33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

MODEL FIDELITY AND ITS INFLUENCE ON AIRCRAFT CONCEPTUAL DESIGN

Alexandra Oprea[1], Robert Hällqvist[1], Ingo Staack[2], Magnus Eek[3] & Hampus Gavel[1]

[1]Saab Aeronautics, Bröderna Ugglas Gata, 581 88 Linköping, Sweden; +46 13 18 00 00
[2]Linköping University, 581 83 Linköping, Sweden; +46 13 28 10 00
[3]Swedish National Road and Transport Research Institute; +46 13 20 40 00

Abstract

This study investigates the effects of including industry-grade legacy system simulation models already at the conceptual stage in the aircraft development process. Through a novel method for Legacy Model (LM) integration based entirely on open standards, two systems simulation models of different levels of fidelity, one based on handbook methods representing a low fidelity simulation model, and one based on legacy data from a previous aircraft project of a similar design, considered to be of a higher level of fidelity, are included in an aircraft sizing framework. The resulting aircraft designs are evaluated in terms of wing reference area and engine size. The LM and the Handbook Model (HM) are assessed in terms of integration and, respectively, development efforts, and in terms of execution time. It was found that the choice of model fidelity impacts the resulting design of the aircraft, with the low fidelity HM generating a design with a bigger wing reference area and engine size than the high fidelity LM. The assessment of the results turned out to be more time-consuming for the HM than for the LM. Therefore, although the integration effort of the LM took longer than the development time of the HM, the benefit of increased confidence in the results generated with the LM outweigh the initial cost of LM integration wrapper development. Moreover, once the LM integration wrapper has been built, the integration of other LMs with the same interface is similar to "plug-and-play", allowing for a more thorough design space exploration, albeit in the area restricted to the models' Operational Domain (OD). In terms of execution time, the optimisation process based on the LMs is twice as long as the one based on the HM. However, the execution time is low enough to not be an impediment to LMs inclusion already at the conceptual stage.

Keywords: modelling and simulation, model reuse, system conceptual design, validation, fidelity, FMI, SSP

1. Introduction

This paper is on the topic of modelling and simulation in the early phases of aircraft development. It targets the problem of including detailed legacy system simulation models, from here on referred to as "high fidelity models", already at the conceptual design stage. The presented research aims to investigate whether this change in model development strategy would have an influence on the aircraft conceptual design process. Traditionally, this process starts with exploiting handbook methods and the corresponding models with a lower level of detail, from here on referred to as "low fidelity models", in order to generate a rough aircraft design. The resulting concept is iteratively developed into a finished product, and system simulation models are developed, verified, and validated in the process. However, for a new design, the process may start again at the handbook methods stage. Model reuse is not uncommon, as the Model Based System Engineering (MBSE) methodologies [1] encourage the practice of system modelling by reusing components from pre-existing component libraries. However, the reuse of fully developed system models is restricted to purposes similar to the original purpose of the models. Many actors within the aerospace industry have, as a result, developed libraries of high fidelity system simulation models that have been verified and validated in previous projects. Provided that the LMs are similar enough to the targeted future development activities, they could be reused. The efforts required for model development, verification, and validation can be significantly reduced if model reuse is considered from the start of a model development effort. The goal of this paper is to investigate the feasibility of using, and whether it is possible to take advantage of, the existing knowledge through the inclusion of existing LMs earlier in the development process and whether this inclusion influences the results of the design. If so, are the computational time losses associated with a higher level of detail extensive enough to justify the existence of the lower fidelity models when the higher-fidelity ones are already available? An assessment of the LM integration effort in comparison with the efforts for development of a HM is also performed.

1.1 Contributions

There are three primary contributions of this paper. First, the presented research aims to devise a method for investigating the impact of the level of fidelity of aircraft system models on the aircraft concept design stage. Second, the paper investigates the usability of the Modelica standard FORTRAN interface in order to facilitate reuse of LMs with the existing Functional Mock-up Interface (FMI) support in Modelica tools. Finally, the aircraft sizing is conducted in a FMI and System Structure and Parametrisation (SSP) standard supporting simulation tool. As a result, additional modelled aircraft systems can be accounted for without significant tailoring of the LMs in question.

2. Background

The aircraft development process is a lengthy, multidisciplinary one, involving several teams of engineers collaborating over the course of many years and many disciplines. A set of requirements coming from the project stakeholders and the certification authorities is iteratively transformed into a concept, and further developed into a system, that is tested, produced, and finally deployed [2, 3]. The concept phase is of critical importance, as it can make or break a project. To correct the flaws of a concept later in the development stage is costly, if not impossible. Therefore, at this stage it is critical to make use of as much information as possible so that the best design is generated. The difficulty lies in that, traditionally, the knowledge about different systems involved is quite low at this stage, and many simplifications are made. There is an aspiration to move towards set-based design, where design parameters are kept open for as long as possible in the product development process [4]. This implies including models with a higher degree of representativeness, which will be here referred to as a higher level of fidelity. For a more exact definition, *model fidelity* is regarded by Ref. [5] as

"The degree to which a model or simulation reproduces the state and behaviour of a real world object or the perception of a real world object, feature, condition, or chosen standard in a measurable or perceivable manner; a measure of the realism of a model or simulation; faithfulness."

As *fidelity* is a term that is difficult to quantify, we will restrict its definition. Model fidelity is here seen to typically increase with the increase in the level of detail of the model, or with its level of representativeness, as shown in Fig. 1. An increase in the number of model parameters and variables relevant to the intended use of the model would generate a model of higher fidelity. However, this is not the only property that affects model fidelity, since this also depends on the context of the modelling task. Part of the context is the system/model OD, that can be regarded as the domain in which a system, or a system simulation model, is designed to be used [6, 7]. An increase in the overlap between the intended OD of the model and the OD of the LM would also increase the overall model fidelity. To better illustrate the concept, four different hypothetical OD are visualised in Figure 2: a legacy system OD, a corresponding LM OD, the assumed OD of the target system to be developed, and the OD of the target model. The area of the overlap between the target system of interest OD and the LM OD has bearing on the model fidelity relative the future model use.

