

# Modeling and Simulation for the Automation of Aerial Refueling of Military Transport Aircraft with the Probe-and-Drogue System

Nicolas Fezans\* and Thomas Jann\*

This paper presents the modeling and simulation activities currently conducted at DLR to support aerial refueling automation research activities. During the air-to-air refueling maneuver the tanker and receiver aircraft need to fly very close to each other, which induces a very significant aerodynamic interaction between them. Both aircraft also influence the motion of the probe-and-drogue, which needs to be modeled as well. The paper gives an overview of the developed model and simulation infrastructure, their main functionalities, and of the work needed to produce the data supporting this modeling. Unlike in many other work with similar modeling needs, RANS CFD computations were preferred to more simple techniques for the modeling of the aerodynamic interaction between tanker, receiver, hose, and drogue. The real time capable dynamic model is based on two complete aircraft dynamic model. A complete set of simulation programs (including all the complexity of modern airliner flight control systems) is used in connection with each of the aircraft. The coupled simulation programs are deployed in the DLR AVES simulator. For this, many modifications in the AVES core programs and their configurations were needed to extend it from to enable the simulation of a formation of two aircraft: some of these modifications are mentioned throughout the text from a functional point of view but without diving too much into the AVES-specific implementation details.

## I. Air-to-Air Refueling

Air-to-air or aerial refueling (AAR) is a challenging maneuver of strong strategic relevance in present military operations and air power concepts. AAR consists in transferring fuel from one aircraft to another during flight. The transferred fuel is generally meant to be used by the receiver aircraft during the same flight with the aim to remain airborne longer (thereby extending its range and endurance). Many military missions as organized and flown nowadays do require one or several in-flight refueling(s).

There are two main air-to-air refueling technologies currently in use: the so-called probe-and-drogue and boom-and-receptacle technologies.<sup>1</sup> The probe-and-drogue system (see figure 1a) consists of a hose (which can be deployed and retracted in flight) equipped with a cone-shaped refueling basket. The pilot of the receiver aircraft maneuvers the receiver such that the probe is inserted into this refueling basket. This permits to connect both fuel systems and allows the transfer of fuel from the tanker to the receiver. In the probe-and-drogue system, the main part of the refueling system on the tanker's side is usually contained in a pod mounted under the wing. The hose can be deployed through an opening on the aft of the pod (case shown in figure 1a). In the A400M, the refueling system can also be mounted at the aft of the cargo compartment and the hose can be deployed through an opening in the cargo door (door remains closed).

The boom-and-receptacle – also called flying boom – system consists of an articulated tube located under the rear of the tanker aircraft fuselage. It can be deployed and controlled by a dedicated operator by means of two movable airfoils that act as control surfaces as it can be seen in figure 1b. The pilot of the receiver aircraft does not actively establish the contact with the boom, but need to position the receiver within a small volume located under the tail of the tanker and to follow the instructions of the boom operator. A receptacle for the boom is located on the upper side of the receiver aircraft. Once the receiver aircraft is in position, the boom operator drives the boom into that receptacle and establishes the contact. In most

---

\*Research Scientist, DLR, Institute of Flight Systems, Lilienthalplatz 7, 38108 Braunschweig, Germany, AIAA Member.

tanker aircraft, the boom operator station is placed in direct-sight of the boom and receiver aircraft. In recent tankers (e.g. Airbus A330 MRTT and Boeing KC767) the boom operator stations are not located at the rear of the aircraft anymore. The boom operator can still observe the receiver and the boom via video cameras, while benefiting of more comfortable operator station.



(a) Probe-and-drogue system. Author: SSgt. Suzanne M. Jenkins, U.S. Air Force.



(b) Boom-and-receptacle system during the refueling of an F-16 by a KC135 seen from the boom operator station. Author: MSgt John E. Lasky, U.S. Air Force.

**Figure 1. View of the two main air-to-air refueling technologies currently used: probe-and-drogue and boom-and-receptacle. Pictures from the U.S. Air Force, in the public domain.**

The work presented in this paper is part of a broader activity at DLR in which the potential of new automation functions and technologies for air-to-air refueling tasks is investigated. Even if aerial refueling has been used for decades, this remains a very difficult piloting task for which almost no assistance is provided to the pilot in spite of all the possibilities of modern flight control systems. The aim of the whole program is therefore to investigate novel pilot assistance systems for aerial refueling and also to investigate the possible system concepts enabling a fully automatized aerial refueling. The latter is especially interesting for UAV receivers for which the delays in the datalink is making it very difficult (if not impossible) to perform such a high-gain piloting task remotely. In order to be able to develop and assess the various assistance concepts that will be investigated in this on-going research activity, good simulation models are required. Such a model shall include realistic aircraft models (both tanker and receiver) as well as the various components of the AAR system itself. Additionally, representative refueling scenarios have to be defined and the simulation infrastructure must be able to fit all the needs of these scenarios. This paper presents an overview on the on-going AAR modeling and simulation activities performed at DLR with the aim of enabling future research on AAR pilot assistance systems.

The work currently performed at DLR on air-to-air refueling focuses on the probe-and-drogue refueling system. An argument against the probe-and-drogue system and for the boom-and-receptacle system was that the majority of the tanker fleet of the NATO nations belongs to the USA. Another argument is that the maximum fuel rate is higher with the boom-and-receptacle system. However, the fact that the German Air Force and many other NATO partners, European partners, and many other NATO "partner for peace" countries use almost exclusively this system (and will probably continue to use this system in the years/decades to come) motivates to focus on the probe-and-drogue system in the present work. Even if the portfolio of European tankers include tankers equipped with the boom-and-receptacle system (e.g. the different variants of the A330-MRTT and A310-MRTT), these aircraft can be (and usually are) equipped with the probe-and-drogue system. Finally, the A400M which is currently entering into service is expected to represent soon the majority of the European tanker-capable aircraft fleet. The A400M can (at least until now) only be equipped with the probe-and-drogue system. As it can be seen in figure 2, the A400M can also be equipped with a probe and be refueled with this system.

This paper consists of two main parts. Section II focuses on the considered scenarios and the flight physics modeling. Section III presents an overview of the way this model is implemented as well as some selected aspects for which further details are provided.



**Figure 2.** Side and front views of the probe on the A400M. Pictures taken on May 22<sup>nd</sup> 2014 during the ILA Air Show in Berlin, Germany. Copyright N. Fezans.

