

VIRTUAL TEST ENVIRONMENT FOR COMPREHENSIVE ANALYSIS OF AIRCRAFT LANDING GEAR IN PRELIMINARY DESIGN USING CO-SIMULATION

L.-H. Lemke*, J. Jobmann*, L. Böhme*, F. Thielecke*

* Hamburg University of Technology, Institute of Aircraft Systems Engineering, Nesspriel 5, 21129 Hamburg, Germany

Abstract

In early design stages, virtual system integration at aircraft level can provide a crucial understanding of cross-system effects. Thereby, cost-intensive and time-consuming redesigns after hardware integration tests are less likely. A particular challenge is modeling landing gear loads and their cross-system effects within virtual system integration tests. This paper presents a virtual test environment for the automatized analysis of landing scenarios. Within this test environment, virtual research aircraft models are used for system integration tests. The multidisciplinary model of the research aircraft, including aerodynamics and fundamental system dynamics, is extended by a co-simulation of the landing gear, optionally including structural dynamics. The relevance of landing gear structural dynamics for system integration tests is discussed. Finally, the applicability of the test environment is discussed for two use cases, the development and evaluation of a landing gear loads observer as well as the determination of target values for active load alleviation functions.

Keywords

Virtual Test, Co-Simulation, Landing Gear, Virtual System Integration

1. INTRODUCTION

Due to the high complexity of modern commercial aircraft, some cross-system effects can only be identified or examined in detail by expensive and time-consuming hardware integration tests in late development stages of the classic V-model development process. In the worst case, this leads to complex and costly design changes or not optimized overall systems.

Model-based virtual methods extend and improve the V-model-based development process. By expanding the V-model to include early virtual validation and verification, cross-system effects can already be recognized and investigated in the preliminary design stages of the subsystems. During this virtual aircraft-level system integration, various physical and hence model domains must be linked. The usage of multidisciplinary simulation models enables the connection of specialized simulation methods for different subsystems. Often these specified simulation methods are created in their own software environments. In these cases, co-simulations are a powerful option for linking software tools and multidisciplinary simulation models. Based on the interaction of these subsystems, realistic behavior of the overall model, including cross-system effects, is achievable.

This principle is used within the virtual integration platform VIPER (Virtual Integration Platform for Education and Research), developed by the Technical University

of Hamburg (TUHH). VIPER enables the virtual integration of aircraft systems within virtual research aircraft via system and component libraries as well as co-simulations. In this way, novel systems can virtually be coupled with adjacent systems. They can be analyzed towards their behavior within the overall system and towards coupling effects across systems, independently of the individual system designer and domain. In addition to the modular integration and simulation within virtual research aircraft, VIPER offers a comprehensive test, evaluation and visualization environment for the detailed virtual analysis of the overall system and its simulation results [1]. For example, in VIPER, the integration of electric flight control actuators (EMA) and the effects of a multifunctional fuel cell as a replacement for the auxiliary power unit or the ram air turbine at the overall aircraft level were investigated [2] [3].

A particular challenge modeling multidisciplinary system couplings is the simulation of interactions of precise landing gear dynamics with system and flight dynamics. Multidisciplinary simulation models are a substantial prerequisite to achieve a precise analysis of the complex landing gear loads within various operating states [4]. These multidisciplinary simulations form a basis for optimizing the design of landing gear dependent components and systems. In order to realize those potentials of VIPER, this publication presents the extension of the test environment

by an extensive virtual landing gear model. Both, a landing gear model with and without structural dynamics are generated to evaluate the necessity for these model fidelities at system integration. Furthermore, this publication demonstrates the subsequent utility of the virtual test environment for two use cases. At first, the test environment is demonstrated for generating realistic sensor data and corresponding landing gear loads to design and test loads observer methods. Additionally, the capability of the test environment to define an evaluation basis for load alleviation functions is shown by determining landing load cases for a holistic load envelope. Although the test environment can be used for arbitrary operating states, this paper will focus only on the landing impact.

First, the modeling and the coupling techniques for the simulation of the complex movement of the aircraft at landing touchdown are presented in chapter 2. To evaluate the extension of the presented test environment, the simulation models are analyzed, and two use cases are discussed in chapter 3. This paper closes with the conclusions in chapter 4.

2. METHODOLOGY AND MODELING

Virtual Research Aircraft (ViRAC) are used at the TUHH for virtual system integration. The ViRAC models are generally based on actual aircraft configurations in order to make the system integration analysis more representative, although they do not represent the exact real-world counterpart. Therefore, basic models are assembled using FlySim library elements. The in-house FlySim library at the FST consists of various models for subsystems with different model fidelities and standard interfaces. Thereby, the fidelity of the simulation can be adjusted for each subsystem according to the needs of the current analysis. This enables the optimization of the ratio between model fidelity and computation effort. Section 2.1.1 presents the basic model used for this paper.

The extension of this basic model by a multibody model of the landing gear is also based on this library approach. Therefore, the aircraft model must be executable with and without the multibody landing gear model, i.e., the landing gear model is another optional subsystem within the FlySim library. This generates special requirements for the coupling strategy of the models. Section 2.1.2 presents the multibody simulation model of the landing gear. Section 2.1.3 and section 2.1.4 present the runway model and the coupling strategy for the co-simulation of the landing gear model, respectively. Finally, section 2.2 introduces the test environment for the simulation and analysis of high numbers of operating states.

2.1. Multidisciplinary model for flight mechanics, system dynamics and landing gear dynamics

Within this publication, the flight mechanical and system dynamics model ViRAC-1 is coupled with a multibody simulation model of the landing gear. ViRAC-1 describes an aircraft configuration based on the VFW-614-ATD by Fokker. The VFW-614 is a short-range aircraft with a seating capacity of 40 to 44 passengers and above the wings mounted engines [5]. The ATD-Version was equipped with a fly-by-wire system and used as a technology demonstrator by Airbus [6]. A flight mechanical model in MATLAB/Simulink for the basic aircraft was already generated and validated in previous works at the FST [7]. Therefore, this model is used for the generation of the test environment.