A complete aircraft simulation model consists of a limited number of system simulation models, that vary according to the scope of the simulations. For instance, Fig. 3 shows the system simulation models included in the ARES Simulation Environment in use at Saab Aeronautics for flight dynamical

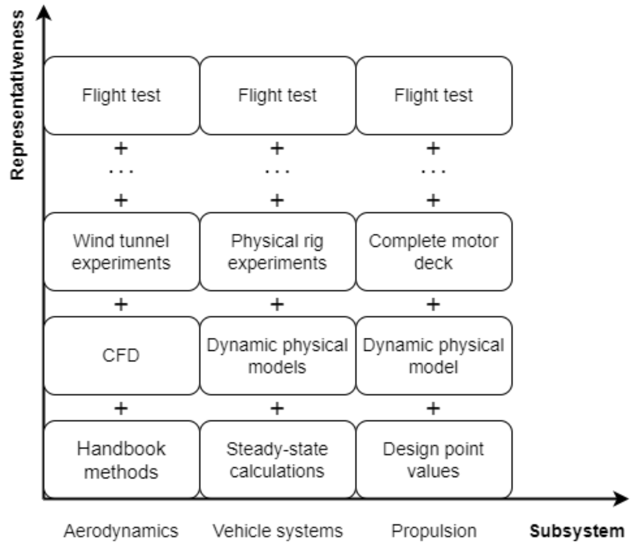

Figure 1 – An example of how model representativeness increases as information is gained through iterative model development and testing steps for three aircraft system simulation models

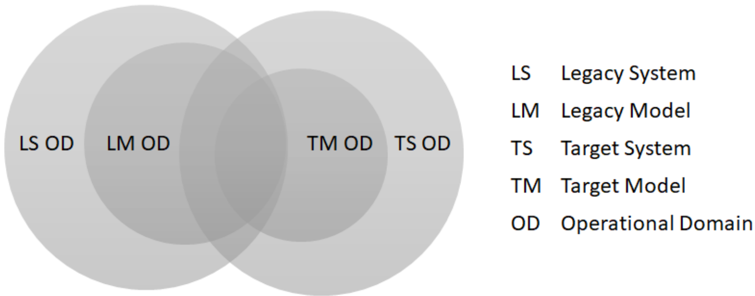

Figure 2 – Hypothetical ODs for the legacy/target systems/models. The dark grey area in the middle is the OD that is common to both the legacy and the target system and where the information gain is the highest

simulations and flight control law development. From the simulator's intended use perspective, the design space is limited by the bounds of the design parameters incorporated in the included models. As a result, many design choices can be excluded already at the requirements interpretation stage in the development process, before the system development concept stage is initiated. At the beginning of a project, assuming that the target aircraft is not the manufacturer's first of its kind, and that there exist simulation models that have been used during previous development efforts, detailed system models with ODs overlapping the ODs could already be used. Therefore, since there are not that many models involved, and since the design space of interest is not infinitely big, it might be worth to consider the time and information gain in reusing existing models, compared to building new ones. Using models of a higher fidelity is not cost-free, though, since they usually come with an increase in simulation execution time and require some overhead activities for integration purposes and understanding the influence of the legacy OD on the simulation results.

In order to investigate the influence of including a higher fidelity LM on the concept development process a simple aircraft sizing task has been performed. This was performed first by using a HM based on methods developed by Raymer in Ref. [9] and implemented in the Modelica language, and then by integrating LMs from the ARES Simulation Environment and written in FORTRAN. The resulting concepts have then been evaluated in terms of wing reference area and engine size, and the model development and integration efforts have been estimated.

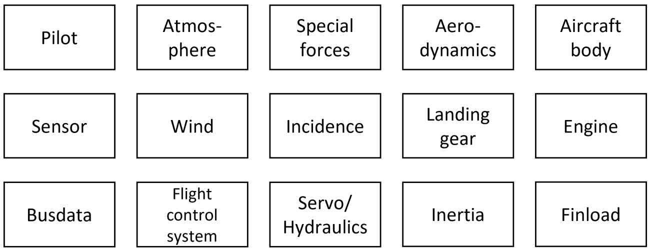

Figure 3 – The models contained in the Saab Aeronautics ARES Simulation Environment. Adapted from Ref. [8]

3. Methodology

The aircraft model included in the sizing process is based on five simple system simulation models: Geometry, Aerodynamics, Propulsion, Mission and Atmosphere. The Mission, Atmosphere, and Propulsion models are identical in both instances of the sizing process. The lower level of detail aerodynamics and geometry models are based on the handbook methods developed by Raymer in [9], and the higher level of detail aerodynamics and geometry models are incorporated as interpolation tables populated with data from Computational Fluid Dynamics (CFD) analysis. The resulting aircraft designs are compared in terms of wing surface and engine size. The model development process is then evaluated in terms of development time, simulation time and reliability.

The sizing is performed using the full aircraft mission simulation framework developed in an earlier publication of the authors [10]. The framework, implemented in the Modelica language, takes advantage of FMI [11] and SSP [12] opportunities provided by *OMSimulator* [13], *OpenModelica* [14], and *Dymola* [15], and is, therefore, suitable for the incorporation of LMs in a system simulation.