## II. Scenario and Modeling

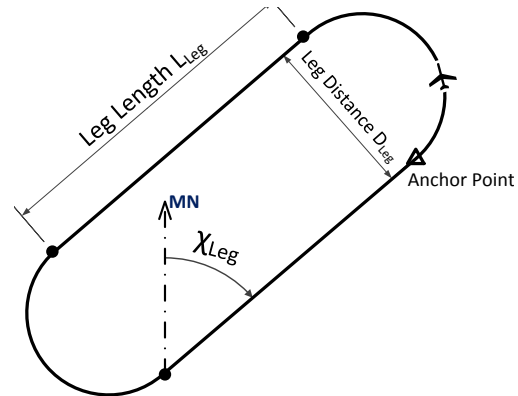
### II.A. Considered Scenario

In terms of aircraft pairing, there are mainly two kinds of operationally relevant refueling scenarios: a large tanker aircraft refueling a significantly lighter aircraft (often a fighter) and a large tanker aircraft refueling another large and heavy airplane (possibly even larger than the tanker). In terms of automation the latter case appears to be more challenging, due to the fact that the large receiver aircraft are usually less agile than the lighter receiver aircraft. Consequently, the considered scenario involves a large tanker aircraft and a large receiver aircraft. Previous work made at DLR led to define the so-called “Future Military Transport Aircraft” (FMTA) as an A400M-like configuration with sufficient differences from the original A400M to benefit from a certain amount of freedom for research and publications. Quite detailed dynamic models of FMTA have been developed during previous projects.<sup>2-6</sup> Besides, a scenario with an A400M refueling another A400M is meaningful from an operational point-of-view. Consequently, only the air-to-air refueling of an FMTA with another FMTA is considered in the manned-aircraft part of this work.

This work is part of a research project in which the refueling of unmanned aircraft is also addressed, however this paper focuses on the manned receiver case. The flight point chosen for this scenario corresponds to a typical refueling flight point for both large aircraft and fighter aircraft (both as receivers): FL200 with  $VCAS = 260$  kts.

### II.B. Operational Procedure

There are plenty of standardized air-to-air refueling procedures<sup>7</sup> within the NATO alliance and among all these procedures the so-called RendezVous Alpha (RVAAlpha) is one of the most commonly used. As a consequence, the RVAAlpha procedure has been selected for the present work. The RendezVous Alpha procedure takes place along a predefined holding pattern that is specified by an anchor point, a leg orientation ( $\chi_{Leg}$ ), a leg length, and the distance between the legs (see figure 3). The used procedure is expected to have no or only very little influence on the evaluations made of the various assistance/automation functions developed during this work, such that – at the current stage of the investigations – it does not appear useful to consider more than one procedure.



**Figure 3.** Race-Track Holding Pattern Followed by the Tanker and the Receiver.

## II.C. Tanker and Receiver Flight Dynamics and Their Aerodynamic Interaction

### II.C.1. Origin of the Interaction and Use of Computational Fluid Dynamics to Produce the Required Data

When flying far from each other, both aircraft can be seen as separated models with no interaction whatsoever. When both aircraft begin to fly relatively close to each other but still with no significant aerodynamic interaction (e.g. slightly different altitudes or lateral positions) some coupling must be introduced between both models. In this case, their dynamics stay uncoupled but the external conditions encountered by each aircraft (weather, temperatures, turbulence, etc.) are now partly correlated and this correlation must be modeled.

Finally, when one of the aircraft (usually the receiver) approaches or flies into areas in which the wind field velocities induced by the other aircraft (usually the tanker) are large enough to induce a significant change of the aerodynamic forces and moments, these effects must be modeled. Most of the aerodynamic interaction between both aircraft during aerial refueling occurs at distances (along the flight path) lower than a few wingspans of the leading/tanker aircraft. In this type of situations, approximations of the induced flow field using the various well-known wake-vortex models reach their limits because the roll-up process is not achieved at such short distances and secondary effects (e.g. due to the engine blow/propellers and due to the fuselage) cannot be fully neglected. The approach shown in Ref. 8 shall be able to tackle the vortex roll-up issue and constitute a good compromise between complexity and precision but still lacks the modeling of the secondary effects.