2.1.1. Flight mechanical model with flight controller and basic system dynamics

The main part of the virtual basic model ViRAC-1 is a flight mechanical model for the simulation of the equations of motion covering all 6 degrees of freedom of the aircraft. Inputs to the equations of motion are the forces and moments from the aerodynamics, the inertia, the gravity, and the engines. In addition to the flight mechanical model, the basic model by default includes some basic aircraft system models, e.g. models of the flight control actuator dynamics. In total, the basic model comprises the following components:

- 6 degrees of freedom equations of motion
- Nonlinear 1-point aerodynamic model including time delay effects of the wing downwash at the horizontal tailplane
- Stall model according to Jategaonkar [8]
- Ground effects on aerodynamics
- Actuator dynamics
- Nonlinear engine model-based on a nonlinear calculation of the N1 shaft dynamics
- Interface for the connection of landing gear systems

Additionally, the model includes a flight controller model with various autopilot modes and a simulation of fly-by-wire control logic. Within this work, an instrument landing system (ILS)-based automatic landing function was added to the flight controller. In the first phase of the ILS approach, the aircraft is vertically guided along a glideslope and laterally guided along a localizer signal. Position deviations from the lateral approach path are settled by means of heading changes using coordinated turns. Combined with continuous disturbances, e.g. crosswinds, this results in a crabbed approach. Altitude deviations from the glideslope are settled by a change in pitch angle. The approach speed is controlled via an automatic thrust system. The controller changes into flare mode after descending below the pre-defined flare height. This mode gradually reduces the sink rate until touchdown by spline interpolation between the sink rate at flare activation and the desired touchdown sink rate.

For the simulations in this work, the de-crab function was not used. That means lateral deviations during

the flare are still corrected by coordinated turns, i.e., sideslip is avoided. This enables drifted landings, as the aircraft's longitudinal axis is neither aligned with the runway centerline nor with the direction of flight in crosswind scenarios. The amount of drift can be defined by the approach speed and the crosswind component. After touchdown, the controller changes into rollout-mode. At activation of this mode, the spoilers are extended, the engines are set to reverse thrust, and the wheel brakes are activated. As only the landing impact is analyzed in this publication, the wheel brake function was disabled. During the rollout mode, lateral deviations from the lateral approach path are settled by means of combined deflections of the nose wheel steering and the rudder.

Additionally to the aircraft model, the environmental model is also part of the basic model and is composed of FlySim library components analogously. The environment model comprises the following components:

- Model of the atmosphere according to the international standard atmosphere (ISA) [9] plus the option of temperature deviations
- Earth, gravitational and magnetic field model using the definitions of WGS84 [10] and EGM96 [11]
- Runway reference point at the runway threshold with selectable elevation above mean sea level
- Idealized ILS system with selectable glide slope angle and threshold crossing height
- Various wind models for continuous wind, discrete wind gusts, wind shear and stochastic turbulence fields according to [12]

Within this publication, only static crosswinds were used. The approach simulations are initialized at pre-defined trim points. Additionally to the trim points, the flight controller settings and target values must be defined initially. Due to the automated changes in controller mode from approach to rollout, no further intervention of the test management is necessary after the simulation start for most scenarios.

2.1.2. Multibody landing gear model

The landing gear of ViRAC-1 is a typical nose gear configuration. The nose landing gear and the main landing gears have two wheels each (twin configuration). In order to achieve a sufficiently detailed modeling of the landing gear components and the inherent dynamics, the landing gear is modeled as a multibody simulation model. This is done using the multibody simulation software ADAMS (Automatic Dynamic Analysis of Mechanical Systems) by MSC Software. All three landing gear legs are simulated in one model, as depicted in Fig 1.

The geometry was approximated using available data from images, flight simulator models, and drawings from the official VFW-614 handbook [13] [14]. The main components of the landing gear are represented as single bodies. An exemplary schematic structure of a main landing gear is depicted in Fig 2. The modeled components include the main fitting (MF) and upper side stay (U-SS), connected to the air-

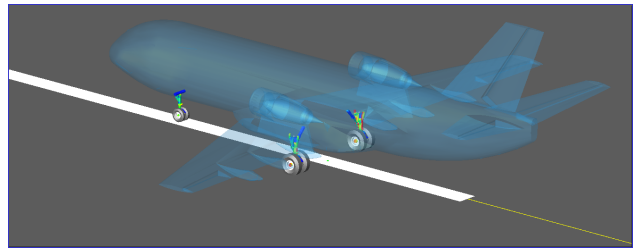


FIG 1. Depiction of all components of the MBS model: landing gear model (with deformations) and runway model as well as an airframe visualization

craft/airframe (AC), the lower side stay (L-SS), upper torque-link (U-TL), lower torque link (L-TL), sliding tube (ST), axle, and the two landing gear wheels (W1 and W2). To connect those bodies, mechanical joints are used. These Joints, represented by the grey elements in Fig 2, are categorized as either revolute (providing one rotational degree of freedom), cylindrical (providing one translational and one rotational degree of freedom), or fixed (providing no degree of freedom). The selected type of the joint is based on its required function. For example, the connection between the upper and lower side stay of the main landing gear is fixed to represent the down-lock mechanism.

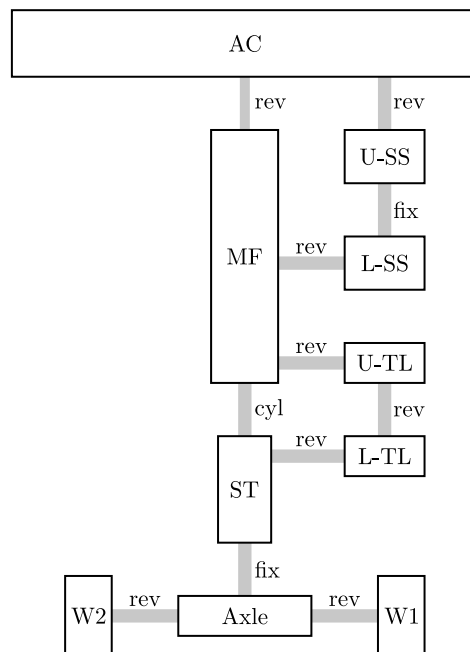


FIG 2. Schematic representation of the bodies and joints of a single main landing gear

In addition to the rigid multibody model of the landing gear, a model including flexible bodies is generated to analyze and demonstrate the difference between a flexible and a rigid simulation of the landing gear bodies. Therefore, Modal Neutral Files (MNF) were generated for each component based on finite element method (FEM) analysis using the software ANSYS. In this work, only flexible components of the landing gear were modeled. The remaining aircraft

structure, like wings and the fuselage, are assumed to be rigid. The effect of a flexible aircraft structure may be analyzed in future work. When generating the flexible landing gear components, modeling flexible effects of the shock absorber comprises a certain complexity. Therefore, the flexible shock absorber is modeled using the add-on LineCon for ADAMS of MSC Software [15]. This enables the calculation of contact forces and moments within the deformed shock absorber. The translational force exerted by the oleopneumatic shock absorber, denoted as F_{SA} , and thus the translational dynamics of the shock absorber, are defined by equation 1. Here, F_{spring} represents the force of the gas spring, which is included using a two-dimensional lookup table as a function of the shock absorber compression s and the ambient temperature T_{amb} . The term F_{damp} depicts the compression and recoil damping force, induced by the existence of oil. The force is modeled using a one-dimensional lookup table dependent on the shock absorber compression rate \dot{s} . The upper and lower stops of the translational shock absorber compression are defined by the force F_{bistop} . It is modeled as a function of the shock absorber compression s , its rate \dot{s} , the lower and upper stop of the compression s_1 and s_2 as well as the terms k , e , c_{max} and d defining the stiffness and damping characteristics of the upper and lower shock absorber compression stops.