The LMs exploited in this paper are FORTRAN-based and they are originally developed for the ARES Simulation Environment, a Saab Aeronautics in-house tool that does not support the FMI standard. In order to mitigate this lack of tool functionality, a general integration wrapper is developed. This general wrapper exploits the standardised FORTRAN interface specified in the Modelica specification [16] in order to accommodate for FORTRAN LMs in Modelica tools. As the FMI standard is developed and maintained by the *Modelica Association* [17], many Modelica tools have mature support of the FMI standard. Once wrapped, the LMs are exported as Functional Mock-up Units (FMUs) for subsequent integration in the optimisation framework. The FORTRAN model steps of integration in Modelica are shown in Figure 4 and the wrapper code is shown in Listing 1.

Listing 1: The Modelica wrapper for integrating FORTRAN functions

```
package FORTRANExternalFunction
  function f
    input Real u[nu];
    input Real x[nx];
    input Real xdot[nx];
    output Real y[ny];
    external "FORTRAN 77" aerodynamics(u,x,xdot,y)     annotation(Library={"external"});
    end f;

  model Test
    Real u[nu] = {...};
    Real x[nx] = {...};
    Real xdot[nx] = {...};
    output Real y[ny];
  equation
    y = FORTRANExternalFunction.f(u,x,xdot,y);
  end Test;
end FORTRANExternalFunction
```

All the non-legacy models included in the sizing process are packaged as FMUs and their interdependencies are expressed in the System Structure Description (SSD) format of the SSP standard. The resulting simulator architecture, including all its constituent executable artefacts and parameter-sets, are packaged in the zip file format specified in the SSP standard. The resulting simulator is then

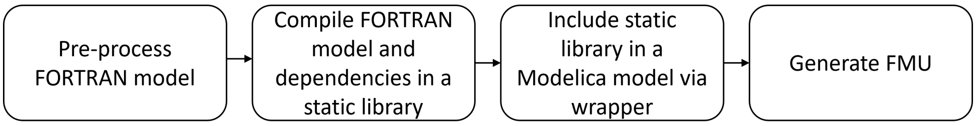

Figure 4 – Stepwise encapsulation of a legacy FORTRAN model in an FMU

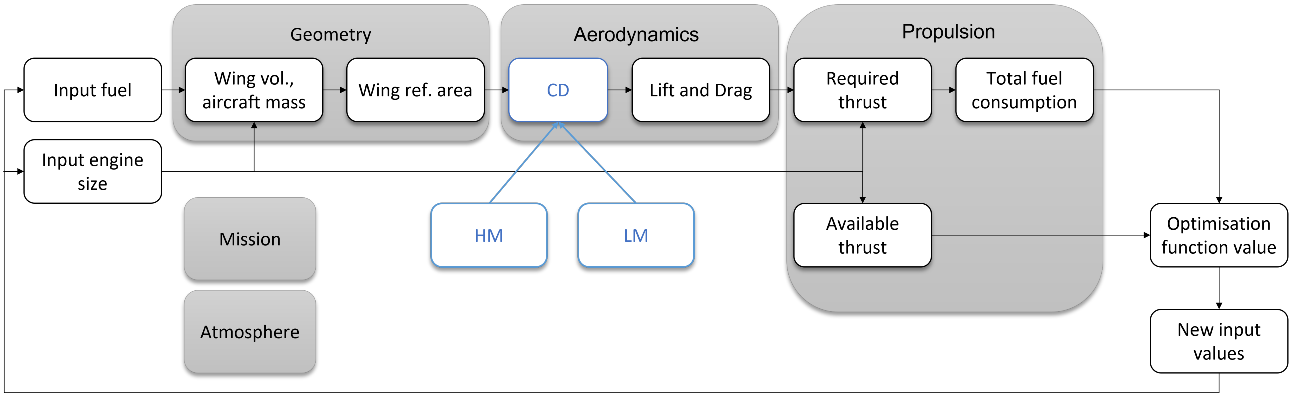

Figure 5 – Input parameters and their influence on the simulation models involved in the sizing process

expressed in a fully portable format, available for integration in the OMSimulator master simulation tool by means of the available OMSimulator Python API. The OMSimulator tool is then responsible for the scheduling of, and communication between, the simulator executable entities in order to drive the simulation forward.

The sizing procedure is based on a user-defined target mission, altitude and velocity profile, and it follows the steps described in Ref. [10]. The sizing process results in a set of parameters defining the aircraft: a wing area, the fuel mass required to complete the mission, and parameters describing a suitable engine size. The wing area calculations, as well as the engine sizing, are presented in Section 3.1. The required fuel mass for mission completion is, simply, the total quantity of burnt fuel during the entire mission, without any extra fuel for diversions or accounting for the non-usable fuel in the aircraft. Including these would generate a slightly bigger wing, but would not impact the overall comparison between the low fidelity and the high fidelity models or the assessment of the usability of the tools and methodologies developed. A schematic diagram of the sizing process is presented in Fig. 5.

3.1 Models

There are five models included in the simulation: a geometry model for the aircraft dimensions, an aerodynamics model for the calculation of aerodynamic forces, a propulsion model, a mission model providing the altitude and velocity of the aircraft and an atmospheric model.

3.1.1 Geometry

The geometry model for the low fidelity sizing process is based entirely on Ref. [9]. As the sought aircraft design is a V-tailed fighter with a double-delta wing, some assumptions are required to calculate the geometrical features. The root, mid, and tip chord $c_{root}, c_{mid}, c_{tip}$ (see Fig. 6), as well as the fuselage length and nose cone length L_f, L_{nc} and diameter D_f are all fixed.

