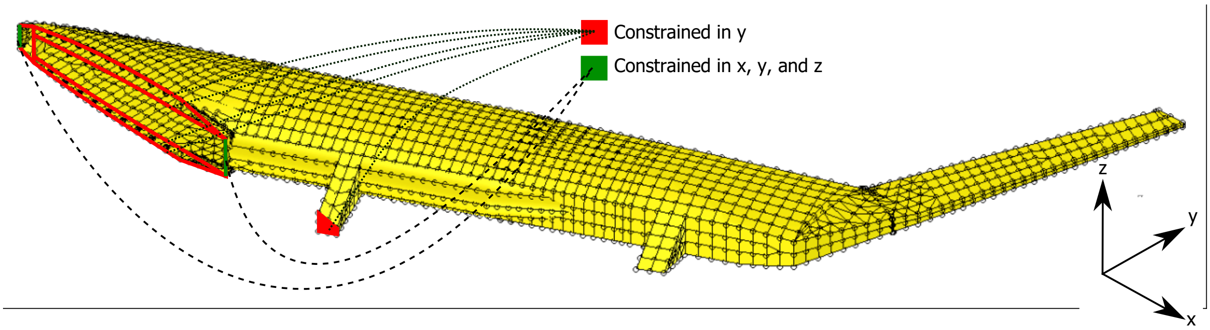


Figure 4.4: Boundary conditions and at which they are imposed on the FE model.

- F_y: imposed along the entire symmetry plane, including at the extension box. As only symmetric load cases are taken into consideration, all internal stresses and deformations are mirrored in this plane. This implicitly restrains rotation around the x-axis and z-axis.
- F_z: Imposed at the nodes as the F_x-constraint, for the same reasons.

These locations are also highlighted in Figure 4.4.

4.2.5. Meshing

Para*Py* provides a meshing module to generate surface and volume meshes using the SALOME meshing algorithm. Control hypotheses must be defined for both the node generation on edges as well as the grid generation on face topology. These are detailed in their respective Sections 4.2.5 and 4.2.5. All hypotheses are tweaked for mesh quality. For FEA meshes, this means that [36]:

- The elements are preferably quadrilateral.
- The aspect ratio of each element is $AR \leq 30$.
- All interior angles of each element are within the range of $2^{\circ} \leq \psi_{interior} \leq 175^{\circ}$.

Face Groups

Face groups allow load application to specific faces as well as selective result visualisation. Multiple face groups are identified based on distributed load application, point load application and result evaluation. All faces must be included in a group and are not limited to a single groups. The identified groups are:

- The pressure cabin consisting of the oval shells and pressure bulkheads, for application of the pressurisation loading.
- The passenger cabin floor, for application of the payload loading.
- The fuel tank floors, for application of the fuel loading.
- The wing box structure of outboard wing, transition wing and fuselage, for application of the flight loads. The fuselage wing box consists of the crown, keel and outboard arcs of the nose, transition and main section. The tapered fuselage section is excluded from the fuselage wing box.
- The wing ribs.
- The fuselage frames, consisting of webs and flanges.
- The passenger cabin non-floor faces.
- The extension boxes.

Face selection is based on the individual parts obtained during the model construction – i.e. the oval faces are selected right after their construction. This is done to eliminate the need for complex selection algorithms in the finalised structure. Selected faces are matched to faces of the final geometric object. This is based on the following hypotheses:

- The face centre points must correspond to within a margin (set at $1mm$).
- The face centre normals must correspond to within a margin (set at $1mm$). If this is not the case, the check is made again with one of the normal vectors multiplied by -1.
- The face areas must correspond to within a relative margin (set at 2%).

Edge Nodes

The aspect ratio of all mesh elements must be as close to $AR = 1$ for a high-quality mesh. Ideally, all elements have the same dimensions. These two requirements are satisfied by providing a desired node spacing. A FixedNumber hypothesis is applied to all edges in the structure, defining the number of nodes. The specific number is determined per edge by approximating the desired node spacing as close as possible. The minimum number of nodes on an edge is set to 2. For the meshing, the desired node spacing is set to 0.5 m.

Face Mesh

The SALOME *QUAD* (quadrilateral element) algorithm is applied to all the structure faces. This algorithm connects nodes on opposing faces linearly. In case the quad element has a different amounts of nodes on the opposing edges, a triangular element is inserted. The non-quadrilateral faces are selected by counting their number of edges and a *TRI* (triangular element) algorithm is applied to them.

4.2.6. ZORRO – Structure Sizer

ZORRO, Airbus' in-house structure sizer, is used to estimate the FEM-mass of the meshed structure subjected to the identified load cases. Prior to commencing a linear analysis, aerodynamic loads are scaled such that $k \cdot \Sigma F_{aero} + n \cdot \Sigma F_{inertia} = 0$. All individual mesh elements are sized to yield stress and local buckling failure criteria. The element thickness values are altered according to criteria exceedance and smoothed such that the difference compared to neighbouring elements is limited to 15%. Iteration continues until the mass values of the defined face groups converge.

Often seen in regular swept wing FEM sizing is an extremely thick aft spar. This happens as the aft spar root is closer to the aerodynamic centre of the wing. This is undesired as the front spar is more efficient in resisting bending given the taller web. For the Flying V this behaviour is not expected nor a problem for the following reasons:

- The sweep change from inboard lifting surface to pure outboard wing effective puts the outboard wing front spar closer to the aerodynamic centre. ZORRO will try to increase the front spar thickness first.
- The high sweep of the inboard lifting surface ensures that the difference is arm length from front spar root attachment (at the front web) and the rear spar root attachment (at the aft web) is relatively small given the long arm to the aerodynamic centre.
- The fuselage elements representing the spars are of equal height and thus roughly equivalent in terms of bending load resistance efficiency.

Global Buckling

By running a non-linear analysis, global buckling effects can be identified. ZORRO cannot account for this. In the ideal sizing procedure, stability of stress values given the deformed structure is assessed and taken into account in automated design. Due to the introduced bending stiffness correction described in Chapter 5, deformation values are not representative. Global buckling is therefore not considered.

4.2.7. Mass Correction

The last undiscussed blocks in the design procedure concern a mass correction, the subsequent static margin analysis and correction method for an unbalanced aircraft. The mass correction is required as there is a discrepancy between FEA-based mass estimation and the true engineered mass. Engineered mass takes into account material overlap, fastener material, and production-limited thicknesses, amongst others. The regular global thickness correction factor lies between $k = 1.7$ and $k = 2.0$ for conventional aircraft, but this may not apply to unconventional configurations. Howe reports penalty factors for BWB secondary structure with order magnitude 10%-20% of total aircraft weight instead [37]. In order to obtain specific correction factors, a detailed reference design must be created for each subcomponent for a range of load cases and dimensions. For the sake of simplicity, this method is foregone and the global correction factor is considered.

5

Verification and Validation of Sizing Procedure

Various aspects of the sizing procedure are verified:

- Implementation of SALOME meshing algorithm within Para*Py* , Section 5.1.
  - Visual inspection.
  - Patran verification.
- Application of loads, Section 5.2.

Afterwards, an attempt at result validation is described in Section 5.3.

5.1. Meshing Verification

A global overview of the meshing result is shown in Figure 5.1. The assumption of non-intersecting pressure cabin and wing transition spar can be observed as the pressure cabin mesh runs through the spar mesh without node commonality. Note that around the wing sweep change at the LE kink, there is a high node density. This is due to the many edges in the structure, resulting from wing profiles and the surface splits at the various ribs in this location.

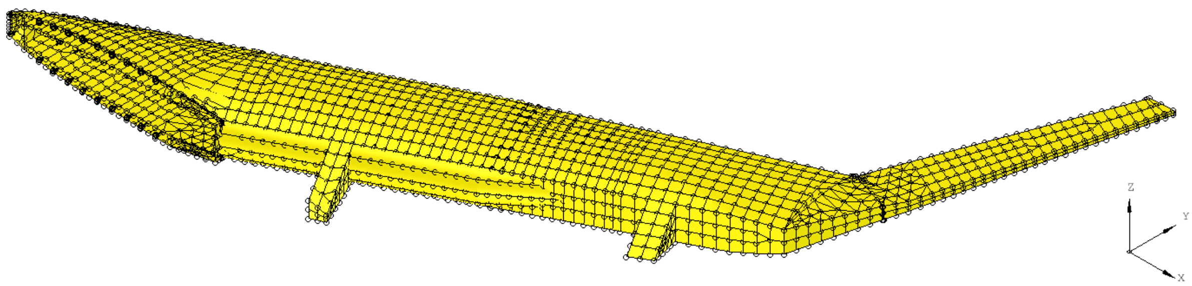
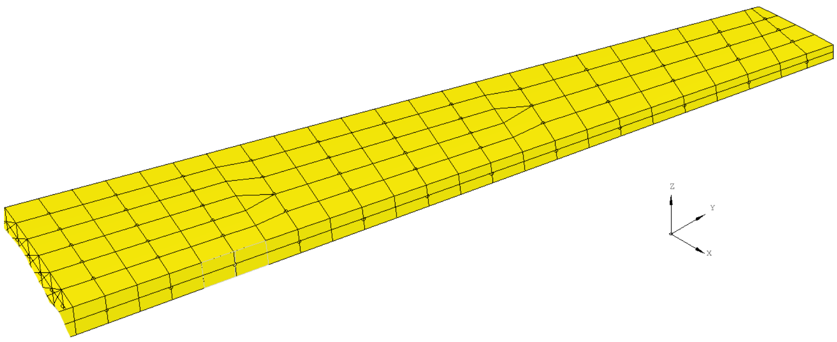


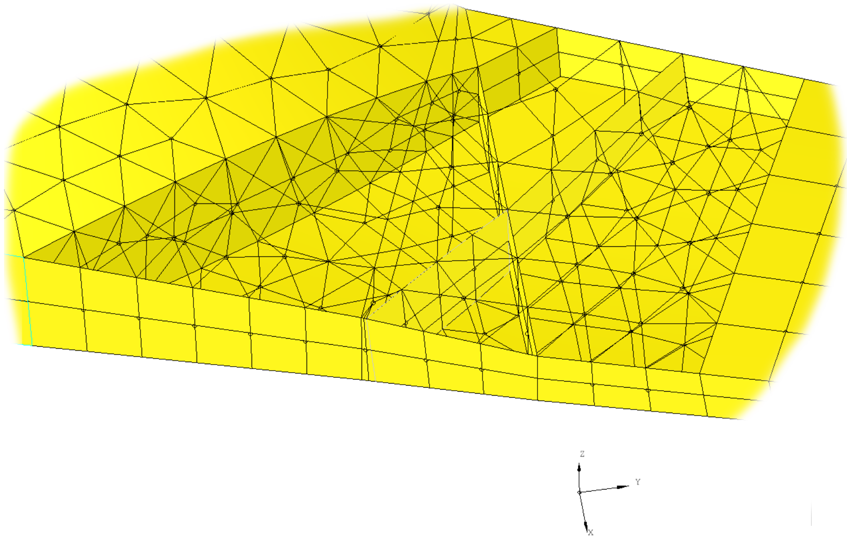
Figure 5.1: Overview of meshing result, node spacing increased to 0.75 m and max element size to 0.7 m^2 for clarity

A detail of the LE sweep change region is provided in Figure 5.2b. The top skin is cut away (the mesh for this is still visible) to provide a look at the rib meshing. The rib shell is meshed with quadrilateral elements and additional triangular as expected. The top and bottom skins feature a mix of element types dominated by triangular elements, as two major faces are built from five edges. Two particular high-density node areas are at the top middle and bottom middle in the figure, where two different ribs connect to the spars close to the wing root profile.

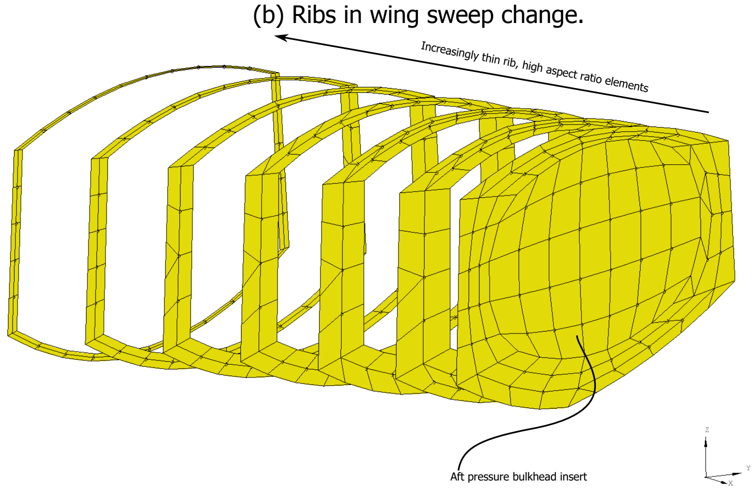
Figure 5.2a shows the effect of quadrangle preference. In case opposing edges feature a different number of nodes, triangle elements are introduced. This effect is also visible along the aft edges of the transition wing rib in Figure 5.2d, as these are longer than the opposing forward edges. The two rib extensions are geometrically different faces with only three edges and therefore feature a fully triangular mesh.



(a) Wing outer shell.

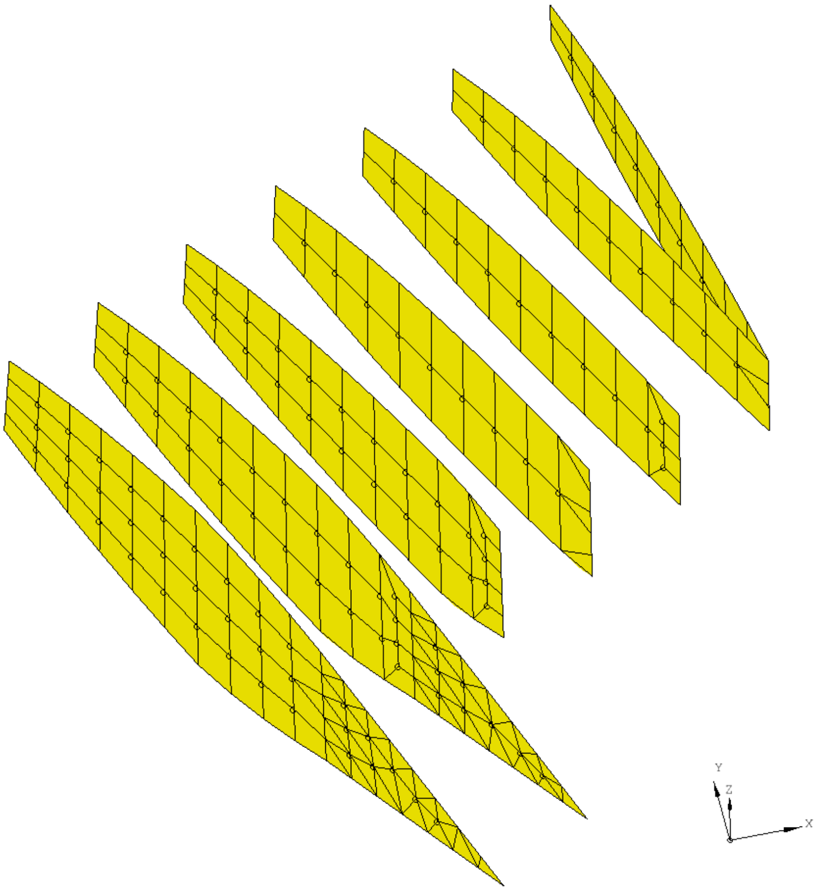


(b) Ribs in wing sweep change.

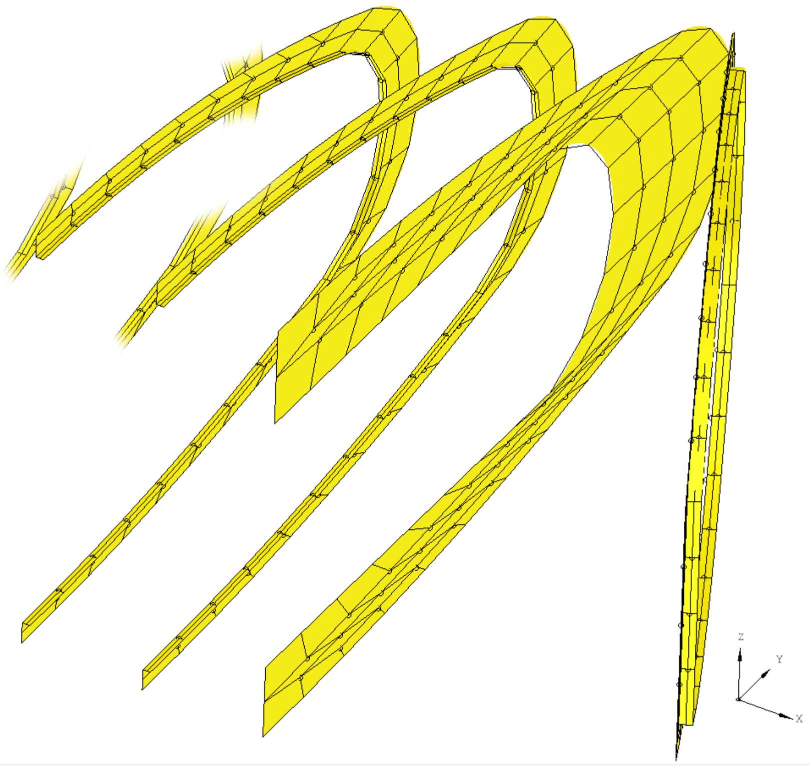


(c) Ribs in fuselage.

Figure 5.2: Resulting mesh from SALOME meshing algorithm.

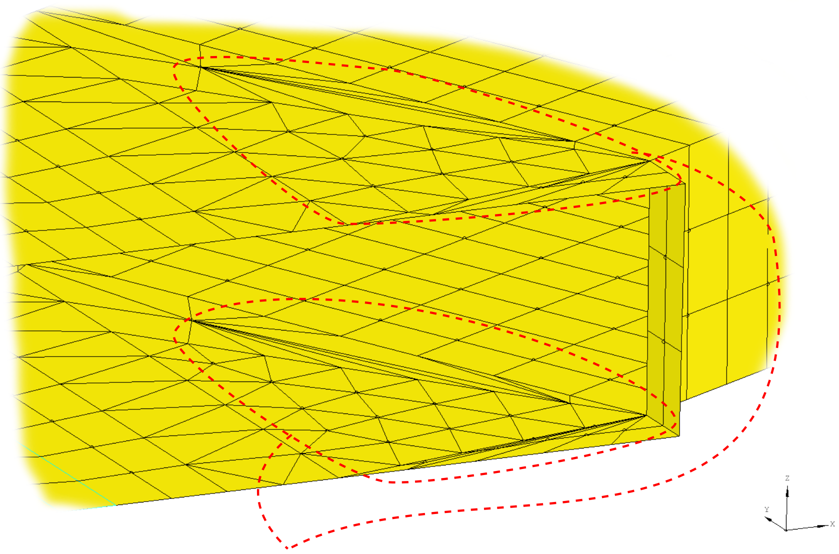


(d) Ribs in transition wing.

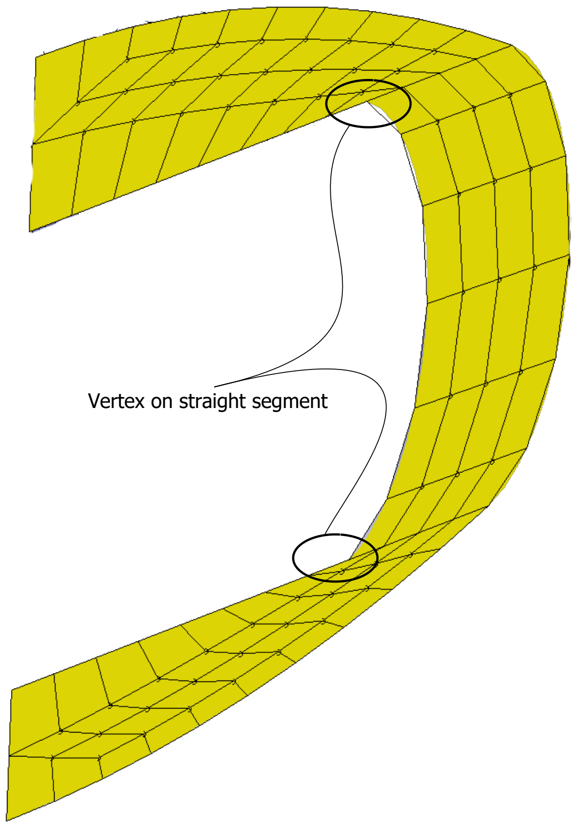


(e) Meshing result for fuselage frames.

Figure 5.2: Resulting mesh from SALOME meshing algorithm.

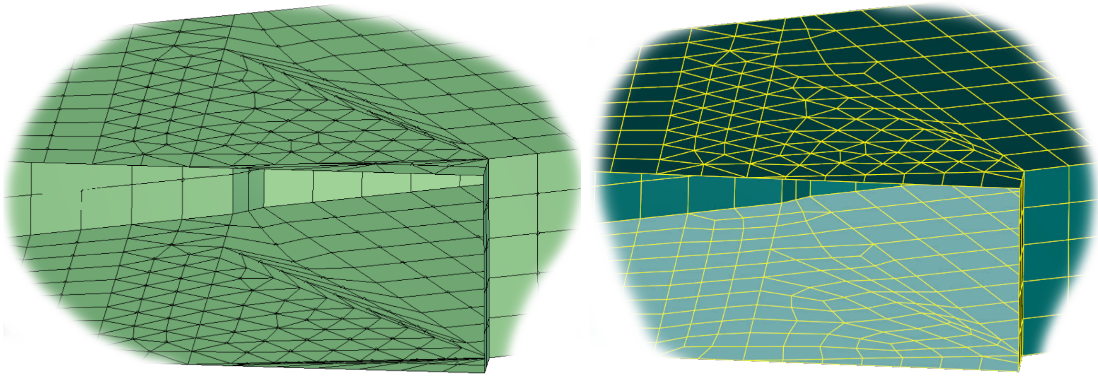


(f) Meshing result for passenger cabin, detail of fuselage transition.



(g) False vertices on front web mesh.

Figure 5.2: Resulting mesh from SALOME meshing algorithm.



(a) Old, automatically generated fuselage transition mesh in Para*Py* .

(b) Fuselage transition old (top surface, triangular) and new (bottom surface, quadrilateral) meshing in Patran.

Figure 5.3: Fuselage transition old (left) and new (right, part-finished)

The limitations of the *FixedNumber* hypothesis show in the rib webs and frame flanges seen in Figures 5.2c and 5.2e respectively. For increasingly short edges, the node spacing becomes increasingly narrow. The cause is a relatively narrow node spacing, which can only be solved by reducing global node spacing. This generates a high number of nodes and degree of fidelity, which is undesired as the resulting mesh fidelity supersedes the accuracy of the model and applied loads.

The passenger cabin transition faces show an odd mesh generation result. Instead of simply connecting opposing nodes and connecting a few triangular elements as was the expectation, it generates many skewed quadrilateral elements converging into a single vertex. This is likely caused by common point inequivalence, in which common points are not considered geometrically similar. However, the meshing algorithm does not return warning or error messages to indicate this.

Patran's mesh verification indicated multiple errors and warnings for the mesh. A fatal error occurred on the front web mesh. As indicated in Figure 5.2g, vertices exist on a straight edge, resulting in an interior angle $\geq 180^{\circ}$. This is remedied by splitting the element over the short diagonal.

Mesh verification also indicates skewed elements with high taper ratio exist in both the fuselage sweep transition and wing sweep transition. As the skewness is limited, these are not expected to introduce large errors. Elements with solely a large aspect ratio (≥ 3) also exist in the tapered fuselage rib faces. As skewness is limited, no large errors are expected. Moreover, Patran verification only marks these as warnings instead of critical errors.

The faulty mesh elements are regenerated manually. Simultaneously, the triangular meshes on the fuselage transition faces and wing top skin at the sweep change are remeshed using Patran-native QUAD algorithms (the *IsoMesh* and *Paver* algorithms). The result of this for the cabin structure is shown in Figure 5.3. The wing bottom skin is not remeshed. This will only be loaded in tension, as explained in Section 5.2. Substitution of original elements with old elements further disables an iterative design and sizing procedure.

5.2. Load Application Verification

Upon application of distributed and point loads, the thin shell elements show extreme deformations. This is due to a lack of stiffeners, as stiffeners are not present in the thin-shell model. Several remedying solutions are identified and described in Appendix C.

The issue of deformation is solved by applying a bending stiffness correction to the mesh elements within Patran. By applying this correction, quantitative results on deformation due to bending is invalidated. Local buckling can still be sized for qualitatively by ZORRO, a global buckling analysis will not produce meaningful results. Quantitative results are no longer aspired.

Table 5.1: Partial selection of load cases used to size the structure.

No.	Name	Load factor	Mass	Gear reaction forces?	Pressurisation
1.	Fatigue	1G	MTOM - ½FM	No	$1.33\Delta p$
2.	Pure Δp	0G	MTOM	No	$2\Delta p$
3.	2.5Full	2.5G	MTOM	No	$1.33\Delta p$
4.	2.5Empty	2.5G	OEM	No	$1.33\Delta p$
11.	Take-off	1.7	MTOM	Yes	0
12.	Braking	1.9	MTOM	Yes, also $N_x = -0.8$	0

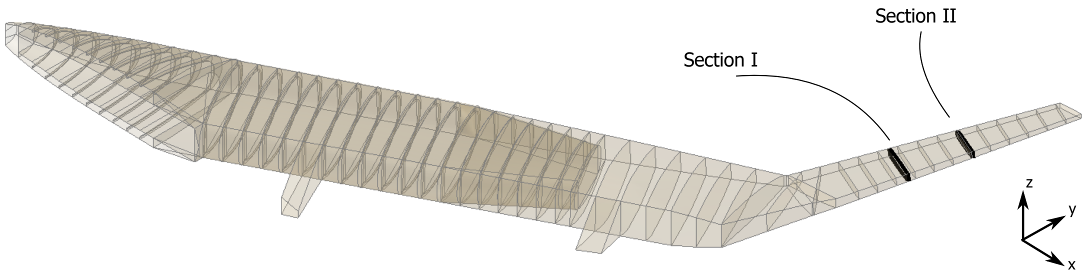


Figure 5.4: Sections for which thickness is manually determined.

Moreover, this correction is only applicable to quadrilateral elements. The mesh of the wing skins at the sweep change feature many triangular elements. The top and bottom skins are expected to be loaded under compression due to bending for +G and -G load cases respectively. Bending stiffness is required to resist buckling under compression. The model is remeshed manually to introduce quadrilateral elements on the the top skin. Negative G flight load cases are omitted as no additional qualitative insight in load propagation is obtained from these. Remaining load cases are provided in Table 5.1.

5.3. Result Validation

Validation of the sizing procedure result is preferably done in a direct manner using a reference model. A TaW with similar dimensions as an Airbus A350 is modelled. A description of this model is provided in Appendix E. Similar modelling and sizing assumptions are applied:

- Mission profile:
    - Payload.
    - Range.
    - Cruise conditions.
- Absence of stiffeners.
- Bending stiffness correction.
- Load case selection.

Due to the arisen issues, described in the previous sections, it is not possible to perform a Nastran analysis for the reference model within the selected time frame. Instead, the sizing result is validated by a manual analysis. These sections, shown in Figure 5.4, are chosen due to relative simplicity of the analytical relations sizing the structure and the common nature of the structure.

These sections are partitioned into four elements, also shown in Figure 5.5:

- Front spar,
- Rear spar,
- Top skin,

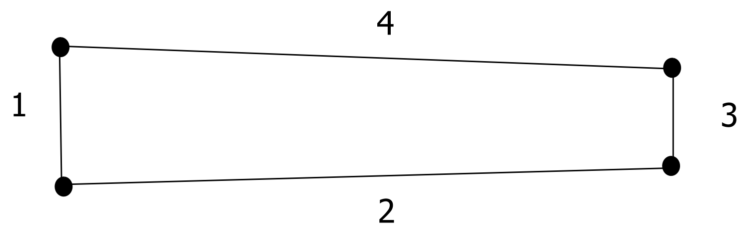


Figure 5.5: Detail of cross-section simplification and numbering convention.

Table 5.2: Minimisation settings.

Parameter	**Input**
Max Iterations	3000
Tolerance	0.001 mm
Constraint tolerance	10 MPa
Minimum change	0.025 mm
Maximum change	0.5 mm

- Bottom skin.

For each section, the elements' required thickness value for resisting the internal loads is determined. This is done using Megson's boom method [38]. For this, several pieces are collected and coded:

- The inertia and aerodynamic loads.
- The wing section geometry.
- The sizing scheme.

ZORRO's output .bdf-file includes all load case forces and linked to grid point. These are extracted and projected on an axis running through the wing structure. The force and moment introductions for the 2.5G manoeuvre are accumulated to obtain six internal loads, namely shear forces along three axes, as well as bending and torsional moments around three axes. By interpolating from the internal load distribution, values are obtained at the relevant wing sections.

Wing section geometry is obtained from the structure model in Para*Py* . For each section, a plane is created at the desired spanwise position with its normal tangential to the wing axis. The model is intersected with this plane. The vertex coordinates are extracted. A rotation transformation is applied in order to obtain a two-dimensional profile, compatible with the boom theory.

The sizing is performed using a constrained minimisation function, incorporating MATLAB's default settings. Altered settings are listed in Table. The design vector contains a thickness value for each section element. The minimisation aims to minimise the total cross-sectional area, as this corresponds to a minimisation of sectional mass. From the thickness values, the area moment of inertia is determined. The normal and shear stresses are calculated and combined into the Von Mises yield criterion. A safety factor of 1.5 is applied to the Von Mises stress value. In order to arrive at a fully stressed design, the Von Mises stress value in each element is allowed to deviate 10% from the yield stress value. An upper bound is not set, the lower bound is set at $1.2mm$ gage thickness.

The resulting thickness values for both sections are provided in Table 5.3, these are normalised with respect to the bottom skin in Table 5.4. The optimiser indicates convergence to within the tolerance of and satisfaction of the Von Mises stress constraint.

As expected, the thickness values decrease going outboard, indicative of load reduction due to a smaller accumulated internal bending moment and shear force. Compared to the numerically obtained results discussed in Chapter 6, discrepancies are observed. The top and bottom skin thickness values (elements 2 and 4) are mostly consistent with the validation result, except for the top skin of the inboard section.

Table 5.3: Validation results.

Element	**Section I**			**Section II**		
$y[m]$		$24m$			$27m$	
	Validation	ZORRO	$^{Valid}/_{ZORRO}$	Validation	ZORRO	$^{Valid}/_{ZORRO}$
$t_1[mm]$	5.44	3.7	1.47	4.54	2.2	2.06
$t_2[mm]$	4.07	3.8	1.07	2.01	2.1	0.95
$t_3[mm]$	4.46	3.2	1.39	5.57	2.1	2.65
$t_4[mm]$	18.26	5.0+	3.65-	2.19	2.7	0.81

Table 5.4: Validation results, normalised to bottom skin thickness.

Element	**Section I**			**Section II**		
$y[m]$		$24m$			$27m$	
	Validation	ZORRO	$^{Valid}/_{ZORRO}$	Validation	ZORRO	$^{Valid}/_{ZORRO}$
$t_1[mm]$	1.34	0.97	1.38	2.25	1.05	2.14
$t_2[mm]$	1.00	1.00	1.00	1.00	1.00	1.00
$t_3[mm]$	1.10	0.84	1.31	2.77	1.00	2.77
$t_4[mm]$	4.49	1.32+	3.40-	1.09	1.29	0.85

Both spars are sized thinner by ZORRO than by analytical sizing. The bending stiffness correction affects the effective area moment of inertia by increasing the effective width. This reduces the normal stresses and the principal stresses due to bending. Therefore, a thinner thickness value is determined by the numerical analysis. This is also the case when the element is subjected to a bending moment whilst simultaneously carrying shear loads. These elements will be sized thinner than realistically required.

The elements of each individual section are however sized in proportion. Therefore, the results can be used in a qualitative manner to interpret stress concentrations in individual regions in the structure. Additionally, load paths can be interpreted between stress and thickness concentrations, though load magnitudes are disregarded. From these observations, recommendations with respect to structure updates are made in Chapter 6. The results' quantitative aspect is doubtful and will will be omitted from conclusions.

6

Model Analysis Results

ZORRO is ran with the corrected mesh model and a selection of the earlier determined load cases from Table 5.1. Sizing is done for material failure and local shell buckling criteria. Element thickness values are smoothed based on a 15% maximum difference between neighbouring elements. The resulting data provides qualitative insight in the stress distribution, thickness distribution and the critical load cases per structure region. These will be presented separately in Sections 6.1 and 6.2. The returned structure mass is discussed in Section 6.3. Integration of the oval fuselage into a lifting body showed previously unobserved effects. These, together with a structure concept evaluation, are discussed in Chapter 7.