$$(1) \quad F_{SA} = F_{spring}(T_{amb}, s) + F_{damp}(\dot{s}) + F_{bistop}(s, \dot{s}, s_1, s_2, k, e, c_{max}, d)$$

Tire dynamics, in addition to shock absorber dynamics, significantly contribute to the overall behavior of the landing gear dynamics. Therefore, it is crucial to accurately represent tire dynamics using an appropriate tire model [16]. ADAMS offers different approaches to include such models. A Fiala model modified for aircraft applications and supplied by ADAMS is used in this case. The Fiala model is a brush-type model-based on a physical description of the tire behavior. Compared with other tire models, the Fiala model only requires the definition of a few physically interpretable parameters. Besides good usability, this enables the adjustment of specific model parameters in order to study the effects of degraded tires, e.g. a modification of the friction coefficient to adapt the model properties to a worn tire. The tire parameters for ViRAC-1, e.g. for the longitudinal, lateral and normal forces within the tire footprint or the rolling resistance and reclining moments in longitudinal and normal directions, respectively, are estimated based on extrapolation of available tire data. The fundamental equations of the Fiala tire model are presented below. For a detailed explanation, refer to [17]. The calculation of the normal tire force F_z is conducted by

$$(2) \quad F_z = \min(0, \{F_{z,spring} + F_{z,damp}\}),$$

where $F_{z,spring}$ and $F_{z,damp}$ represent the vertical tire spring and damping forces, respectively. To compute the longitudinal force F_x within the Fiala tire model, a distinction is made based on the magnitude of the longitudinal slip ratio S_s relative to the critical longitudinal slip $S_{critical}$. If S_s is smaller than or equal to $S_{critical}$, indicating that the tire is in an elastic deformation state, the longitudinal force can be expressed as:

$$(3) \quad F_x = -C_{slip} \cdot S_s$$

using a linear estimation with a local linearization at zero longitudinal slip. Alternatively, when $|S_s|$ exceeds $S_{critical}$, indicating that the tire is fully sliding in the longitudinal direction, the equation

$$(4) \quad F_x = -\text{sign}(S_s) \cdot \left(U \cdot F_z - \left| \frac{(U \cdot F_z)^2}{4 \cdot |S_s| \cdot C_{slip}} \right| \right)$$

applies. The term U in the defined formula describes the coefficient of friction.

The lateral force F_y depends, among other factors, on the slip angle α and C_α , representing the partial derivative of the longitudinal force concerning the longitudinal slip ratio at zero longitudinal slip [17]. Similar to the calculation of the longitudinal force, the Fiala tire model differentiates between two scenarios for computing the lateral force. In the case where the tire is considered to be in an elastic deformation state ($\alpha \leq \alpha_{critical}$), F_y is determined by:

$$(5) \quad F_y = -U \cdot |F_z| \cdot (1 - H^3) \cdot \text{sign}(\alpha),$$

where

$$(6) \quad H = \left(1 - \frac{C_\alpha \cdot |\tan(\alpha)|}{3 \cdot U \cdot |F_z|} \right).$$

For the assumption of a lateral sliding state ($|\alpha| > \alpha_{critical}$), F_y is computed as:

$$(7) \quad F_y = -U \cdot |F_z| \cdot \text{sign}(\alpha).$$

The calculation of the aligning moment M_z within the Fiala tire model also considers both elastic deformation state ($\alpha \leq \alpha_{critical}$) and sliding state ($|\alpha| > \alpha_{critical}$). In the elastic deformation state, M_z is determined by:

$$(8) \quad M_z = U \cdot |F_z| \cdot w_{tire} \cdot (1 - H) \cdot H^3 \cdot \text{sign}(\alpha).$$

where w_{tire} denotes the tire width. Conversely, when in a complete sliding state ($|\alpha| > \alpha_{critical}$), M_z simplifies to:

$$(9) \quad M_z = 0.$$

The rolling resistance moment M_y is also considered under two conditions, both dependent on the rolling resistance R_{tire} . When the tire rolls forward, it is given by:

$$(10) \quad M_y = -R_{tire} \cdot F_z$$

In the case of backward rolling, M_y is computed as:

$$(11) \quad M_y = R_{tire} \cdot F_z.$$

In order to fully calculate the ground reaction of the landing gear, the runway is also included in ADAMS. The relative motion between the landing gear and the runway is achieved by attaching the landing gear to the inertial coordinate system and translational and rotational movement of the runway. This separation of motion is chosen to prevent numerical instabilities and problems within the shock absorber model at the sudden re-positioning of the landing gear at simulation initialization.

In addition to the landing gear structure and the shock absorber, as well as the tire behavior, a nose wheel steering and wheel brakes are implemented. For the nose wheel steering, the steering angle is defined by an input variable and the resulting steering moment is subsequently calculated by ADAMS. This enables the independent modeling of a steering actuator outside of ADAMS. Analogously, braking moments for the wheel brakes are defined by input variables and the rotational speeds of the wheels are subsequently calculated by ADAMS. The brake system or even a direct drive system for the gears can be modeled independently of the MBS model.

2.1.3. Runway model

In order to simplify the co-simulation with respect to the interaction between the runway and the landing gear, the runway is modeled within ADAMS, as this significantly reduces the necessary amount of data to be transferred. At the time of writing this paper, the runway is implemented as a 2-dimensional flat plane without bumps or curvature, as depicted in Fig 1. Nevertheless, those parameters can easily be added for further analysis of ground loads in the future. Other data, e.g. runway stiffness and friction coefficients, are defined within the tire contact model, described in section 2.1.2.

2.1.4. Method of coupling

MATLAB/Simulink and ADAMS are both capable of solving the aircraft's equations of motion. Thereby, two basic options exist for the coupling of both programs. In order to maintain and optimally use the library concept presented at the beginning of chapter 2, here MATLAB/Simulink solves the equations of motions. The MBS model only provides additional forces and moments, which are then included in the equations of motion. Hence, the landing gear model can be seen as an optional subsystem of the total model.

Another advantage of this coupling method is the excellent compatibility of the simulation with a test management using Simulink Test.