One of the purposes of the geometrical model is to provide the aerodynamics model the wing dimensions and the wetted areas of the aircraft in order to compute the aircraft drag. These are obtained from a total required wing volume, which is calculated from the input fuel mass M_{fuel} (which is an

Section	a_1	a_2	b_1	$h_{section}$
Inner	$0.5c_{root}$	$0.5c_{mid}$	$t/c_{inner}c_{root}$	h
Outer	$0.5c_{mid}$	$0.5c_{tip}$	$t/c_{outer}c_{mid}$	$2h$

Table 1 – Values for the semi-major axes a_1 and a_2 and semi-minor axis b_1 for the ellipses defining the inner and outer sections of the wing, as well as the height $h_{section}$ of the truncated cones the ellipses define

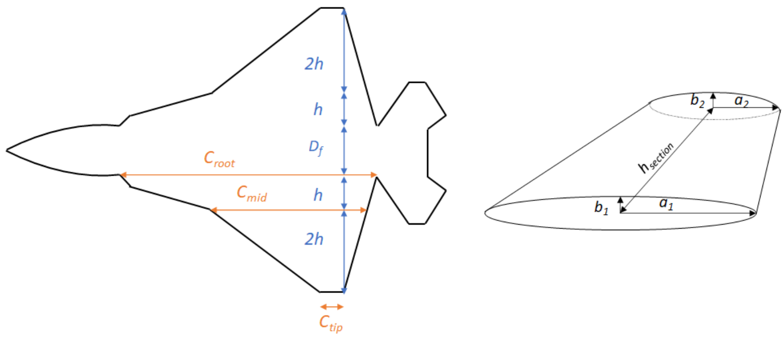

Figure 6 – Geometrical parameters of the aircraft and of the wing section viewed as a truncated elliptical cone

optimisation parameter). For this sizing task, it is assumed that 50% of the total mission fuel is stored in the wing. Knowing the fuel density ρ_{fuel}, the wing volume can be described as

$$W_{vol} = 4 \cdot 0.5 \cdot \frac{M_{fuel}}{\rho_{fuel}} \tag{1}$$

if assuming that only 25% of the wing volume is available for the fuel tank. Furthermore, some assumptions about the wing are required to calculate the wing surface S_{ref}. The wing is divided into four segments, two inner and two outer volumes. Each segment is regarded as an elliptic truncated cone, as shown on the right-hand side of Fig. 6. The volume of each section is then given by

$$V_{section} = \frac{\pi b_1 h_{section}}{3a_1}\left({a_1}^2 + a_1 a_2 + a_2^2\right) \tag{2}$$

where the values for the semiaxis a_1, a_2, b_1 and the section height h are given in Table 1.
Knowing that

$$W_{vol} = 2\left(V_{inner} + V_{outer}\right) \tag{3}$$

then the value for h becomes

$$h = \frac{6W_{vol}}{\pi\left[0.5\left(c_{tip}^2 + c_{tip}c_{mid} + c_{mid}^2\right)t/c_{outer} + 0.25\left(c_{mid}^2 + c_{mid}c_{root} + c_{root}^2\right)t/c_{inner}\right]} \tag{4}$$

and allows for the calculation of b, S_{ref} and AR.

The baseline aircraft geometry related to the CFD-based aerodynamics model is already set, and the only parameter that needs to be calculated is the scaling factor SF to be applied. The SF is given by Eq. 5.

$$SF = \left(\frac{Mass_{fuel,input}}{Mass_{fuel_{baseline}}}\right)^{1/3} \tag{5}$$

3.1.2 Mission

The design mission for the aircraft is simple: taxi, take-off and climb, cruise, loitering around the surveyed area, cruise back to base, descent, and landing. The start and final altitude and velocity

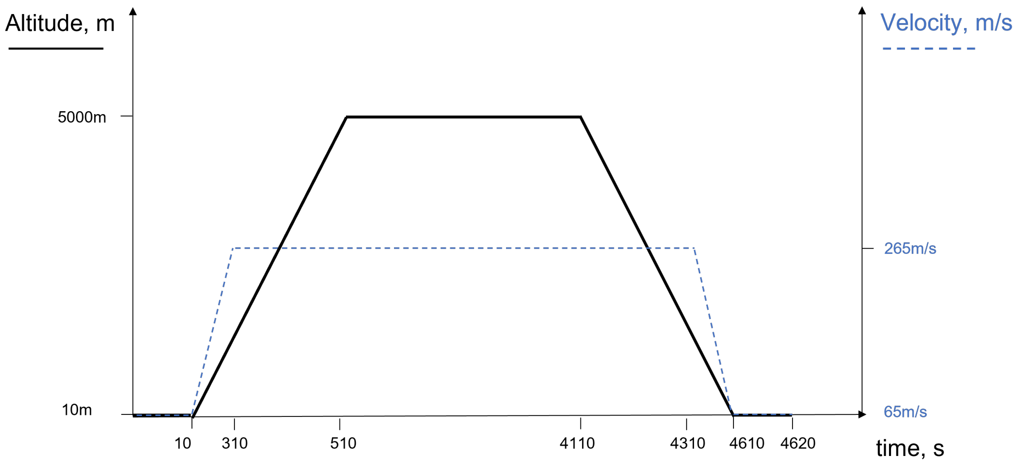

Figure 7 – Altitude and velocity profile for the given mission. The segments are not to scale

are the same for all missions. The maximum altitude h_{max} is specified as 5000m, 5500m, 6000m, 6500m and 7000m and the maximum velocity v_{max} is 265m/s for all altitudes. Fig. 7 shows a sketch of the given mission. The length of each of the mission segments is the same for all five versions of the mission, effectively resulting in a climb/descent rate of 10m/s, 11m/s, 12m/s, 13m/s and 14 m/s respectively.