6.1. Stress Values

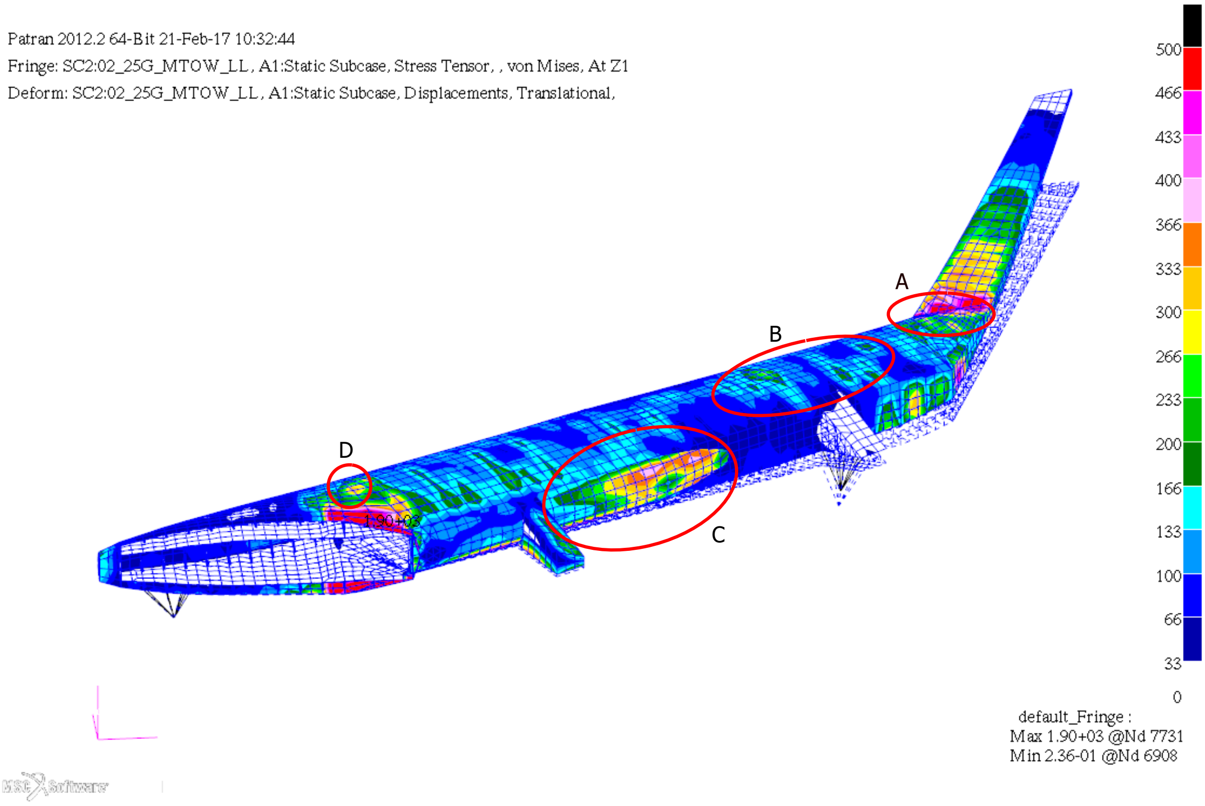
Stresses are obtained using a structure with uniform thickness. These are shown in Figures 6.1 through 6.4. Stress concentrations are consistently observed at the front rib and fuselage transition aft wall as a result of imposed boundary conditions. These concentrations, as well as the thicknesses resulting from them, will be ignored for all result interpretations.

6.1.1. Stress Results for Load Cases 3 & 4, Figure 6.1 and Figure 6.2

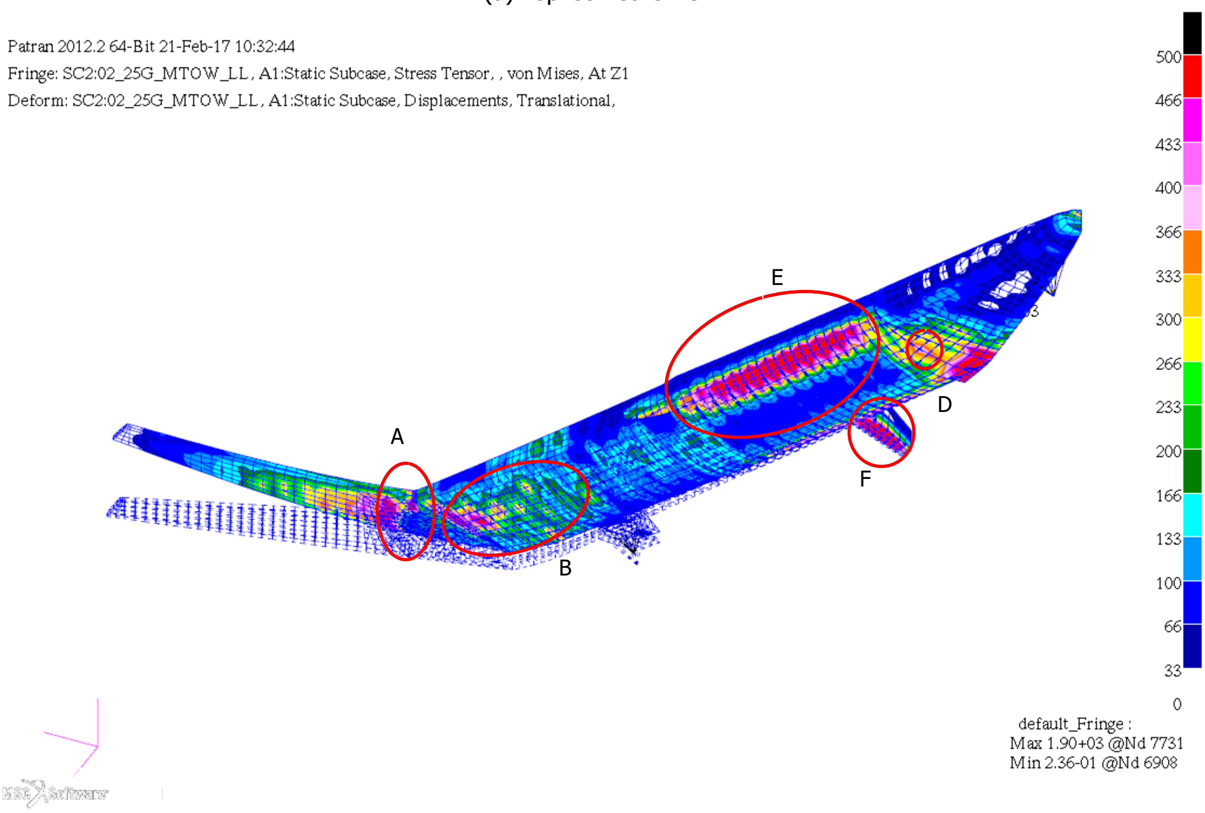
Stress concentrations are observed in region E, at the connection with the passenger cabin members, for all load cases. These increase with load factor and with absence of aerodynamic loading, as seen when comparing results between load cases 3, 4 and 12. are the effect of the payload inertia introduction into the outer shell. Absence of aerodynamic loads increase the stresses, as aerodynamic loads in this region have a negative z-component.

The wing shows conventional stress patterns for the flight load cases. Stress increases towards the root due to increase in internal bending moment. Stresses are especially pronounced in region A due to the discontinuity of the skin smoothness. A pair of stress concentrations is observed at the front spar and rear spar kinks due to the combined effect of internal bending and torsional moments over the orientation change. The front spar's concentration is more extreme as it coincides with the skin kink in region A. Between region A and the spar kinks, loads propagate mildly over the top skin but strongly over the bottom skin. Similarly, the front spar just forward of the kink shows a low stress value, whereas the aft spar shows high values in comparison. These differences in both the spar stresses and skin stresses around the kink likely have the same cause. The observation described next is used to explain this.

A stress concentration is identified in region C. A similar effect is not observed on the outboard arc opposing it. This concentration is present in both load cases 3 and 4. It is not observed in the braking and take-off load cases. This indicates it is induced by aerodynamic loading. The bottom skin on the forward extension box, region F, shows high stress values. If the concentrations in the bottom skin in the

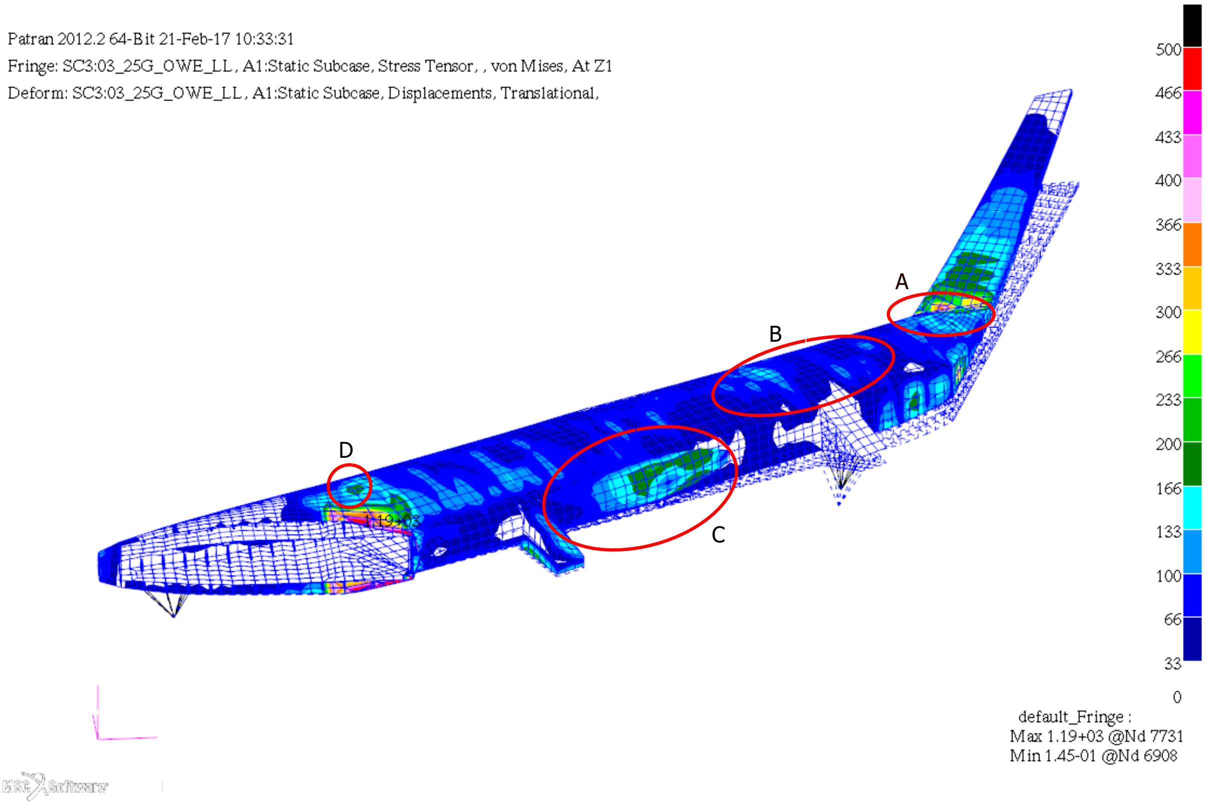


(a) Top isometric view.

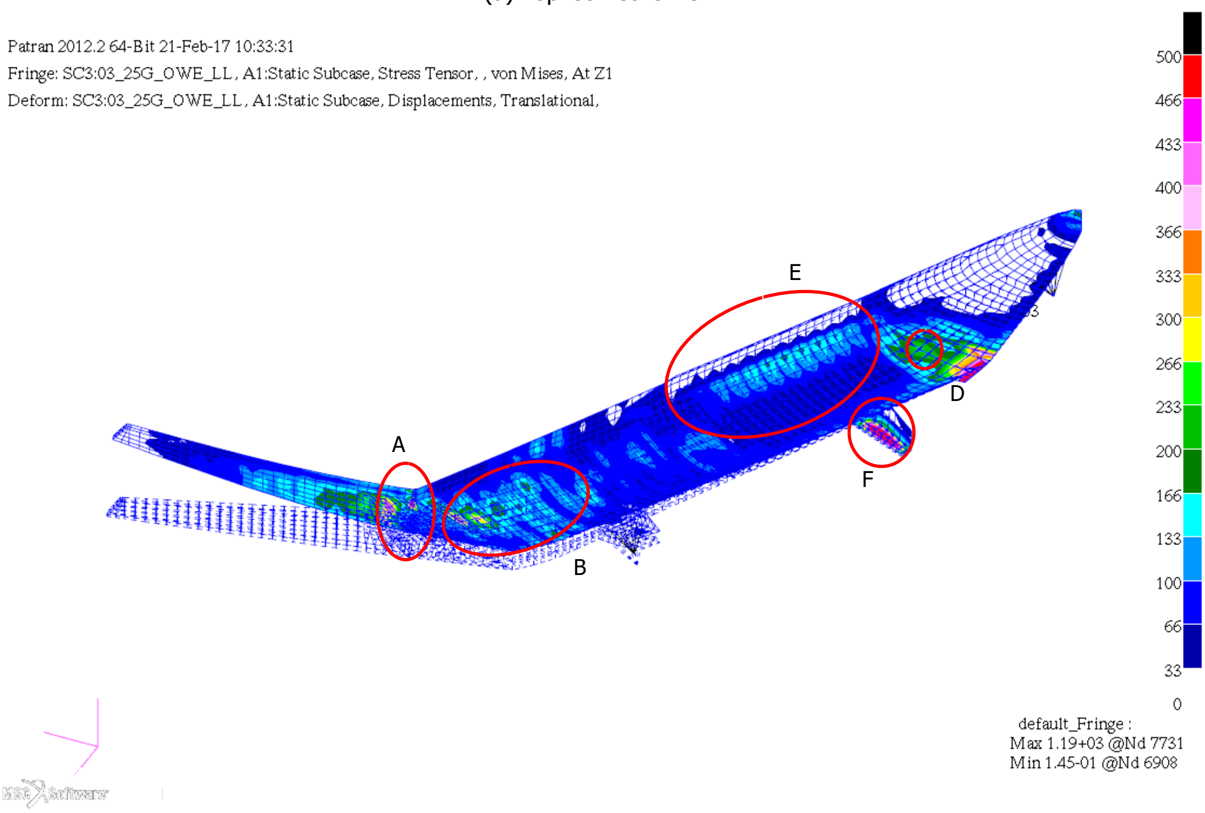


(b) Bottom isometric view.

Figure 6.1: Stresses $[MPa]$ for load case 3, 2.5G at MTOM.



(a) Top isometric view.



(b) Bottom isometric view.

Figure 6.2: Stresses $[MPa]$ for load case 4, 2.5G at OEM.

kink, the aft spar near the wing kink, the inboard arc between the extension boxes and the bottom skin of the forward extension box are connected, a fairly direct path is obtained. This load path contradicts the expectations steering the concept development.

Higher stress values are observed outboard of region A than inboard of it. This is due to the rapid increase in mass moment of inertia of the structure, as the wing box height increases significantly over a short span.

Scanning inboard and forward from the wing kink, several minor stress concentrations are observed in region B. The bottom skin stress concentrations are significantly larger due to load propagation and presence of fuel mass. Further towards the wing root however, the stress concentrations are reduced. The introduction of the tapered fuselage is identified as the cause, though the underlying principle is not yet clear and could be either of the following:

1. The pressurised nature of the fuselage cabin demand large thickness values. These attract loads away from the outer skin.
2. The presence of an additional shell increases the moment of inertia, reducing bending-induced deformation and Von Mises stress values.
3. Load alleviation by an increase in structure mass is introduced by the tapered fuselage. This cancels the internal bending moment.

A later analysis of the resultant thickness values in Section 6.2 must indicate the correct principle.

A minor stress concentration is observed in region D. It does not connect with the stresses induced by boundary conditions and is therefore not a direct effect of them. However, it full constraint conditions have pushed it outboard by artificially supporting the shell on the root plane. Without boundary conditions, the stress concentration on these shells is expected to be positioned in the root plane.

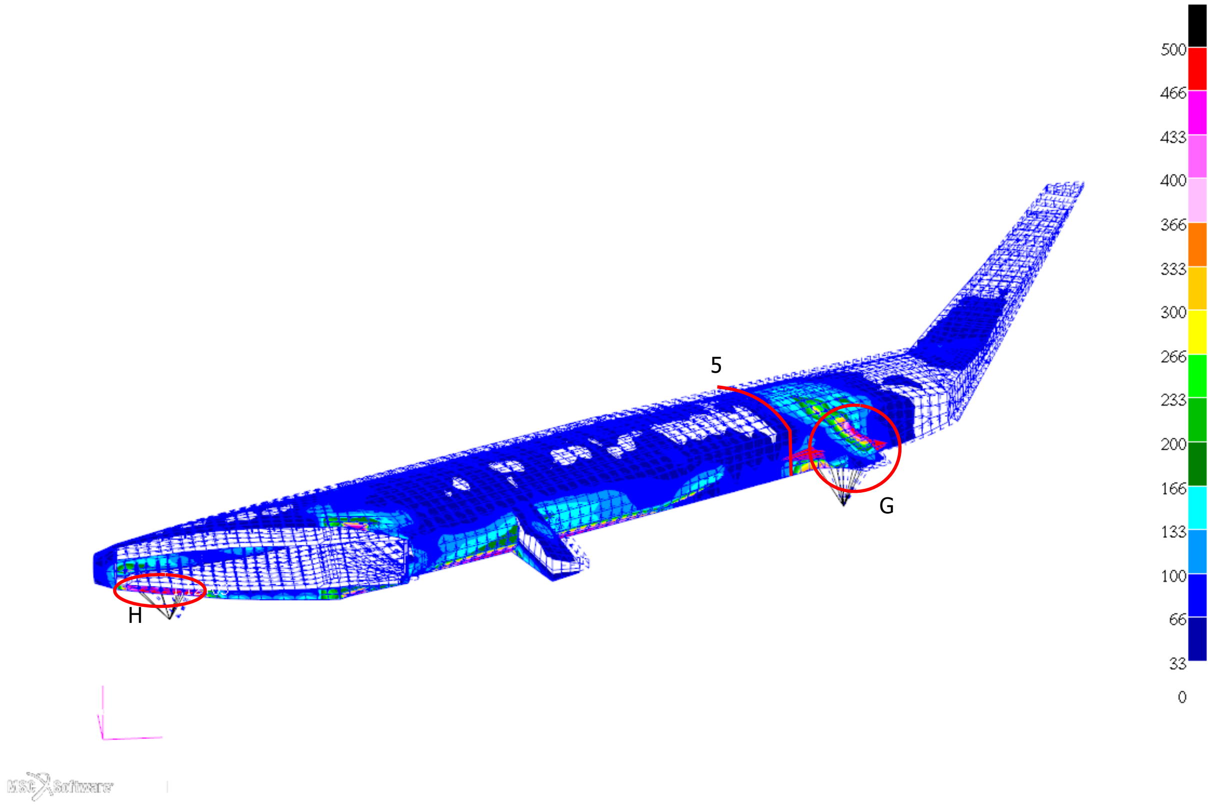
The effect of bending relief is investigated by comparing load cases 3 and 4. Bending relief presents itself through higher local stress values for an OEM scenario than for an MTOM scenario. Payload and fuel inertia effects on local stress values can also be observed. Between Figures 6.1a and 6.2a, few differences are seen. Lift-induced stress concentrations have decreased, but their location and shape remain similar. In the rear half of region B, the stress concentrations on the top skin have disappeared. This is the effect of removal of the fuel mass. Along the LE and TE of the main fuselage section, the stress concentrations introduced at the cabin floor and ceiling connections have been significantly reduced. However, they still exist as the cabin structure members carry pressurisation residual loads. Despite removal of the payload inertia, the forward extension box' bottom skin stress values are still large. This is yet another indication that a large proportion of the wing loads propagate to the symmetry plane through this box.

6.1.2. Braking Load Case, Figure 6.3

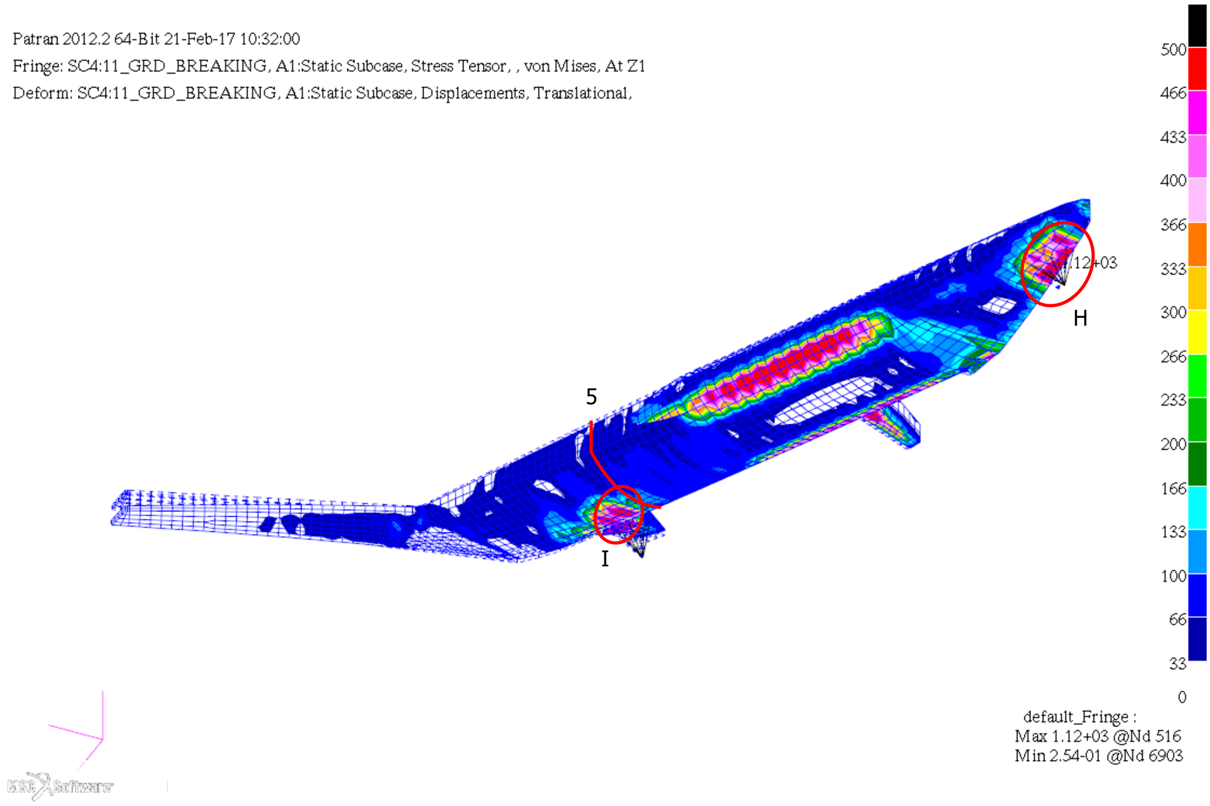
High stress values in and excessive deformation of the aft gear extension box webs are identified in region G. These webs are meshed with triangular mesh elements. The bending stiffness correction could not be applied to these. Local loads are high as they are close to the rigid bar connections of the aft gear. The vertical acceleration induces compression and combines with the forward deceleration inducing transverse loading. For both of these, a stiffness correction is required but cannot be applied.

Stress concentrations are present at the front and rear gear attachments in regions H and I respectively. The front gear concentration is slightly elongated along the longitudinal axis due to the deceleration. The aft gear concentration shows clear propagation forward, but disappears as soon as it reaches section 5. In the top skin adjacent to region G, loads propagate toward the leading edge. This is due to the combined effect of longitudinal and vertical acceleration inducing large shear stresses.

Airframe deformation is minimal. The largest deformation is observed at the forward extension box. At the root plane, is is deformed forward along the longitudinal axis by approximately $30cm$.



(a) Top isometric view.



(b) Bottom isometric view.

Figure 6.3: Stresses $[MPa]$ for load case 12, braking.

6.1.3. Take-off Load Case, Figure 6.4

Just as in the results of load case 11, a large deformation is observed in region L due to compressive loading of thin shells without a bending stiffness correction. Stress values induced in region E by the cabin floor and at the gear positions in regions H and L are bigger compared to those in the braking load case due to a higher vertical acceleration. Another effect of the high vertical acceleration is observed in the fuselage transition. In this structure region, loads propagate along the front web marked by J1 and J2 as if it were a carry-through spar.

6.2. Resultant Thickness Values

The thickness results are combined with the critical load case results. For each structure group, the results are presented and discussed. As with the stress values of Section 6.1, the concentrations of structure thickness close to the boundary conditions are ignored.

6.2.1. Outer Shells, Figure 6.5

The outer shell thickness results are provided in Figure 6.5. Critical load cases for each element are shown in 6.7. The outboard wing skin thicknesses on the top skin are as expected and increase towards the root. The wing front spar shows a similar thickness pattern. The aft spar is thin between the aft extension box and the wing tip. This is the likely result of the applied bending stiffness correction.

A large discrepancy between element thicknesses is observed in region A. The decrease in thickness occurs simultaneous with internal introduction of a second closed cross-section profile due to the tapered fuselage. A similar effect was described for the stress values in Section 6.1. For this, three potential causes were identified. Each cause is reconsidered in combination with the thickness results of the outer shells and pressure cabin (Figure 6.6).

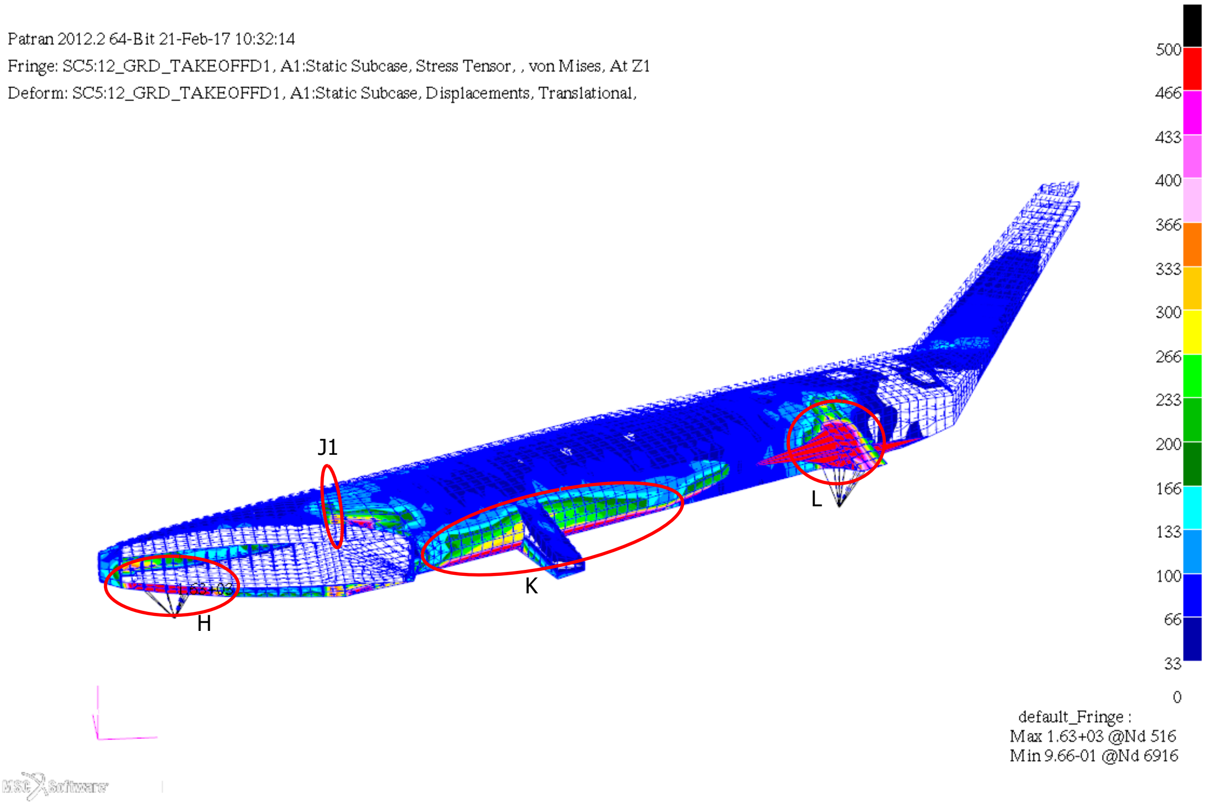
The thickness values of the pressure cabin elements aft of region A are approximately 75% of those in front of region A. The outer shell thicknesses reduce by approximately 50% between these regions. Addition of the element thickness values just aft of region A results in a value approximately 25% higher than the shell thickness before region A. The following conclusions can be drawn with respect to the potential causes:

1. The pressure shell reduces in thickness by 25% and reduces further to 50% of the thickness just ahead of region A. It does not (significantly) attract load from the outer shell.
2. The tapered fuselage cross-section is introduced close to the outer skin. The combined thickness is increased. Therefore, the moment of inertia os nearly doubled compared to just forward of region A.
3. Additional load alleviation is present by an increase in structure mass smaller than the combined thickness increase of 25%. This is because the fuselage tapers and has a smaller cabin floor surface area. The bending stress relief should be less pronounced further aft of region A. Stress values indicate the opposite.

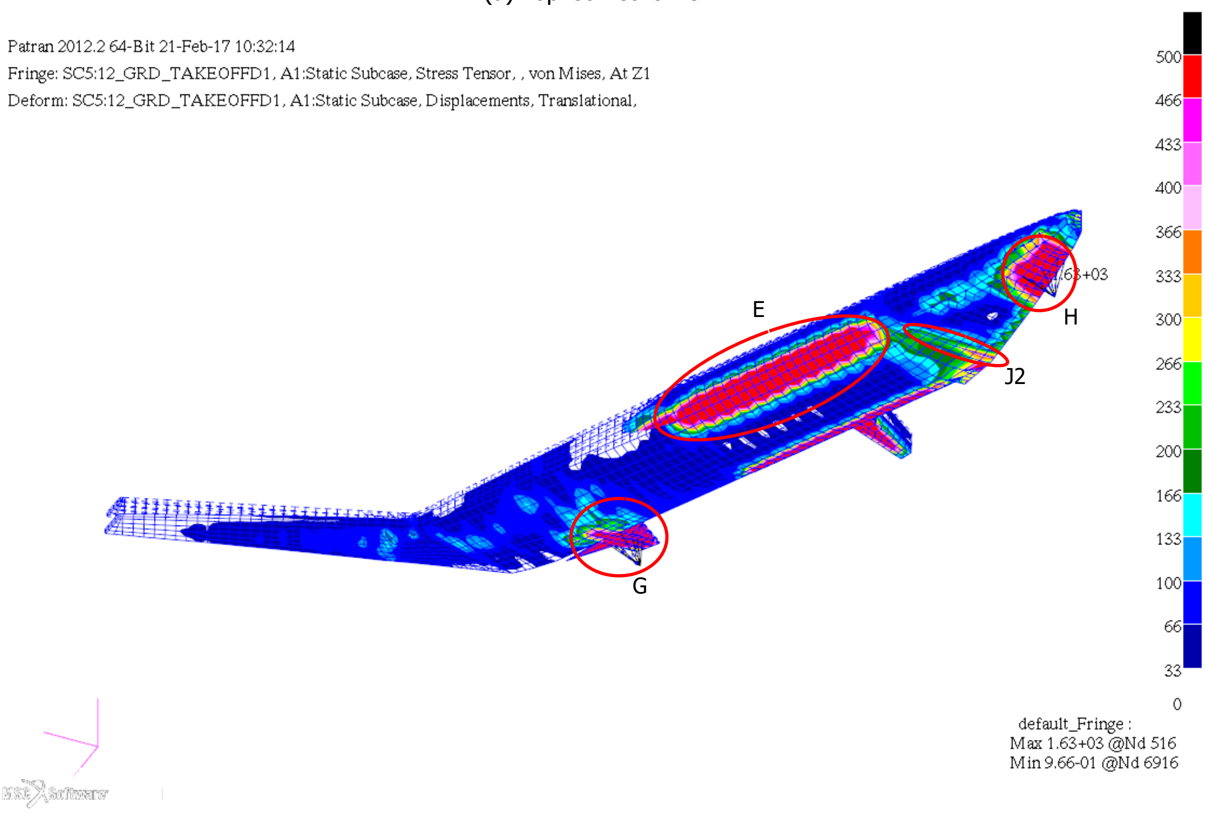
The reduced stress and skin thickness values of the outer shell just aft of region A are thus the result of increase in moment of inertia with respect to the horizontal plane.

Along the main fuselage in region B, a thickness increase is observed compared to where the residual force cabin structure members connect. For the side arcs, a decrease is observed instead. This is due to the vertical distance from the neutral axis. The increase combines with pressurisation-induced stresses.The top and bottom arc thickness values are larger than those of the sidewalls. The top skin is thicker than the bottom skin. This contradicts expectations based on the respective arc radii.

An explanation is found when studying the critical load cases. The keel shells are primarily sized by load case 1, indicating their tension state is decisive given the reduced yield stress for fatigue. The crown shells are sized for a maximum compression state from load case 3, except in the centre of the main fuselage. Inclusion of a -1G load case in future research might shown a different critical load case and thus increased thicknesses for (part of) the bottom shell.



(a) Top isometric view.



(b) Bottom isometric view.

Figure 6.4: Stresses $[MPa]$ for load case 11, take-off.

For the performed analysis, the bottom skin critical load cases are load case 11 for the majority of structure not containing the fuselage and load case 1 for the fuselage structure. This distinction is attributed to the pressurisation of the fuselage. Load case 12 is the critical load case at the aft extension box connection and directly upstream of it. This is due to forward deceleration upon braking.