Both, MATLAB/Simulink and ADAMS have their own solver and support the Functional Mock-up Interface (FMI) standard for model exchange between different software programs. FMI is an open-source standard for co-simulation, independent of the software [18]. Both models are able to run as the FMI Master or FMI Slave. In this case, MATLAB/Simulink simulates in master mode, and ADAMS will calculate the resulting landing gear reaction in slave mode. The MBS model of the landing gear is exported from ADAMS as a Functional Mock-up Unit (FMU) of version 2.0 according to the FMI standard. The exported FMU is then embedded into MATLAB/Simulink to enable the latter to command the simulation. The concept of coupling can be seen in Fig 3.

Using the FMI interface, current information on the aircraft position is given to the landing gear model. This includes the position $[x, y, z]_{rwy}^T$ of the attachment point of the left main gear to the aircraft structure. This point is given in reference to the runway reference point in the runway coordinate system depicted in Fig 4. Additionally, the aircraft Euler angles $[\Phi, \Theta, \Psi]^T$ are transferred. The resulting landing gear ground reaction is then calculated within ADAMS, and the resultant forces $[F_x, F_y, F_z]_{LDG}^T$ and moments $[M_x, M_y, M_z]_{LDG}^T$ at the three landing gear attachment points are returned to MATLAB/Simulink using the landing gear coordinate system, depicted in Fig 4. Thereby, the landing gear integration into the aircraft's equation of motion is achieved. In addition to this primary coupling loop, secondary coupling loops are generated for system interactions with the landing gear. Therefore, further interface signals to ADAMS are a braking moment for each wheel, the steering angle of the nose wheel steering and the ambient temperature. As secondary outputs, the wheel rotational speeds, the steering moment and the weight-on-wheel signals are returned to MATLAB/Simulink. This enables an independent system modeling of the brakes, potential electric taxiing or the nose wheel steering actuator.

2.2. Test environment

Due to the high number of parameters to vary in the process of development of landing gears (center of gravity position, flap settings, airport elevation, etc.), a great amount of test cases arise. Further use cases for system integrations, e.g. the generation of sensor data for various landing scenarios, provide a high number and high variation of test cases, as well. Therefore, a test chain for the automatized simulation and evaluation of test cases based on Simulink Test is presented in this section. An overview of this test chain is given in Fig 5.

The main element of the test chain is the test management in Simulink Test (SL Test). SL Test is used to simulate predefined scenario groups consisting

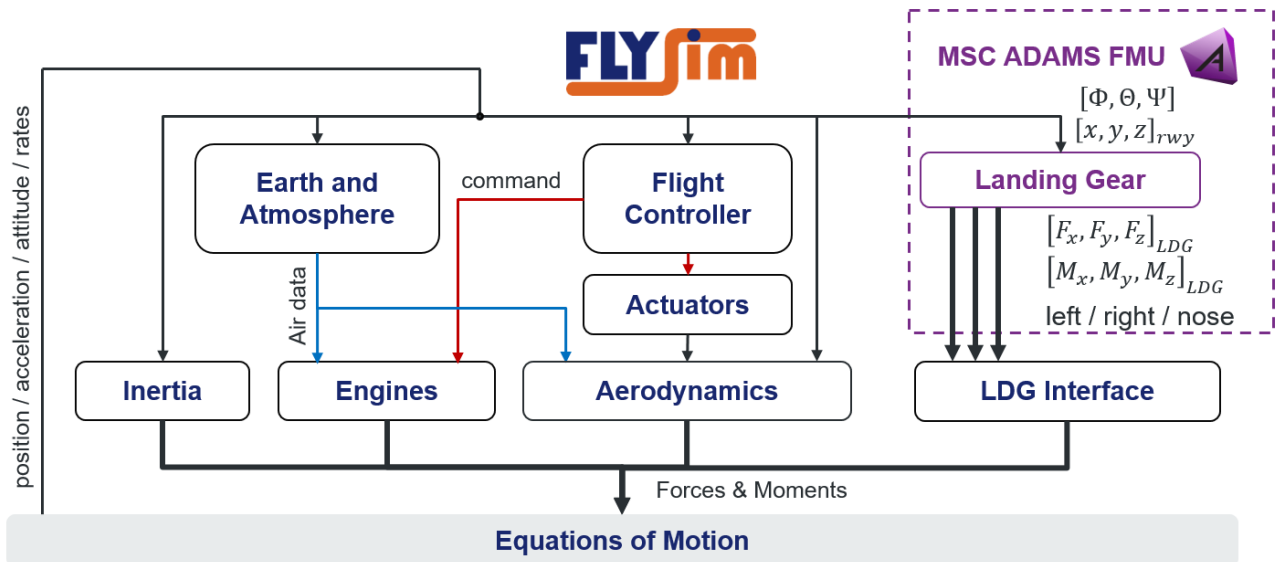


FIG 3. Coupling concept for the co-simulation of MATLAB/Simulink and MSC ADAMS with primary in- and outputs

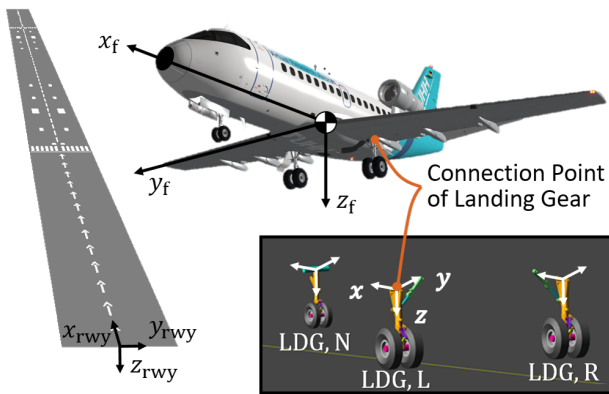


FIG 4. Reference frames of the co-simulation interface

of multiple test cases. The individual test cases are defined in Test Files within Excel sheets. SL Test extracts the initial values and settings for the current simulation using the Scripted Iteration option and writes them into the MATLAB workspace via Post Load Functions. All simulations in the context of this publication were designed such that a complete definition by those initial values and settings is achieved and no further intervention of SL Test during the simulation is necessary, see section 2.1.1. After one simulation has stopped, SL Test runs the following three checks in order to ensure the plausibility of the results:

- 1) Ground contact of the landing gear only between the two runway thresholds?
- 2) Ground contact of the landing gear only between the left and the right runway edges?
- 3) Sink rate at touchdown of the main landing gear within acceptable limits with reference to the pre-defined goal for the touchdown sink rate?