3.1.3 Aerodynamics

Two types of aerodynamic models are included in the aircraft simulation: a handbook coefficient approximation model based on the methods presented by Raymer in [9] and an aerodata model based on CFD analysis. The main purpose of these models is to provide data about the aircraft drag D, which is then used for the thrust calculations.

The handbook coefficient model

The conceptual design methods from Raymer [9] are the basis of the handbook coefficient model used in this work. An initial version of the model used in this paper is described in [10]. For clarity, some of the essential aspects are described below.

The aircraft drag coefficient

$$C_D = C_{D,0} + C_{D,i} \tag{6}$$

can be split into two terms: the parasitic drag coefficient, given by the parasite drag coefficient at zero lift $C_{D,0}$, and the lift-induced drag coefficient $C_{D,i}$. The lift-induced drag coefficient $C_{D,i}$ is dependent on the lift coefficient C_L, and the aircraft aspect ratio AR according to

$$C_{D,i} = \frac{C_L^2}{\pi e AR} \tag{7}$$

where e is the Oswald efficiency factor. The value of C_L is calculated as

$$C_L = \frac{2L}{\rho v^2 S_{ref}} \tag{8}$$

where the lift L is obtained from

$$L = Mgcos\gamma \tag{9}$$

where the air density ρ from Eq. 8 is given by the atmospheric model, and the flight path angle γ in Eq. 9 is given by

$$\gamma = arcsin\frac{\dot{h}}{v} \tag{10}$$

where the altitude h and the velocity v are supplied by the incorporated mission model.

Raymer provides a methodology to approximate a value for $C_{D,0}$ from the geometry of the aircraft. There are several dimensions that are included in the estimation process, of which the most important are the fuselage length L_f including the nose cone length L_{nc}, the fuselage diameter D_f, the aspect ratio AR, the sweep angle Λ, the mean aerodynamic chord $\bar{c}$, and the thickness-to-chord ratios t/c for all lifting surfaces. The wingspan b is also required, and in this paper it will be a result of the optimisation process. The reader is advised to consult Ref. [9] for a full description of the estimation process.

As the aircraft design features a double-delta wing, some geometrical approximation methods were required to obtain the reference area of the wing S_{ref} and the wetted areas required by the drag calculations.

CFD-based aerodata model

The complete CFD aerodata model includes coefficients for aerodynamic forces and moments that are dependent on the angle of attack α, the angle of sideslip β, control surface deflections δ_{cs}, and Mach number. The data is tabulated and integrated into a FORTRAN model that interpolates the data and provides the force and moments coefficients.

The CFD model is based on a baseline aircraft geometry. In order to be used during a sizing process, some sort of scaling is required. For the purpose of this sizing, aerodynamic scaling has been chosen. The dimensions pertaining to this baseline geometry will hereby be denoted with the subscript *base*. The lengths and surfaces of this geometry need to be scaled with an appropriate *scaling factor* in order for the resulting force and moments coefficients to still be applicable. The scaling factor definition was given in Eq. 5.

3.1.4 Propulsion

The propulsion model is based on the Williams FJ44 engine, with performance data from Ref. [18]. The Thrust-Specific Fuel Consumption (TSFC) data is obtained from Saab Aeronautics [19]. The engine data is scaled via an engine scaling factor that acts upon the engine weight

$$W_{engine} = 100(k-1) + W_{eng_{base}} \tag{11}$$

and on its maximum output thrust

$$T_{avail} = k \cdot T_{max} \tag{12}$$

where k is the scaling factor. This scaling factor is not related to the the aerodynamics SF from Eq. 5. Knowing the thrust-specific fuel consumption allows for the calculation of the total fuel consumed during the entire mission $Mass_{fuel,total}$.

3.2 Optimisation

Due to the nature of the problem, gradient-based methods could not be used for the optimisation routine. Instead, the direct search Nelder-Mead method [20] implemented in the Scipy.Optimize [21] package was used. The optimisation function is

$$f_{opt} = \lambda_1 f_1(\vec{x}) + \lambda_2 f_2(\vec{x}) + w_1 G_1(\vec{x}) \tag{13}$$

and the two subfunctions f_1 and f_2 are

$$f_1 = |m_{tf} - m_{if}| \tag{14}$$

and

$$f_2 = \max_t |T_{req}(t) - T(t)| \tag{15}$$

where λ_1 and λ_2 are weights selected so that they would influence the overall objective function in equal parts. G_1 is a penalty function ensuring that there is always more available thrust than required thrust throughout the entire mission

$$w_1 G_1(\vec{x}) = 500(1 + \mathrm{sign}(T_{req} - T_{avail})) \tag{16}$$

and the weight w_1 ensures that it has a similar influence on the overall objective function as the other two terms in Eq. 13.

4. Results

The results are divided into two sections. The first part will analyse the impact of the LM and the HM on the design of the aircraft, and the second will present and evaluate the developed integration method for the LM.

4.1 Model fidelity impact on design

The total fuel mass required for mission fulfilment is shown in Fig. 8. The fuel mass $M_{fuel_{norm}}$ and reference area $S_{ref_{norm}}$ values are normalised according to

$$M_{fuel_{norm}} = \frac{M_{fuel}}{M_{fuel_{min}}} \tag{17}$$

and

$$S_{ref_{norm}} = \frac{S_{ref}}{S_{ref_{min}}} \tag{18}$$

where the minimum value is obtained for the LM with no control surface deflections $\delta_{cs} = 0.0$. The engine scaling factor, resulting from the sizing optimisations, is shown in Fig. 9 and the reference wing area S_{ref} is shown in Fig. 10.