The top shell critical load cases feature a combination of load case 3 for the majority of the elements, load case 1 for the main fuselage and part of the nose fuselage and load case 11 for the front half of the nose fuselage. Load case 11 is present due introduction of the gear loads. Presence of load case 3 is due to wing bending. The combined effect of payload inertia and vertical aerodynamic load (normal to the surface) in the main fuselage induce large tensile forces in the crown and keel. Therefore, the main fuselage crown shell is sized by load case 1.

The crown arc maximum thickness is smaller in the forward half of region B than in the aft half. This is not observed on the bottom shell. It is likely the cause of the wing load path propagating to the extension box. The front region therefore is loaded less.

Smoothing on the wing kink bottom skin in region C is not working correctly. ZORRO's smoothing algorithm only works for quadrilateral elements. The triangular elements in the wing kink are thus not smoothed. Instead, a small number of elements is thick to carry the majority of the bending.

The bottom skin of the nose fuselage shows two thickness concentrations. The small one correspond to load case 11 as critical load case and as such is attributed to the presence of the nose gear. Load case 1 is the critical case for the major thickness concentration. This indicates it is sized for tensile stresses at reduced yield stress. The tensile stresses increase toward the centre of the concentration due to increase in keel arc radius.

In region D, the thickness distribution is not centred around the wing box axis. Instead, it is slightly skewed. The critical load case plot indicates load case 12 is decisive directly upstream of the aft gear along the longitudinal axis.

6.2.2. Frames

The frame thickness values vary from $1.2mm$ to $9.3mm$ between the nose and main fuselage, and the tapered fuselage. The difference between these is that in the tapered fuselage, the webs are clamped between the pressure cabin shells and the wing box structure. This is an indication that the nose and main fuselage frames are not providing sufficient stiffness. They deform too much when the structure is loaded. Their web height and flange width dimensions are too small. No quantitative conclusions can be drawn from the nose and main fuselage frame thickness values.

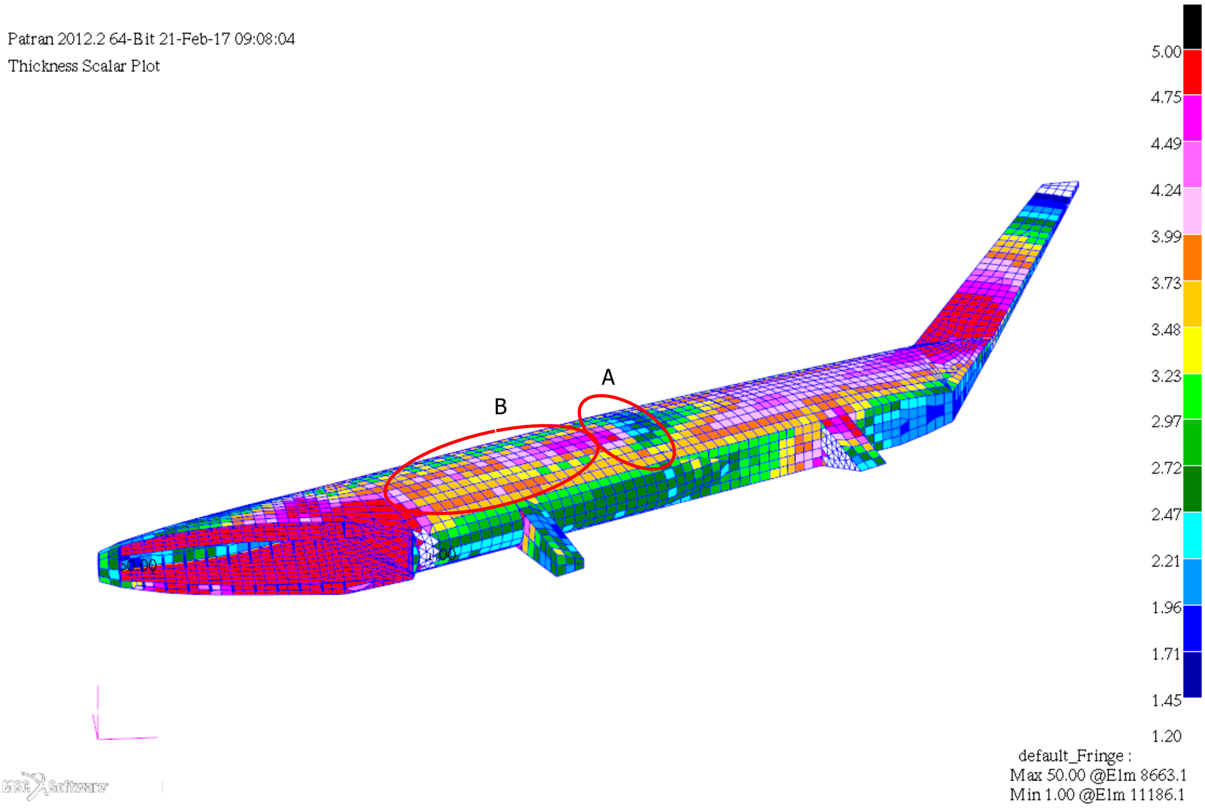
Several qualitative observations are made. There are stress concentrations at the passenger cabin structure member attachments. The concentrations are lower at floor connections than at ceiling connections. Payload inertia reduces stress magnitude.

In the tapered fuselage frames, larger thickness values are observed in streamwise direction ending at the inboard edge of the last frame, highlighted in Figure 6.8a. Referring to the critical load cases, it is clear these are *not* due to the braking deceleration introduced at the last frame. Instead, they are the effect of the load path propagating from the front spar the the rear spar.

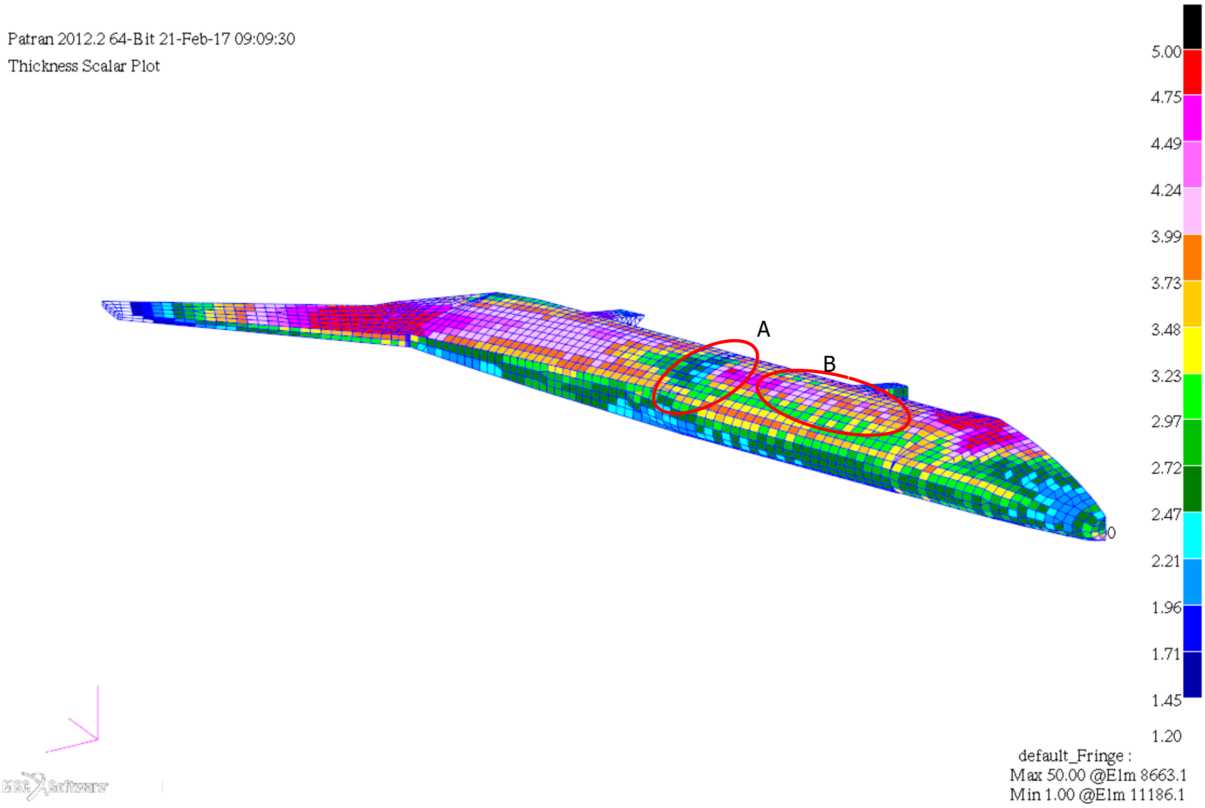
The thickness values of the extension box webs reduce toward the trailing edge. The internal shear load reduces due to increasing distance towards the aerodynamic centre.

6.2.3. Spars, Figure 6.10

The outboard wall of the nose fuselage shows a thickness concentration. Load case 11 is critical. Therefore, the concentration is the result of nose gear shear force introduction. Continuing along the wall, a low constant thickness is observed. Shear is carried by both the wall and the outboard arc. The combined thickness of these is approximately $4\ 4.5mm$ up until section 4. Given the continuous distributed payload, the shear force and thus thickness should not be constant. Aft of section 4, the spars seem to be sized as expected. Therefore, the thick and stiff floor and ceiling shells are likely the cause of the constant nature of the nose and main fuselage wall thickness.

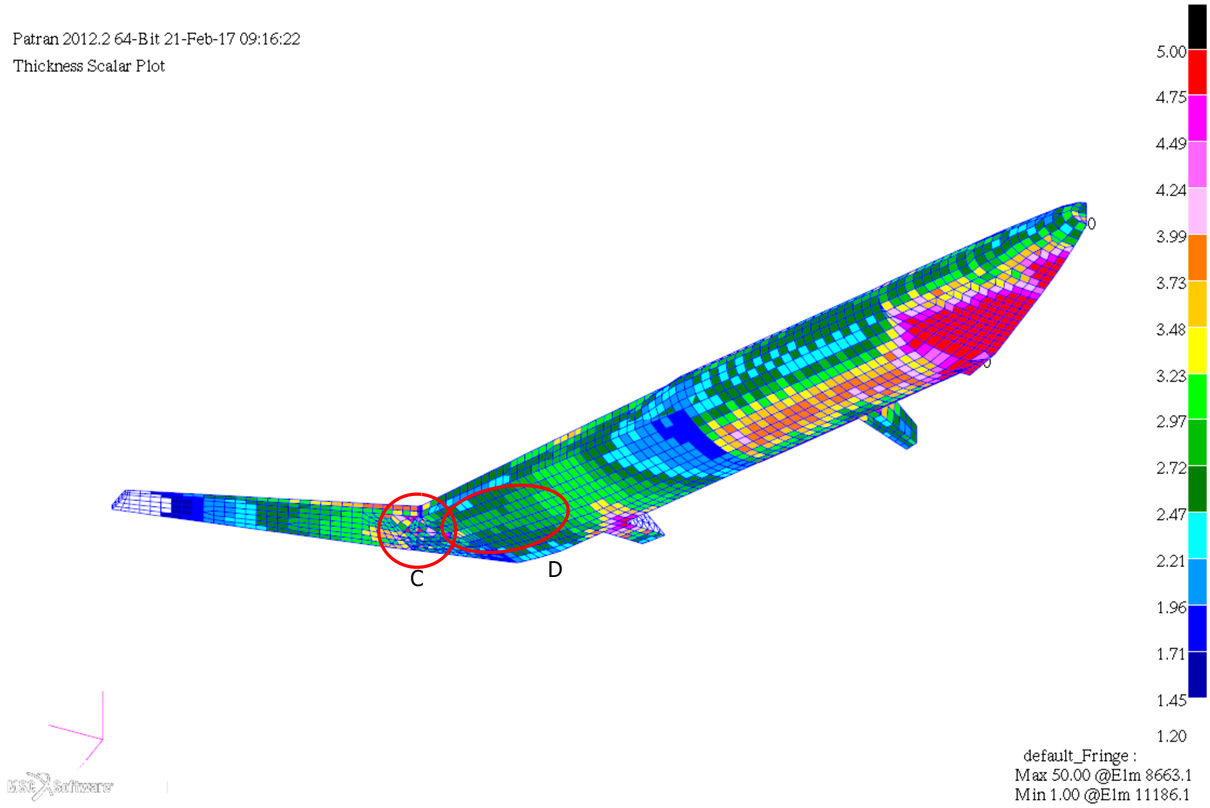


(a) Top isometric view.

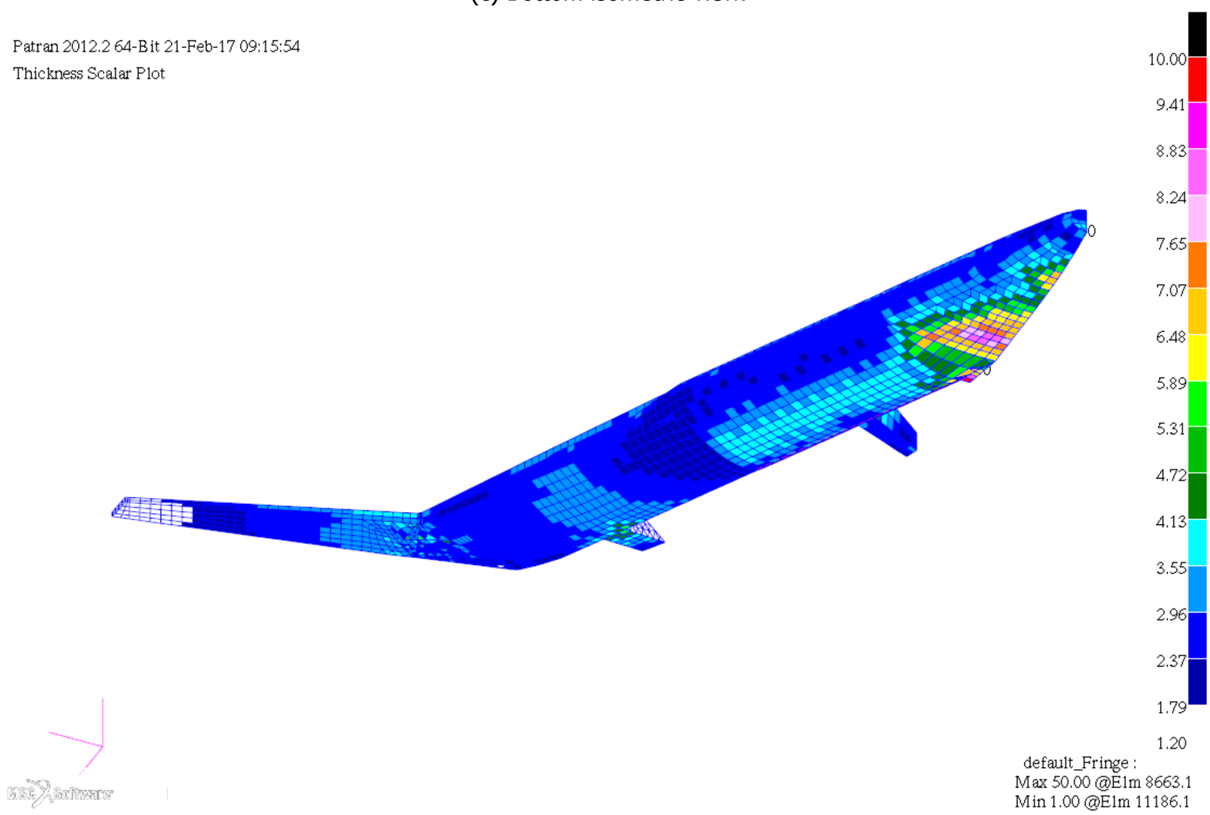


(b) Top reversed isometric view.

Figure 6.5: Sized thickness values $[mm]$ of outer shells.



(c) Bottom isometric view.



(d) Bottom isometric view, 10mm scale upper limit.

Figure 6.5: Sized thickness values $[mm]$ of outer shells.

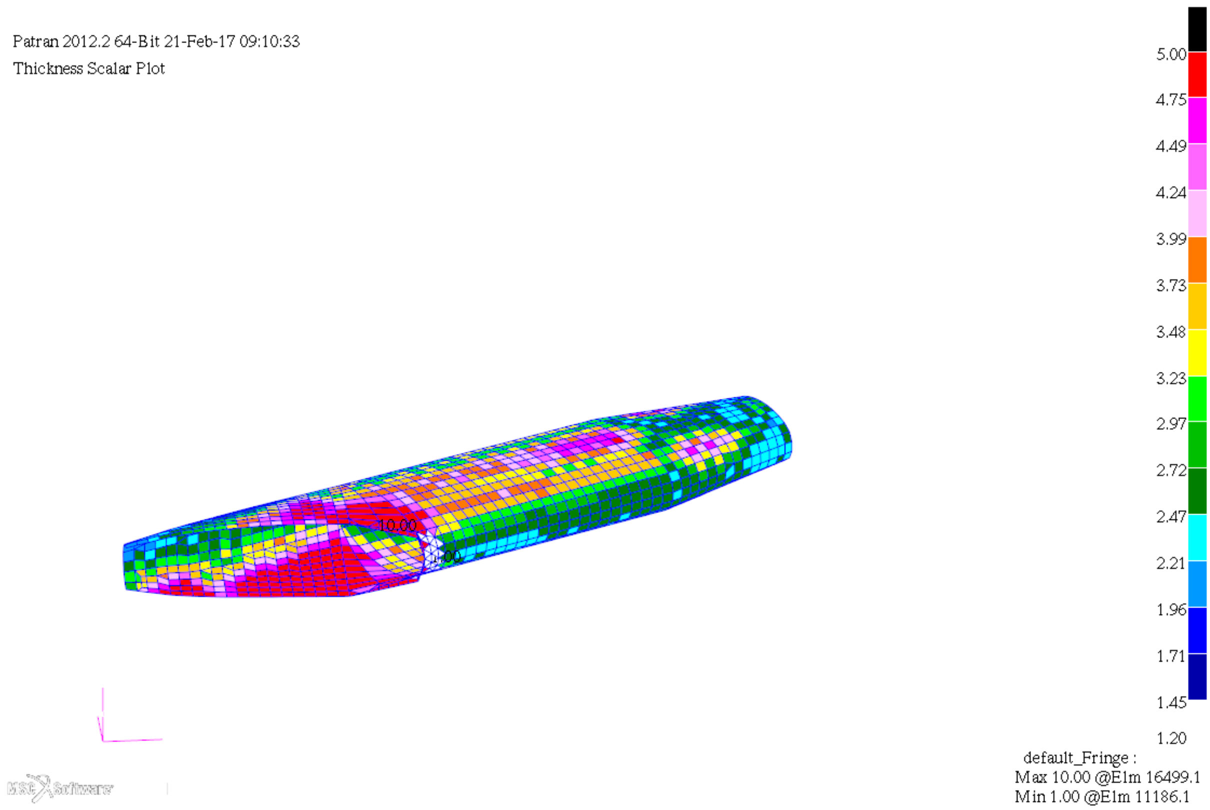
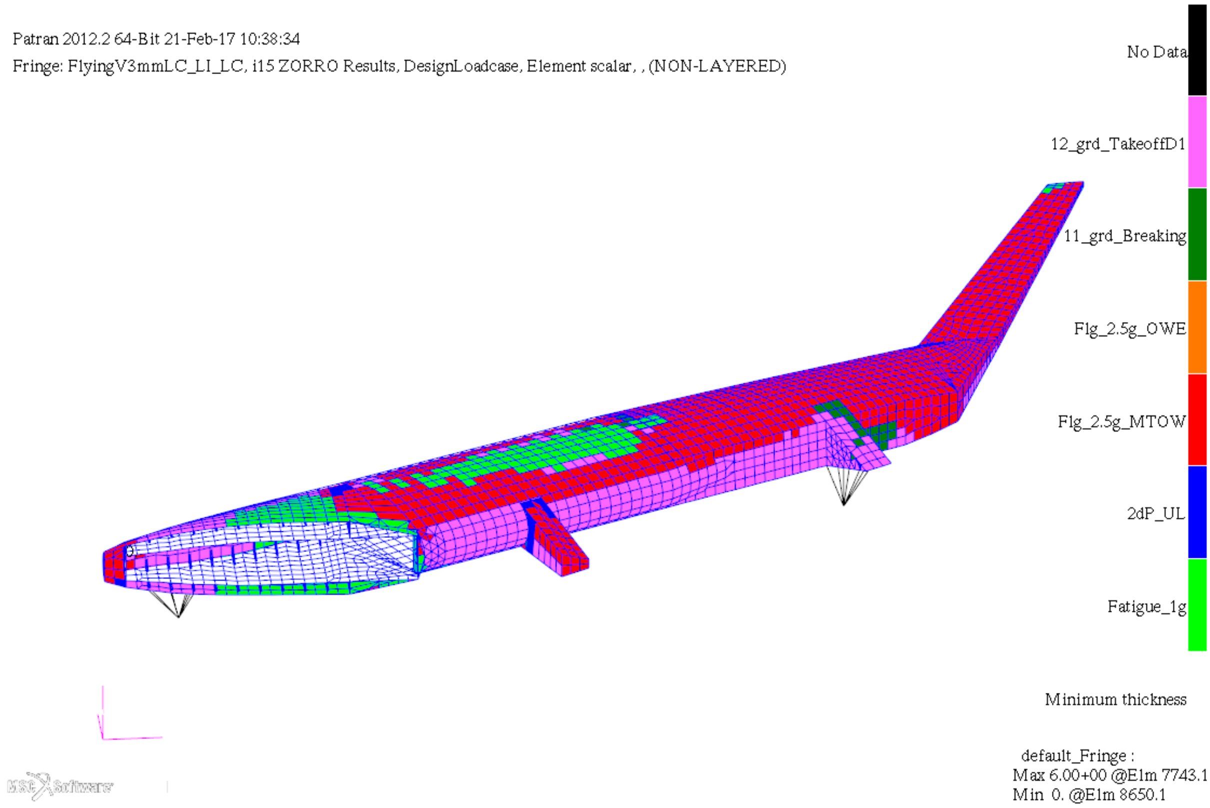
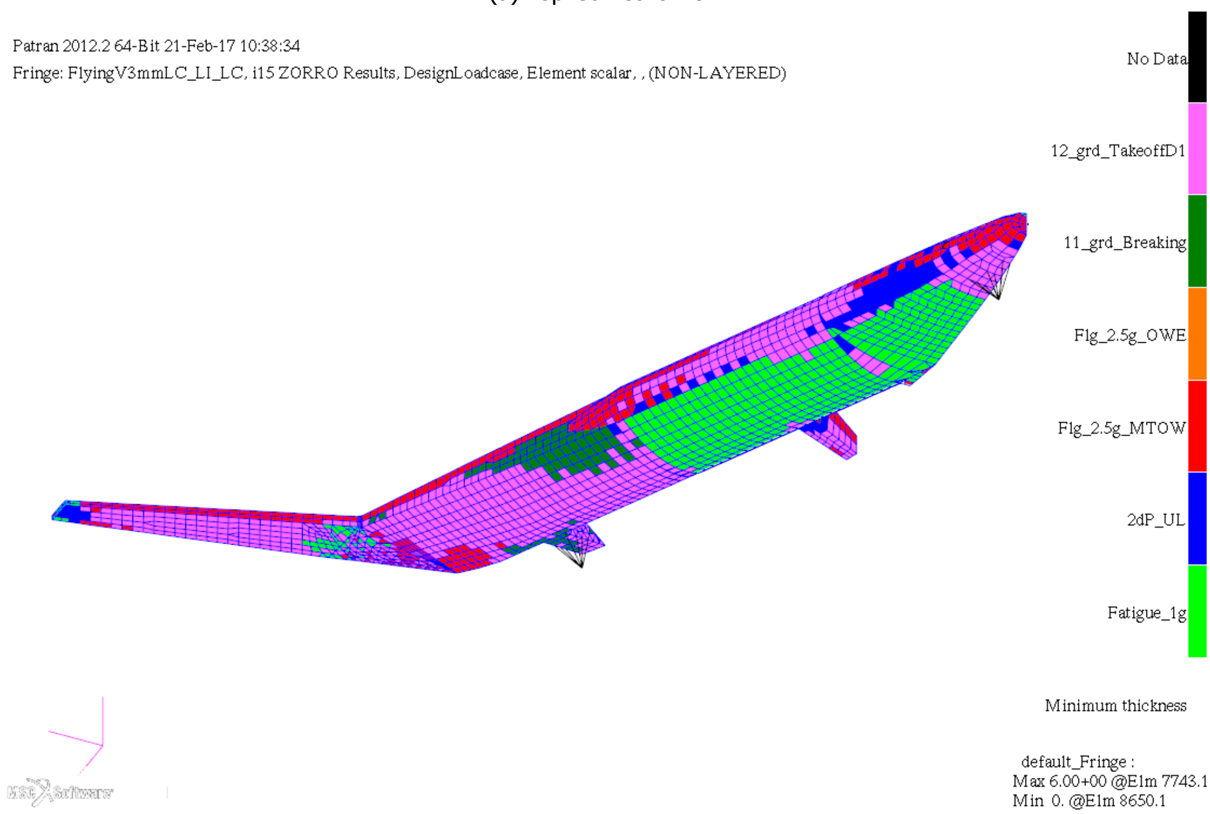


Figure 6.6: Sized thickness values of pressure cabin shells.

At section 4, a small reduction in thickness is observed in the lower half of both spars. The tapered fuselage walls are sufficiently close to be considered extra spars. From the critical load case plot, it is seen that load case 11 is critical for the majority of the spar elements between the gears. An exception is the top row of elements, which are sized by load case 3. This increases the thicknesses of the top half of the spar and explains why the spar thickness reduction is only observed in the bottom half.

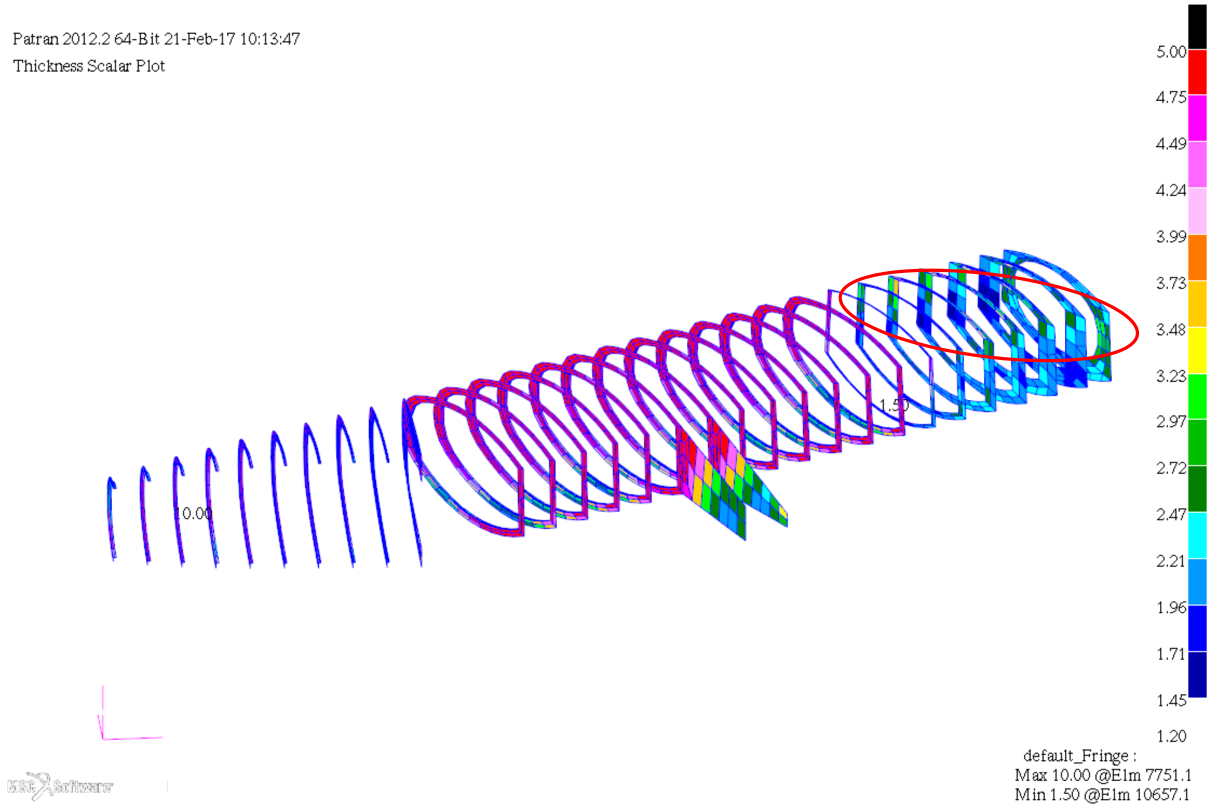


(a) Top isometric view.

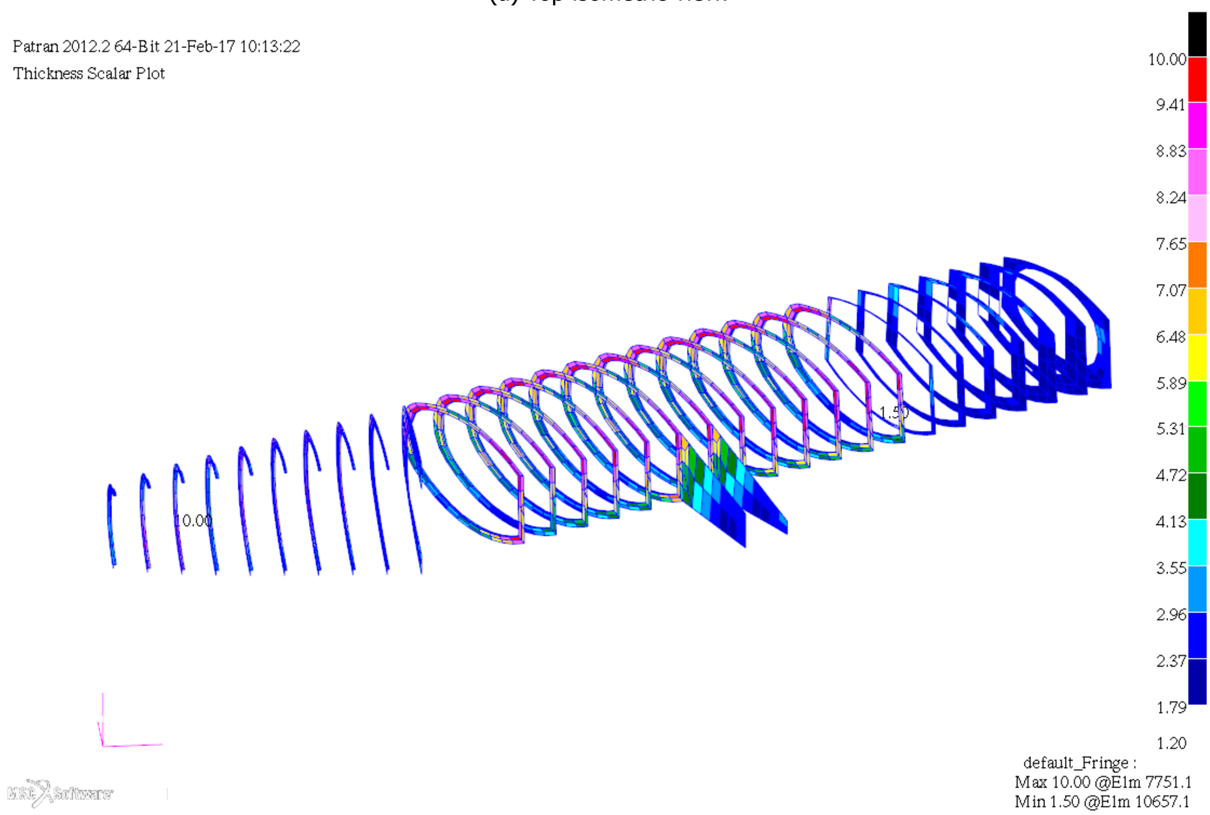


(b) Bottom isometric view.

Figure 6.7: Critical load cases for outer shells.



(a) Top isometric view.



(b) Top isometric view, 10mm scale upper limit.

Figure 6.8: Sized thickness values $[mm]$ for frame shells.