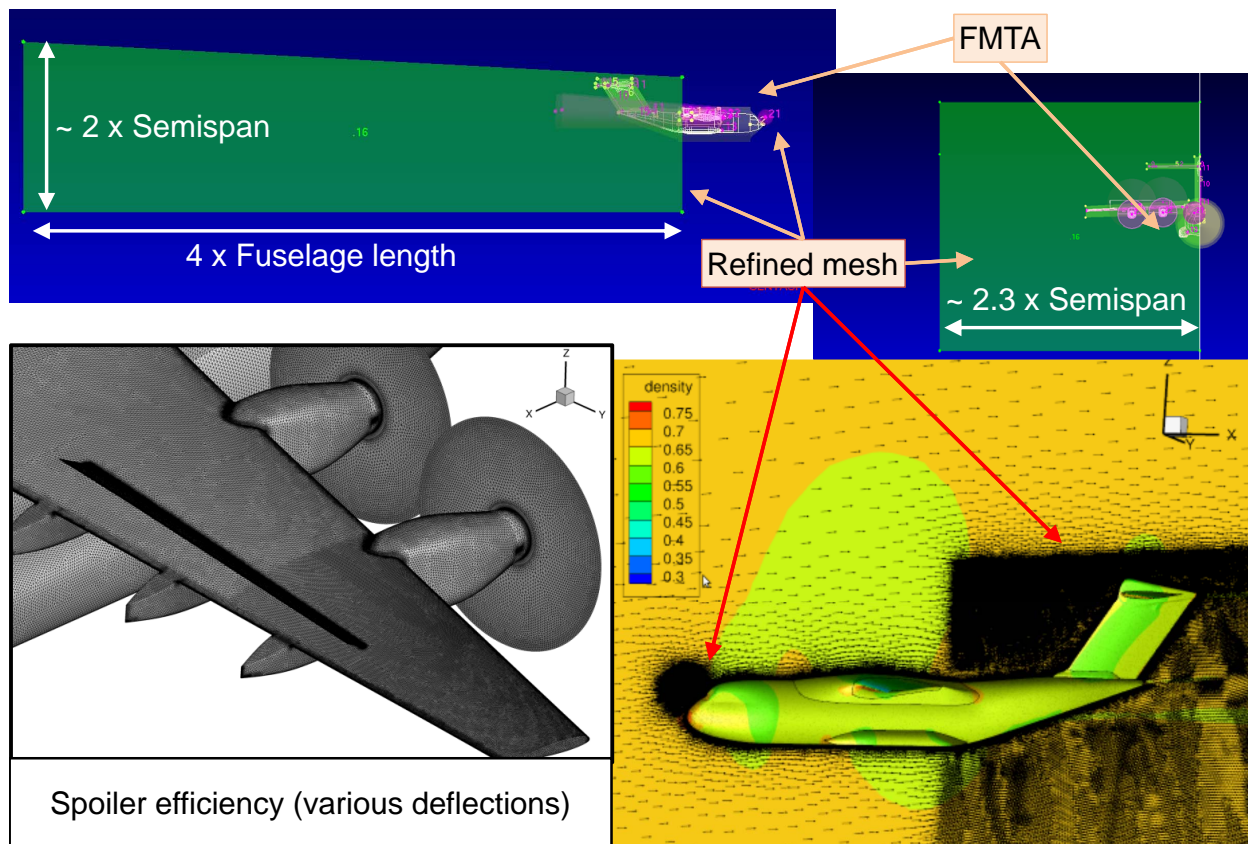
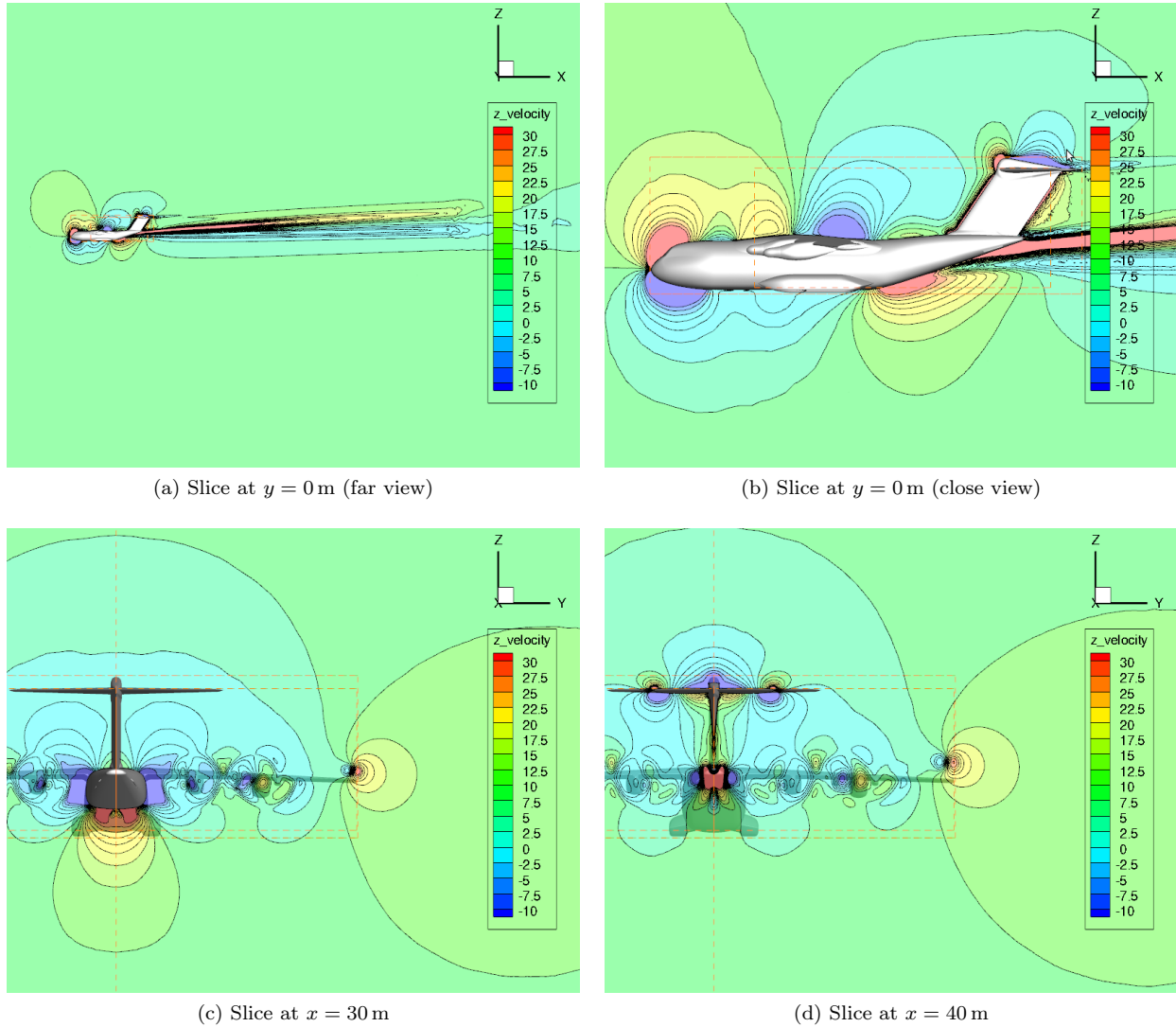


Figure 4. Overview of the CFD Computations Supporting the Modeling Activities.

Due to the need for a wide ramp door, the aft of the fuselage of a military transport aircraft is generally less aerodynamically efficient than those of a typical airliner and the corresponding effects are not very easy to compute with simplified approaches. For the proper computation of the flow field in the case of the air-to-air refueling with a probe-and-drogue system at the aft of the cargo compartment (i.e. not attached to a pod under the wing) and whose hose passes through the ramp door, RANS CFD seems to be the most appropriate technique. Panel methods might provide interesting results, but in the absence of adequate



validation data, RANS CFD was preferred. A series of steady RANS CFD computations were performed with the DLR TAU flow solver<sup>9–11</sup> to compute the complete wind field around and behind the tanker. In order to prevent the dissipation of the wake in the computed solution, the mesh was specially refined in the interesting areas, as shown in figure 4. A large box of roughly 4 fuselage lengths  $\times$  2 semispans  $\times$  2.3 semispans was refined in the mesh (half model computations). Additionally, the domain around the aircraft nose was also refined in order to compute precisely the so-called bow wave effect. The bow wave describes the disturbed flow field around the nose of the receiver aircraft close to the probe, which tends to push the drogue away from the aircraft nose when trying to establish contact with a probe located near to the aircraft nose/bow.<sup>12–17</sup> Finally, in these CFD computations, the propellers are modeled as actuator disks.



**Figure 5. Illustration of the Typical Flow Characteristics That Can Be Found in the Wind Fields Computed with RANS-CFD.**  $V_{CAS} = 260$  kts,  $H = 20,000$  ft.