Afterward, all results are passed to the data management tool. The data manager collects all generated data, including e.g. the simulation metadata

and the ADAMS results files of the MBS model and generates a data table. Each row in this table represents one test case and contains all relevant data for further evaluation. Those tables are then saved on a data server. Finally, the results can be visualized using a visualizing tool, which enables a synchronized simulation replay in different software, including a time-pacing function. The visualizing tool currently includes the following components:

- Visualization of the total aircraft movement within the flight simulator XPlane 10
- Visualization of arbitrary signals within the Simulink Data Inspector
- Visualization of typical cockpit gauges using the Aero-Toolbox of MATLAB
- Visualization of the MBS landing gear model within ADAMS view

3. ANALYSIS AND EVALUATION

At the beginning of this chapter, the aircraft behavior at the landing approach will be analyzed in section 3.1. In section 3.2, a comparison of the flexible and the rigid landing gear structural model is made in order to evaluate the necessity of a model including flexible structures for the simulation of different use cases. Finally, in section 3.3, the application of the test environment for two exemplary use cases is discussed.

3.1. Analysis of the approach and landing phase

An autopilot, including an autoland mode based on an ILS system, is implemented in order to simulate different load conditions of the landing gear. Different crosswind scenarios can be simulated, resulting in various landing gear side load conditions and touchdown sink rates. This, in turn, leads to a variety of vertical load conditions. Fig 6 shows the landing approach paths with and without crosswind

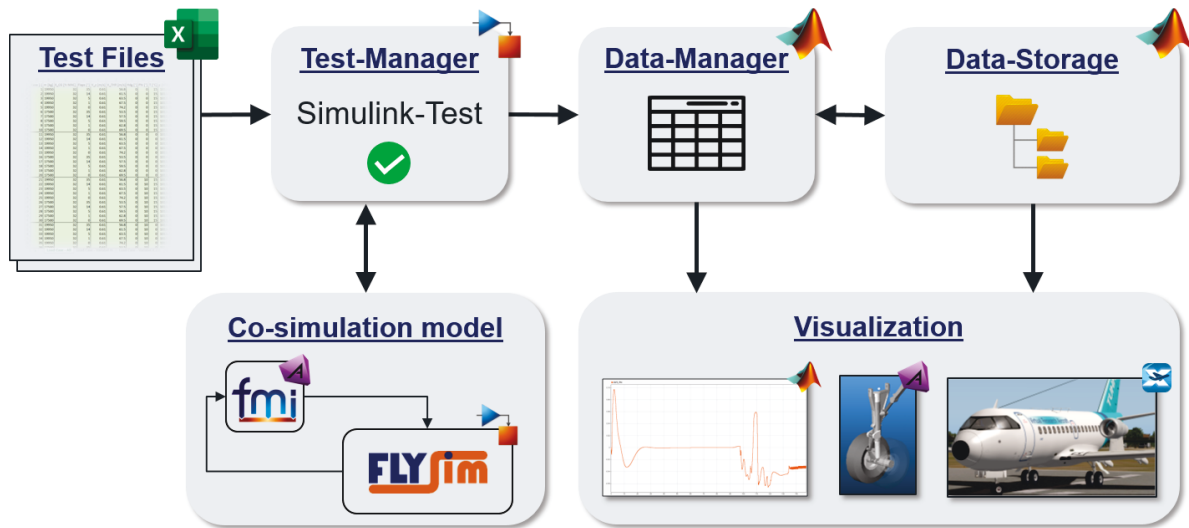


FIG 5. Test environment process chain for the automatized simulation and evaluation of landing scenarios

from the right side. Trim points with a lateral and vertical offset from the ILS path are chosen as initial aircraft positions to demonstrate the ILS capture phase of the controller. For crosswind scenarios, the flight controller adjusts the aircraft's heading by coordinated turns to stay aligned with the runway centerline. This results in a crabbed approach, where the nose of the aircraft points into the wind, and no aerodynamic sideslip is obtained, as depicted by the aircraft symbols in the lower plot of Fig 6. Following this, the de-crab maneuver can be disabled such that no alignment of the aircraft's longitudinal axis with the runway centerline is made prior to touchdown. This is used for high loads tests of the landing gear as drifted landings result in high side loads. Due to the drifted landing, the direction of motion changes shortly after the touchdown. The rollout mode, activated at touchdown, then aligns the aircraft motion with the centerline using a combination of nose-wheel steering and rudder deflection. This can be seen by the temporary deviation from the centerline between approximately 900m and 1400m behind the runway threshold for the crosswind case.

3.2. Comparison of simulations with and without flexibility of the landing gear structure

In the following, results from landing simulations using the model with flexible landing gear structure (FlexLDG) and with rigid landing gear structure (RigLDG), respectively, are compared. This gives an estimation of the influence of landing gear flexibility on the dynamic landing gear loads. Thereby, a recommendation can be given on when to use the FlexLDG-model. This evaluation is of particular interest as the computation effort for a simulation including the flexibility of the landing gear structure is approximately 26.6 times higher than for a simulation of the RigLDG-model.

As this paper only focuses on the landing touchdown, the comparison is performed for landing scenarios

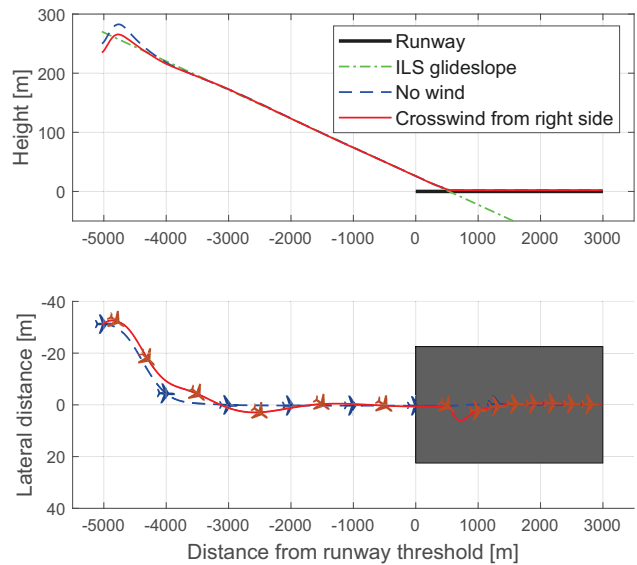


FIG 6. Vertical and lateral paths for the ILS approach with and without crosswind from the right side; grey rectangle displays the runway; Aircraft symbols qualitatively demonstrate the aircraft's heading

without wheel brakes. The results are presented for the left main landing gear. Due to symmetry, they are likewise applicable for the right main landing gear. For the nose landing gear, qualitatively similar results were observed. In particular the aircraft and flight parameters listed in Tab 1 were used for the following test cases.

aircraft mass m_{AC}	18500 kg
approach speed V_{App}	57 m/s
touchdown sinkrate $V_{z,TD}$	1.83 m/s
crosswind (Fig 9 only) $V_{w,cross}$	6.35 m/s