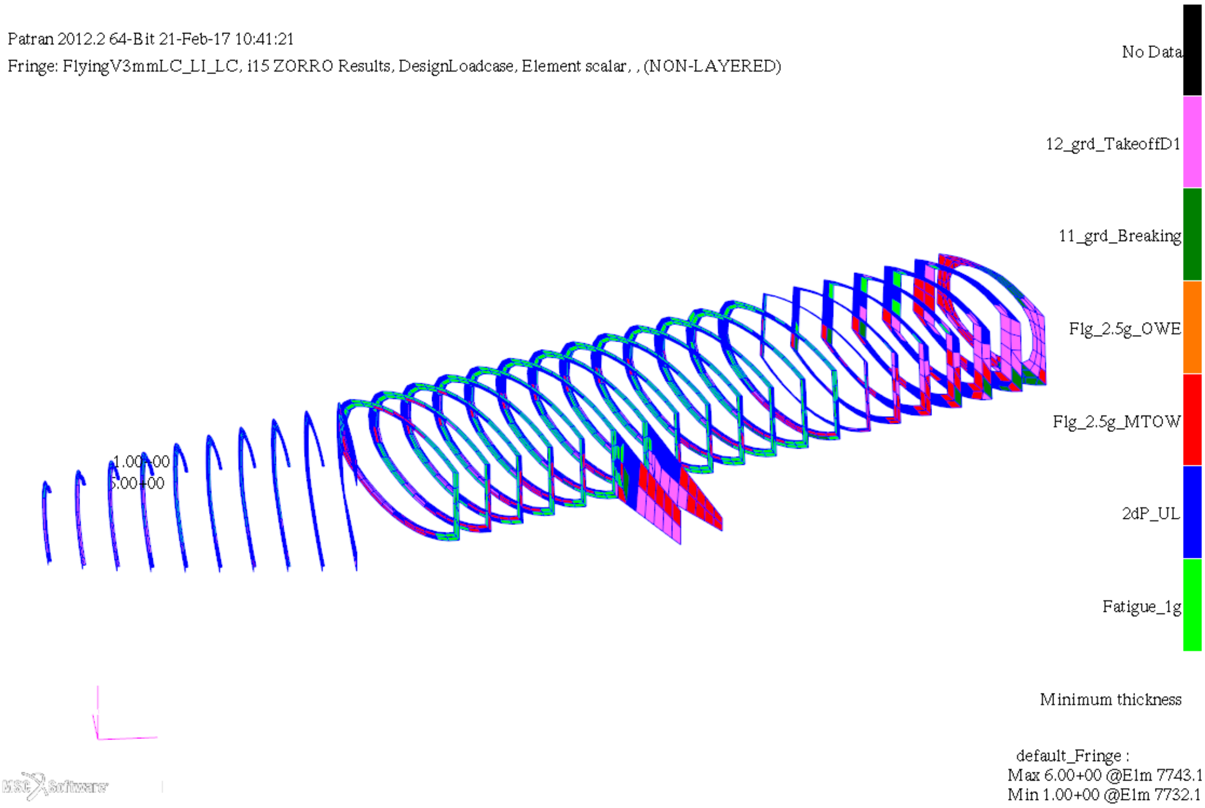
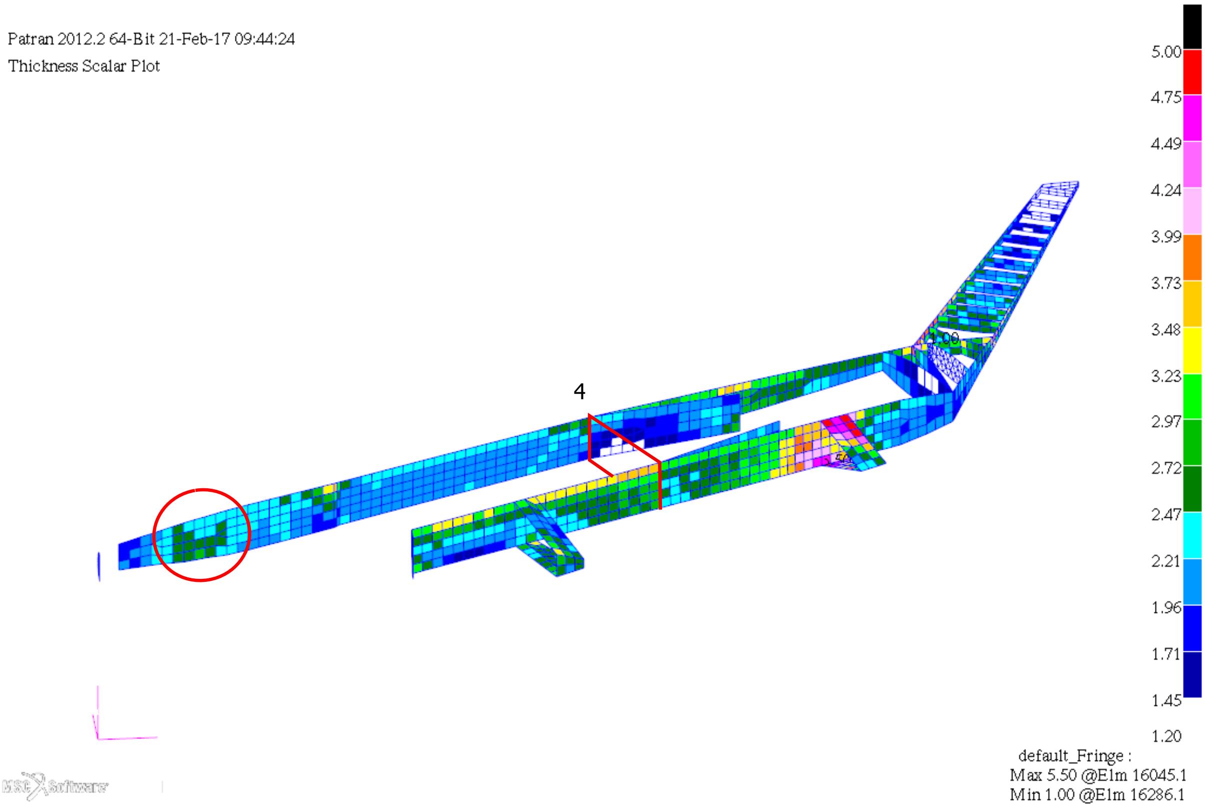
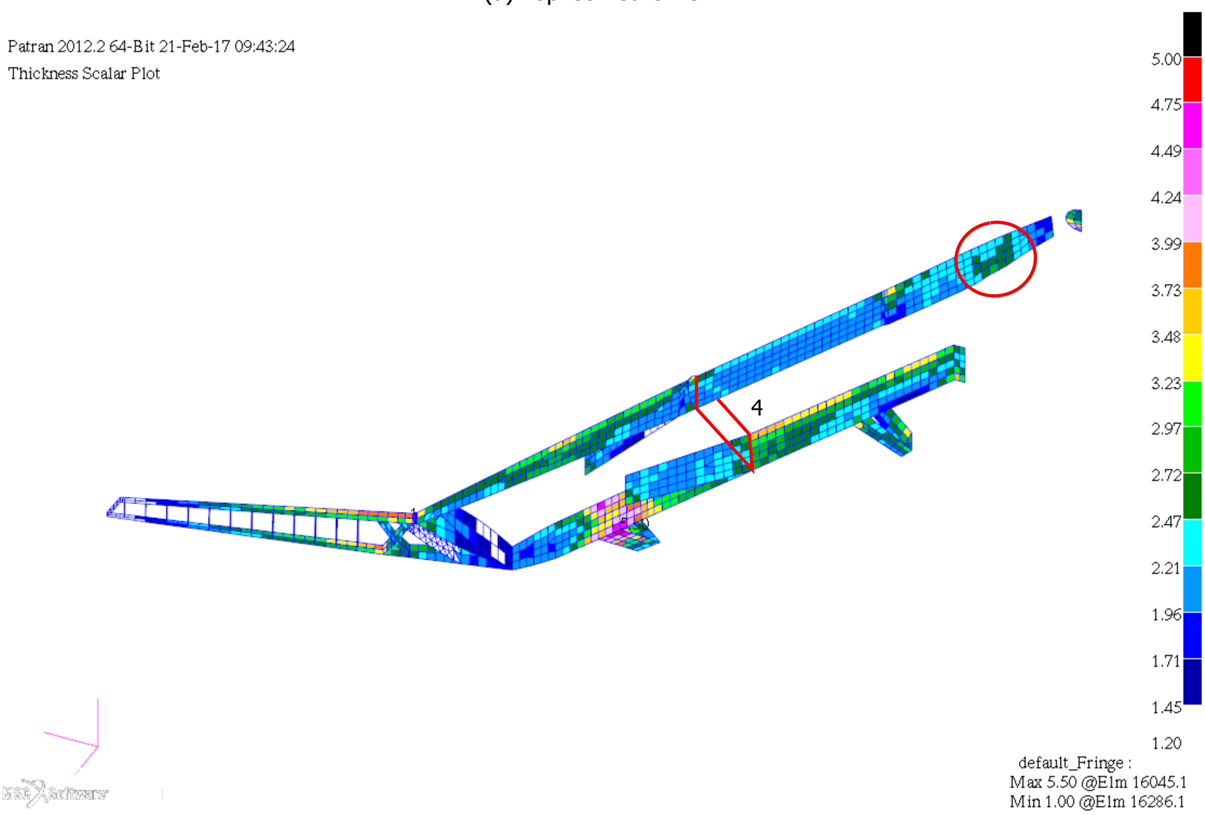


Figure 6.9: Critical load cases for frame shells.



(a) Top isometric view.



(b) Bottom isometric view.

Figure 6.10: Sized thickness values $[mm]$ for spar shells.

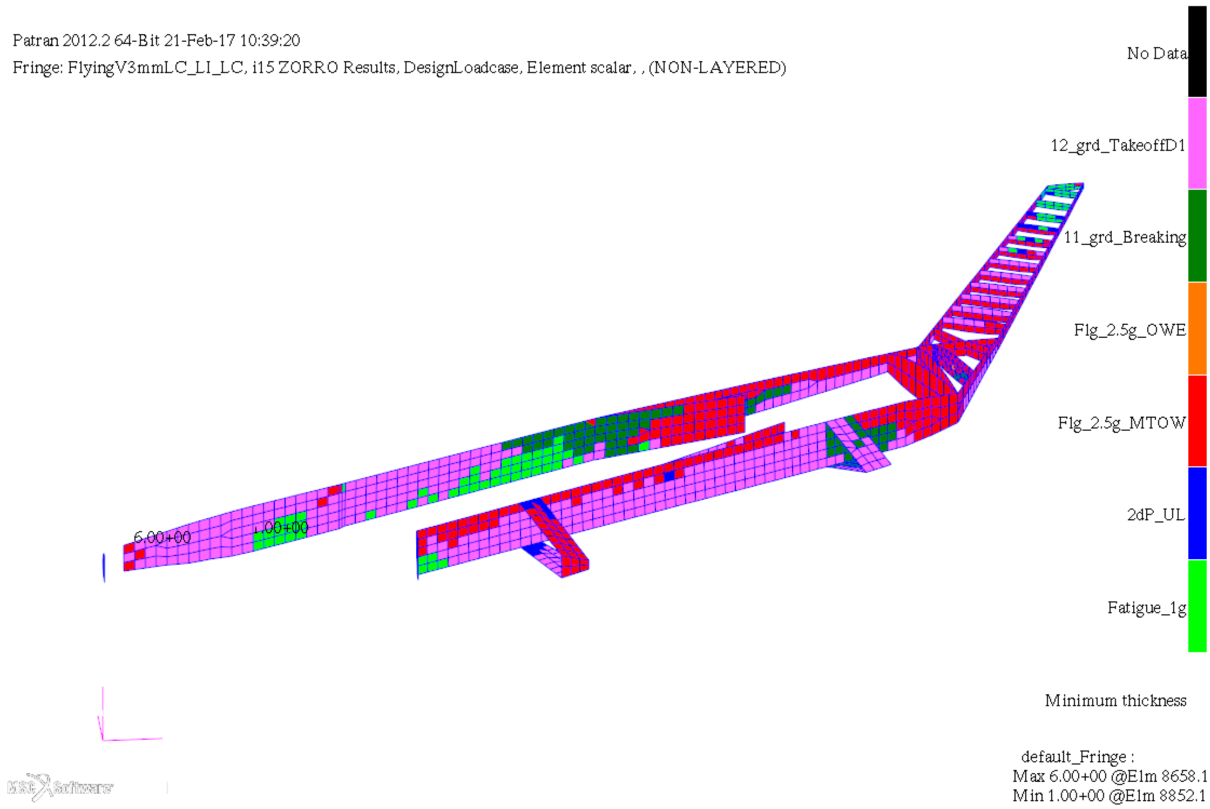


Figure 6.11: Critical load cases for spar shells.

Table 6.1: Sized structure mass results.

Source	**Mass *[kg]***		
Estimated		99781	
Benad		73953	
ZORRO		13790	
FEM-to-Engineered factor	1.67	2.0	2.5
ZORRO corrected	23029	27580	34475
Difference from estimated	76752	72201	65306
Difference from Benad	50924	46373	39478

6.3. Mass and Balance

The mass of the sized structure has been determined in three different ways:

1. An initial estimation given the design mission.
2. An estimation of Benad.
3. An estimation using ZORRO.

ZORRO's sizing procedure has converged on a mass to within 0.05‰after 15 iterations. This estimation must be corrected to an engineered mass, which includes secondary structure elements not captured in the FEA. The FEM mass usually lies between 50% and 60% of the engineered weight. An additional factor is introduced which assumes 40%.

All the resulting masses are provided in Table 6.1. A difference of more than 50% between ZORRO engineered mass and the reference mass is observed. This occurs even for the highest correction factor and the lowest reference structure mass. No conclusions can be drawn regarding the magnitude of the masses without a reference aircraft model analysis. This model must be constructed using the same design procedure, feature the same assumptions and have a known engineered mass. Such a model was set up as described in Appendix E, but abandoned when ZORRO's results appeared skewed.

The static stability for a 2.5G load is assessed. The centres of lift and mass are determined within Patran and shown in Figure 6.12. This centre of lift is determined from the aerodynamic distribution in Figure 4.2a, in which control surface deflections are not taken into account. The CoG for MTOM and OEM conditions are located at approximately $27.05m$ and $27.91m$ along the longitudinal axis respectively. The CoG position was estimated by Faggiano at $30.92m$, this ensured a static margin of 2.5% MAC with $MAC = 18m$, meaning $s.m. = 0.45m$. With the newly determined CoG, the static margin is increased to 16.7%.

In the sizing results, thin thickness values were observed in the outboard wing structure. An improvement in sizing results without the bending stiffness correction will likely shift the CoG aft. On the other hand, introduction of stiffeners will likely shift it forward. Boundary-induced thickness values will reduce for a balanced aircraft and shift the CoG aft. Future research must indicate what the combined effect is on the static margin and the aircraft stability.

The centre of lift moment arm around the CoG for MTOM conditions is $8.3m$. This indicates a large pitch-down moment is present. As stated, control surface deflections are not taken into account. For analysis of sustained 2.5G pull-up and -1G push-over manoeuvre, control surface deflections must be modelled into the wing profile passed to LatticeBeta.

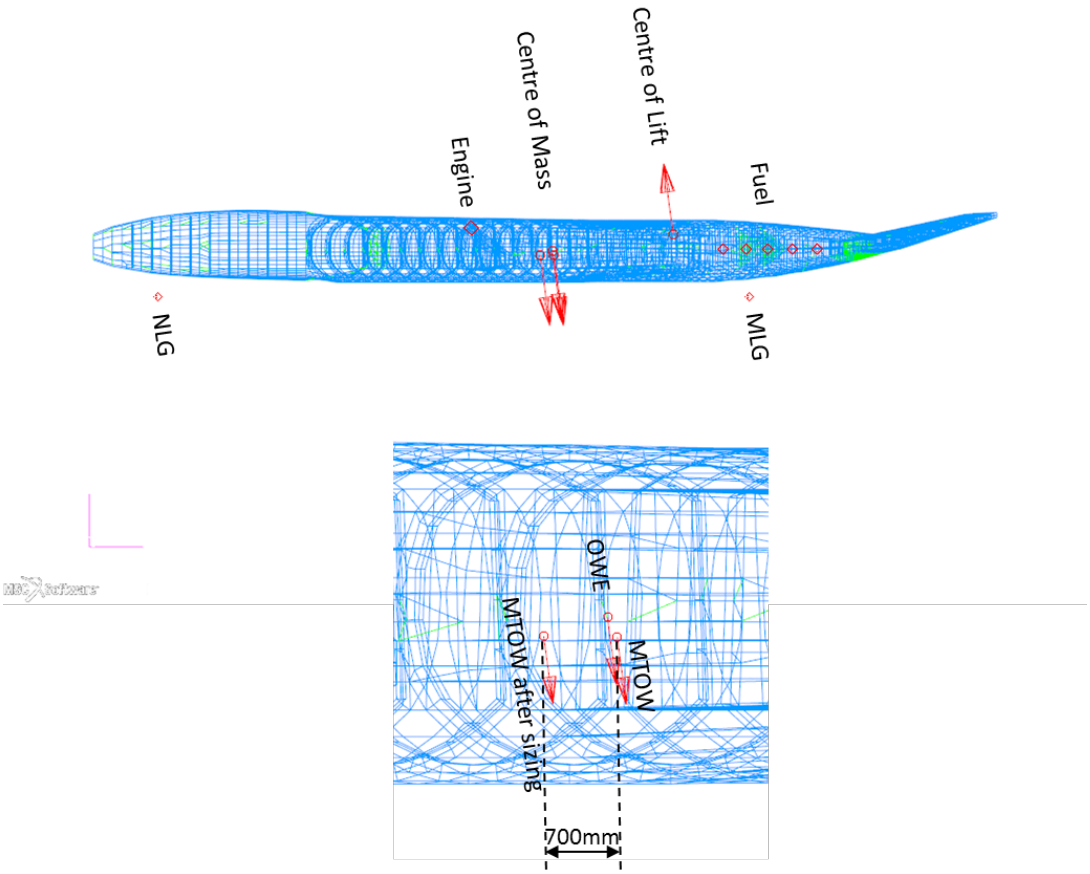


Figure 6.12: CoG for MTOM and OEM as well as aerodynamic centre longitudinal positions.

7

Recommendations for Flying V Research Continuation

Based on the results of the structural analysis and sizing, as well as from the model generation process, recommendations are made for the various steps in the process towards obtaining a valid mass estimation. This is done for attempted sizing procedures in Sections 7.1 and 7.2, for the model generation in Section 7.3 and for the structure concept and integration of systems therein in Section 7.4.

7.1. Analytical Solution

Analytical solutions offer the benefit of fast integration of novel technologies, as well as a low computational expense [20]. Para*Py* offers a fast model generation invalidating these two items of concern.

Every variation in cross-sectional geometry is captured in an additional higher level primitive. Each primitive requires its own implementation of an analytical sizing method. For compatibility between the various primitives' analyses, a framework must be constructed in which each of these methods can be coupled. It is found that for a configuration concept with a degree of dependency between design disciplines of aerodynamics and structural mechanics such as the Flying V, an analytical sizing procedure approaches a FEA.

Visual inspection of the airframe model shows various double-spheroid surfaces in the fuselage transition and wing transition sweep change regions. Analytical analysis of these requires further discretisation, approaching a true FEA. For these reasons, it is not recommended to attempt an analytical sizing method to configurations with a multitude of cross-sectional definitions.

7.2. Ideal Sizing Procedure

An iterative sizing procedure must be applied to obtain a feasible design. Structural analysis and sizing results are fed back to the model generator in order to update the design. This ideal procedure is detailed in Figure 7.1.

There are several limiting factor preventing this design loop from being implemented:

- Para*Py* 's has limited functionality:
  - Nastran coupling
    - Writing of Natran-compatible .bdf-files exports only mesh information.
    - Mesh element IDs are not consistent between Nastran and Para*Py* .

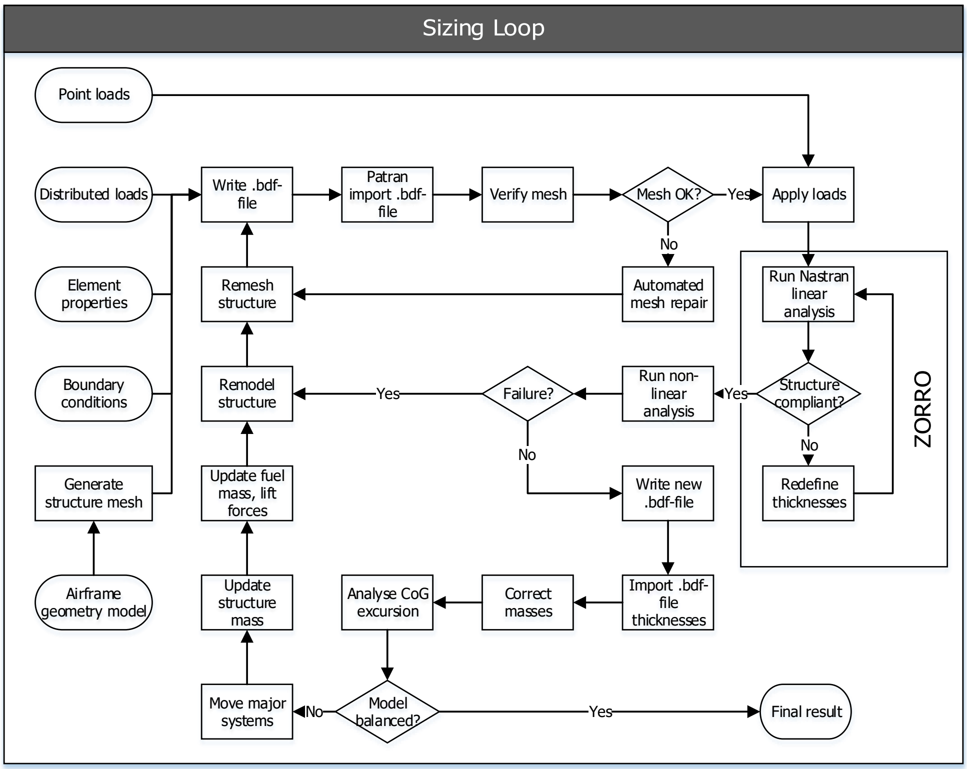


Figure 7.1: Ideal, iterative design procedure.

> Face topology cannot be defined for a non-planar face generated from a Wire.

> The top-level oval cabin definition does not allow for efficient implementation of stiffener geometry.

> No reference mass values are available for components of the airframe. These are needed to arrive at an engineered mass from the FEM mass.

7.3. New Model Generation Procedure

The earlier presented model generation is set up based on:

- The Flying V structure concept.
- The oval fuselage concept.
- The generation of an analysis model for a Flying V:
  > The structure model generation flowchart.

  > The application of a knowledge-based engineering approach.

Various observations of the model generation are made based upon these measures:

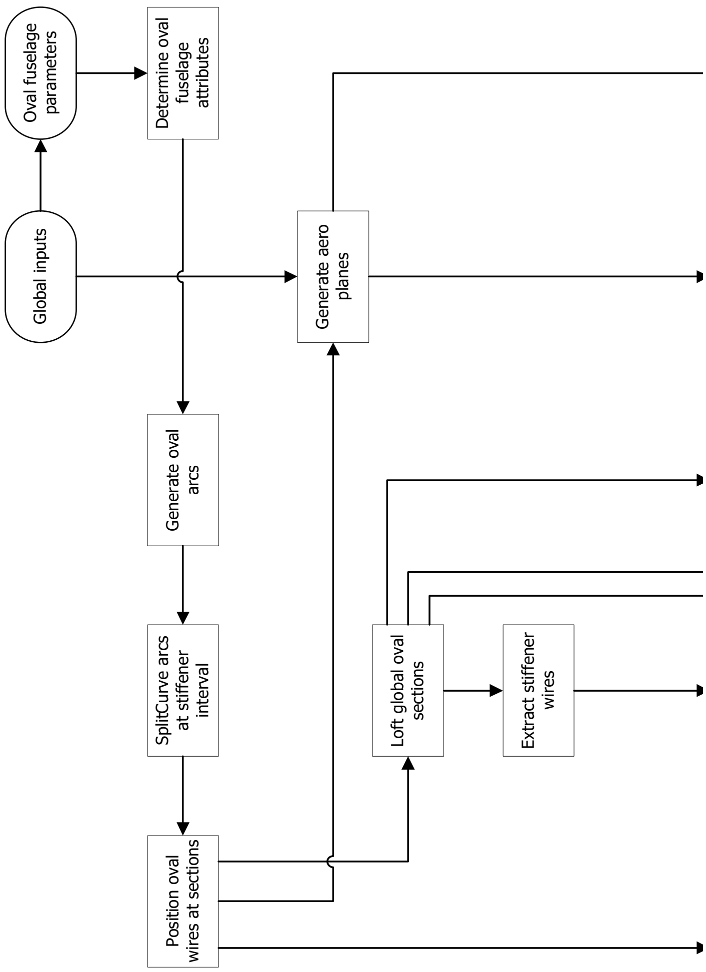
- The model generation only functions correctly for a limited range of input parameters. This occasionally prevents feasible solutions from being generated.
- Debugging code for incorrect model generation proved tedious, due to a *large* number of *complex* geometric operations.
- Model generation from nothing until mesh takes 276 seconds on a Intel Core i7-4500 clocked at 1.8 GHz and automatically over-clocked to $2.4 GHz$ using Para*Py* 1.0.7 on Python 2.7. Again, the large number of geometric operations is the likely cause of the extended duration.
- The generated model was not correct in terms of geometric surface definitions – this may have caused increased mesh generation time.

With these observations in mind, recommendations for a new geometry generation procedure are given. It is assumed that the same version of Para*Py* will be used, in which no new operations are added. The following characteristics are sought after:

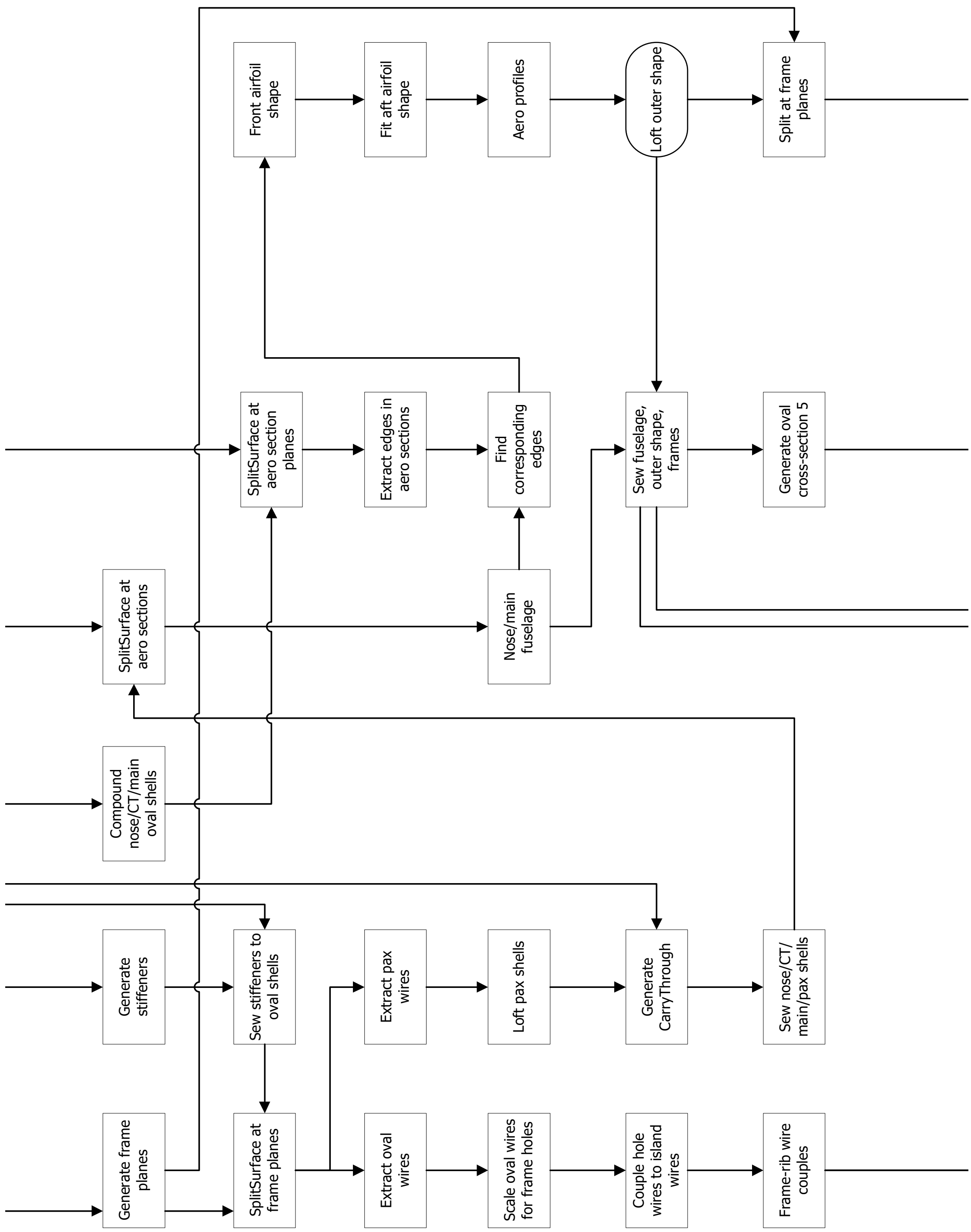
- For geometric equivalence, model generation speed, and memory consumption, calls to geometric operations must be kept to a minimum. Existing geometry must be used rather than creating new geometry.
- The code should be clean with a logical flow in mind. Similar operations are identified and may be consolidated into a single operation. This ensures a larger valid range of input parameters can be tested and, in case model generation fails for a specific parameter set, debugging is simplified.
- An option for stiffeners must be included, such that the FE shell model features stiffness.

A new model generation procedure is presented. The largest change to the modelling procedure is aimed at maintaining geometric equivalence between the structure and aerodynamic shell models. Geometry parts are used and modified, instead of serving as a basis for new parts. In addition, the fuselage structure generation is separated from the wing structure generation. Both are grouped together for clarity and consistency. Figure 7.2 outlines the generation of the structure in the fuselage-section. Wing structure generation for both the transition and outboard wing is shown in Figure 7.3. Note that the wing transition structure generation is no longer treated as an individual part.