Various slices across one of the CFD solutions are shown in figure 5. Note that in this figure, the coordinates are body-fixed, with the origin roughly at the aircraft nose, and the signs as typically defined in aerodynamics ( $x$  pointing backward,  $y$  to the right, and  $z$  upward). Consequently, a  $z$ -velocity that is higher than the average, far-field  $z$ -velocity corresponds to an induced upwash, whereas a  $z$ -velocity that is lower than the average, far-field  $z$ -velocity corresponds to an induced downwash. For instance, the bow-wave effect<sup>12–17</sup> can be seen in figures 5a and 5b by the induced downwash (i.e. cyan/blue) under the aircraft nose and the induced upwash (i.e. yellow/orange/red) above it. In these figures, it can also be observed that directly behind the ramp door, a significant upwash is induced, even if the wake vortices eventually (i.e.

further behind the aircraft) are dominating (downwash). The receiver aircraft wings will be located in this downwash during the refueling, but quite different flow conditions are encountered along the refueling hose. Figures 5c-5d also show the vertical velocity in two separate planes at the tail of the aircraft. The beginning of the vortex roll-up can be observed through the upward induced wind right from the right wing tip and the slight downwash behind the wings. The latter is more difficult to observe due to the propeller swirls and due to the effects of the high-lift fairings and of the engine nacelles.

The high-quality CFD results are a great support for the modeling of the flow induced by the tanker aircraft and around the nose of the receiver aircraft. After having imported the TAU solutions into the software packet TecPlot 360, the wind velocities in the CFD results are exported for all points of some rectangular grids. These data will be imported in the real time simulation model and currently various grids and resolutions are still being tested. The amount of data is such that a compromise must be made between the precision and the memory usage to load all these data. At this stage of the investigations it seems quite likely that the final implementation will use representation with a location-dependent resolution: A higher resolution will probably be used around particular flow field features (e.g. the wake vortex cores), whereas a lower resolution will probably be used in areas where the induced wind is relatively homogeneous.

### *II.C.2. Modeling the Impact of Inhomogeneous Flow Onto the Receiver, the Hose, and the Drogue*

Once the inhomogeneous flow field induced by the tanker had been determined, its impact onto the aerodynamic forces and moments for the receiver need to be modeled. This corresponds exactly to what is known in the wake-vortex community as “Aerodynamic Interaction Model” (AIM). A good overview of the different approaches for generating an AIM can be found in Ref. 18 (section IV) and the references therein. The strip method from Refs.19,20 is adapted to the FMTA geometry and used in the current work.

### *II.C.3. Improved Spoiler Effectiveness Model*

As suggested by the lower-left part of figure 4, CFD computations were also used to improve the FMTA model regarding the spoiler aerodynamic efficiencies. This was required because some of the flight control concepts that will be investigated for the aerial refueling phase involve the use of the spoilers as a mean for direct lift control (DLC). This allows the receiver to partly control the lift independently from the angle of attack. A symmetrical deflection of the spoilers (similar but not necessarily equal in behavior to the speedbrake function) reduces the lift generated on the wing: by commanding a permanent deflection of the spoilers and by dynamically increasing or decreasing this deflection the lift can be respectively decreased or increased. Symmetrical spoiler deflections have also other effects on drag and on the pitching moment, which can be both compensated such that a pure direct lift control can be realized in practice.

## **II.D. Probe-and-Drogue Modeling**

The selected scenario presently considers the centerline hose and drogue refueling (figure 6), but can also be adapted to refueling systems using PODs. The flexible hose is controlled by the so-called hose drum unit (HDU), which is located inside the FMTA tanker aircraft directly behind the closed ramp door. When fully deployed the hose has a trailing length of 24 m ( $\approx 80$  ft), while a diameter of 0.068 m (2.6 inch) and a specific “wet” weight of 4 kg/m were assumed according to Ref. 21. The high speed drogue is modeled with a drag area of 0.186 m<sup>2</sup> (2 ft<sup>2</sup>) and a weight of 30 kg (65 lbs).

Two modeling approaches are used:

1. a physically deduced multi-body model and
2. a simpler replacement model,

which are implemented as a subsystem in the overall model including tanker and receiver aircraft. The first approach is a lumped mass model with 50 coupled point masses. In figure 7 the masses are located in the joints 1 to 50 connected by rigid or semi-rigid (elasticity only in longitudinal direction) and mass-less segments. Each of these point masses has a drag coefficient associated with it and reacts accordingly to the local airflow at its position. For reeling the hose out or in, only the closest segment to the HDU are modified between zero and maximum length, until the next joint appears or disappears at the fairing. The drogue is modeled as rigid body, having a conical geometry and adequate aerodynamics creating also side and lift forces. This is important in order to respond realistically to the bow wave. A contact model takes into account the conical inner and outer geometry of the drogue creating the proper reaction in case of collision

between the probe and the inner or outer drogue surfaces. This approach is quite similar to the modeling made in Refs. 21–25 for the whole assembly. Refs. 14, 26 contain interesting data for the aerodynamic modeling of the drogue. A higher-order modeling approach of the hose – based on flexible elements – was proposed in Ref. 27: this approach shall permit to reduce the number of elements, compared to a model based on rigid or semi-rigid elements. However, the equations are significantly more complicated when using flexible elements and therefore this approach was not selected for the present work.

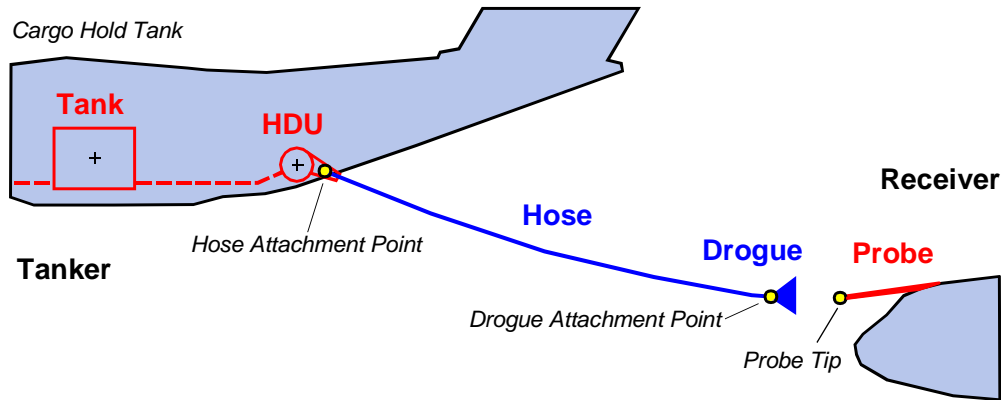


Figure 6. Sketch of the aerial refueling configuration with the probe-and-drogue system.