A general observation can be made for Figures 8 to 10. The results obtained with the LM aerodynamics vary with altitude with a predictable trend, as climbing to a higher altitude requires more fuel, which generates a bigger wing and a bigger drag, and therefore requires a bigger engine. However, the results generated with the HM are relatively constant in all three respects: required M_{fuel}, which impacts the reference area S_{ref} and the engine scaling factor k. The values of these parameters are also consistently higher for the HM-based design than in the LM-based design, with the engine being almost double in size for the HM compared to the LM without trim drag effects ($\delta_{cs} = 0.0$). The reason for this is still under investigation, but one of the factors contributing to this effect could be the higher estimated values of the drag coefficient C_D in the HM compared to the values obtained from the LM aerodata.

The normalised drag coefficient C_{Dnorm} is presented in Fig. 11 for all three models for the mission at 6000m, but the results are representative for all altitudes. The C_D values are normalised according to

$$C_{D_{norm}} = \frac{C_D}{\min C_{D_{\delta_{cs} \neq 0}}} \tag{19}$$

where $C_{D_{min}}$ is the minimum C_D value for the LM with $\delta_{cs} \neq 0$. There is a high difference during the climb and descent phases of the mission between the LM results, with and without trim drag, and the HM results. One of the reasons contributing to this difference is due to extrapolation: when the Mach number falls below the minimum value existent in the tabulated aerodynamics data, the extrapolation routine keeps the C_D value constant and equal to the nearest known value. A different extrapolation routine would probably increase the C_D value by a few percent, but the difference between the legacy and handbook models would still not be eliminated. Although the HM has been used in previous work [22], it was for a more conventional aircraft design. It is therefore reasonable to assume that the rough approximations included in the geometrical model for the double-delta wing might lead to C_D overestimation in the HM.

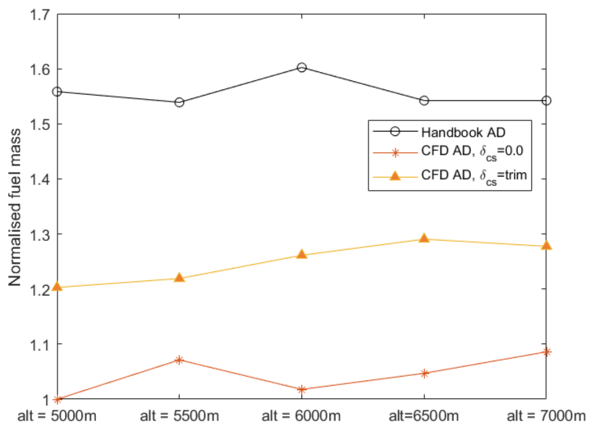

Figure 8 – Total fuel mass $Mass_{fuel}$ required for the three simulation models. Normalised values as in Eq. 17.

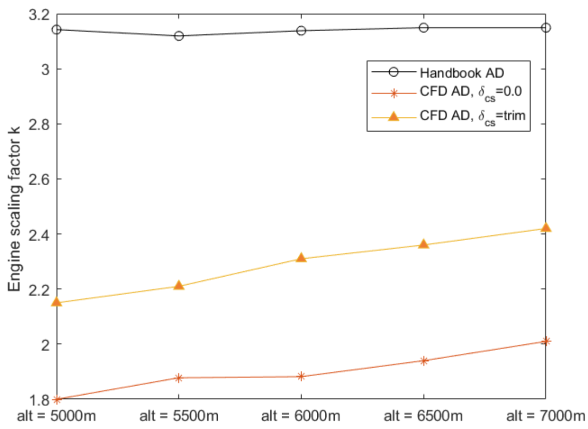

Figure 9 – The scaling factor to be applied on the Williams FJ44 engine on the three simulation models, as in Eqs. 11 and 12

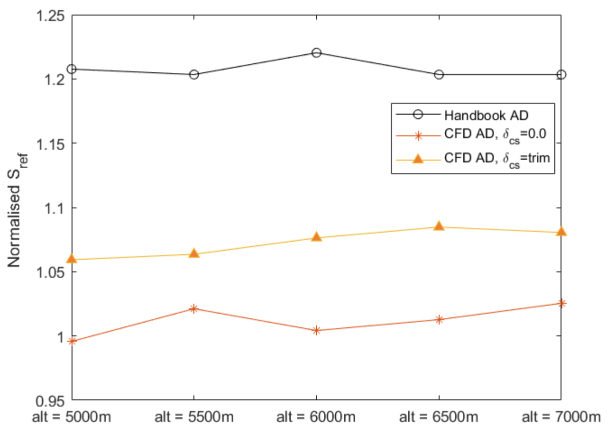

Figure 10 – The reference wing area for the resulting aircraft designs for the three simulation models. Normalised as in Eq. 18

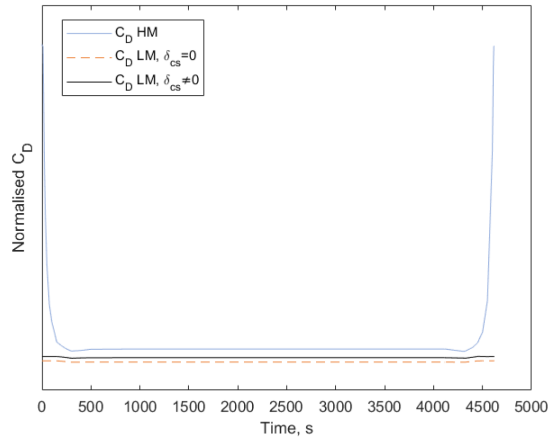

Figure 11 – Drag coefficient values for the three simulation models during the mission at 6000m, normalised as in Eq. 19

Including the trim drag effects ($\delta_{cs} \neq 0.0$) in the LM based simulations increases the aircraft drag, as shown in Fig. 11, and generates designs with higher S_{ref} and higher engine scaling factors, which is in line with the expected behaviour of the model. It is important to note that the δ_{cs} values for trim are constant throughout the mission and are the ones required for steady level flight at the loitering part of the mission. It would be interesting to investigate the behaviour of the LM by adding the δ_{cs} values required for steady climb and descent, as it is difficult to tell whether the current δ_{cs} values lead to over- or under-estimation of the drag in those segments of the mission. Varied δ_{cs} values would increase the level of fidelity of the LM and would lead to more accurate results.