TAB 1. Aircraft and flight parameters for the example landing simulations

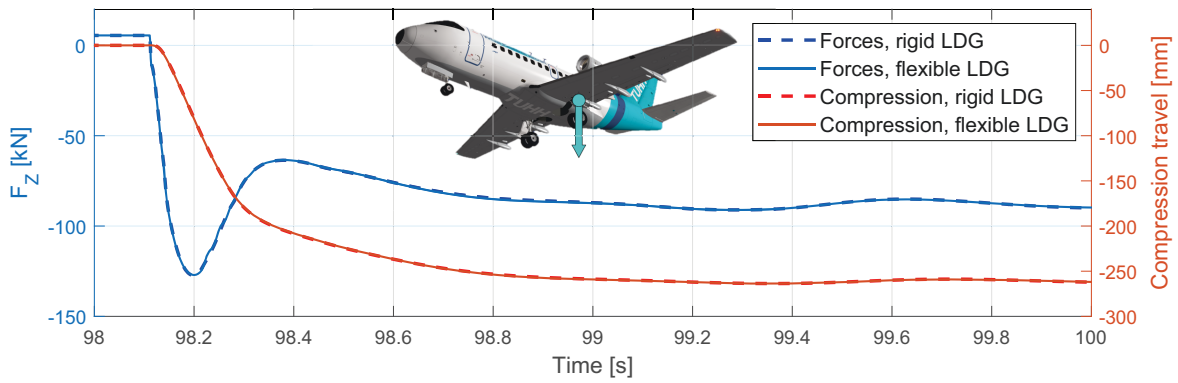


FIG 7. Vertical landing loads at the structure connection point of the left main gear

Fig 7 shows the vertical reaction of the landing gear at touchdown. As can be seen, the vertical load F_Z at the connection point of the landing gear to the wing structure for the RigLDG-model is almost identical to the load calculated with the FlexLDG-model. The measured peak load at touchdown using the RigLDG-model is 0.12% higher than the load using the FlexLDG-model. Therefore, the isolated consideration of the vertical loads does not require the usage of the FlexLDG-model.

Unlike the vertical reaction, the longitudinal response at landing touchdown of the FlexLDG-model significantly differs from that of the RigLDG-model, as can be seen in Fig 8. As expected from the literature [16] [19], the spin-up of the tire causes a significant drag force at the wheel axle. This drag force leads to a backward bending of the landing gear leg, as can be seen by the longitudinal deflection ΔX of the wheel axle from its unloaded position. After the wheels are spun up, the bent leg springs back, which leads to an oscillation of the first landing gear bending mode. The RigLDG-model cannot predict this oscillation and the resulting forces. Therefore, at system integration simulations, when longitudinal oscillating forces are essential for the system of interest, the usage of the FlexLDG-model is necessary.

Similarly, the lateral reaction at landing touchdown differs between the FlexLDG-model and the RigLDG-model, as shown in Fig 9. Here, a crosswind-landing

scenario is depicted, making use of the approach mode and omitting the pre-touchdown de-crab maneuver. Thereby, a drifted landing is generated, which leads to a considerable side loading of the landing gear structure. The first peaks until approximately $t = 100\text{ s}$ result from the drifted landing itself. The slower oscillation from approximately $t = 100.5\text{ s}$ until $t = 102.5\text{ s}$ results from steering controller oscillation at the aircraft alignment with the runway centerline using a combination of rudder and nose wheel steering. Differences in the forces between the two models mainly occur during the drifted landing-induced peaks due to an oscillating bending of the landing gear. The total maximum side load, when including the landing gear flexibility in the simulation, is approximately 12.3% greater than for simulations with the RigLDG-model.

All in all, the RigLDG-model may be used for system integration simulations only depending on the vertical reaction of touchdowns and roughly approximated longitudinal and lateral forces. When dynamic longitudinal and lateral loads are relevant for the system integration simulation, the usage of a model with a flexible landing gear structure is recommended.

3.3. Future use cases of the test environment

In this section, the application of the virtual test environment is briefly demonstrated for two possible use cases. The first use case shows the potential applica-

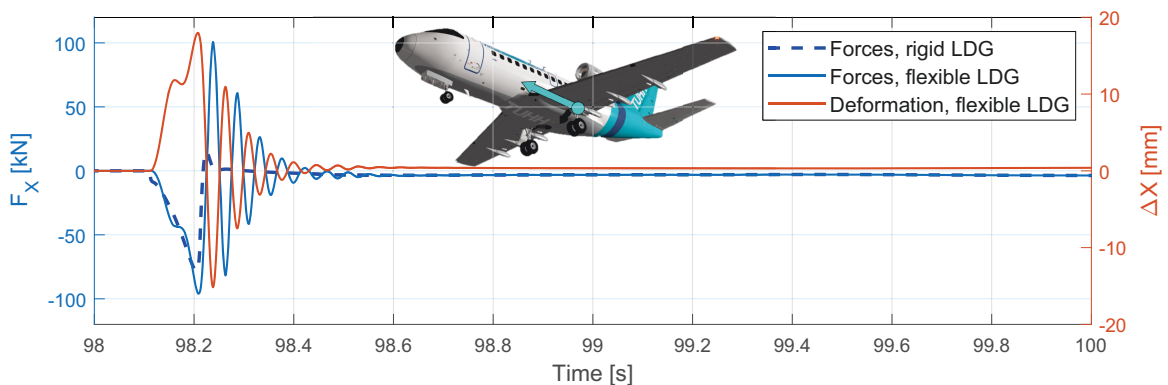


FIG 8. Longitudinal landing loads at the structure connection point of the left main gear

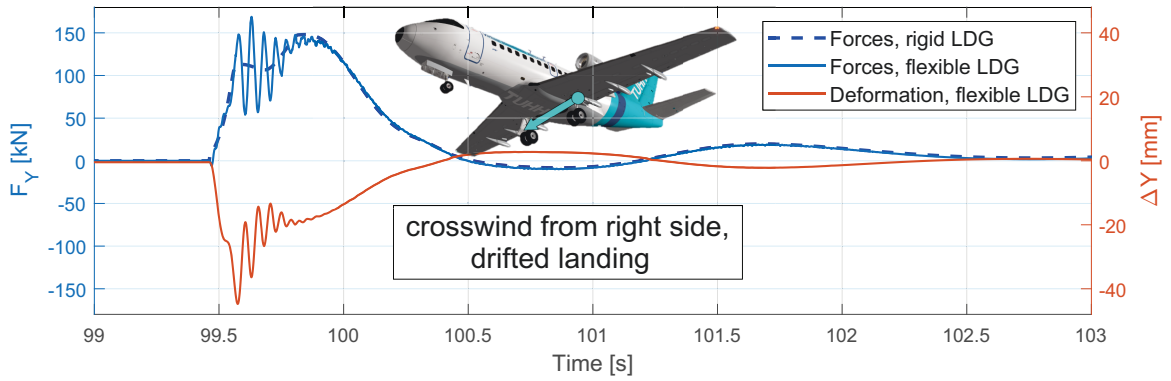


FIG 9. Lateral landing loads at the structure connection point of the left main gear

tion of the test environment to generate realistic and plausible data sets for the development and validation of a loads observer used for monitoring the condition of landing gear structures. In the second use case, the ability to generate landing load envelopes as an evaluation basis for load alleviation functions is showcased. A detailed application and evaluation of the test environment for those use cases will be made in future work.