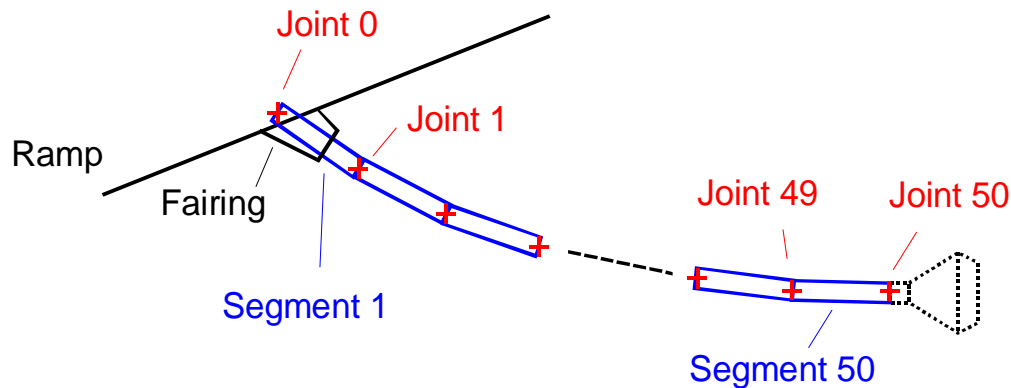


Figure 7. Geometrical representation of hose model using 50 rigid segments connected by 51 joints.

The multi-body model of the hose and drogue allows studying dynamic effects of the hose and drogue system like hose whip,<sup>23–25, 28</sup> whereas the simple model approximates the behavior only in a very simplified manner. In the simple model, the drogue is modeled as a point mass reducing the aerodynamic forces to drag only. The hose is modeled as a mass-less line with a viscoelastic characteristic. The shape of the trailing hose is approximated by a parabola providing also the coordinates of the single joints for the subsequent visualization as illustrated in figure 7. In this simplified model the contact between the probe and the drogue is modeled with a unidirectional magnet-like attractor, which lets the drogue being caught by the probe tip when coming close enough.

Figure 8 shows the modules of the refueling subsystem. The block structure is equal for both the simple and the multi-body hose/drogue models. The external interfaces of both models are identical, such that they can be exchanged easily in the simulation programs. The refueling control signals are connected to the so-called tanker control station, a virtual tanker cockpit, providing (among others) the hose reel in/out command, preset fuel quantities, commanded fuel flows, as well as corresponding indicators for offloaded fuel, signal light status, and many others on an emulated AAR panel. For readability reasons in figure 8, a block called "Wind & Environment" is placed aside of all other blocks (and with no connection to them) to symbolize that the wind field surrounding both aircraft as well as other physical properties – such as gravity and atmospheric quantities – are impacting/interacting with all parts of the model.

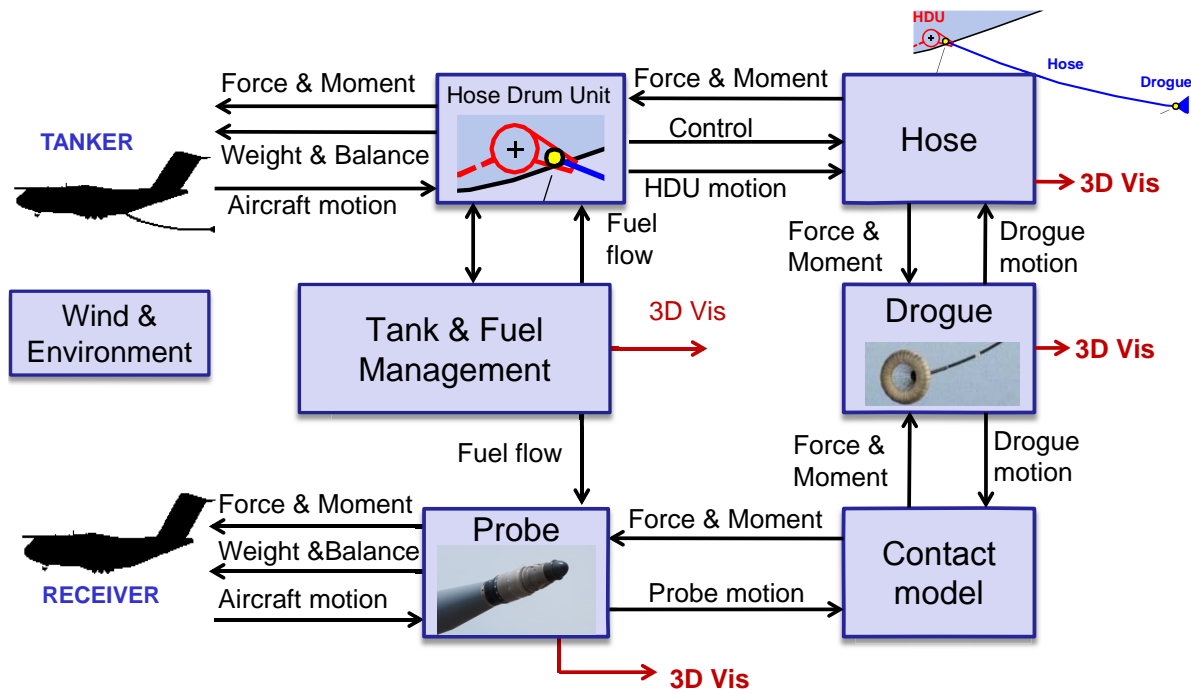


Figure 8. Overview of the probe-and-drogue refueling model internal structure (“3D Vis” = link to the 3D visualization program).

## II.E. Refueling Systems Modeling

To ensure compatibility among the different tanker and receiver aircraft used by NATO nations, the typical air-to-air refueling equipment and procedures are standardized (see section II.B) and described in corresponding documents.<sup>7</sup> For modeling the refueling system here, the refueling process is subdivided into 10 phases numbered from 0 to 9:

- 0: drogue inside pod or HDU
- 1: drogue reeling out
- 2: full trail, clear contact
- 3: contact, but not in refueling zone
- 4: in refueling zone, fuel flow
- 5: in stand-off zone, fuel flow
- 6: in cut-off zone
- 7: refueling finished, disconnect
- 8: separated
- 9: drogue reeling in