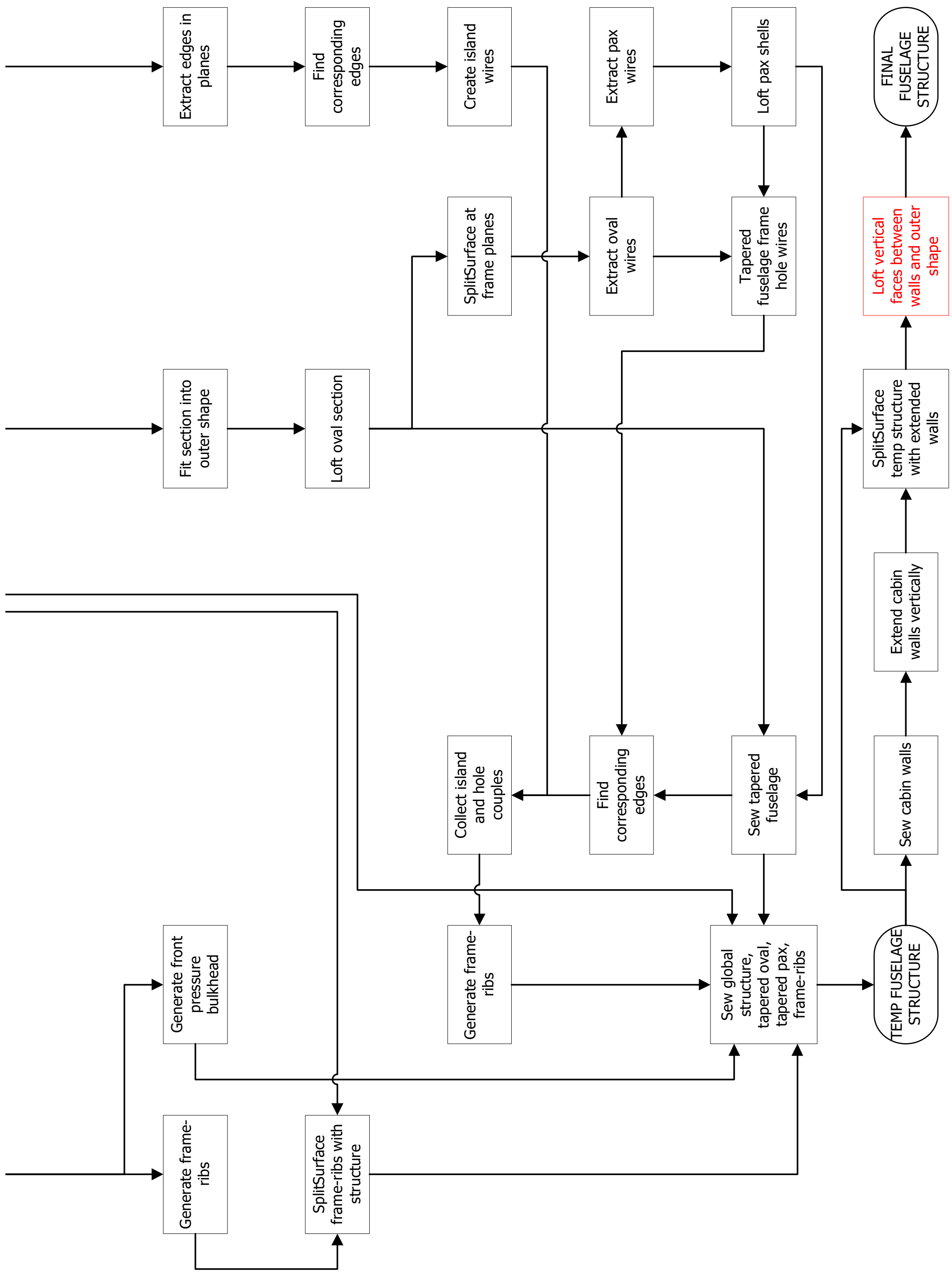
It has been noted in Chapter 5 that an allowance for stiffener introduction is best done in the definition of the oval fuselage arcs. This can be effectuated by implementing a different method for the oval arc generation. Instead of defining control points and fitting a curve, arc segments should be generated using Para*Py*'s *Arc* object. These must be split along set intervals, depending on the number of stiffeners, and collected in a *Wire*. This will ensure an oval shell loft is generated which automatically contains


Oval fuselage parameters
Determine oval fuselage attributes
Global inputs
Generate aero planes
Generate oval arcs
SplitCurve arcs at stiffener interval
Loft global oval sections
Extract stiffener wires
Position oval wires at sections

(a) 1 of 3

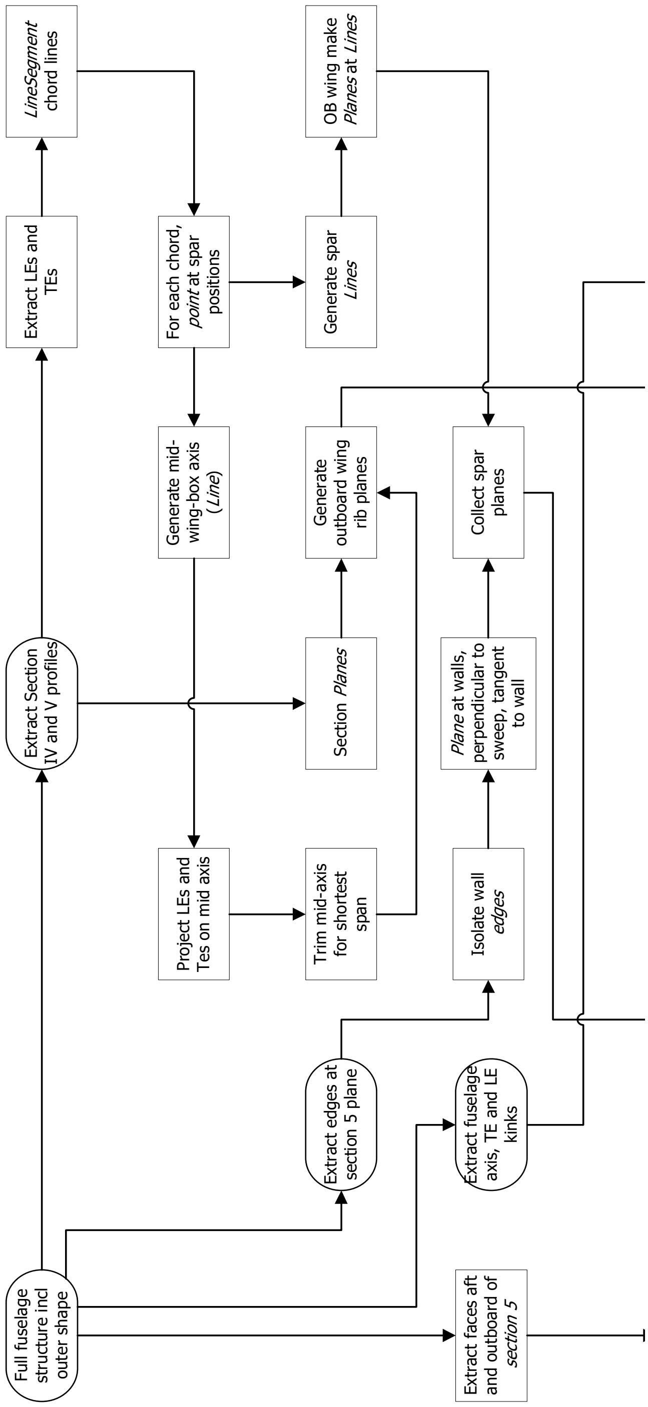

Front airfoil shape
Fit aft airfoil shape
Aero profiles
Loft outer shape
Split at frame planes
SplitSurface at aero section planes
Extract edges in aero sections
Find corresponding edges
Sew fuselage, outer shape, frames
Generate oval cross-section 5
SplitSurface at aero sections
Nose/main fuselage
Compound nose/CT/main oval shells
Generate stiffeners
Sew stiffeners to oval shells
Extract pax wires
Loft pax shells
Generate CarryThrough
Sew nose/CT/ main/pax shells
Generate frame planes
SplitSurface at frame planes
Extract oval wires
Scale oval wires for frame holes
Couple hole wires to island wires
Frame-rib wire couples

(b) 2 of 3

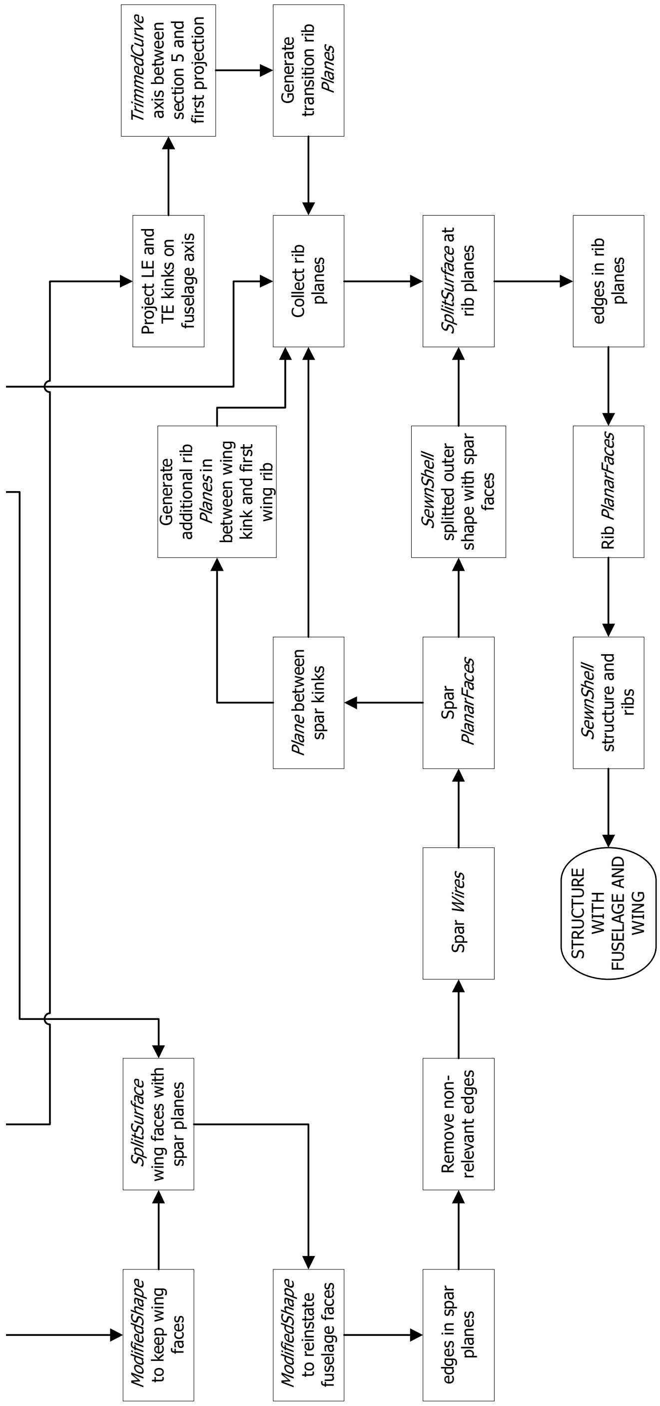


(c) 3 of 3

Figure 7.2: Recommended fuselage structure generation procedure for future Flying V model generation tool.


Full fuselage structure incl outer shape
Extract Section IV and V profiles
Extract LEs and TEs
LineSegment chord lines
For each chord, point at spar positions
Generate spar Lines
OB wing make Planes at Lines
Generate mid-wing-box axis (Line)
Section Planes
Generate outboard wing rib planes
Collect spar planes
Project LEs and Tes on mid axis
Trim mid-axis for shortest span
Plane at walls, perpendicular to sweep, tangent to wall
Isolate wall edges
Extract edges at section 5 plane
Extract fuselage axis, TE and LE kinks
Extract faces aft and outboard of section 5

(a) 1 of 2



(b) 2 of 2

Figure 7.3: Recommended wing structure generation procedure for future Flying V model generation tool.

edges for stiffeners. For the passenger shell, the same procedure must be applied. First the profiles must be split, only then lofting should be done. Lastly, stiffeners can also be implemented on the airfoil skin with relative ease by splitting the aft airfoil profile, similar as must be done to the oval arcs and passenger cabin edges. After the stiffener's on-shell edges are collected, stiffeners are generated using the procedure in Figure 7.4. A working example for this procedure is presented in Appendix E for a conventional aircraft.

A coupling between the outer shape and internal oval fuselage structure is ensured by splitting the oval fuselage section along the airfoil sections using Para*Py* 's *SplitSurface* operation. The fuselage is split with planes located at the airfoil sections. The resulting edges are selected. A sketch of the expected geometry is provided in Figure 7.5. Whereas the aft airfoil profile in the current model was limited to horizontal attachment, the new definition allows for more options due to the oval fuselage section being broken into numerous parts. With the increased flexibility in aerodynamic design, new methods may be tried for aft airfoil parametrisation and generation.

Generation of frames and ribs remain similar as described before in Section 3.4. Minor modifications to these concern adjusting for the increased number of profile vertices. The correct vertices must be selected by efficient sorting methods. A new carry-through generation procedure is proposed to handle additional vertices. More importantly to create geometrically sound faces which will be meshed properly by removing zero-length edges. This is further described in Section 7.3.

Connecting shells must be introduced between the fuselage structure and the airfoil skin. This is included in the last steps of the fuselage structure construction. As of now, the preferred method seems to project each top and bottom wall edge onto the outer shape. The direction with which this must be done is still unclear. Resulting projected curve start and end points must coincide with the neighbouring projections. Alternatively, each of the inter-wall edges can be projected along the edge tangent onto the outer shape using *Point.project*. A line segment tracing over the surface between the projection (u,v)-locations on the outer shape face may be used in a *SplitSurface* operation. However, this is not tested and is therefore represented in red in Figure 7.2c.

The generation of the wing structure features fewer defining profiles. The outer shape including the fuselage structure is split into two along *section 5*. Faces ahead of the plane at *section 5* are temporarily removed using *ModifiedShape*. This is done such that a split along the transitional wing spar planes does not split the fuselage structure. After the spar plane *SplitSurface* operation is performed, these faces are reinstated. In a procedure similar to that seen in the current model generation, ribs are generated by defining rib planes. The structure must be split along these planes, edges must be selected and composed in a *Wire*. A rib *PlanarFace* can be generated from this *Wire*. The ribs can then be sewn into the structure, which results in the final model which can be used for aerodynamic and structural analysis.

Note that no definite procedure has been implemented for positioning and sizing the fuel tanks. The most promising method is to extract the skin surfaces between two rib planes and use these to create a *LoftedSolid* representing a section of the fuel tanks. This solid then provides a volume for local fuel mass. If *RBE3*-elements are implemented in Para*Py* , these can be used to connect the geometrically determined CoG to the solid vertices. Otherwise, a uniformly distributed load may be applied to the bottom skin surface. This causes a slight misrepresentation of fuel loading, as the cross-sectional area along the structure's axis is not uniform. Local fuel loads are slightly larger towards the wing root. This effect is more pronounced for higher taper values of inter-rib sections.

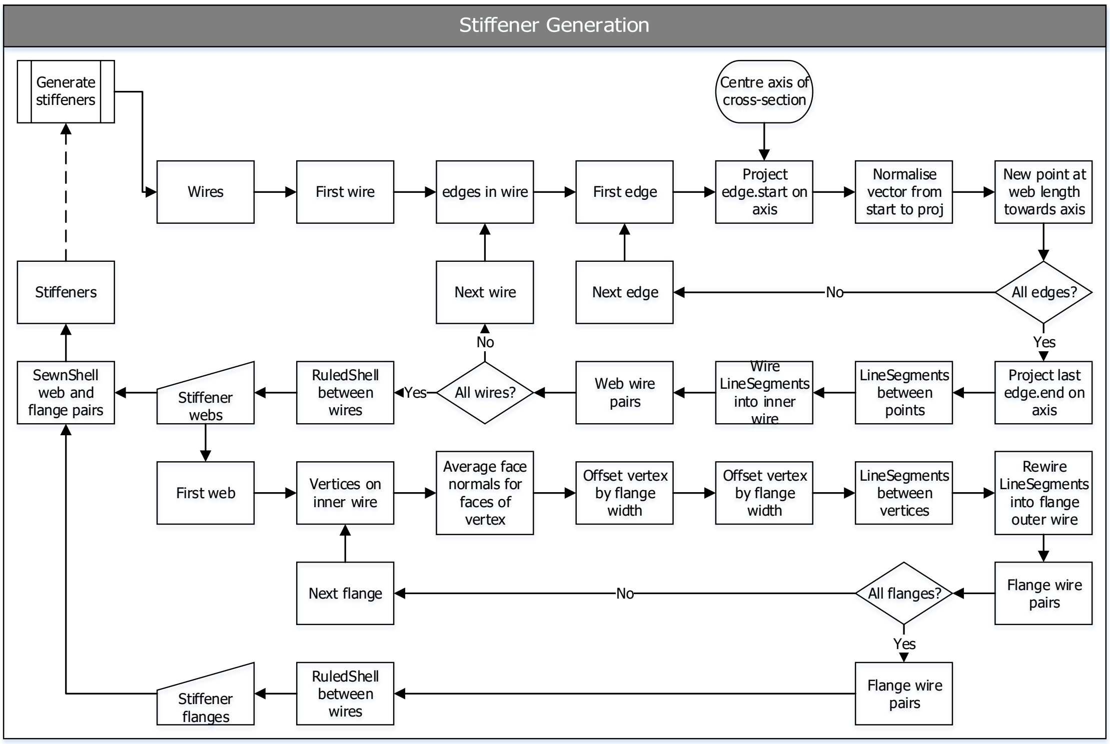


Figure 7.4: Procedure for stiffener generation.

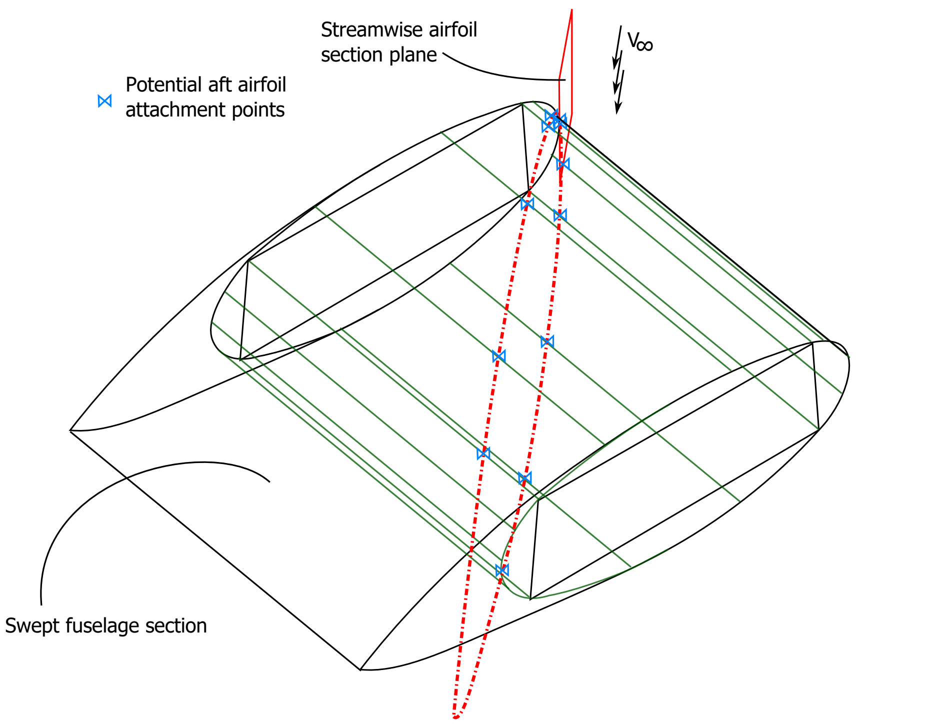


Figure 7.5: Extended number of points for aft airfoil attachment with stiffener-enabled oval fuselage definition, $n_{stfnr} = 2 + 3 + 2 + 3 = 10$.

Mesh Error Solution

Mesh errors are identified in the wing kink and root transition. The wing kink structure will be properly meshed using the new procedure. The carry-through mesh errors occur due to improper face definition in the ceiling, floor and oval side arc faces. The floor and ceiling faces are split to allow for sewing the front web into the carry-through. Using *SewnShell*'s *cut_free_edges*-option, this should also be achieved. The resulting face will have a minimum of 5 edges, to which a Quad-algorithm cannot be applied. Instead, a Tri-algorithm must be applied with quadrilateral-preference.

Other mesh errors present themselves in the oval side arcs. The outboard arcs of *section 2* and *section 3* have a topologically common point, though this is not marked as geometrically equivalent. The result of a *Fused*-operation is the introduction of a zero-length edge on a quadrilateral face. A similar problem is found at the inboard arc of *section 3*. By projecting the arc profile onto the symmetry plane and creating a *LoftedShell* between this, a common non-equivalent point was introduced.

The preferred solution is to extract the proper edges, rewire them and make a *Face* from this. However, this wire is not planar and no method is available to migrate the face topology. Instead, a recommendation is made to change the section definitions and allow for a gap between *section 2* and *section 3* as shown in Figure 7.8a. This also implies aerodynamic *section* $II.a$ is added.

7.4. Structure Concept

The structure concept is evaluated on two aspects. Integration of the oval fuselage concept is discussed and new recommendations are made regarding the airframe concept.

7.4.1. Oval Fuselage Concept

In applying the oval fuselage concept integration into a FW concept, various hurdles are identified. These concern implementation of the cabin panels as an integral part of the load-carrying structure and deformation of the fuselage leading to a disturbed aerodynamic profile.

Integration of Cabin Panels into Primary Structure

It is recommended to investigate new residual force-carrying members in the oval fuselage. In these, panels are to be removed from the primary structure and replaced by beam elements between the frames. Two concepts for this are provided in Figure 7.6. Although the oval arcs are no longer continuously supported by the panels, stiffeners transfer the bending loads to the beams and frames.

Removing the tensile and compressive loading of the cabin panels due to pressurisation will also introduce additional benefits. Once the panels are not part of the primary structure, cut-outs result in a lower mass penalty. This enables a wider cabin width to be used for payload stowage, and allows more possibilities for overhead bins in the crown volume. Maintaining the payload and cabin height restrictions, the fuselage dimensions can be reduced for a thinner airfoil, shorter airfoil chord and a subsequent reduced structure mass.

One of the primary notes of concern for oval fuselage integration as noted by Airbus' employees is sizing for rapid decompression. In case the crown, keel and side arc volumes are sealed, a rapid decompression will put significant pressure forces on the respective cabin panels. Instalment of alternative load-carrying beams removes the seal and allows for blow-out panels. In addition, existing furnishing systems can be installed. Cross-members doubling as seat rails are placed on each of the beams to carry the cabin loads, as shown in Figure 7.6.

Full-chord Integral Frames

As mentioned in Chapter 2, the frames are connected to the inboard wall of the fuselage. Frame extensions inboard towards the trailing edge are implied. By applying the modelling procedure described in Section 7.3, creation of frames and ribs will be less complicated. A full-chord integrated frame design

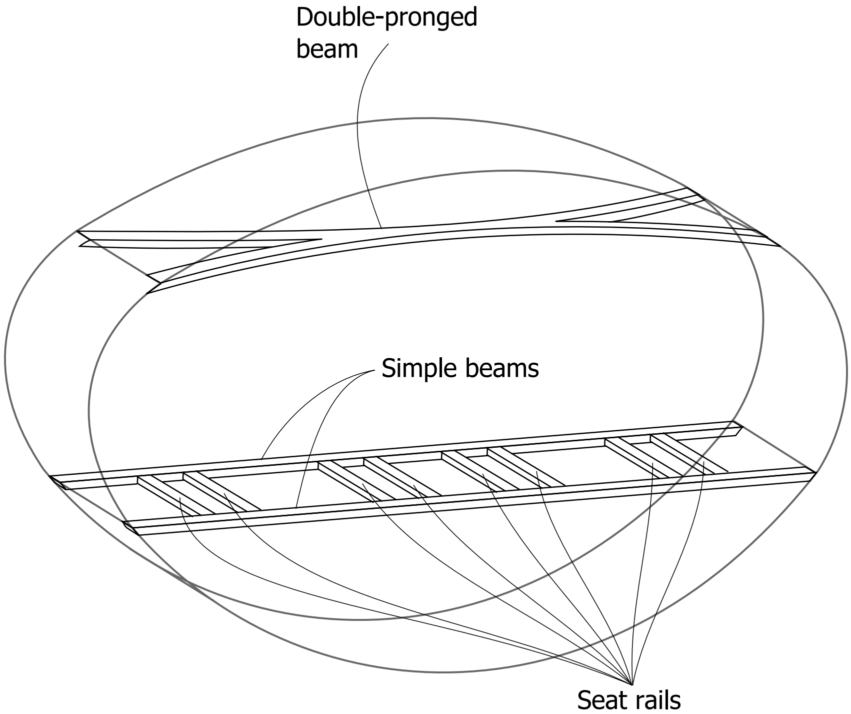


Figure 7.6: Solutions for compression beams in floor and ceiling members of oval fuselage.

may be applied. The manufacturing constraint described before no longer holds when using a full-chord frame.

A concept for a full-chord frame design is presented in Figure 7.7. In this, the frame is continuous over the entire chord line. The shown cut-out is optional. The oval arcs attach to the frames outward from the flange. Depending on the chosen design solution for the passenger cabin panels, either the beams or panels connect at the vertices of the oval shells.

7.4.2. Airframe Concept

The fuselage frame extensions to the symmetry plane are initially excluded from the primary structure and not modelled. Introduction of the forward extension box enabled identification of an unforeseen load path to the symmetry plane. In a FEA, this box appears highly loaded. Addition of all frame extensions and wing skins to the symmetry plane, as seen in Figure 7.8b, will alleviate the single forward extension box. On top of that, they increase the bending stiffness around the z-axis. This will decrease the longitudinal displacement seen in the deformations for the braking load case.

Introduction of the full-chord frame described in Section 7.4.1 requires a proper connecting element between the transition wing aft spar and the inboard oval arc shell to be included in the structure. A structure element similar to that on the outboard oval arc described in Section 2.6.1 is introduced at *section 4*.

Thick nose fuselage crown and keel shells are observed. Due to pressurisation loads on their small curvature shells, internal stresses are inherently large. The curvature is reduced by decreasing the floor width and cabin height. This provides larger keel and crown heights for a given root airfoil profile. Alternatively, the total width can be reduced by increasing the LE sweep angle.

System Integration

Fuel tank integration in the transition wing part provides sufficient fuel volume. However, in order to extend the ferry range and to allow for fuel trim, wing tanks may be considered. Fuel tank placement

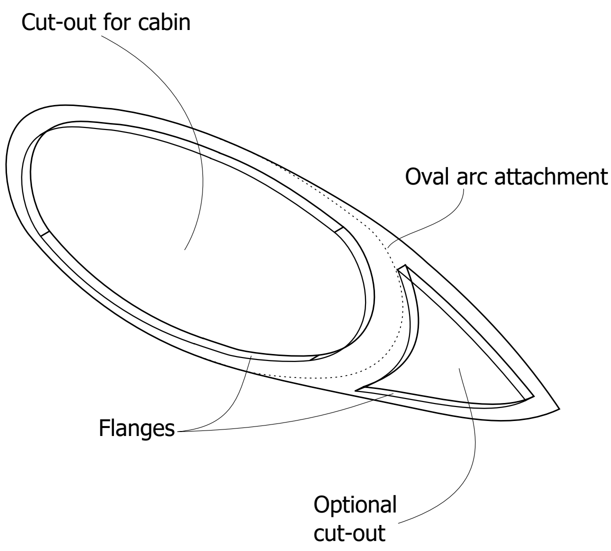
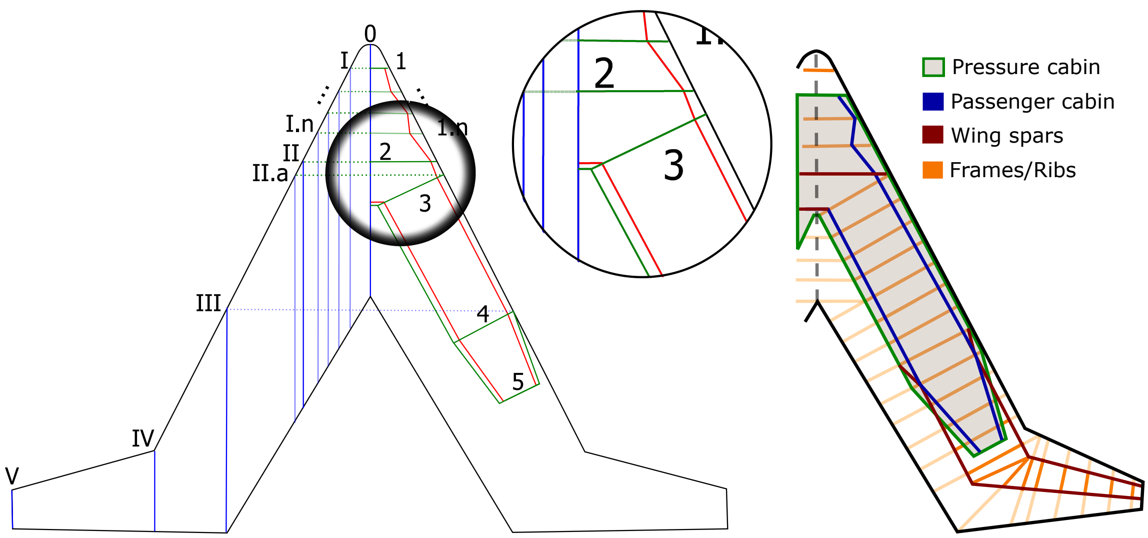


Figure 7.7: Full-chord integrated frame design concept.



(a) Recommended repositioning of planform sections – decoupling of *section 2* and *section 3*.

(b) Recommended update on the structure concept.

Figure 7.8: Planform recommendations.

in the fuselage keel will very likely result in heavy structures to conform to crash-worthiness requirements. Alternative fuselage fuel tank placement must be investigated, such as aft and inboard of the fuselage.

Based on ZORRO results, the CoG is further forward and the static margin is larger than initially assumed. By repositioning the engine further outboard, the CoG will shift aft:

- The engine mass is shifted aft.
- The reinforced structure around the engine is shifted aft.

However, this shift aft may not be sufficient for obtaining the proper static margin as required by the aerodynamically optimised planform defined by Faggiano.

8 Conclusions

A structure concept for the Flying V was researched in collaboration with Airbus' Future Project Office. The Flying V planform, proposed in a conceptual design study of TU Berlin and Airbus was taken as a basis. The fuselage integration was changed from an interior dual-tube fuselage configuration to an oval fuselage, partly integrated in the skin. The oval fuselage was conceived at TU Delft and consists of multiple constant-radius arcs. These are supported by structure members carrying resultant forces from internal bending moment differences. Implementation of this fuselage concept follows the design philosophy of the original Flying V concept; it provides a higher wing volume efficiency, introduces better airfoil shaping parameters and allows a wide and open cabin configuration.

The integrated fuselage is separated into multiple sections: a streamwise-oriented nose section, a main section, a tapered section and a transition structure between the nose and main section. The main and tapered sections are oriented parallel to the leading edge. The transition structure doubles as a conventional wing carry-through structure. To effectuate this, the defining fuselage profiles are connected to the symmetry plane. Additional members are introduced to provide a front and rear carry-through spar. The open cabin concept is maintained by introducing a cut-out in the front spar.

Between the main and tapered fuselage sections, a wing transition structure is incorporated. This structure enables a smooth change from integrated fuselage cross-section to pure-wing cross-section. The aft section of the fuselage is tapered as it cannot accommodate a full-size fuselage. The wing transition features a conventional wing structure with spars and ribs. The ribs feature cut-outs in the region where the fuselage is contained.

The outboard wing section features a conventional structure. A large change in sweep angle is present between the wing transition and outboard wing. Both the transition and outboard wing spars are extended until they intersect. A kink rib spanning the spar kinks is introduced. Additional ribs are positioned between this kink rib and the outboard wing's ribs. An extra rib is placed at the connection between transition and outboard wing to support a kink in the skin surface. A different parametrisation of this wing section will remove this kink and increases skin smoothness. This was not incorporated such that Faggiano's optimisation research could use simple aerodynamic analysis tools.

An identical geometry model of an integrated-fuselage configuration must be used for every applied analysis. Parametrisation of the aerodynamic and structure model was based on several sections. Starting from a main fuselage definition, root profile and leading edge profile, a majority of the structure sections was defined. The streamwise aerodynamic sections followed from these. Lastly, the fuselage definition is completed. The internal structure was based on few global parameters, including rib and frame pitch as well as spar positioning.

It has been observed that not only in the joint Flying V project, but also in other aircraft design projects in the industry, a local change in geometry is not communicated to all the disciplines. This can be prevented by always employing a single reference model definition to which different disciplinary views are applied. To accomplish this, Para*Py* is used as a KBE platform and geometry generator. With the application of KBE, the challenge lies in capturing not only correct knowledge, but also a complete collection of the knowledge.

The applicability of higher-level primitives (HLPs) used in KBE modelling is closely tied to the applicability of an analytical analysis of a structure. They both aim to capture component similarities [29], and for each class (a cross-sectional configuration) a geometry definition and analysis implementation must be customised. As such, higher degrees of functional integration produce a higher number of HLPs. For a conventional TaW, or even segregated pressure shell and outer skin configuration such as Liebeck's BWB and the BWB example by La Rocca [8, 29], this implies KBE can provide high value to model generation and analysis. Fully integrated solutions between design discipline lack the recurrence of structure cross-sections. For the Flying V with oval fuselage integration, this is exemplified in the variety of airfoil and fuselage definition methods.

If a collection of structure parts can be contained within a single HLP, it is also possible to contain them in the same analysis group. For increased model complexity, HLPs are more difficult to identify. In addition, the consideration must be made whether the gains of an analytical sizing method – reduction in computational expenses – are worth the effort of setting it up for each model section. It has been shown model and mesh generation for numerical analysis can be automated with currently available tools if done correctly. This comes with the added benefit of allowing higher flexibility in and better extendibility of the model, as well as variable fidelity by means of few mesh generation settings. A Nastran analysis can be run quickly, even for multiple design iterations. The disadvantage often named for numerical solutions, namely the computational expense, is less pronounced for the calculation of solutions given a rough grid and rather lies with the geometrical operations required to generate a CAD model.

Automation of CAD model generation removes the bottleneck in a numerical design process. Para*Py* provided a platform in which to automate the generation. The primary structure model used was initially built for an analytical sizing method. Several components were omitted and geometric continuity was not adhered to. System masses are accounted for by empirical relations. A switch to a numerical sizing method was made after model complexity and increase of HLPs proved too impractical for an analytical sizing method. The model generator was recoded to take into account geometric continuity such that a single, multi-face mesh could be generated.

An FEA mehs was generated using the SALOME meshing algorithm and exported to Patran using Para*Py* 's integrated tools. The mesh export and import tools are not sufficiently developed to incorporate a fully automated design loop. Load cases generated within the model generator using Faggiano's aerodynamic analyses could not be exported. Instead, Airbus' LatticeBeta was used to obtain aerodynamic load distributions. Additionally, the mesh element IDs in Para*Py* and Patran do not correspond. Mesh element groups cannot be exported to Patran disabling automatic element selection for distributed loads. All loads were manually reintroduced in Patran pre-processing. A single design was sized for material failure and local buckling using Airbus' ZORRO structure sizing tool. Thickness values were smoothed to a maximum difference between neighbouring elements of 15%.

Stiffeners were simulated using a bending stiffness correction in order to prevent excessive deformation of mesh elements. ZORRO proved partially incompatible with this correction. This combined with unavailibility of FEM-to-engineered mass correction factor resulted in a flawed mass estimation. The estimated FEM mass shows discrepancies greater than 50% with respect to design mission parameters as well as to that found by Benad. Alternative corrective methods were investigated, these proved incompatible with the current functionalities of Para*Py* .

Resulting element thickness and stress values are indicative for load propagation. From these, several qualitative observations were made of the structure concept behaviour. The envisioned primary structure did not sufficiently attract and carry loads. The main load path instead runs from the outboard wing box towards the root profile trailing edge. Exclusion of a majority of the non-fuselage root structure concentrated loads. Moreover, it reduced bending stiffness of the model in the symmetry plane. This is necessary to resist longitudinal global deformation with respect to the symmetry plane under braking.

The aerodynamic and pressurisation loading reinforce each other's effect on tensile and compressive residual forces in the cabin structure members. Floor loading by payload inertia aggravates this further. It was found an ultimate pressurisation load case is more critical than the combined loading.

Integration of the oval cabin concept into the Flying V nose section pushes the boundaries of the concept working principles. The combination of long span and short crown and keel heights results in large crown and keel arc curvatures. This effectuates high stress values and thick skins. An optimisation combining structure and aerodynamic efficiency will likely significantly change the aerodynamic parameters for the planform, compared to those established by Faggiano.

8.1. Future Research

For continuation of the Flying V research project, an updated model parametrisation and generation procedure are provided. This takes into account:

- Limitations of the current Para*Py* software version.
    - Non-planar face generation.
    - Limited functionality of the BDFWriter exporting the mesh to Patran.
- Limitations of ZORRO.
- Requirements for model generation:
    - relatively easy implementation of additional components such as ribs,
    - geometric continuity and equivalence of the various components in order to sew components into a single model.
- Incorporation of the model in different types of analyses.