4.2 LM integration method

An important result is the developed method of integrating LMs into simulation frameworks different than their native one, based entirely on open standards [11, 23]. This method has been successfully applied on industry-grade LMs (FORTRAN models originally built for the ARES Simulation Environment that does not support the FMI standard) that have been integrated in a different simulation framework [10]. The method is easy to implement and requires only five steps:

1. Assessment of model suitability for the target system, as per the discussion from Section 2.

This includes identifying which parts of the design space, and the target system ODs, that can be explored using the LM under evaluation

2. Model interface development to ensure use-case compliance and, if native tool FMI support is unavailable, model export according to the FMI standard

3. Integration testing of the exported FMU

4. Model verification to identify any numerical errors. If such errors are present, model or solver adjustments may be needed to mitigate the problems

5. Once exported and verified, the LM is ready to be used.

At the start of this work, it seemed that the most important step to overcome was Step 2. Indeed, developing the required interface to ensure FMI compatibility for the LMs took several weeks and the issues arisen were from the software development domain entirely. However, once all the software dependencies were understood, and the integration interface was built, the process, from an integration perspective, is close to "plug-and-play" for all the other models developed for the ARES Simulation Environment. Moreover, the method can be extrapolated to other tools that use a standardised interface and are based on models developed in C or FORTRAN, as these are, at the time of writing, the only two languages the Modelica Standard supports the integration of.

One of the main time-consumers in the process proved to be Step 4. Depending on the nature of the model, different numerical problems can arise. For instance, a tabulated version of the mission profile, allowing for better mission parametrisation, generated signal continuity problems; adding a first-order delay on the output signals generated signal output errors. However, switching to a simple trapezoid-type of mission, which is included as a Modelica Standard Library component, and adding the same type of first order delay on the output signal, did not generate the same numerical issues, for reasons not completely understood. It is safe to assume that similar unforeseeable issues might arise with the inclusion of other types of models, and their successful solution depends on the experience of the integrator, on their knowledge of the models and of their native and target simulation environment, and even on luck. Such issues highlight the importance of verification and regression testing activities within the integration process.

In comparison, developing the low fidelity aerodynamics model took a few days only. The base model already existed from Ref. [10], however the model needed adjustments in order to fit the double-delta wing configuration that was sought. Although the resulting geometrical model seems to fit rather well with the target design, there is no guarantee that a new design concept iteration would fit similarly well. The approximation of wetted areas and a reasonable calculation of the wing area require extensive trial and error, and a general method to fit all possible designs is difficult to develop (if even possible).

5. Discussion

The assumed sizing task performed in this paper is of a systems engineer developing their own system at a conceptual level. To investigate the impact of the aircraft design on their own system design, and, conversely, to investigate the impact of their own system modifications on the overall aircraft design, the systems engineer has, in this case, two modelling methods: the handbook methods, as in [9] (but others are available, as well), or, as we have shown, reusing existing models from similar projects with similar ODs. The handbook methods have the general advantage that they are easy to understand and provide rules and methods that are easy to implement. However, it turned out that they can be problematic when it comes to uses that require higher accuracy than that of a "rough sketch", and understanding the influence of the different approximation the methods assume on the simulation results can be time-consuming.

The other option, that is the main contribution of this paper, is the reuse of LMs already at the conceptual stage. The are several benefits to this option. First, it partly relieves the systems engineer of the Verification and Validation (V&V) activities that have already been performed for the LM. Second, knowing that the model was verified and validated for the OD that matches the target OD increases the model credibility [24] and the confidence in the generated result; it is important to note, though, that it does not completely eliminate the need for result scrutiny. Third, it opens up the possibility of a more thorough design space exploration already at the conceptual stage, as more LMs could be included in the simulation. Their inclusion might highlight effects and interactions that are difficult to capture with the traditional handbook methods. However, these exploration possibilities are restricted to the part of the design space where the ODs of the target and the legacy systems and models match. It can be concluded, then, that using higher fidelity LMs allows for a more thorough investigation of an already known design, whereas the lower fidelity handbook methods seem to be more suitable for the generation of different versions of a concept.

One of the concerns when including higher fidelity models in any simulation is that the simulation time would be increased. Indeed, an optimisation round with the HM took approx. 600s and 75 iterations, while the LM took approx. 1300s and 50 iterations. Although the increase is more than 100%, the absolute values are still low enough to not argue against the usage of high fidelity LM. Furthermore, the availability of High Performance Computing (HPC) clusters in modern development, as well as efficient methods for paralellisation, can significantly reduce the risk associated with increased computational time.

6. Conclusion

The inclusion of higher fidelity LM already at the conceptual design stage of the aircraft development process resulted in a different, less conservative design than the designed obtained with the traditional handbook methods. The integration process of the LM took longer than the setting up of a HM suitable for the design task; however, once the integration wrapper was developed, the integration of other LM with the same interface became close to "plug-and-play". At a first glance, the computational losses associated with the higher level of fidelity of the LM relative the HM are significant, as the total optimisation time increased by 100%. In absolute values, though, the required optimisation time increased from 10 to 20 minutes, which is negligible, especially given the possibilities of parallelisation and HPC of today. The possibility of a more thorough design space exploration through the inclusion of LM outweighs the computational cost and the initial cost of wrapper development. A possible improvement to the proposed LM integration method is the inclusion of a step evaluating the credibility of the LM in the context of the target system OD through the incorporation of a credibility-assessment component [25]. This would enable a better overview of the exploration opportunities and of the limitations that the inclusion of a LM early in the development process entail. Hopefully, the insights they provide will lead to faster, cheaper, safer and overall better aircraft designs in the future.