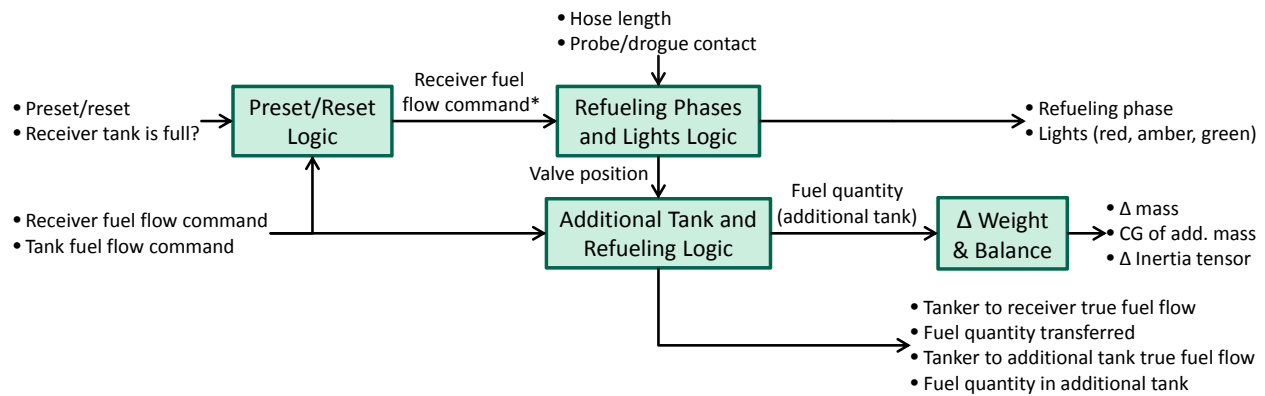
Phases 0, 1, and 9 are active before and after the actual refueling process. The refueling process starts with phase 2 when the receiver actively enters the so-called refueling box. This is the safe and permitted area behind the tanker where probe-drogue contact and refueling shall occur. The transitions between the phases are controlled by the hose retraction status, the contact status, and the (true) fuel flow to the receiver. The refueling status is indicated to the receiver pilot using red, amber and green signal lights near the opening through which the hose is deployed (see later in figure 12a). Phase 2 and 3 are indicated by a steady amber signal light.

To enable fuel flow, the receiver has to push the drogue until the hose is retracted by the HDU and reaches the refueling zone, which is indicated by markings on the hose. The hose retraction is accomplished by controlling the tension in the hose. In the model this controller is located together with the actuator model in the HDU block. Within the refueling zone (phase 4) the fuel valve is opened and the transfer of



fuel is indicated by a steady green light. If the drogue is pushed in too far, the hose leaves the refueling zone and enters the stand-off zone (phase 5). In this phase fuel is still transferred, but an additional flashing amber light indicates the receiver pilot to fall back towards the refueling zone again. Beyond the refueling and stand-off zones the green light disappears, the amber light remains flashing, and the fuel flow is cut off (phase 3 and phase 6). When the refueling for one receiver is finished (phase 7), a flashing green light informs the receiver pilot to disconnect (phase 8) and leave the refueling box, allowing the next receiver - if applicable - to be refueled. All in all the phases presented before are quite similar to the phases used in Refs. 29,30, which is not very surprising since a very detailed NATO standard<sup>7</sup> has been used and kept up-to-date for a very long time.

The “tank and refueling management” block of the diagram of figure 8 is further detailed in figure 9. The core of the “tank and refueling management” is a state machine that is located in the block called “refueling phases and lights logic.” The three other blocks shown in figure 9 (“preset/reset logic,” “tank and refueling logic,” “tank weight and balance”) are responsible for managing the fuel transfer from tanker to the receiver and its implications.



**Figure 9. Internal structure of the “Tank and Refueling Management” subsystem from figure 8.**

The first block allows using a preset value for fuel quantity that shall be transferred. When this value is reached, the refueling system automatically stops the fuel transfer and signals “disconnect” to the receiver. The preset value is set in the AAR panel of the tanker, where also a reset can be commanded in order to refuel the next receiver. The second block named “refueling phases and lights logic,” as explained earlier, is responsible for stepping through the phases, setting the light signals accordingly and opening or closing the valve for fuel transfer. However, fuel can only be transferred if enough fuel is available in the tanker and the receiver tank is not full. These conditions are checked and handled in the “tank and refueling logic” block.

In real AAR scenarios, fuel can be transferred either from additional cargo hold tanks or directly from the own fuel compartments in the tanker. In the model, a cargo hold tank is always present and working as buffer between the tanker and receiver tanks (see figure 6). Accordingly not only the fuel flow to the receiver, but also the fuel flow to the cargo hold tank has to be considered and controlled. Depending on the fuel level in the cargo hold tank the fuel flows to and from these tank are limited and cut off, if one of the limits is reached. Mass, inertia and CG location of the additional cargo hold tank are finally computed in the block “Tank Weight and Balance” and provided to the overall weight and balance subsystem of the tanker model. Note that the mass of fuel in the regular tanks is directly handled (in terms of weight and balance) by the basis FMTA models (tanker and receiver). However, the additional masses related to the Hose-Drum Unit, possible wing pods, additional tanks (and their content) and the probe (as it can be unmounted) are computed in the AAR subsystem and transmitted to the FMTA models such that they are properly taken into account in the equations of motion.

### III. Infrastructure for the Simulation of Aerial Refueling Maneuvers Between Two Future Military Transport Aircraft

This section presents the simulation infrastructure as well as some selected aspects related to the implementations of the dynamic model and of some of the other modules.

#### III.A. High-Level View of the Infrastructure

The simulation infrastructure designed for the FMTA-FMTA case is represented in the diagram of figure 10. The usual software architecture of the DLR AVES simulator<sup>31–33</sup> is used for the receiver aircraft and most of the software modules instantiated a second time for the tanker aircraft. The duplicated modules are identical for both aircraft, but have separated states such that both aircraft can be flown independently from each other. Note that figure 10 shows rather the high-level functional decomposition of the overall architecture and its main modules. The whole architecture is somewhat comparable to the architecture presented in Refs. 34,35, even though the different graphical representations used here and in these references might give the impression that the architectures are completely different. The graphical representation used in the aforementioned references is rather based on the working stations (e.g. cockpit, instructor, operator, simulation observer stations) and the physical computers.