Setting up a new version of the model generator, enables investigation of different component structure concepts as well as a coupling of different analyses. For the structure concept, it is especially worthwhile to investigate new concepts of the oval fuselage and integration thereof in a lift-generating body.

Furthermore, efficient methods of balancing the structure must be explored, as it was observed that the CoG is positioned further forward than was assumed by Faggiano. For the new CoG, the planform might not be as efficient as was shown. In order to arrive at a reliable mass estimation through an FEA method, all primary structure elements must be modelled and a FEM-to-engineered mass correction factors must be obtained. For this, representative reference structures are needed.

In order to arrive at a complete sizing method, design loop automation is a hard requirement. In order to link the necessary aerodynamic, structure and performance analyses, Para*Py* 's Nastran coupling must be matured and the analyses modules must be integrated in a new model generator. Design automation consequently enables design multidisciplinary optimisation. With such a model generator and optimiser in place, conclusive design feasibility and optimisation research for the Flying V concept can be conducted.

Bibliography

[1] K. Bender, *Airbus FPO preliminary wing design expert, interview,* (2017), not recorded.

[2] Alcoa, *Alcoa material property data,* http://www.matweb.com.

[3] J. Benad, *Design of a commercial aircraft for high-subsonic speed as a flying wing*, Tech. Rep. (Airbus GmbH, 2015) public report; sensitive information censored.

[4] J. Benad, *The Flying V: An entirely new aircraft configuration,* Presentation at Technische Universität Berlin: Seminar für Mechanik, Systemdynamik und Reibungsphysik (2015), first public presentation of configuration. Accessed October 2015 through http://www.jbenad.com/news/2015/2/4/flyingv.

[5] M. Roelofs, *Semi-Analytical Composite Oval Fuselage Mass Estimation*, Master's thesis, Delft University of Technology (2016).

[6] R. Schmidt, *A Semi-Analytical Weight Estimation Method for Oval Fuselage in Novel Aircraft Configurations*, Master's thesis, Delft University of Technology (2013).

[7] R. Martinéz-Val, E. Perez, and J. Roa, *Optimization of planform and cruise conditions of a transport flying wing,* Proc. IMechE **224**, 1243 (2010).

[8] R. Liebeck, *Design of the Blended Wing Body Subsonic Transport,* Journal of Aircraft **41**, 10 (2004).

[9] R. Martinéz-Val, *Flying Wing. A New Paradigm for Civil Aviation,* Arta Polytechnica **47**, 32 (2007).

[10] R. Martinéz-Val, J. Palacin, and E. Perez, *The evolution of jet airliners explained through the range equation,* Proc. IMechE **222.G: J. Aerospace Engineering**, 915 (2008).

[11] A. Bolsunovsky, N. Buzoverya, B. Gurevich, V. Denisov, A. Dunaevsky, L. Shkadov, O. Sonin, A. Udzhuhu, and J. Zhurihin, *Flying wing — problems and decisions,* Aircraft Design **4**, 193 (2001).

[12] R. Martinéz-Val, E. Perez, P. Alfaro, and J. Perez, *Conceptual design of a medium size flying wing,* Proc. ImechE **222.G: J. Aerospace Engineering**, 57 (2007).

[13] E. Torenbeek, *Synthesis of Subsonic Airplane Design: An Introduction to the Preliminary Design of Subsonic General Aviation and Transport Aircraft with Emphasis on Layout, Aerodynamic Design, Propulsion and Performance* (Springer, 1982).

[14] V. Denisov, A. Bolsunovsky, N. Buzoverya, and B. Gurevich, *Recent Investigations of the Very Large Passenger Blended-Wing-Body Aircraft,* 21st ICAS Congress (1998).

[15] V. Li and A. Velicki, *Advanced PRSEUS Structural Concept Design and Optimization,* 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference (2008).

[16] V. Mukhopadhyay and J. Sobieszczanski-Sobieski, *Analysis, Design and Optimization of Noncylindrical Fuselage for Blended-Wing-Body Vehicle,* Journal of Aircraft **41**, 925 (2004).

[17] R. Vos, F. Geuskens, and M. Hoogreef, *A New Structural Design Concept for Blended Wing Body Cabins,* 53rd Structures, Structural Dynamics and Materials Conference , . (2012).

[18] M. Hoogreef, *The Oval Fuselage - A New Structural Design Concept for Blended Wing Body Cabins*, Master's thesis, Delft University of Technology (2012).

[19] R. Vos and M. Hoogreef, *Semi-Analytical Weight Estimation Method for Fuselage with Oval Cross-Section,* 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference (2013).

[20] M. Ardema, M. Chambers, A. Patron, A. Hahn, H. Miura, and M. Moore, *Analytical Fuselage and Wing Weight Estimation of Transport Aircraft*, Technical Memorandum 110392 (National Aeronautics and Space Administration, 1996).

[21] J. Roskam, *Airplane Design Part V: Component Weight Estimation*, Airplane Design (Roskam Aviation and Engineering Corporation, 1989).

[22] D. Howe, *Aircraft Loading and Structural Layout*, Aerospace Series (Professional Engineering Publishing, London, 2004).

[23] P. Eustace, *Analytical method for concept aircraft structural weight and balance prediction,* AIAA **98**, 1023 (1998).

[24] A. Elham, *Weight Indexing for Multidisciplinary Design Optimization of Lifting Surfaces*, Ph.D. thesis, Delft University of Technology (2013).

[25] V. Mukhopadhyay, *Blended-Wing-Body (BWB) Fuselage Structural Design for Weight Reduction,* 46th AIAA Structures, Structural Dynamics and Materials Conference (2005).

[26] E. Carrera, L. Mannella, G. Augello, and N. Gualtieri, *A two-level optimization feature for the design of aerospace structures,* Proc. Instn Mech Engrs **217**, 189 (2003).

[27] M. Nawijn, M. van Tooren, J. Berends, and P. Arendsen, *Automated Finite Element Analysis in a Knowledge Based Engineering Environment,* 44th AIAA Aerospace Sciences Meeting and Exhibit (2006).

[28] G. La Rocca and M. van Tooren, *A Knowledge Based Engineering Approach to Support Automatic Generation of FE Models in Aircraft Design,* 45th AIAA Aerospace Sciences Meeting and Exhibit (2007).

[29] G. La Rocca and M. van Tooren, *Knowledge-based engineering to support aircraft multidisciplinary design and optimization,* Proc. IMechE **224.G: J. Aerospace Engineering**, 1041 (2010).

[30] R. van Dijk, *ParaPy 1.0.7,* http://www.parapy.nl (2013).

[31] B. Kulfan, *Universal Parametric Geometry Representation Method,* Journal of Aircraft **45**, 142 (2008).

[32] R. Schmidt and R. Vos, *A Semi-Analytical Weight Estimation Method for Oval Fuselages in Conventional and Novel Aircraft,* AIAA SciTech 52nd Aerospace Sciences Meeting (2014).

[33] M. Niu, *Airframe Structural Design – Practical Design Information and Data on Aircraft Structures* (Conmilit Press Ltd, 1988).

[34] Anonymous, *Certification Specifications for Large Aeroplanes: CS-25, Amendment 3,* (2007).

[35] D. , *Aircraft Initiator V2.8,* (2015).

[36] A. unknown, *Mesh quality,* https://www.rocscience.com/help/phase2/webhelp9/phase2_model/Mesh_Quality.htm (Year unknown), accessed on 24.12.2016.

[37] D. Howe, *Blended wing body airframe mass prediction,* Proc. Instn Mech Engrs **215**, 319 (2001).

[38] T. Megson, *Aircraft Structures for Engineering Students* (Elsevier Science & Technology, 2007).

[39] F. Faggiano, *Aerodynamic Design Optimization of a Flying V Aircraft*, Master's thesis, Delft University of Technology (2016).

[40] Jane's, *Jane's Aero-Engines,* https://janes.ihs.com (2015), accessed on 12 December, 2015.

A

Assumed and Estimated System Masses

In this appendix, various mass-related values used in the Flying V design are listed in Section A.1. The empirical relations of Roskam's method used to estimate system masses are provided in Section A.2.

A.1. Design Masses

Design masses are the global masses used for preliminary sizing and analysis. The MTOM and payload mass are loosely based on those of the A350-900. The fuel mass has been approximately determined from the Breguet range equation, using the A350 typical mission range, the L/D determined by Faggiano [39] and the specific fuel consumption of the engine [40].

Table A.1: Design masses.

Type	**Abbreviation**	**Mass *[kg]***
Maximum take-off mass	MTOM	260000
Payload mass	MPL	60000
Fuel mass	MF	70000
		-
Operational empty mass	OEM	130000
Combined system mass	CSM	30219
		-
Structure mass	MS	99781

A.2. Empirical System Weights

Landing Gear Weight

Toorenbeek method for wing mounted main gear and fuselage mounted nose gear for low-wing aircraft with retractable gears. The wing position affects the gear length, a low-wing configuration is deemed most representable for Flying V.

$$W_{g,nose} = 20 + 0.10 \cdot MTOW^{0.75} + 2 \cdot 10^{-6} \cdot MTOW^{1.5} \tag{A.1}$$

Table A.2: System masses.

System		Mass *[kg]*
Engine	W_e	7372
Auxiliary power unit	W_{apu}	650
Nose gear	$W_{g,nose}$	1347.7
Main gear	$W_{g,main}$	7205.5
Instrumentation, avionics, electronics	W_{IAE}	480.3
Paint	W_{pt}	390
Electrical systems	W_{els}	113.5
Onboard baggage and cargo handling	W_{bc}	61.1
Hydraulic systems	W_{hydr}	780
Oxygen systems	W_{oxy}	29
Flight control systems	W_{fc}	470.3
Climate and de-icing systems	W_{api}	67.0
Cabin furnishing	W_{fur}	11222.8
		-
Total system mass		30219

$$W_{g,main} = 40 + 0.16 \cdot MTOW^{0.75} + 0.019 \cdot MTOW + 1.5 \cdot 10^{-5} \cdot MTOW^{1.5} \tag{A.2}$$

Power plant Weight

The power plant weight is split up in weights of the engine, air induction system, fuel system and propulsion system. The propulsion system in turn compromises all control systems and provisions for the propulsion, except for the engine system itself.

$$W_{pwr} = W_e + W_{ai} + W_{fs} + W_p \tag{A.3}$$

Engine

Roskam provides no estimation of engine weight using known parameters. Instead, engine weight is related to the required take-off thrust, an unknown parameter. As the Flying V is in the same size class as the A350-900 [3], the engines of the A350 will be used as reference with data taken from the Jane's database [40]. These engines are rated at $T = 374kN = 84,000lbs$ of thrust.

$$W_{e,A350} = 7,277kg = 16,042lb \tag{A.4}$$

Air Induction System

The engine is assumed externally podded, as is suggested by Benad [3]. Additionally, this simplification allows for easier structural sizing by introducing point forces where the engine is mounted to the Flying V. Roskam tells this weight is included in the nacelle weight, which in turn is already accounted for in the engine weight specification provided by Jane's.

Fuel System

K_{fsp} is constant factor taking into account type of fuel. $K_{fsp} = 5.87lbs/gal$ for aviation gasoline and $K_{fsp} = 6.55lbs/gal$ for JP-4. For an integral fuel tank such as that assumed for the Flying V, the fuel system weight is given as:

$$W_{fs} = 80 \cdot (N_e + N_t - 1) + 15 \cdot \sqrt{N_t} \cdot \left(\frac{W_F}{K_{fsp}}\right)^{0.333} \tag{A.5}$$

Propulsion System

The Torenbeek method provided by Roskam uses several unknown parameters in the weight estimation of the propulsion system. As such, the GD method will be used for estimating this weight.

$$W_p = 9.33 \left(W_{e,total} \cdot 10^{-3}\right)^{1.078} \tag{A.6}$$

Fixed Equipment Weight

Flight Control System

$$W_{fc} = K_{fc} \cdot MTOW^{0.667} \tag{A.7}$$

in which $K_{fc} = 0.64$ for powered flight controls.

Hydraulic/Pneumatic System

Roskam provides a range of values from 0.006 to 0.0120 for the multiplier. Howwever, assuming technological progress and relatively few hydraulically powered surfaces (Benad leaves out spoilers, flaps etc), a multiplier value of 0.006 is assumed appropriate.

$$W_{hydr} = 0.006 \cdot MTOW \tag{A.8}$$

Electrical System

$$W_{els} = 10.8 \cdot V_{pax}^{0.7} \cdot \left(1 - 0.018 \cdot V_{pax}^{0.35}\right) \tag{A.9}$$

in which V_{pax} is the pressure cabin volume in ft^3

Instrumentation, Avionics and Electronics

$$W_{iae} = 0.575 \cdot OEW^{0.556} \cdot R^{0.25} \tag{A.10}$$

Air-conditioning, Pressurisation, De-icing Systems

$$W_{api} = 6.75 \cdot l_{pax}^{1.28} \tag{A.11}$$

in which l_{pax} is the pressure cabin length in ft.

Oxygen System

The weight of the oxygen system depends on the service ceiling as well as the overwater flight duration. The Flying V is expected to have a service ceiling of FL350, similar to the A350. Additionally, it will be capable of performing trans-ocean flights – however, it is unclear whether this qualifies as an *extended overwater flight* such as introduced by Roskam (in the Torenbeek method). Both relations are provided below.

$$W_{ox,+FL250} = 30 + 1.2 \cdot N_{pax} \tag{A.12}$$

$$W_{ox,ex.wtr} = 40 + 2.4 \cdot N_{pax} \tag{A.13}$$

Auxiliary Power Unit

Roskam suggests using manufacturer data for the auxiliary power unit if possible. However, as no APU has been defined for the Flying V, the relation between APU weight and MTOW will be used.

$$W_{apu} = 0.005MTOW \tag{A.14}$$

Furnishings

Furnishings include seats, insulations, panels, sound proofing, instrument panels, control stands, lighting, wiring, the pantry structure and provisions, the lavatory and associated systems, overhead luggage compartments, wardrobes, escape provisions and fire fighting equipment. These are widely spaced throughout the fuselage, however, Roskam provides no detailed information on location nor segregation of weight for each of these parts. As such, it will be assumed the weight is uniformly distributed throughout the pressure hull.

$$W_{fur} = 0.211\,(MTOW - W_F)^{0.91} \tag{A.15}$$

Baggage and Cargo Handling Equipment

$$W_{bc} = 3 \cdot S_{ff} \tag{A.16}$$

in which S_{ff} is the freight floor surface area in ft^2. Additionally, cargo containers may be assumed to have a density of $1.6lbs/ft^3$.

Paint

$$W_{pt} = 0.003 \cdot MTOW \tag{A.17}$$

B

Structure Generation Process Flow for Numerical Sizing

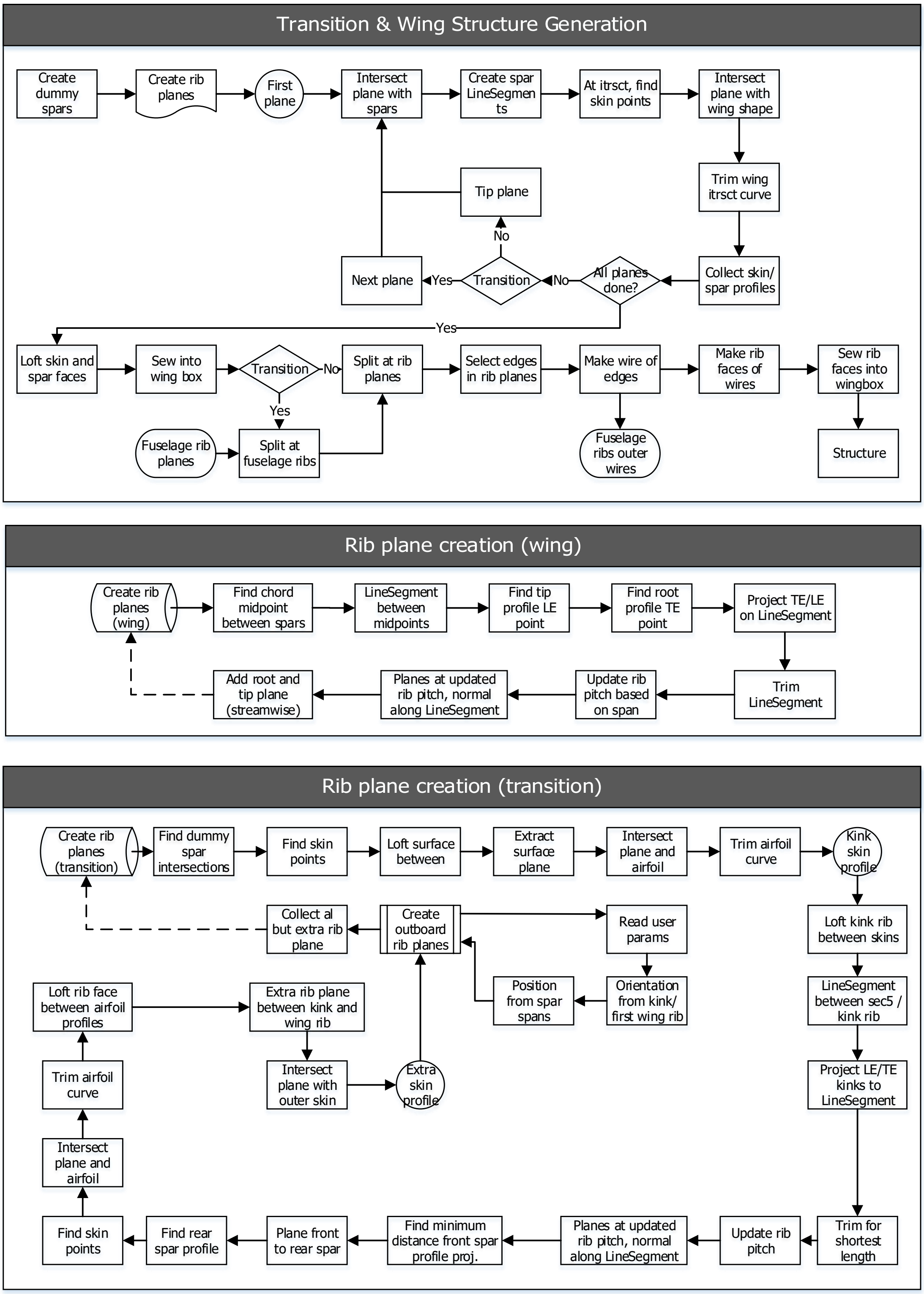


Figure B.1: Overview of wing structure generation process.

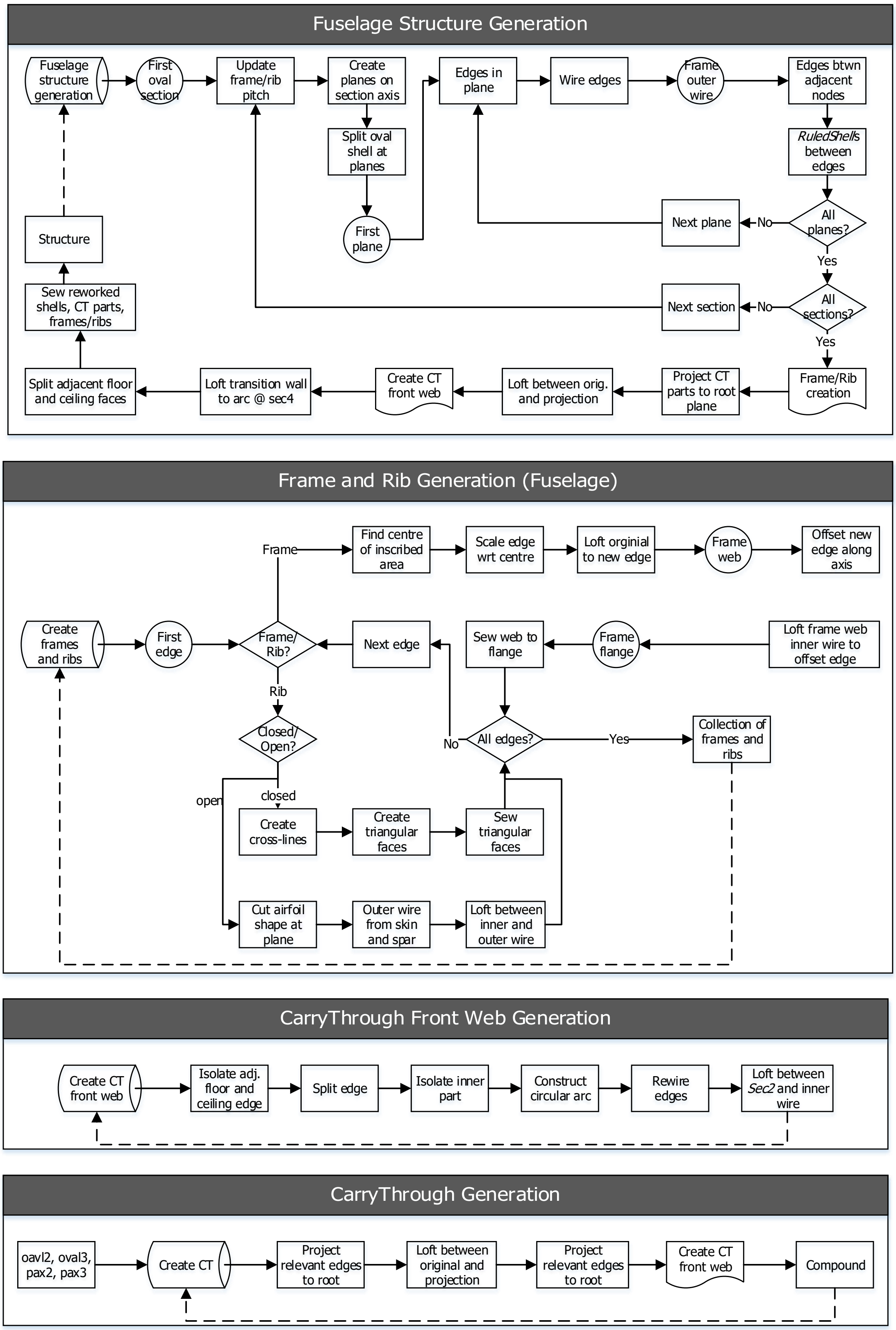


Figure B.2: Overview of fuselage structure generation process.

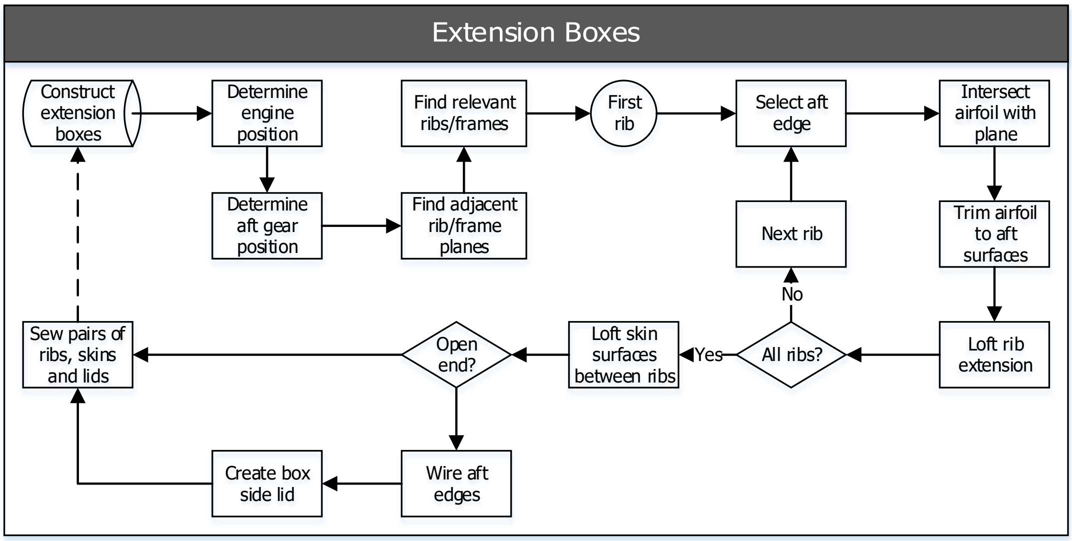


Figure B.3: Overview of rib extension boxes generation process.

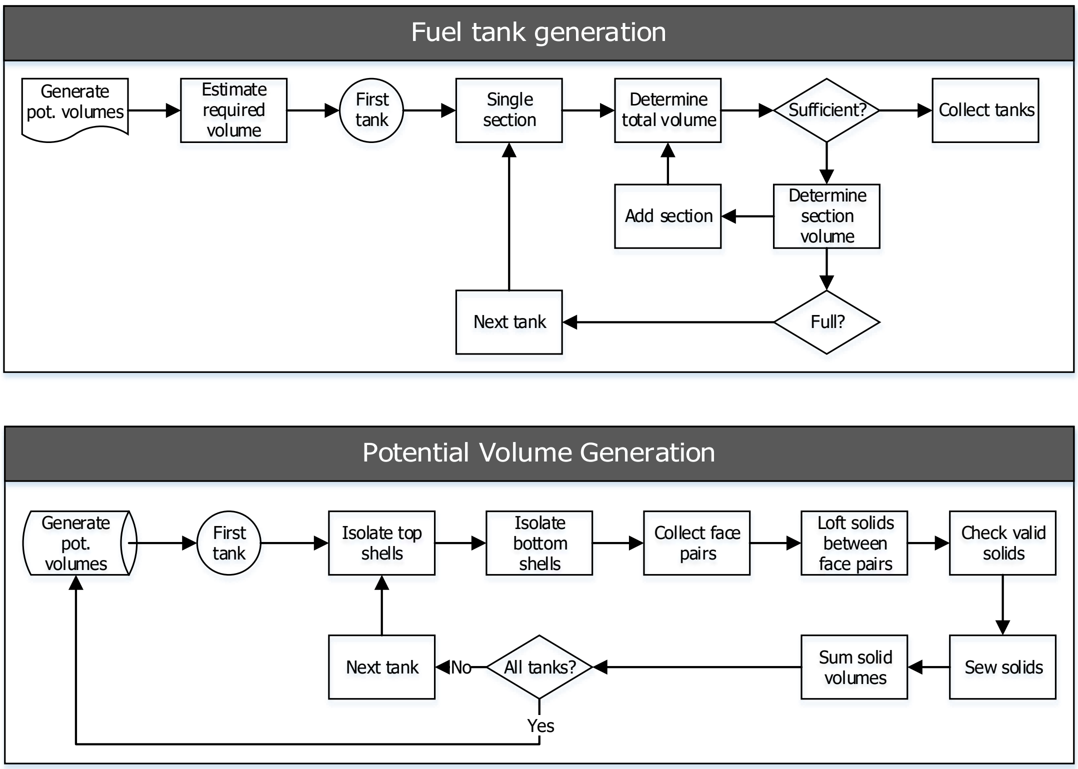


Figure B.4: Overview of fuel tank generation process.

C

Transverse Loads Application Infeasible Solutions

In Chapter 5, it is observed that transverse load application to the thin shell model causes extreme deformation of the shells. This is due to absence of bending stiffness, usually provided by stiffeners. Several potential solutions which can be used to solve the arising problems are identified:

- Add stiffeners to the model.
- Thicken the transversely loaded shells.
- Model shells as foam core plates consisting of three shells.
- Add rigid bar elements between each of the shell mesh nodes.
- Add beam elements between each of the shell mesh nodes.
- Apply an extremely large shell thickness.
- Increase the shell material equivalent bending stiffness.
  - Isotropic material
  - Orthotropic material

Addition of Stiffeners

Addition of stiffeners can be accomplished in similar fashion as with the frames and will limit the global deformations. Local deformations may still be large on mesh elements not connected to a stiffener. In order to generate stiffeners, on-the-shell profiles must be extracted from which a web face is lofted, which can subsequently be used to loft a flange face. Initial observations in Patran post-processing of the frame behaviour under loading shows that such shell constructions provide the desired additional stiffness.

Geometric equivalence constraints were adhered in the model. This proved particularly important for the generation of rib and frame faces. A different method of face creation functioned by splitting the existing structure along planes and extracting the edges in the plane. This could be done because the frames and ribs cover the full width of the primary structure. Stiffeners on the other hand are oriented along various axes on various shells. For these, splitting of the full structure is not an option. Alternatively, the stiffeners can be added in the individual fuselage and wing sections by splitting the respective sections along planes. This however still splits all faces intersecting the plane, creating more faces and thus increasing mesh fineness.

A best effort can be made by redefining the profiles at top-level to include vertices at the stringer locations. When a shell is then lofted between them, the stiffener edges are automatically generated *along the face topology*. The wing profiles are constructed from intersections with the outer shape, which have a variable amount of variables. Besides the rib edges, no further geometry is dependent on these profiles. The floor profiles are generated simultaneously with the frame wires. By splitting these using a *SplitCurve* operation, stiffeners can be accounted for. However, the surfaces of the keel fuel tanks as well as fuselage transition interface profiles will change geometrically, which implies every operation using these must be altered. An overview of a redesigned model generation procedure, taking geometric equivalence and continuity of elements into account, is presented in Chapter 7.

The oval shell profiles are at the root of the entire model. They are currently constructed by creating *Point*s along the various arcs and generating a *FittedCurve* through this. To provide better options for stiffener creation, this single curve must be remodelled. The several arcs are generated using Para*Py* 's *Arc* object. These are split at certain intervals, depending on the amount of desired stiffeners, using *SplitCurve*. The splitted arcs are then rewired to provide the top-level oval fuselage profiles with which modelling is started. Implementing this solution requires a revision of the entire model, specifically concerning edge selections. This exceeds time constraints set for this particular project.

Thickening of Transversely Loaded Shells

Para*Py* provides a *ThickShell* operation which may be applied to the transversely loaded shells. In order to accomplish structurally sound connections, all shells must be thickened. The full shell model is too complex for the *ThickShell* operation to handle. Individual thickening requires the correct direction to be selected for each face. Given the differing nature of face orientations, this may prove difficult. The most feasible option is to include a multi-dimensional axis through the structure, toward which all faces are thickened. This will likely result in singularities for faces of which the normal is parallel to this axis locally.

The thick shells will also present difficulties in meshing the structure. Currently, face groups are selected based on the intermediate shells created during construction. By fusing thick shells, the final model faces will be different from the shell faces such that the hypotheses presented in Section 4.2.5 no longer return the corresponding face pairs.

Lastly, although Nastran can analyse 3D-object meshes, Para*Py* has not yet included the relevant cards for these. As such, thick geometry cannot be captured in the .bdf-file.

Shell Collection Representing Thick Plate

In order to circumvent Para*Py* 's inability to export thick shells, an effort can be made to approximate thick shells with multiple shells. By offsetting these and connecting them rigidly, the bending stiffness is increased. A further approximation is made by introducing a third, inner shell of different material to approach a foam-core composite plate. This method is limited by Para*Py* 's *BDFWriter* module, as Nastran's *RBAR3* card and multiple material definition are not supported.

Introduce Rigid Bars between Nodes

Instead of applying rigid bars to a connection between two shells, the nodes of the shell mesh may instead be connected using rigid bars. These are laid over the existing elements and prevent the absolute distance between nodes to change, creating in effect an infinitely stiff shell. The downside of this is twofold:

1. The infinitely stiff bar will likely attract more loads, ZORRO will keep increasing its thickness.
2. Shell deformation is not captured and therefore global buckling checks cannot be performed.

These two limitations, coupled with the unsupported *RBAR3* elements, mean this solution is not feasible.

Introduce Beam Elements between Nodes

Instead of rigidly connecting all the nodes as discussed above, the nodes may also be connected using finitely-stiff beam elements. This effectively remodels stiffeners purely into the mesh, instead of into the entire model. Their respective stiffness can thus be taken directly from the proposed stiffeners. ZORRO cannot size these beam elements and will forego them. Additionally, Para*Py* does not support Nastran's *CBEAM* card, such that all beam elements will have to be modelled manually. This solution is therefore not chosen.

Increase Shell Thickness

Instead of introducing a volume from the shell in the model, the shell thickness may be increased in the mesh data. This combined with the isotropic material attribution results in invalid displacements caused through bending moments to be calculated by Nastran. This makes a non-linear analysis not possible. To offset these specific out-of-plane displacements, the material's Young's modulus can be altered. This would however alter the in-plane strain. Therefore, this solution is invalidated.

Increase Equivalent Bending Stiffness

Continuing on the previous solution, Nastran provides the option to alter the bending stiffness specifically by introducing an equivalent Young's modulus without changing the material's true Young's modulus. This approach works as bending stress and moment are linear with the Young's modulus through the flexure formula in Equation C.1 and bending deformation approximation in Equation C.2 respectively.

$$\sigma = \frac{My}{I} \tag{C.1}$$

$$M = -EI\frac{\partial^2 \phi}{\partial x^2} \tag{C.2}$$

The downside of this approach is that it can only be applied to quadrilateral mesh elements. In the fuselage and wing sweep transitions, triangular elements were applied initially. However, the manual re-mesh of the fuselage transition features quadrilateral elements. Therefore, this solution can be applied to all elements of shells which are transversely loaded. The equivalent bending stiffness is chosen such that excessive deformation is constrained to expected extent. The equivalent Young's modulus value is set to $1 \cdot 10^{13} N/m^2$ for all QUAD elements.