3.3.1. Use case 1: Realistic and plausible data set for the generation and validation of a landing gear loads observer

Structural components of landing gears are subjected to various loads during take-off, landing and ground operations of the aircraft. During these different operating conditions, overloads can occur, e.g. at hard landings. In addition, the loads of the mentioned operating conditions contribute to constant fatigue of the landing gear structural components. If all the loads that have occurred on the landing gear during the aircraft operation are known, the actual remaining useful life of the landing gear components can be calculated, which is a prerequisite for a condition based maintenance.

In order to be able to evaluate possible overload incidents on the one hand and to be able to monitor landing gear fatigue on the other, landing gear loads observers are necessary. These loads observers usually use the aircraft sensors to determine the landing gear loads without having to attach special load sensors (e.g. strain gauges) to any landing gear components. The loads observers can be validated using flight test data, which comprises data from standard aircraft sensors as well as additional data from sensor hardware, which was explicitly used during flight tests.

However, this flight test data is only available in very late phases of the aircraft development process and can be very cost-intensive and time-consuming to generate due to the large number of tests to be performed. The VIPER test environment presented in this paper enables an early and cost-efficient test and verification of the developed loads observers. With

the included coupled flight dynamics and landing gear model, it is possible to plausibly generate the test scenarios, various operating states and dedicated loads required for validating the loads observers in the virtual environment.

Here, in addition to the recorded default aircraft in-service sensor data, the actual landing gear loads are calculated using the extensive multibody simulation. The actual landing gear loads can then be compared to the observer-predicted loads for verification.

3.3.2. Use case 2: Evaluation basis for load alleviation functions

Within the design process, the aircraft structure in general is designed and sized based on the maximum loads to expect during the life cycle. Load alleviation functions (LAFs), e.g. gust load alleviation and maneuver load alleviation, can be designed in order to reduce specific load cases. Thereby, the structural sizing is affected and a reduction in structural weight can be possible [20] [21]. A possible evaluation variable for LAFs is the load alleviation factor, i.e. the ratio of the resulting load increment $\Delta L_{LAFactive}$ with LAF and the total load increment ΔL_{noLAF} without LAF of the same relevant load case compared to the steady 1-g flight, as defined in equations 12, 13 and 14.

$$(12) \quad K = 1 - \frac{\Delta L_{LAFactive}}{\Delta L_{noLAF}}$$

$$(13) \quad \Delta L_{LAFactive} = L_{LAFactive} - L_{1g}$$

$$(14) \quad \Delta L_{noLAF} = L_{noLAF} - L_{1g}$$

Thereby, an alleviation factor of $K = 1$ corresponds to an alleviation function reducing the loads at the current load case to the level of the equivalent 1-g flight. Likewise, an alleviation factor of $K = 0$ corresponds to an ineffective load alleviation function. Active load

alleviation functions often require an additional system effort, such as faster actuator rates or additional control surfaces on the wing. Hence, a good knowledge of the total system mass increment for different load alleviation factors is important for evaluating LAFs during preliminary aircraft design.

With respect to weight reduction, a higher load alleviation factor of a specific load case is only beneficial when this load case still defines the outer shape of the load envelope of a structural element. Only thereby, a structural weight reduction is enabled. For example, the total reduction of the wing bending moment to the level of the steady 1g-flight case during gust encounter is not reasonable, as the structure must still be designed for the loads at landing touchdown. A reliable reduction of the loads at landing touchdown is very complex to achieve. Therefore, those loads can be used as an upper limit for the aimed load alleviation factor.

The evaluation of specific local loads due to landing touchdown is a particular use case of the here presented virtual test environment. A reasonable set of landing test cases, including all the extreme landing scenarios the aircraft shall be certified for, can be used to estimate those local loads. For demonstration, Fig 10 shows the local loads at the connection point of the left main gear to the wing structure for touchdowns with different flap settings and crosswind components up to a maximum crosswind component of 25 m/s . The here used variations of aircraft and flight parameters are listed in Tab 2. The approach speed is calculated to fit the respective center of gravity position, flap setting and aircraft mass. Those local loads at the landing gear connection point can be used in future work to estimate landing loads at arbitrary points within the aircraft structure when the total aircraft structure is modeled as a flexible body.

As a second option for a faster estimation of the limit loads at touchdown, the aircraft model within the test environment can easily be reduced to depict the basic landing load requirements of the certification specification CS25 (CS25.471 – CS25.485, [22]). In this case, the aerodynamic model is reduced to a single predefined lifting force and the simulation is initialized shortly before touchdown at specified speeds and configurations according to CS25.473 [22]. Both options are implemented within the virtual test environment, enabling its use for this use case in future work when further structural dynamics of the aircraft are included.

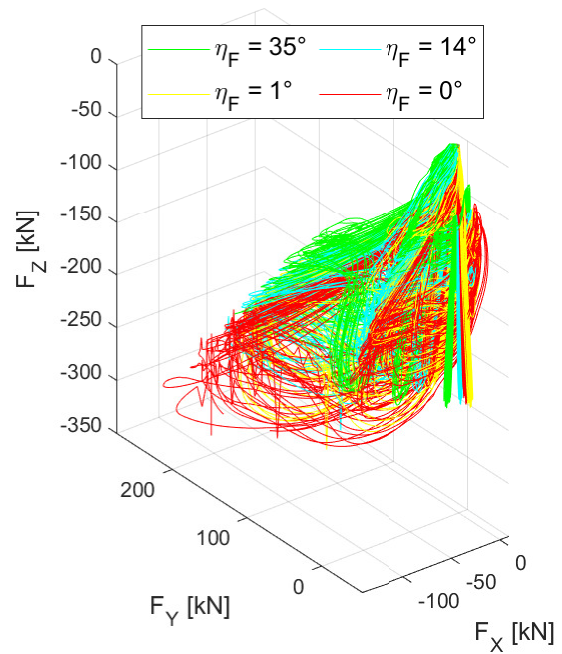


FIG 10. Touchdown induced load envelope of the left main landing gear for a variety of crosswind components and flap settings