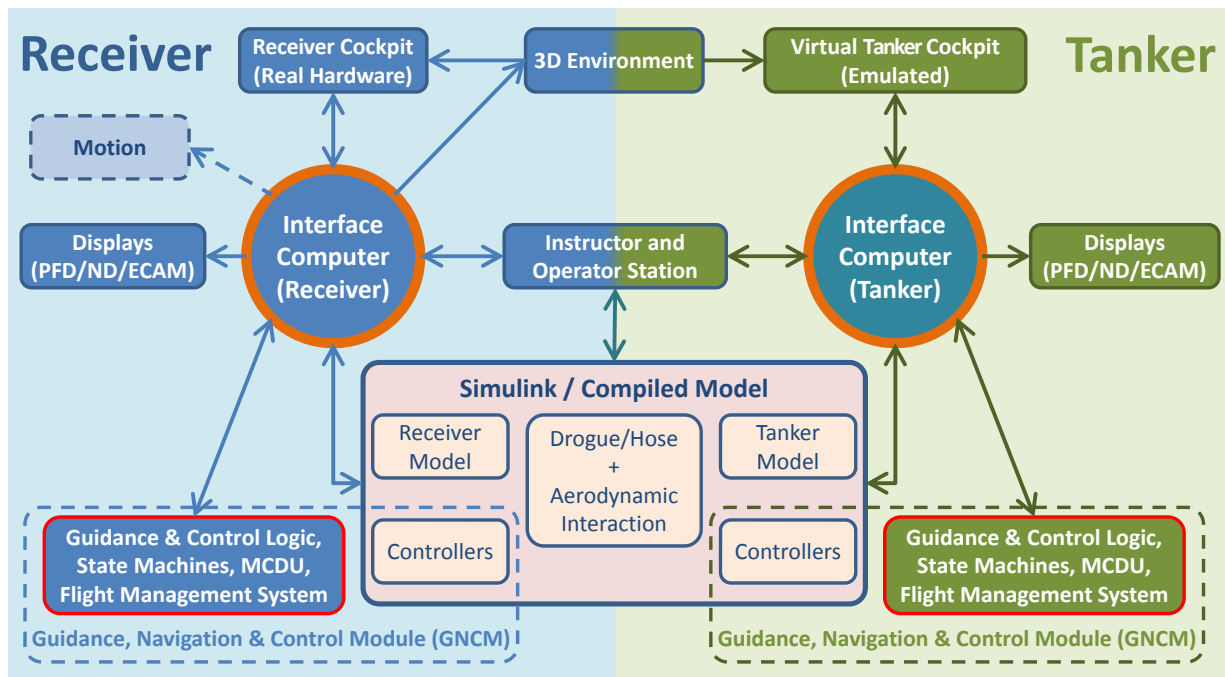


Figure 10. Overview of the different modules of the air-to-air refueling simulation environment

In order to easily ensure that the physical couplings between the aircraft are properly realized and simulated, both aircraft as well as all the flight physics are contained in a single Simulink model. This eliminates the need for a very tight synchronization mechanism (apart from the standard functionalities of Simulink) between the aircraft models since the right order for all computations can directly be determined by Simulink at model compile time. This also permits to avoid having to define data exchanges required for the quite complex couplings between the aircraft in an interface between two modules: these couplings remain inside the Simulink model. As a consequence, the resulting Simulink model is connected to two interface computers (IC): the one that is the central node of the “Receiver” simulation and the one that is the central node of the “Tanker” simulation. The interface definition is the same for both the Receiver-IC and the Tanker-IC connections.

In addition to having only one common Simulink model for both aircraft, two other modules are shared and not duplicated. The first one is the 3D environment and visualization program which drives (among

others) the visual system of AVES (view for the receiver pilots) as shown in section III.C. The representation of the tanker and of the probe-and-drogue in the visual system is crucial for most automation concepts that will be tested later on in the project. Even for concepts where the complete maneuver would be automated, the visual impressions provided to the pilots will influence the evaluation of the system made by the pilots.

The second module that is shared by both aircraft is the so-called “Instructor and Operator Station” (IOS). The IOS is basically the module that controls the whole simulation: it permits to initialize, start, pause, and stop the simulation. It permits to modify parameters during the simulation, to trigger some faults, or just to select and observe some variables in real-time during the simulation.

Due to the fact that only one “real” cockpit is available, a “virtual cockpit” was implemented for the tanker whereas the AVES A320 cockpit is used for the receiver. Consequently, only the receiver pilots will benefit from the motion platform. The priority is given to the receiver regarding the real cockpit and the motion due to the fact that the interesting piloting task when considering the aerial refueling using the probe-and-drogue system takes place in the cockpit of the receiver. The tanker only flies as steady as possible during the maneuver. Nevertheless, the tanker can turn, as for instance to follow the race-track pattern shown in figure 3. Even if active control of the tanker would be used in some of the considered automation concepts, then the control actions would only be taken by the flight control system and not by the tanker pilots, such that these concepts can be tested without a fully equipped tanker cockpit on a motion platform.

### III.B. Simulink Extension for Wind Field and Relative Kinematics Management

MATLAB/Simulink is a very widespread and powerful software package for the modeling and simulation of dynamic systems. Simulink is a graphical programming environment tightly integrated with the numerical computing environment of MATLAB. It is mainly based on a block diagram representation of the system equations (usually ordinary differential equations). Numerous additional packages offer additional functionalities. These functionalities would often be very difficult or time-consuming to program as block diagrams (e.g. large state machines, discrete-event simulation, or component-based modeling). Some additional packages can extend the scope of the analyses made on the modeled systems.

MATLAB/Simulink is an expensive package for research organizations (not considered as academia) and the decision to use additional packages in models that are shared across a large number of people would quickly lead to massive additional license costs. Apart from that none of the proposed additional package was offering a very satisfying way to model combinations of complex kinematics and tabular data.

Other modeling approaches and especially component-based modeling and simulation packages (e.g. Modelica/Dymola, LMS Imagine.Lab Amesim) have been on the market for many years. These approaches are offering a different paradigm than the signal-based paradigm Simulink is based on: both have their advantages and disadvantages. Although several high-quality packages like Amesim have been there for 15 years, the simulation world has not massively adopted a non-signal-based approach. Some of the underlying reasons are certainly to found in the conservatism of many organizations. However, the already existing base of models and the need for engineers to learn a set of new tools and techniques are clearly other obstacles for a paradigm change. Model conversion is not a real option and working with several – costly – tools for years or decades is not affordable either.

All these elements lead to decide to continue using the MATLAB/Simulink environment but with an in-house developed extension that is tailored for the needs of the considered application. This extension is a relative kinematics module, which eases the numerous kinematic computations by providing in the Simulink signal-based block-diagram environment a high-level access to all kinematic quantities. In order to illustrate some of the uses of this extension, two use cases are explained hereafter.

In the first use case, let assume that the receiver is equipped with a camera system that is able to track to motion of a particular marker placed on the drogue. The correct computation of the 3D position and velocity of this marker in a coordinate system attached to the camera is required for building a model of the camera system (and its post-processing algorithms). This computation would involve a series of coordinate transformations and the combination of several simultaneous movements which can certainly be implemented in Simulink, but it might be slightly time-consuming to gather all the required information in the appropriate subsystem, to compute the right transformation matrices, translations/rotations, etc. without dropping terms or making approximations. Besides, the risk of missing one term in the process is not negligible.

The developed Simulink extension enables the users to declare rigid-bodies attached to the different part of the system and to label them. When they need to access some particular kinematics quantity, they only need

to use one of the developed Simulink blocks and to configure it (through its GUI) by declaring the relative motion that they are interested in as well as the coordinate system in which this motion shall be expressed in the output of the block. The corresponding block takes care of analyzing the kinematics paths between these rigid-bodies and of computing the requested quantities by means of all required transformations. In this case (camera on receiver, marker on the drogue) the exact computation can be added to the model within a few minutes, instead of one to two hours when dealing with fairly complex relationships between the involved bodies. This first use case is, admittedly, a case that would be relatively easy to solve with a good multi-body simulation program/additional toolbox.