As the direction of the elements is not determined, an isotropic material is still applied. The equivalent bending stiffness therefore acts isotropically. This approximation is not a true representation of the pursued aim to model stiffener-contributed stiffness. However, it prevents excessive deformation of the shells under transverse loading. Because the displacements are still considered a realistic representation, a non-linear analysis is still possible.

After the mesh was checked for a $2\Delta p$ load case combined with floor inertial loading, it became clear that the floor and ceiling shells required an even higher stiffness. This was modelled by assigning orthotropic material properties, a $10cm$ thickness and lower density to these shells. With this, an approximation of a honeycomb structure was attempted.

ZORRO showed various limitations upon implementation of the equivalent bending stiffness workaround:

- Equivalent bending stiffness was not written to the new .bdf-file.
- Shell inner and outer faces are spaced far apart from the shell. Principal stresses used by ZORRO are higher than true.

An experimental version of ZORRO is developed in order to address the encountered problems. The latter was solved by using the average of the principal stresses. Due to this inclusion, elements which are simultaneously subjected to in-plane normal and shear forces may not be sized correctly. The readily available Von Mises stress could not be used due to omission of difference between tension and compression. This distinction is required for the different compressive and tensile yield stress. Averaging the principal stress magnitudes and assigning a sign based on the largest magnitude does not decrease the magnitude to true values.

D
Code Documentation

D.1. Code Structure

The model generation is broken into several sections. The code is broken into several parts due to a mid-project update of Para*Py* and following experimental approaches. As it is recommended to implement a new modelling approach, the code structure was not updated.

The relevant modules of the code are captured in a UML diagram:

- Hybrid planform geometry generation.
- Internal structure geometry generation.
- Implementation of mesh automation.
- Writing of the point forces to .txt-files Patran inclusion.

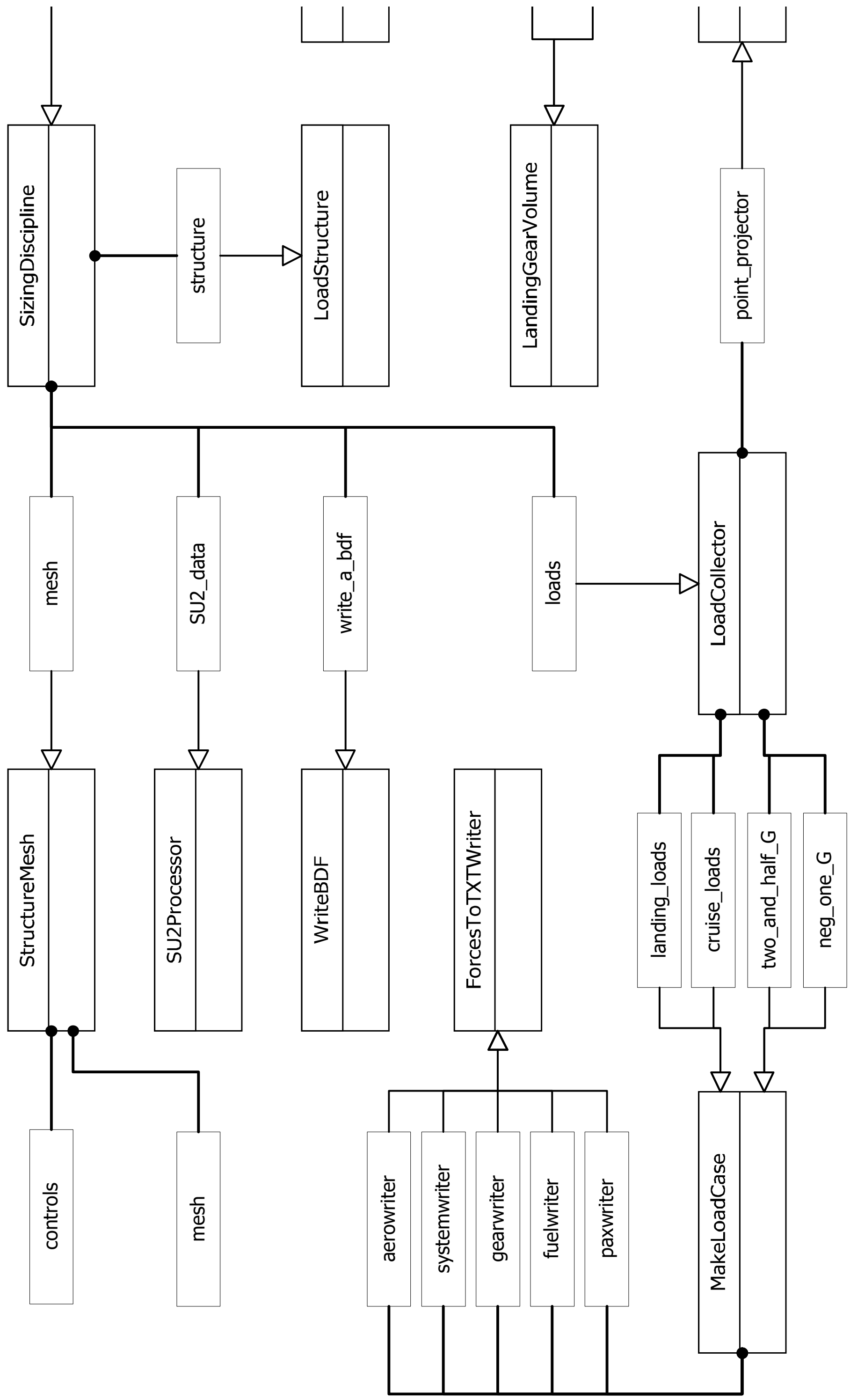
The UML diagram is provided in Figure D.1. Many objects in the diagram have been intentionally left out to simplify the diagram. Their presence follows from the modelling procedure described in Appendix B.

D.2. Identified Higher Level Primitives

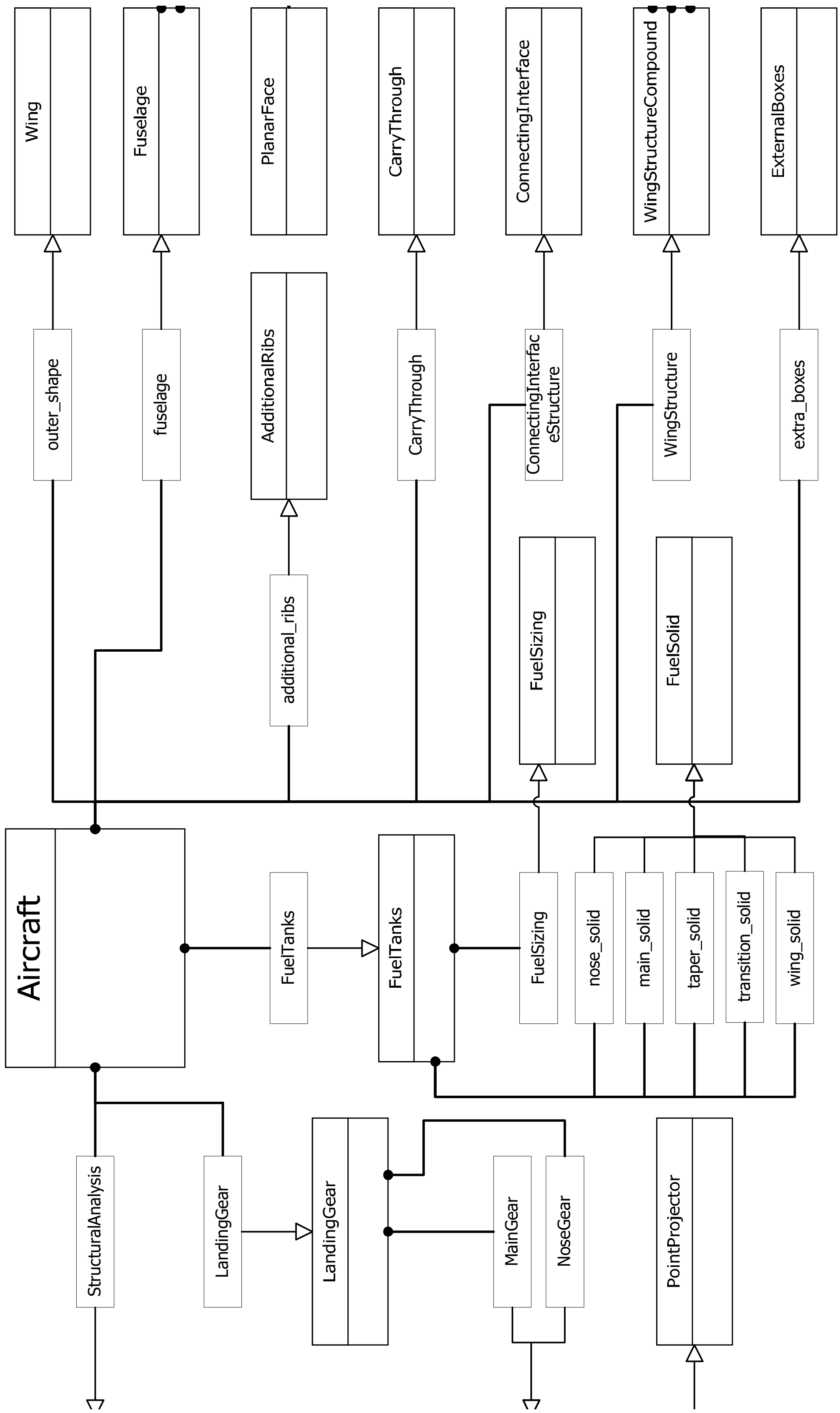
Several HLPs are identified in the existing model. Their structure is presented in Tables D.1 through D.6. Due to a mid-development Para*Py* update, several methods which are now outdated were implemented. These are omitted from the HLP descriptions.

D.3. User Input File

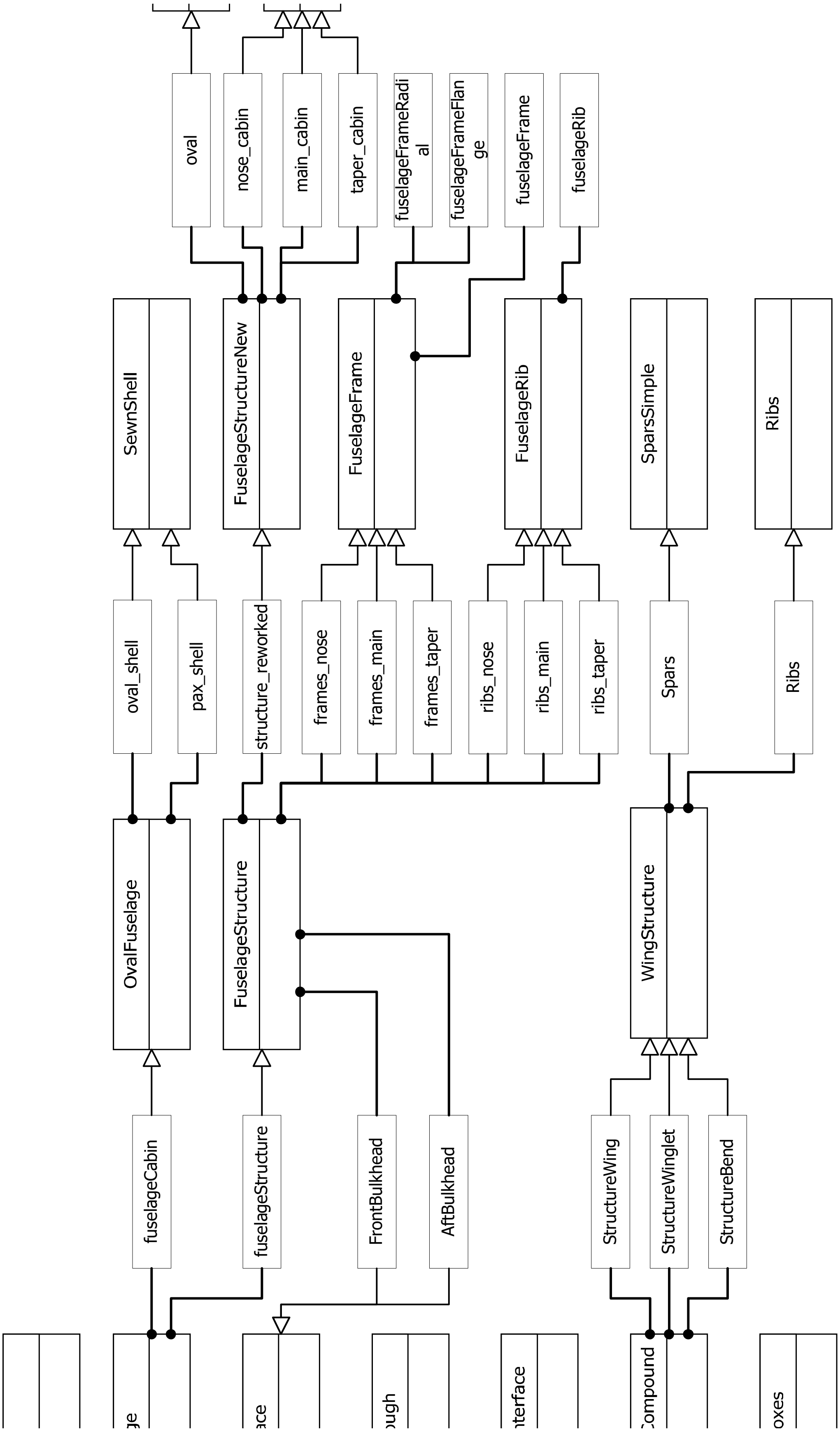
User input parameters are passed to the tool by a .txt-file. Parameters can be edited either through the .txt-file values, or in the Para*Py* graphical user interface. The available inputs are given in Table D.7.



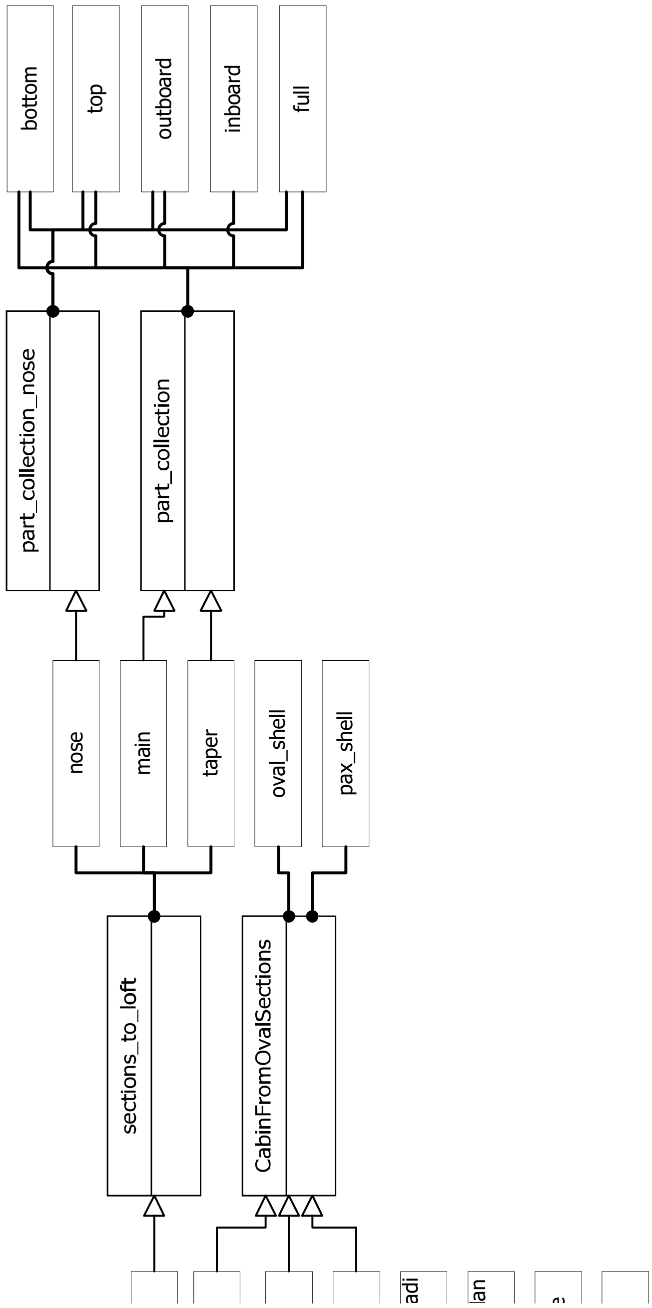
(a) Page 1 of Flying V UML diagram.



(b) Page 2 of Flying V UML diagram.



(c) Page 3 of Flying V UML diagram.



(d) Page 4 of Flying V UML diagram.

Figure D.1: Flying V UML diagram for outer geometry, primary structure geometry, mesh generation and load case force exports.

Table D.1: Fuselage section HLP.

	Name	Description
Inputs	oval	Global oval section *[LoftedShell]*
	frame_pitch	Maximum frame pitch in fuselage *[m]*
	web_scale	Scaling factor for frame webs *[-]*
	flange_width	Frame flange width *[m]*
Attributes	actual_pitch	Recalculated frame pitch *[m]*
	oval_edges	Edges in frame planes *[list of lists of Edges]*
	oval_edges_sorted	Edges sorted similarly *[list of lists of Edges]*
	oval_wires	Wires of oval_edges *[list of Wires]*
	pax_wires	Wires of pax_shell edges in frame planes *[list of Wires]*
Parts	section_axis	Section longitudinal axis *[LineSegment]*
	frame_planes	Planes at frame positions *[Plane]*
	oval_split	Oval section split at the planes *[SplitSurface]*
	oval_shell	New oval shell with only edges at frame planes *[SewnShell]*
	pax_shell	Cabin shell *[list of Faces]*
	frames	Section frames *[list of Faces]*
	full_section	Combined shell of oval, cabin and frames *[SewnShell]*

Table D.2: Oval cabin shell HLP.

	Name	Description
Inputs	oval_wires	Wires of oval section at frame planes *[list of Wires]*
Attributes	top_parts	Crown shells *[list of RuledShells]*
	outboard_parts	Outboard shells *[list of RuledShells]*
	inboard_parts	Inboard shells *[list of RuledShells]*
	bottom_parts	Keel shells *[list of RuledShells]*
Parts	oval_parts	Individual shells between wires *[list of RuledShell]*
	top	Single crown shell *[SewnShell]*
	outboard	Single outboard shell *[SewnShell]*
	inboard	Single inboard shell *[SewnShell]*
	bottom	Single keel shell *[SewnShell]*
	full	Sewn oval shell from oval_parts *[SewnShell]*

Table D.3: Passenger cabin shell HLP.

	Name	Description
Inputs	oval_shell	Full oval shell *[SewnShell]*
	frame_planes	Frame planes *[list of Planes]*
Attributes	lateral_edges	Edges of oval shell not in frame planes *[list of Edges]*
	lateral_edges_root	Only for nose. Lateral edges projected onto root *[list of Edges]*
	edges_sorted_clockwise	Lateral edges lists sorted clockwise *[list of Edges]*
	faces	Faces between the edges *[list of RuledShells]*
Parts	pax_shell	Full passenger cabin shell *[SewnShell]*

Table D.4: Frame collection HLP.

	Name	Description
Inputs	oval_wires	Oval wires in frame planes *[list of Wires]*
	pax_wires	Pax wires in frame planes *[list of Wires]*
	frame_planes	Frame planes *[list of Planes]*
	web_scale	Web scaling factor *[-]*
	flange_width	Flange width *[m]*
Attributes	outer_wires	Web outer wires *[list of Wires]*
	scale_points	Scaling reference point *[list of Points]*
	inner_wires	Web inner wires *[list of ScaledShape]*
	flange_outer_wires	Flange outer wires *[list of TranslatedShapes]*
Parts	frames	Frame instances *[sequence of Frames]*

Table D.5: Frame HLP.

	Name	Description
Inputs	outer_wire	Web outer wires *[list of Wires]*
	inner_wires	Web inner wires *[list of ScaledShape]*
	flange_outer_wires	Flange outer wire *[list of TranslatedShapes]*
Attributes		
Parts	web	Web *[LoftedShell]*
	flange	Flange *[LoftedShell]*
	full	Full frame *[SewnShell]*

Table D.6: Wing structure HLP.

	Name	**Description**
Inputs	outer_shape	Wing outer shape *[LoftedSolid]*
	root_profile	Root profile *[FittedCurve]*
	tip_profile	Tip profile *[FittedCurve]*
	spar_front_xc	Relative front spar position *[-]*
	spar_rear_xc	Relative rear spar position *[-]*
	rib_pitch	Maximum rib pitch *[m]*
Attributes	front_spar_profiles	Front spar profiles *[list of LineSegments]*
	rear_spar_profiles	Front spar profiles *[list of LineSegments]*
	faces_between_spars	Faces between spars *[list of Faces]*
	rootTE	TE of root profile *[Point]*
	tipLE	LE of tip profile *[Point]*
	wing_axis	Axis through middle of wing box *[LineSegment]*
	trimmed_axis	Trimmed axis between rootTE and tipLE *[LineSegment]*
	rib_param	Number of ribs and updated rib pitch *[-, m]*
	rib_edges	List of edges in rib planes *[list of list of Edges]*
	rib_wires	Wires of edges in rib plaes *[list of Wires]*
Parts	front_spar	Front spar *[LoftedShell]*
	rear_spar	Rear spar *[LoftedShell]*
	wing_splitted_at_spars	Wing split at extended spars *[SplitSurface]*
	wing_only_between_spars	Wing between spars *[SewnShell]*
	rib_planes	Rib planes *[list of Planes]*
	wingbox_splitted_at_ribs	Wingbox faces split at ribs *[list of SplitSurfaces]*
	rib_faces	Rib faces *[sequence of Faces]*
	structure	Wing structure *[SewnShell]*

Table D.7: Available user input parameters.

Input Group	Variable	Description
Wing Planform Variables	w_area	Wing area $[m^2]$
	w_span	Wing span $[m]$
	w_sweep_LE1	Inboard wing LE sweep $[deg]$
	w_sweep_LE2	Outboard wing LE sweep $[deg]$
	w_kink_position1	Kink 1 spanwise position $[-]$
	w_kink_position2	Kink 2 spanwise position $[-]$
	w_root_c	Wing root chord length $[m]$
	w_taper_ratio	Wing taper ratio $[-]$
	w_dihedral_in	Inboard wing dihedral $[deg]$
	w_dihedral_out	Outboard wing dihedral $[deg]$
	w_kink2_twist	Kink 2 twist angle $[deg]$
	w_tip_twist	Tip twist angle $[deg]$
Winglet Variables	winglet_A	Winglet aspect ratio$[-]$
	winglet_dihedral	Winglet cant angle $[deg]$
	winglet_sweep	Winglet LE sweep angle $[deg]$
	winglet_taper_ratio	Winglet taper ratio $[-]$
Oval fuselage parameters	noval_x	Nose fuselage sections x-position $[m]$
	noval_h2	Nose fuselage sections h_2 $[m]$
	height_up_2	h_1 of section 2 $[m]$
	height_cabin_2	h_2 of section 2 $[m]$
	height_low_2	h_3 of section 2 $[m]$
	height_up_3	h_1 of section 3 $[m]$
	height_cabin_3	h_2 of section 3 $[m]$
	height_low_3	h_3 of section 3 $[m]$
	width_floor_3	w_f of section 3 $[m]$
	height_up_4	h_1 of section 4 $[m]$
	height_cabin_4	h_2 of section 4 $[m]$
	height_low_4	h_3 of section 4 $[m]$
	width_floor_4	w_f of section 4 $[m]$
	height_up_5	h_1 of section 5 $[m]$
	height_cabin_5	h_2 of section 5 $[m]$
	height_low_5	h_3 of section 5 $[m]$

Continued on next page

Table D.7 – continued from previous page

	Variable	**Description**
	width_floor_5	w_f of section 5 $[m]$
Nose cone airfoil parameters	a_upper	Coefficient a_{up} of front cone $[\sqrt{m}]$
	a_lower	Coefficient a_{low} of front cone $[\sqrt{m}]$
	e_upper	Coefficient e_{up} of front cone $[{}^1/_m]$
	e_lower	Coefficient e_{low} of front cone $[{}^1/_m]$
Nose cone structure parameters	nose_droop	Nose point vertical offset fraction $[-]$
	control_pts_LE	LE Bezier profile additional control points $[list]$
	control_pts_root	Root Bezier profile additional control points $[list of two lists]$
Structure parameters	fuselage_frame_pitch	Maximum frame pitch $[m]$
	wing_frame_pitch	Maximum rib pitch $[m]$
	frame_web_scale	Scaling factor for frame web inner wire $[-]$
	frame_flange_width	Frame flange width [1] $[m]$
	wing_spar_front_xc	Front spar relative chord position $[-]$
	wing_spar_rear_xc	Rear spar relative chord position $[-]$
Rear airfoil parameters	c_up_root	c_{up} of section I $[1/m]$
	c_low_root	c_{low} of section I $[1/m]$
	z_TE_root	z_{TE} of section I $[m]$
	c_up_inb	c_{up} of section II $[1/m]$
	c_up_inb	c_{up} of section II $[1/m]$
	z_TE_inb	z_{TE} of section II $[m]$
	c_up_kink1	c_{up} of section III $[1/m]$
	c_up_kink1	c_{up} of section III $[1/m]$
	z_TE_kink1	z_{TE} of section III $[m]$
Outboard airfoils	w_kink2_CST	CST coefficients of section IV
	w_tip_CST	CST coefficients of section V
	data_file_name_airfoil_winglet_tip	Path to airfoil coordinates file for winglet
Engine parameters	engine_length	Engine nacelle length $[m]$
	engine_diameter	Engine nacelle diameter $[m]$
	engine_position_y	Engine relative spanwise position $[-]$
	data_file_name_airfoil_pylon	Path to airfoil coordinates file for engine pylon
	engine_sfc	Engine specific fuel consumption $[{}^{kg}/_{Ns}]$
Mission	cruise_mach	Cruise Mach $[-]$

Continued on next page

[1] Not settable through top-level input

Table D.7 – continued from previous page

	Variable	Description
	cruise_altitude	Cruise altitude $[m]$
	max_altitude	Operation ceiling $[m]$
	cabin_altitude	Cabin pressure altitude $[m]$
	takeoff_speed	Take-off velocity $[m/s]$
	DesignRange	Design range $[nm]$
Masses	MTOM	Maximum Take-Off Mass $[kg]$
	cruise_mass	Median cruise mass $[kg]$
	OEM	Operating Empty Mass $[kg]$
	MF	Fuel mass $[kg]$
	MPL	Payload mass $[kg]$
	pax_transverse_load	Passenger cabin floor loading $[N/m^2]$
	cargo_transverse_load	Cargo cabin floor loading $[N/m^2]$

Table D.8: Default input parameter values.

Parameter	Value	Parameter	Value
Wing Planform		**Winglet Planform**	
w_area	883.33	winglet_A	2.2
w_span	65.	winglet_dihedral	90.
w_sweep_LE1	64.48	winglet_sweep	35.
w_sweep_LE2	37.85	winglet_taper_ratio	0.6
w_kink_position1	0.387		
w_kink_position2	0.629	Nose Cone	
w_root_c	24.	a_upper	0.9
w_taper_ratio	0.132	a_lower	0.6
w_dihedral_in	0.	e_upper	-2.
w_dihedral_out	7.	e_lower	-1.
w_kink2_twist	-4.33		
w_tip_twist	-4.368	**Rear Airfoils**	
		c_up_root	-5.468 e-5
Oval Sections		c_low_root	-7.646 e-5
height_up_2	0.8	z_TE_root	-5.016 e-2
height_cabin_2	2.1	c_up_inb	-1.099 e-4
height_low_2	0.6	c_low_inb	1.463 e-4
height_up_3	0.6	z_TE_inb	-6.927 e-2
height_cabin_3	2.1	c_up_kink1	-1.277 e-5
height_low_3	0.7	c_low_kink1	2.302 e-4
width_floor_3	6.	z_TE_kink1	-4.201 e-2
height_up_4	0.4		
height_cabin_4	2.1	**Outboard Airfoils**	
height_low_4	0.7	w_kink2_CST	[...]
width_floor_4	6.	w_tip_CST	[...]
height_up_5	0.3	data_file_name_winglet_tip	'NACA_0012.dat'
height_cabin_5	1.8	data_file_name_airfoil_pylon	'NACA_0012.dat'
height_low_5	0.3		
		Engine Parameters	
Mission		engine_length	6.4
DesignRange	14350000	engine_diameter	3.94
cruise_mach	0.85	engine_position_y	0.1
cruise_altitude	13000	engine_sfc	1.53 e-4
takeoff_speed	77.17		
		Masses	
		MTOM	260000
		cruise_mass	200000
		OEM	130000
		MF	70000

D.4. Introduced Methods and Tools

Several useful methods and tools were introduced for the model generation of the Flying V. These may also be useful for other model generators based in Para*Py* . These rely specifically on integration of OpenCascade in Para*Py* . For other model generators, the logic may therefore be applicable, but the code itself not.

PointProjector

The PointProjector class is used to project SU2 data points on the outer shape, in order to obtain a unit vector for the local pressure force. This may be adapted to project any list of points onto any shape.

```
class PointProjector(GeomBase):
        points = Input()
        pressure_magnitude = Input()

        @Attribute
        def outer_shape(self):
                return self.parent.parent.parent.outer_shape

        @Attribute(in_tree=True)
        def wing_solid(self):
                return self.outer_shape.WingSolid
        @Attribute(in_tree=True)
        def wing_edges(self):
                return self.wing_solid.edges

        @Attribute
        def wing_faces(self):
                return sorted(self.wing_solid.built_from[1:-1], key=lambda f:f.uv_center_point
                    .y)

        @Attribute
        def point_pressure_couples(self):
                lst = []
                for i, point in enumerate(self.points):
                        lst.append((point, self.pressure_magnitude[i]))
                return lst

        @Attribute
        def forces(self):
                lst = []
                points=self.point_pressure_couples
                for point, pressure in points:
                        point2 = Point(point.x, point.y, point.z+0.01)
                        line = LineSegment(point, point2)
                        distance = []
                        for face in self.wing_faces:
                                u, v = face.parameter(point)
                                if face.point(u, v).distance(point) < 1e-3:
                                        normal = face.normal(u, v)
                                        break
                        force = pressure * normal
                        force = Vector(force[0], force[1], force[2])
                        lst.append((force, face.point(u, v)))
                return lst
```

run_AVL.read_output_elements

A method was added to the run_AVL class to read element-wise output.

```
def read_output_elements(filename, panels_spanwise, panels_chordwise):
        print "Reading AVL element-wise output..."
        with open(filename,'r') as f:
                strip_width = []
                strip_LE_pos = []
```

```
            strip_dihedral = []
            element_pos = []
            element_dx = []
            element_slope = []
            element_dCp = []
            headers = []
            data = [[],[],[],[],[],[],[]]
            for line in f:
                    values = line.split()
                    if line.lstrip().startswith('Xle'):
                            strip_x = float(line.split()[2])
                    elif line.lstrip().startswith('Yle'):
                            strip_y = float(line.split()[2])
                            strip_width.append(float(line.split()[6]))
                    elif line.lstrip().startswith('Zle'):
                            strip_z = float(line.split()[2])
                            strip_dihedral.append(float(line.split()[6]))
                            strip_LE_pos.append(Point(strip_x, strip_y, strip_z))
                    elif line.lstrip().startswith('I') and headers == []:
                            headers = line.split()
                    elif len(values) == len(headers) and not values == headers:
                            for i in range(len(values)):
                                    if values[i] == '***':
                                    break
                            elif i == 0:
                                    data[i].append(int(values[i]))
                            else:
                                    data[i].append(float(values[i]))
    f.close()
    print "    Retaining AVL Surface 1 (starboard) data points. Removing AVL Surface 2 (
        port) data points..."
    strip_data = [strip_LE_pos, strip_width, strip_dihedral]
    for i in range(len(strip_data)):
            del strip_data[i][panels_spanwise:]
    for i in range(len(data)):
            del data[i][panels_spanwise*panels_chordwise:]
    element_data = data
    return [element_data, strip_data]
```

find_corresponding_edges

This method was used to select edges in a structure based on previously collected edges.

```
def find_corresponding_edges(new_edges, old_edges,margin=0.0001):
        corresponding_edges = []
        for edge in old_edges:
                mid = edge.midpoint
                pt1 = edge.start
                pt2 = edge.end
        for index, shape_edge in enumerate(new_edges):
                if edge == shape_edge:
                        corresponding_edges.append(shape_edge)
                        break
                elif shape_edge.midpoint == mid and ((shape_edge.start == pt1 and shape_edge.
                    end == pt2) or (shape_edge.start == pt2 and shape_edge.end == pt1)):
                        corresponding_edges.append(shape_edge)
                        break
                else:
                        mid_diff = mid - shape_edge.midpoint
                        pt1_diff = pt1 - shape_edge.start
                if pt1 - shape_edge.end < pt1_diff:
                        pt1_diff = pt1 - shape_edge.end
                        pt2_diff = pt2 - shape_edge.start
                else:
                        pt2_diff = pt2 - shape_edge.end
                d = [abs(el) < margin for el in mid_diff]
                hypo1 = True if sum(d) == 3 else False
                d = [abs(el) < margin for el in pt1_diff]
                hypo2 = True if sum(d) == 3 else False
```

```
            d = [abs(el) < margin for el in pt2_diff]
            hypo3 = True if sum(d) == 3 else False
            if hypo1 and hypo2 and hypo3:
            corresponding_edges.append(shape_edge)
        return corresponding_edges}
```

find_corresponding_faces

This method was used to select faces in the final structure based on faces collected in intermediate construction phases.

```
def find_corresponding_faces(new_shape, old_faces, margin=1e-3):
        corresponding_faces = []
        if not isinstance(new_shape, list):
                face_lst = new_shape.faces
        else:
                face_lst = new_shape
        for face in old_faces:
                normal = face.uv_center_normal
                point = face.uv_center_point
                area = face.area
        for indfex, shape_face in enumerate(face_lst):
                if face.uv_center_normal != shape_face.uv_center_normal:
                        nm_diff = face.uv_center_normal - shape_face.uv_center_normal
                else:
                        nm_diff = [0., 0., 0.]
                d = [abs(el) < margin for el in nm_diff]
                hypo1 = True if sum(d) == 3 else False
                if not hypo1:
                        nm_diff = face.uv_center_normal - -1 * shape_face.uv_center_normal
                        d = [abs(el) < margin for el in nm_diff]
                        hypo1 = True if sum(d) == 3 else False
                pt_diff = face.uv_center_point - shape_face.uv_center_point
                c = [abs(el) < margin for el in pt_diff]
                hypo2 = True if sum(c) == 3 else False
                hypo3 = True if abs(shape_face.area - area) < area/50. else False
                if hypo1 and hypo2 and hypo3 and shape_face not in corresponding_faces:
                        corresponding_faces.append(shape_face)
        return corresponding_faces
```

edges_in_plane_fix

This function extends the Para*Py* experimental method *edges_in_plane* such that it also functions when degenerate (improper) edges are present in the shape or list of edges.

```
def edges_in_plane_fix(shape_or_lst, plane):
        if shape_or_lst.edges:
                edges = shape_or_lst.edges
        else:
                edges = shape_or_lst
        proper_edges = []
        for edge in edges:
                if not edge.is_degenerate:
                        proper_edges.append(edge)
        return edges_in_plane(proper_edges, plane.reference, plane.normal)
```

distance_pt_plane

This method can be used to sort points based on distance from a plane. It was used in selecting oval shell points for passenger shell generation per inter-frame section.

```
def d_pt_plane(plane, pt):
        coef = plane.coefficients
        d_pt = coef[0] * pt.x + coef[1] * pt.y + coef[2] * pt.z
        return abs(coef[3] - d_pt)
```

E

Reference Aircraft Model and Mesh

After the decision was made to switch to a non-automated design procedure but before it became clear that the mesh generation was not functioning correctly, the conventional TaW was constructed for two purposes:

1. To provide a reference model such that the Flying V FEM mass could be compared to that of a known aircraft.
2. To serve as a mock-up and demonstrator of techniques to create geometry such as stiffeners and rib faces.