4. CONCLUSION

The virtual integration platform VIPER enables an early verification and validation of aircraft systems within the design phase. Especially the investigation of cross-system effects at these early design stages can reduce the required effort for an optimized overall system design. For system integrations at aircraft level within VIPER, virtual research aircraft are used as testbeds. In this work, the virtual research aircraft ViRAC-1 has been extended by a co-simulation of a multibody landing gear model using MSC ADAMS. This is a substantial prerequisite to achieve a precise analysis of the complex landing gear loads within various operating states. Lastly, this enables system integration tests of landing loads-dependent systems. By comparing landing touchdown-induced loads, it has been shown that the inclusion of landing gear structure dynamics is relevant for the modeling of longitudinal and lateral loads. For modeling vertical loads only, a model with a rigid representation of the landing gear structure is sufficient. As the consideration of the structural dynamics of the landing

aircraft mass m_{AC}	[16000 kg, 18500 kg]
CoG position x_{CG}	[9.8428 m, 10.3483 m]
flap configuration η_F	[0°, 1°, 14°, 35°]
touchdown sinkrate $V_{z,TD}$	[0.61 m/s, 1.83 m/s, 3.05 m/s]
approach speed V_{App}	[53 m/s...96 m/s]
crosswind $V_{w,cross}$	[0 m/s...25 m/s]

TAB 2. Aircraft and flight parameters for landing loads envelope generation

gear in the simulation increases the simulation time by more than a factor of 25, a careful evaluation of the requirements at system integration is needed. The analysis of landing loads scenarios leads to a high number of test cases due to the many variables involved, like airport elevation, wind, temperature, glide slope angles, center of gravity positions, etc. Therefore, a test chain has been implemented for the automatized simulation and evaluation of test cases based on Simulink Test. The test chain includes all steps from test case definition to results visualization. The application of the test environment has been demonstrated for two future use cases. First, the generation of plausible data sets for the development and validation of loads observers for the landing gear structure was discussed. In contrast to data sets from real flight tests, the virtually generated sets can already be made available during the design phase, enabling a concurrent development of the observer. Additionally, the applicability of the test environment for the estimation of target performances for load alleviation functions has been demonstrated.

Acknowledgments:

This work was funded by the German Federal Ministry for Economic Affairs and Climate Action within the project OBSERVATOR (contract code: 20D1903C) in the national LuFo program. Their support is greatly appreciated.

Supported by:



on the basis of a decision
by the German Bundestag

Contact address:

lars-hendrik.lemke@tuhh.de

References

- [1] Matthias Peikert, Robert Doering, Lennard Nordmann, and Frank Thielecke. Adapting co-simulation techniques to the virtual integration platform viper. *Proceedings of the 7th International Workshop on Aircraft System Technologies*, 2019.
- [2] Tobias Kreitz, Robert Doering, Sergej Jaeger, and Frank Thielecke. System-level virtual integration study of an aircraft electrical power network. *Proceedings of the 5th International Workshop on Aircraft System Technologies*, 2015.
- [3] Jan Grymlas, Tobias Kreitz, and Frank Thielecke. Scenario-based testing of multifunctional fuel cell systems using the virtual integration platform viper. *International conference on fundamentals and development of fuel cells FDFC*, 2015.
- [4] John Öström, Juha Lähteenmäki, and Tomi Juha. F-18 hornet landing simulations using adams and simulink co-simulation. *AIAA Modeling and Simulation Technologies Conference and Exhibit*, 2008.
- [5] VFW-FOKKER. Vfw 614 brochure.
- [6] Freundeskreis VFW614 e.v. Projekt atd / efcs - vfw614 - der bremer jet. <https://www.vfw614.de/dievw614/forschung-und-entwicklung/projekt-atd-efcs/>.
- [7] Tobias Kreitz, Riko Bornholdt, Matthias Krings, Karsten Henning, and Frank Thielecke. Simulation-driven methodology for the requirements verification and safety assessment of innovative flight control systems. *SAE International*, 2015. DOI: 10.4271/2015-01-2478.
- [8] Ravindra Jategaonkar. *Flight Vehicle System Identification: A Time Domain Methodology: A Time Domain Methodology*. American Institute of Aeronautics and Astronautics, Reston, Virginia, 2006.
- [9] U.S. Government. U.s. standard atmosphere, 1962.
- [10] National Imagery and Mapping Agency. Nima tr8350.2 - department of defense world geodetic system 1984.
- [11] F.G Lemoine, S. C. Kenyon, J.K Factor, R. G. Trimmer, N. K. Pavlis, D. S. Chinn, C. M. Cox, S. M. Klosko, S. B. Lutcke, M. H. Torrence, Y. M. Wang, R. G. Williamson, E. C. Pavlis, R. H. Rapp, and T. R. Olson. The development of the joint nasa gsf and the national imagery and mapping agency (nima) geopotential model egm96.
- [12] U.S. Department of the Air Force. Military specification - flying qualities of piloted airplanes (mil-f-8785c).
- [13] Peter Hager. Peters aircraft. petersaircraft.com.
- [14] Fokker-CFW International B.V. Flight training manual, 1976.
- [15] MSC Software. Adams/linecon: Flexible shock absorber generator, 2016.
- [16] Robert Kyle Schmidt. *The Design of Aircraft Landing Gear*. SAE International, Warrendale, Pennsylvania, 2021. DOI: 10.4271/9780768099430.

- [17] MSC Software. Adams tire user's guide, 15.02.2024. file:///C:/Users/larsl/Downloads/Adams_2021.0.2_Adams_Tire_User_Guide.pdf.
- [18] Modelica Association. fmi functional mock-up interface - the leading standard to exchange dynamic simulation models, 24.02.2023. <https://fmi-standard.org/>.
- [19] Jan R. Wright and Jonathan E. Cooper. *Introduction to aircraft aeroelasticity and loads*. Aerospace series. Wiley, Chichester West Sussex England, second edition edition, 2014.
- [20] Vega Handojo. Contribution to load alleviation in aircraft pre-design and its influence on structural mass and fatigue.
- [21] Kautuk Sinha, Thomas Klimmek, Matthias Schulze, and Vega Handojo. Loads analysis and structural optimization of a high aspect ratio, composite wing aircraft. *CEAS Aeronautical Journal*, pages 233–243, 2021. DOI: [10.1007/s13272-021-00494-x](https://doi.org/10.1007/s13272-021-00494-x).
- [22] European Union Aviation Safety Agency. Certification specifications and acceptable means of compliance for large aeroplanes cs-25: Amendment 26.