7. Acknowledgements

The presented research was conducted within the frame of the NFFP7 projects *Flexible flight mechanics simulation framework* (grant nr. 2017-04889) and *Digital Twin for Automated Model Validation and Flight Test Evaluation* (grant nr. 2019-02760) and the ITEA4 project *EMBrACE*. The research was funded by Vinnova and Saab Aeronautics. The authors would specifically like to thank Adrian Pop, PhD., for his support on the FORTRAN/Modelica integration.

8. Contact Author Email Address

mailto: alexandra.oprea@saabgroup.com

9. Copyright Statement

References

[1] International Council on Systems Engineering. *Systems Engineering Handbook*. 4th ed. John Wiley and Sons, Inc., 2015. ISBN: 9781118999400.

[2] Steven Eppinger, Karl Ulrich, and Maria C. Yang. *Product design and development*. New York, NY: McGraw-Hill Higher Education, 2020. ISBN: 9781260043655.

[3] ISO. *ISO/IEC/IEEE 15288:2015 Systems and software engineering — System life cycle processes*. Standard. ISO, 2015.

[4] Boris Toche, Robert Pellerin, and Clement Fortin. “Set-based design: a review and new directions”. In: *Design Science* 6 (2020), e18. DOI: 10.1017/dsj.2020.16.

[5] SISO. *SISO-REF-039-2013 Reference for Generic Methodology for Verification and Validation (GM-VV) to Support Acceptance of Models, Simulations and Data GM-VV Vol. 3: Reference Manual*. Standard. The Simulation Interoperability Standards Organization, Inc., 2013.

[6] S. Atamturktur et al. “Defining coverage of an operational domain using a modified nearest-neighbor metric”. In: *Mechanical Systems and Signal Processing* (2015), pp. 349–361.

[7] Robert Hällqvist. “On Standardized Model Integration : Automated Validation in Aircraft System Simulation”. Licentiate thesis. Linköping University, Dec. 2019. DOI: https://doi.org/10.3384/lic.diva-162810.

[8] Daniel Simon, Ola Härkegård, and Johan Löfberg. “Command Governor Approach to Maneuver Limiting in Fighter Aircraft”. In: *Journal of Guidance, Control, and Dynamics* 40 (Dec. 2016), pp. 1–9. DOI: 10.2514/1.G002272.

[9] Daniel P. Raymer. *Aircraft design : a conceptual approach*. AIAA education series. American Institute of Aeronautics and Astronautics, 2018. ISBN: 978-1-62410-490-9.

[10] Alexandra Oprea et al. “Connecting System Simulation to Aircraft Concept Development”. In: *Proceedings of the 32nd Congress of the International Council of the Aeronautical Sciences*. 2021.

[11] Modelica Association Project “FMI”. *Functional Mock-up Interface for Model Exchange and Co-Simulation*. Standard. Modelica Association, 2021.

[12] Modelica Association. *System Structure and Parameterization*. Report 1.0. Mar. 5, 2019.

[13] Lennart Ochel et al. “OMSimulator – Integrated FMI and TLM-based Co-simulation with Composite Model Editing and SSP”. In: *Proceeding of the 13th International Modelica Conference*. 2019. DOI: 10.3384/ecp1915769.

[14] Peter Fritzson et al. “The OpenModelica Integrated Environment for Modeling, Simulation, and Model-Based Development”. In: *Modeling, Identification and Control: A Norwegian Research Bulletin* 41.4 (2020), pp. 241–295. ISSN: 1890-1328. DOI: doi:10.4173/mic.2020.4.1.

[15] Dassault Systemes AB. *Dymola User Manual*. 1st ed. 2016.

[16] Modelica Association. *Modelica® – A Unified Object-Oriented Language for Systems Modeling*. Language Specification. 2021.

[17] The Modelica Association. *Modelica and the Modelica Association*. URL: https://www.modelica.org/ (visited on 05/30/2022).

[18] Maido Saarlas. *Aircraft performance*. John Wiley & Sons, 2007. ISBN: 978-0-470-04416-2.

[19] Robert Hällqvist et al. "Optimal Selection of Model Validation Experiments: Guided by Coverage". In: *Journal of Verification, Validation and Uncertainty Quantification* 6.3 (July 2021). ISSN: 2377-2158. DOI: 10.1115/1.4051497.

[20] J. A. Nelder and R. Mead. "A Simplex Method for Function Minimization". In: *The Computer Journal* 7.4 (Jan. 1965), pp. 308–313. DOI: 10.1093/comjnl/7.4.308.

[21] Pauli Virtanen et al. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python". In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[22] Alexandra Oprea et al. "Aspects of flight simulation development in relation to scientific software: a literature review". In: *AIAA AVIATION 2021 FORUM*. 2021, p. 2998. DOI: 10.2514/6.2021-2998.

[23] Modelica Association Project "SSP". *System Structure and Parametrization 1.0*. Standard. Modelica Association, 2019.

[24] NASA. *Standard for Models and Simulations*. Washington, DC, USA: National Aeronautics and Space Administration, 2008.

[25] Magnus Eek. "On credibility assessment in aircraft system simulation". PhD thesis. Linköping University Electronic Press, 2016. ISBN: 978-91-7685-780-9. DOI: 10.3384/diss.diva-127380.