The reference model was to be built in similar fashion as the updated Flying V model. An engineered weight was then only required for comparison to a conventional aircraft configuration. Instead of applying a loosely estimated global correction factor, it was instead decided to construct a TaW model. For such a model, engineered weights are already collected in a database by Airbus and thus easily-accessible reference material could be used.

Secondly, the demonstrator could also be used to investigate other techniques of geometry generation, as TaW design procedures are relatively straightforward due to lack of integrated design. On top of that, they are well documented. This allowed to quickly build a knowledge database for model generation. From this, it was verified that:

- Creation of a wing structure from the wing profile, including integration of the spars and ribs, using *SplitSurface*-operations at spar and rib planes work well and allow for a high degree of flexibility in the structure configuration.
- By splitting the fuselage profiles, before the shells are generated, the stiffener edges are automatically generated.
- Stiffener generation is possible using the design procedure recommended in Chapter 7.
- Frame generation is not affected by the introduction of split fuselage profile curves.
- Intersecting structure, such as the wing and fuselage components of the TaW configuration, can be merged together geometrically – edges are created at the intersections and the components are fully sewn together – using a *Fused*-operation. No settings for the operation had to be changed from the default. This may prove useful once the engine pylon and landing gear structures will be introduced in the Flying V model.

After the Flying V mesh was imported in Patran, it became clear that the estimated weight was significantly flawed by bending stiffness approximations and unbalanced moments in the model. As such, the TaW model was dropped for further analysis.

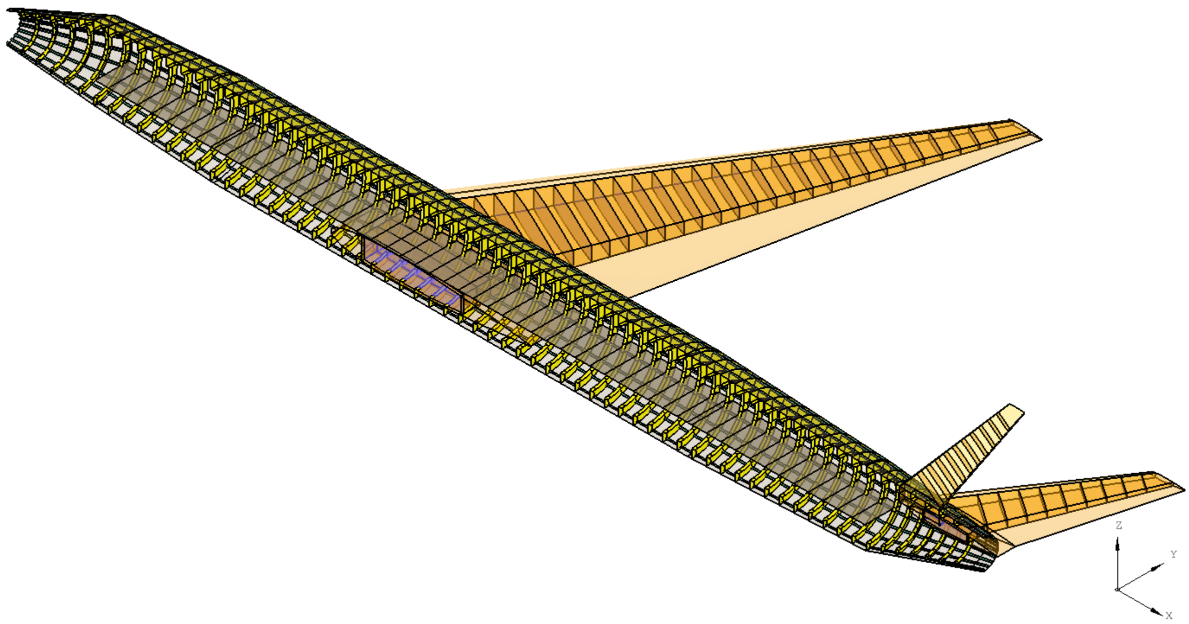


Figure E.1: Isometric view of starboard half of TaW structure model.

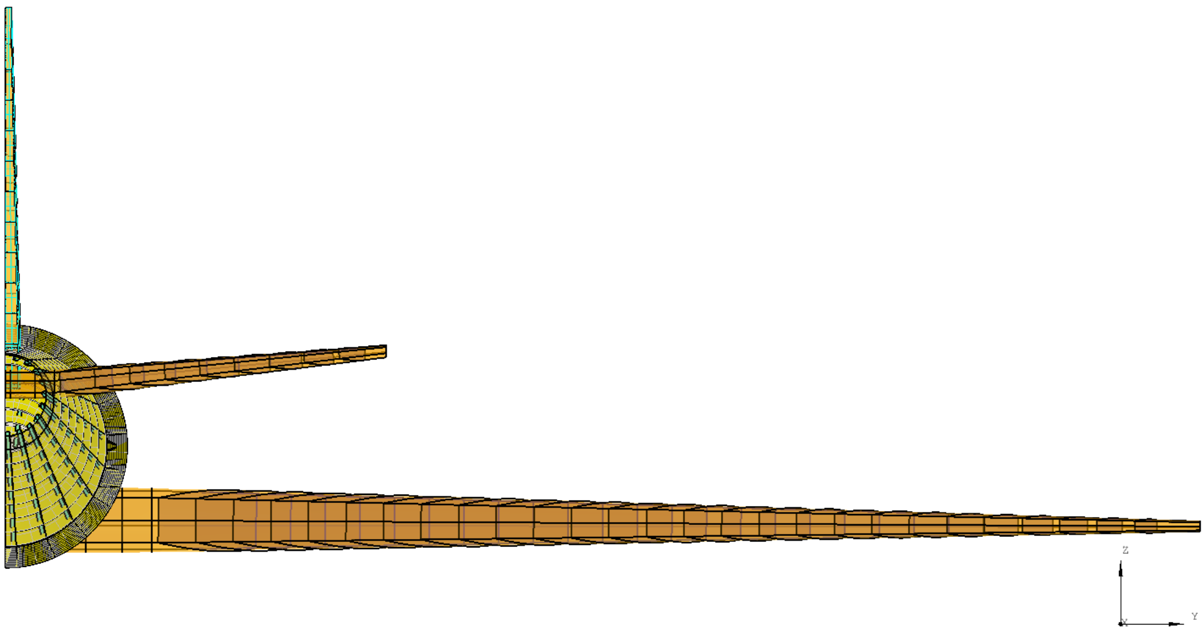


Figure E.2: Rear view of starboard half of TaW structure model.

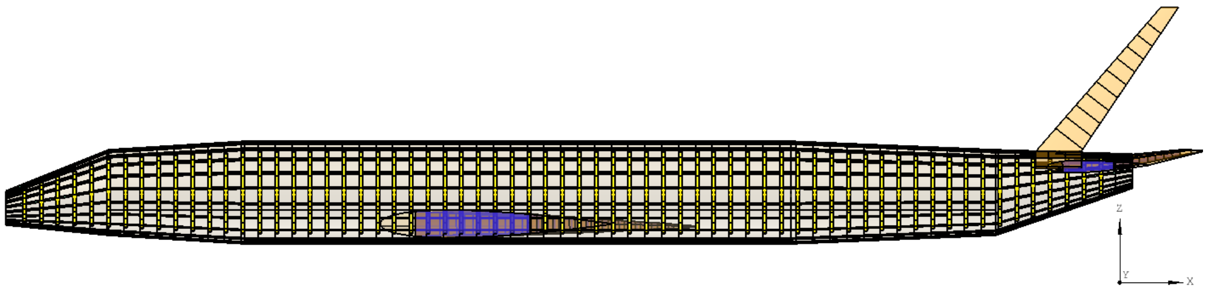


Figure E.3: Side view towards wing tip of starboard half of TaW structure model. Note the VTP profile has not been translated and is left out.

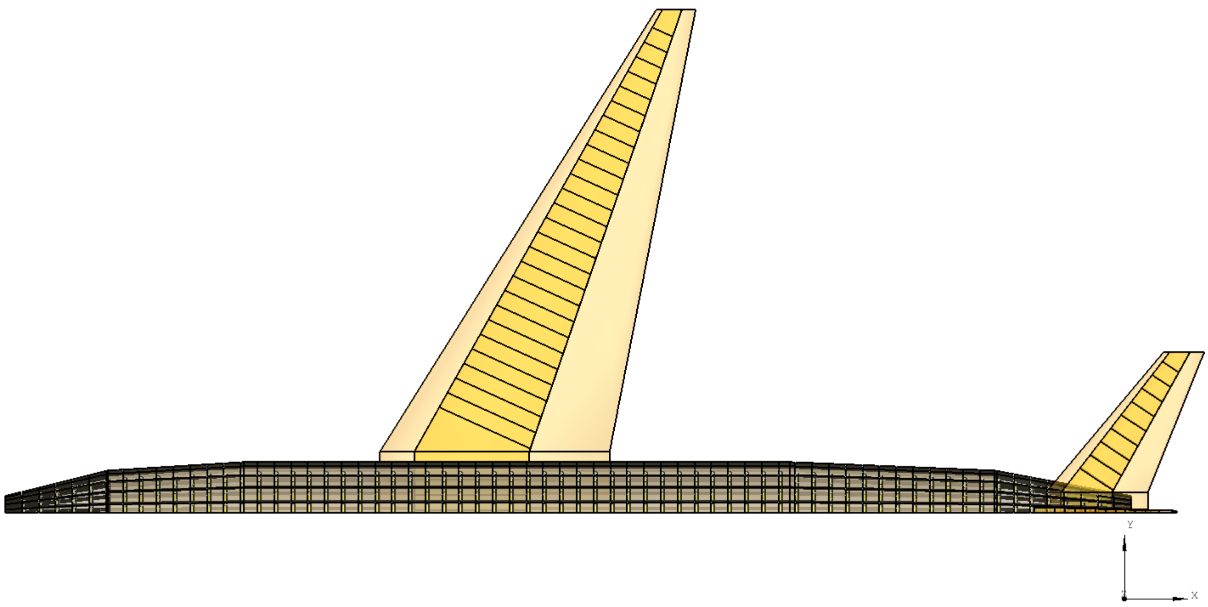


Figure E.4: Top view of starboard half of TaW structure model.

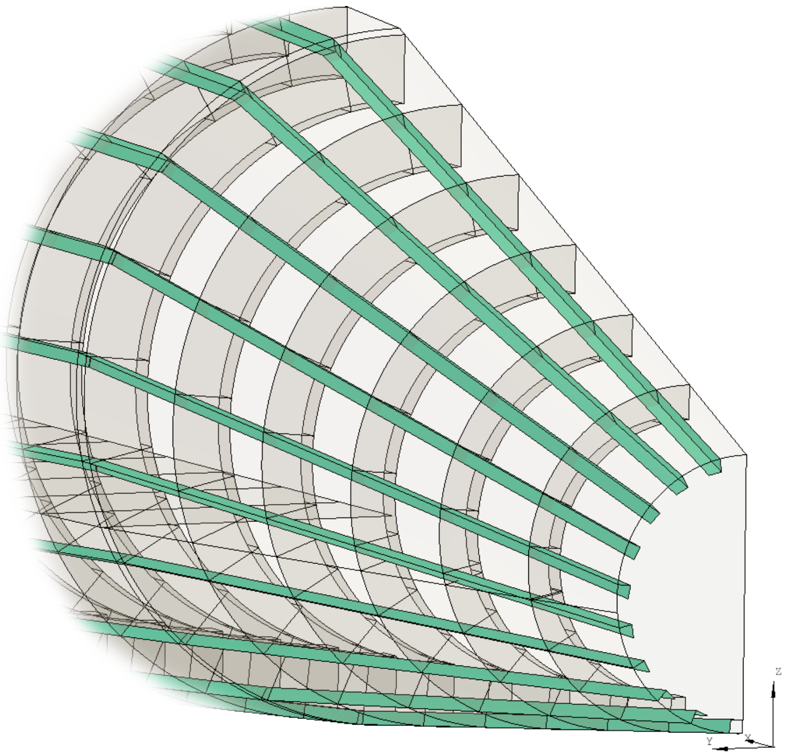


Figure E.5: Example result of stiffener generation procedure.

E.1. Higher Level Primitives

Table E.1: Example plane root class.

	Name	Description
Inputs	nose_radii	Radii of nose section at defined stations *[list of [m]]*
	main_radii	Radii of main section at defined stations *[list of [m]]*
	tail_radii	Radii of tail section at defined stations *[list of [m]]*
	nose_vert	Vertical offset of nose section at defined stations *[list of [m]]*
	main_vert	Vertical offset of main section at defined stations *[list of [m]]*
	tail_vert	Vertical offset of tail section at defined stations *[list of [m]]*
	nose_long	Longitudinal position of nose section at defined stations *[list of [m]]*
	main_long	Longitudinal position of main section at defined stations *[list of [m]]*
	tail_long	Longitudinal position of tail section at defined stations *[list of [m]]*
	frame_pitch	Fuselage maximum frame pitch *[m]*
	airfoil_profile	File containing canonical airfoil coordinates *[string]*
	span	Wing span *[m]*
	taper	Wing taper *[-]*
	dihedral	Wing dihedral *[°]*
	root_twist	Wing twist at root *[°]*

Continued on next page

Table E.1 – continued from previous page

	Name	Description
	tip_twist	Wing twist at tip *[°]*
	root_chord	Wing chord at root *[m]*
	sweep	Wing LE sweep outboard of kink *[°]*
	LE_long	Wing LE longitudinal position *[m]*
	LE_vert	Wing LE vertical offset *[m]*
	rib_pitch	Wing maximum rib pitch *[m]*
	htp_span	HTP span *[m]*
	htp_taper	HTP taper *[-]*
	htp_dihedral	HTP dihedral *[°]*
	htp_root_twist	HTP twist at root *[°]*
	htp_tip_twist	HTP twist at tip *[°]*
	htp_root_chord	HTP chord at root *[m]*
	htp_LE_sweep	HTP LE sweep outboard of kink *[°]*
	vtp_span	VTP span *[m]*
	vtp_taper	VTP taper *[-]*
	vtp_dihedral	VTP dihedral *[°]*
	vtp_root_twist	VTP twist at root *[°]*
	vtp_tip_twist	VTP twist at tip *[°]*
	vtp_root_chord	VTP chord at root *[m]*
	vtp_LE_sweep	VTP LE sweep outboard of kink *[°]*
Attributes	centers	Fuselage defining cross-section centres *[List of Points]*
	airfoils	List of airfoil coordinates for each section
	htp_long	HTP longitudinal position *[m]*
	htp_vert	HTP vertical offset *[m]*
	vtp_long	VTP longitudinal position *[m]*
	vtp_vert	VTP vertical offset *[m]*
Parts	fuselage	Fuselage *[Fuselage]*
	wing	Wing *[Wing]*
	htp	Horizontal tail plane *[Wing]*
	vtp	Vertical tail plane *[Wing]*
	aircraft_w_wing	Aircraft structure of fuselage and wing *[Fused]*
	aircraft_w_wing_htp	Aircraft structure of fuselage, wing and HTP *[Fused]*
	full_aircraft	Full aircraft structure *[Fused]*
	mesh	Structure mesh *[Meshing]*

Table E.2: Fuselage class.

	Name	Description
Inputs	radii	Cross-section radii *[list of dicts]*
	vertical_offset	Cross-section vertical offsets *[list of dicts]*
	longitudinal_position	Cross-section longitudinal positions *[list of dicts]*
	frame_pitch	Maximum fuselage frame pitch *[m]*
	n_stringers	Number of stiffeners in fuselage shell *[-]*
Attributes	total_fuselage_length	Total fuselage length *[m]*
	nose_point	Fuselage nose points *[Point]*
	frames_to_sew	Frames to be sewn into shell *[list of SewnShells]*
	fuselage_structure	Compete fuselage structure *[SewnShell]*
	faces_to_remove	Faces on port side *[list of Faces]*
Parts	shell	Complete fuselage shell *[SewnShell]*
	stringers	Stiffener parts *[FuselageStringer]*
	frames	Frame parts *[FuselageFrames]*
	fuselage_mid_split	Fuselage split on XZ-plane *[SplitSurface]*
	fuselage_half	Complete starboard-side fuselage structure *[Modified-Shape]*

Table E.3: Fuselage shell class.

	Name	Description
Inputs	radii	Cross-section radii *[list of dicts]*
	vertical_offset	Cross-section vertical offsets *[list of dicts]*
	longitudinal_position	Cross-section longitudinal positions *[list of dicts]*
	frame_pitch	Maximum fuselage frame pitch *[m]*
	n_stringers	Number of stiffeners in fuselage shell *[-]*
	floor_offset_vert	Floor plane vertical offset *[m]*
Attributes	centre_axis	Fuselage centre axis *[Wire]*
	stringer_edges	Stiffener edges on fuselage shell *[list of Edges]*
	floor_edges	Floor edges *[list of Edges]*
Parts	nose	Nose shell *[FuselageShell_partial]*
	main	Main shell *[FuselageShell_partial]*
	tail	Tail shell *[FuselageShell_partial]*
	complete_shell	Complete fuselage shell *[SewnShell]*
	shell_with_stringers	Complete shell with stiffeners *[SewnShell]*
	floor_plane	Floor plane *[Plane]*
	shell_split_at_floor	Fuselage shell split at floor plane *[SplitSurface]*
	shell_split_at_floor_at_frames	Fuselage shell split at floor and frame planes *[SplitSurface]*
	floor_parts	Partial floor shells *[list of LoftedShells]*
	floor_split	Floor *[SewnShell]*
	floor_mid_split	Floor split at XZ-plane *[SplitSurface]*
	complete_sewn	Structure of shells, stiffeners, frames and floors *[Sewn-Shell]*
	shell_mid_split	complete_shell split at XZ-plane *[SplitSurface]*

Table E.4: Fuselage partial shell class.

	Name	Description
Inputs	radii	Cross-section radii *[list of [m]]*
	vertical_offset	Cross-section vertical offsets *[list of [m]]*
	longitudinal_position	Cross-section longitudinal positions *[list of [m]]*
	has_floor	Boolean to introduce floor *[Boolean]*
	n_stringers	Number of stiffeners *[-]*
Attributes	profile_split_for_stringer_tool	Fractions of curve at which to split for stiffeners *[list of [-]]*
	offset	Scale point *[list of Vectors]*
	shell_parts	Intermediate shell faces *[list of Faces]*
Parts	new_profiles0	Non-split shell profiles *[sequence of Circles]*
	new_profiles	Split shell profiles *[sequence of SplitCurves]*
	shells	Sequence of partial shells *[sequence of LoftedShells]*
	full_shell	Complete sectional shell *[SewnShell]*

Table E.5: Fuselage stiffener class.

	Name	Description
Inputs	on_shell_wire	Stiffener wires on fuselage shell *[list of Edges]*
	centre_axis	Axis through fuselage shell centre *[Wire]*
	stringer_dim	Stiffener web and flange dimensions *[m]*
Attributes	projection_inward	Inner stiffener web edges *[list of LineSegments]*
	projection_side	Outer stiffener flange edges *[list of TranslatedCurves]*
Parts	web	Stiffener webs *[sequence of LoftedShells]*
	flange	Stiffener flanges *[sequence of LoftedShells]*

Table E.6: Fuselage frame class.

	Name	Description
Inputs	total_fuselage_length	Fuselage length *[m]*
	frame_pitch	Maximum allowed frame pitch *[m]*
Attributes	n_frames	Number of frames *[-]*
	true_frame_pitch	Recalculated frame pitch *[m]*
	profiles	Frame outer wires *[list of Wires]*
	radial_inner_wire	Frame web inner wires *[list of Wires]*
	flange_free_wire	Frame flange outer wires *[list of Wires]*
	bulkhead_wires	Pressure bulkhead wires *[list of Wires]*
Parts	planes	Frame planes *[sequence of Planes]*
	bulkheads	Pressure bulkheads *[sequence of PlanarFaces]*
	radial	Frame webs *[sequence of LoftedShells]*
	flange	Frame flanges *[sequence of LoftedShells]*
	complete	Complete frames *[sequence of SewnShells]*
	bulkhead_mid_split	Bulkheads split on XZ-plane *[sequence of SplitSurfaces]*
	mid_split	Frames split on XZ-plane *[sequence of SplitSurfaces]*

Table E.7: Wing class.

	Name	Description
Inputs	airfoils	Root and tip airfoil canonical coordinates *[dictionary of lists]*
	span	Wing span *[m]*
	taper	Wing taper *[-]*
	dihedral	Wing dihedral *[°]*
	twist	Root and tip wing twist *[dictionary of [°]]*
	root_chord	Root chord length *[m]*
	LE_sweep	Wing LE sweep *[°]*
	LE_longitudinal	Wing LE longitudinal position *[m]*
	LE_vertical	Wing LE vertical offset *[m]*
	kink_y	Wing relative spanwise kink position *[-]*
	is_vtp	Boolean indicating wing is VTP *[Boolean]*
	spar_front	Root and tip relative chordwise front spar position *[dictionary of [-]]*
	spar_rear	Root and tip relative chordwise rear spar position *[dictionary of [-]]*
	rib_pitch	Maximum rib pitch *[m]*
Attributes	airfoil0_coordinates	Canonical coordinates of root airfoil *[list of Points]*
	airfoil1_point	Position of kink airfoil LE *[Point]*
	airfoil1	Kink airfoil *[Airfoil]*
	tip_chord	Wing tip chord length *[m]*
	airfoil2_point	Position of tip airfoil LE *[Point]*
	airfoil2_coordinates	Canonical coordinates of tip airfoil *[list of Points]*
	mir_wing_structure	Wing structure mirrored in XZ-plane *[MirroredShape]*
	full_wing_structure	Full span wing structure *[SewnShell]*
Parts	airfoil0_point	Position of root airfoil LE *[Point]*
	airfoil0	Root airfoil *[Airfoil]*
	airfoil2	Tip airfoil *[Airfoil]*
	structure_wing	Wing structure object *WingStructure*

Table E.8: Airfoil class.

	Name	Description
Inputs	coordinates	Canonical airfoil coordinates *[list of Points]*
	chord	Airfoil chord length *[m]*
	to_point	Airfoil LE point *[Point]*
	spar_front	Relative chordwise front spar position *[-]*
	spar_rear	Relative chordwise rear spar position *[-]*
Attributes	scaled	Airfoil profile scaled to chord length *[ScaledCurve]*
	new_LE	New LE point *[Point]*
	new_TE	New TE point *[Point]*
	spar_plane	Planes at spar positions *[dictionary of Planes]*
	the_long_edges	Wing box skin edges *[list of Edges]*
	the_short_edges	Non-wing box skin edges *[list of Edges]*
Parts	basis_curve	Canonical airfoil profile *[FittedCurve]*
	complete_profile	Airfoil profile at LE, scaled to chord length *[TransformedCurve]*
	top_wire	Top skin profile *[Wire]*
	bottom_wire	Bottom skin profile *[Wire]*
	complete_wire	Complete airfoil profile *[Wire]*

Table E.9: WingStructure class.

	Name	Description
Inputs	profiles	Wing profiles *[list of Wires]*
	rib_pitch	Maximum rib pitch *[m]*
	is_vtp	Boolean indicating if VTP *[Boolean]*
Attributes	is_swept	Boolean indicating if wing sections are swept *[list of Booleans]*
	full_loft	Full wing loft *[SewnShell]*
	wing_structure_full	Full wing structure loft *[SewnShell]*
Parts	loft	Sectional wing lofts *[sequence of LoftedShells]*
	structure_part	Sectional wing structures *[sequence of InternalStructure]*

Table E.10: InternalStructure class.

	Name	Description
Inputs	loft	Wing section loft *[LoftedShell]*
	rib_pitch	Maximum rib pitch *[m]*
	is_swept	Boolean indicating if wing sections are swept *[Boolean]*
Attributes	skin_panels	Outer skin faces *[list of Faces]*
	n_skin_panels	Number of skin panels *[-]*
	bottom_skin	Bottom skin *[SewnShell]*
	top_skin	Top skin *[SewnShell]*
	spars_profiles	Spar profiles *[list of Edges]*
	rib_line	Wing box axis *[LineSegment]*
	TEs	Root and tip TEs *[list of Points]*
	LEs	Root and tip LEs *[list of Points]*
	span	Wing box axis length *[m]*
	number_of_ribs	Number of ribs *[-]*
	new_rib_pitch	Updated rib pitch *[m]*
	rib_line_trimmed	Wing box axis trimmed to minimum span *[Trimmed-Curve]*
	rib_wires	Rib wires *[list of Wires]*
Parts	spars	Front and rear spar *[sequence of LoftedShells]*
	rib_planes	Rib planes *[sequence of Planes]*
	dummy_box	Unsplit wing box *[SewnShell]*
	wing_box	Wing box split at rib planes *[SplitSurface]*
	wing_box_w_ribs	Wing box including ribs *[SewnShell]*
	ribs	Ribs *[sequence of PlanarFaces]*

F

Required Developments for Para*Py* -Enabled Design Automation

Several additions to Para*Py* are required before it can successfully facilitate a full design automation. Note that these are Para*Py* -native modules and exclude analysis tools for aerodynamic, structure and performance analyses. These recommended additions are described in the following sections.

Shape on Face

The geometry generation can be simplified using the *Fused* operation to merge several components without geometrically equivalent edges. Excess faces will likely be introduced where the edges do not coincide fully. These can be removed using *ModifiedShape*. However, selection of faces to be removed proves difficult when based purely on logic. A valuable addition to the global methods is a method which can determine if a face is either a subset or container of another face. A solution may be found in the BRep-representation of both faces.

Implementation of Mathematical Extrema-Method

It was attempted to couple SU2 data on local pressure coefficient to the model in order to obtain a collection of lift force vectors. The pressure coefficient itself only contains the magnitude of the force vector. The direction can be obtained in the most straightforward way by projecting the data coordinates onto the wing outer shape. However, no method exists for this in Para*Py* . An alternative solution was found in finding the extremas between all outer shape faces and the respective grid point. The minimum distance extrema was selected and the surface *(u,v)*-value was extracted. Using this *(u,v)*-value, the local normal vector on the face was found.

The downside of this method is the intensive geometric operations required to determine the extrema. This caused memory issues for a collection of data points larger than 5000. The computation time rapidly increased too. An unconfirmed suspicion exists determination of the surface projection normal vector can be done in a different, more efficient way by using the mathematical description of the surface instead.

Point-to-Surface Projection

As was explained above, point-to-surface projection is currently not implemented in Para*Py* . An elaborate work-around can be achieved by using the computationally intensive extrema-method. Implementation of a direct point-to-surface projection method will simplify the model generator code.

Surface-to-Surface Projection

Surface-to-surface projection may prove useful when cut-outs must be defined. In the current version of Para*Py* , cut-out edges can be obtained by projecting edges onto a surface using ProjectedCurve. The surface outer wires must then be combined with a wire of the projected edges to generate a Face with an island and a hole. The face topology is not yet definable in the current software version.

Non-Planar Face Topology Definition

Face-topology definition is only defined for PlanarFaces. For non-planar Faces, a Face instance is created but the face topology is not introduced. For a LoftedShell or RuledShell, this is done, so there must be a way in which face topology definition can be implemented. This addition will prove useful for:

- Defining cut-outs.
- Defining non-planar, non-quadrilateral faces.

Non-uniform Reference Point Scaling

In the current model generator, the fuselage frame webs are obtained by lofting between an inner and outer wire. The inner wire is obtained by uniformly scaling the outer wire. This does not allow for independent bending stiffness sizing of the various sides (top, bottom, inboard, outboard) of the frames, as only a single parameter is available. Non-uniform scaling is implemented in Para*Py* , but it ignores the reference scaling point. Non-uniform scaling is only possible around the origin (Point(0,0,0)). This requires translation of the outer wire to the origin and back to its original position and breaks the geometric equivalence constraint.

Profile Orientation Check for Ruled Shell

It has been observed that LoftedShells create a fitted surface through profiles, even when only two profiles are provided and the loft is linear. The RuledShell class creates a surface with the exact profiles, but does not check whether these profiles are oriented similarly. For optimal geometry generation, the RuledShell object is used. To use existing profiles reliably, a check for profile orientation must be implemented.

BDFWriter

Several functionalities of the BDFWriter module of Para*Py* were found missing, causing large hindrance in exporting the model to Patran. These lacking functionalities are:

- Writing of the load cases.
- Writing of multiple materal definitions.
- Writing of multiple shell definitions.
- Variety in available cards from Nastran card deck.

Load cases were defined in the model generator. When writing these to a .bdf-file, the module returned an error of which the cause is unknown. The load cases could be defined manually in Patran pre-processing. However, with design updates, the loads will be modified in magnitude and position. Therefore, load case integration in the .bdf-file is essential.

For an initial mass estimate, it was assumed that a single material was used in the entire model. Although such an initial assumption is valid for aircraft design [24], it severely limits the capability to implement design fixes and constraints. In this research, additional materials were required to define orthotropic properties and to approximate honeycomb passenger cabin panels to replace thin shells. This was not possible. Continuing on the topic of material definitions, the BDFWriter only allows for a simple, isotropic material. Introduction of carbon-fibre reinforced laminates with orthotropic material properties might improve the mass of any configuration. Therefore, implementation of orthotropic material Nastran cards is required.

Lastly, thick structures were required to resist transverse bending. As described in Chapter 5, a solution was to use a model featuring thick shells instead of solely thin shells. Another solution was to implement additional beam elements in the mesh model between all the nodes. Neither of these solutions could be tried as the .BDFWriter does not have the functionality to write any other elements than quadrilateral elements, triangular elements and rigid bars. Currently, detailed geometry of stiffeners and frames is generated. The entire model and analysis thereof may be drastically simplified if these can be replaced by beam elements with a certain inertia and material property.

Mesh ID Shuffling and Matching

In the Flying V study, it was observed that mesh element IDs did not correspond between the meshes in Para*Py* and Patran. It must be found out by what program the mesh IDs are changed. If Para*Py* changes the mesh IDs when writing to a .bdf-file, this must be fixed. If Patran is the root cause, a method must be implemented in Para*Py* to match Patran mesh elements back to Para*Py* mesh elements. *If this step cannot be solved, design automation is not possible using a coupling between ParaPy and Nastran.*

Documentation

During the research, many operations were broken up into smaller sections, each of which was programmed individually. Often an available method was not used because of ignorance of the existence hereof. The current code documentation is limited to documentation of native OpenCascade objects and Para*Py* methods added to these. Documentation of shape modification, mesh generation, coupling to open-source software and file exporters does not exist. Everything had to be found out by trial-and-error. A complete documentation, however limited in explanation, would have been beneficial to this research.