The second use case is based on the implementation of the wind fields induced by the tanker and by the receiver aircraft. Both of these wind fields are relatively easy to implement separately in the appropriate coordinate systems: a tanker-carried/fixed system is likely to be adequate for the wind field induced by the tanker, whereas a receiver-carried or receiver-fixed system would seem more adequate for the receiver induced wind field (e.g. for modeling the bow-wave). Finally, these wind fields (possibly plus some turbulence, gusts, etc.) are needed to compute forces on both aircraft and also on the hose and on the drogue. The computation of the total flow (relative orientations and speeds) encountered by a section of the hose or of the drogue is required to properly simulate their motion. This involves also numerous transformations of coordinates, interpolations in tables, superimposition, transformations of the resulting wind velocities, and eventually of the resulting forces and moments. Here again, these operations could all be programmed with the Simulink block-diagrams, but this would be unnecessarily time-consuming and error-prone. The developed module not only provides a more efficient way of programming these operations, but also provides an API (Application Programming Interface) that permits, though the inclusion of C++ s-functions in the model, to directly obtain the required information for their computations. A library of such s-functions is included for most common operations and will be extended over time. This mechanism can be used for the computation of the additional aerodynamic forces and moments induced by the inhomogeneous wind field of the tanker and acting on the receiver. Advanced users can also write their own s-functions and directly call the C++ functions provided in the API. To the best of the authors' knowledge, the presently available multi-body simulation software packages do not permit to implement such a mechanism and provide such advanced APIs across components of the model. Note that it is relatively easy to ensure to the proper order of the read and write accesses across the various parts of the Simulink models, but such a mechanism would violate the symbolic derivations of the model equations made in the component-based simulation packages. In principle, it is certainly technically possible to make the underlying symbolic engines be aware of the additional interactions between the components, but it would certainly be a rather complicated mechanism.

From a functional point-of-view, the proposed Simulink extension handles the two blocks called "Relative Kinematics & Sensors" and "Atmosphere/Wind Field Management" shown in figure 11 at once. In this figure, the logical structure<sup>a</sup> of the complete dynamic model implemented in Simulink is shown. The tanker and receiver motions as well as the required information for the update of the wind field (e.g. tanker lift, receiver angle of attack/sideslip) are gathered. Based on this information, the total wind field at any location and expressed in any of the defined coordinate systems (e.g. if needed for the aerodynamic interaction model, or for the bow-wave) can easily be accessed. The same can be done for the relative kinematics sensor measurements (e.g. camera system or differential GPS). At the time this paper is being written the basic functionality of this Simulink extension has been demonstrated on simple examples, but this module is still actively being developed and will still have to undergo through a very thorough testing and validation process. Further details on its internal working will be published in the future.

### III.C. 3D Visualization

In the AVES simulator, the 3D environment used by the visual system must integrate a realistic representation of the tanker, of the hose, of the drogue, and of the receiver probe in order for the pilots to be able to manually control the receiver airplane towards the tanker, establish contact with the drogue, and proceed to the refueling. Such a visualization is required also when pilots shall assess the way an automatic refueling system is behaving. The in-house 3D visualization software used in AVES has been developed since many years on the top of the OpenSceneGraph open source library and possesses – on the top of all classical 3D scene graph features – many less common features, for instance for brown-out and fog effect or realistic water visualization for maritime scenarios.<sup>36</sup>

<sup>a</sup>The real structure in terms of subsystems does not follow this diagram.

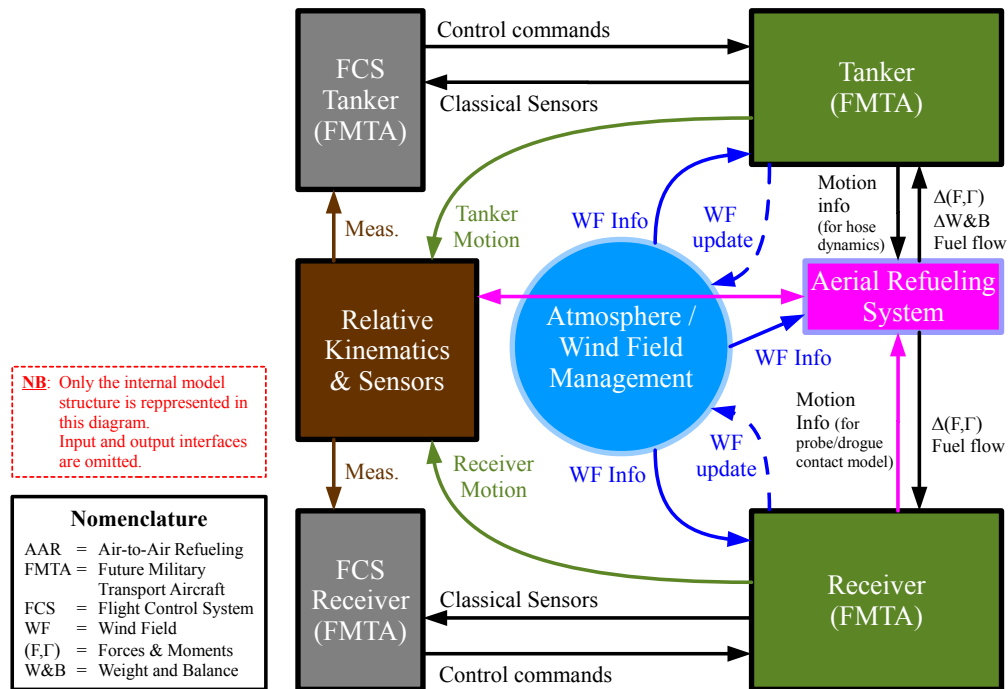
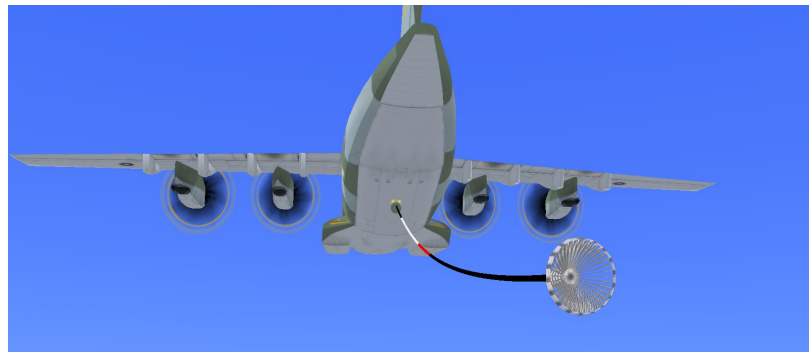
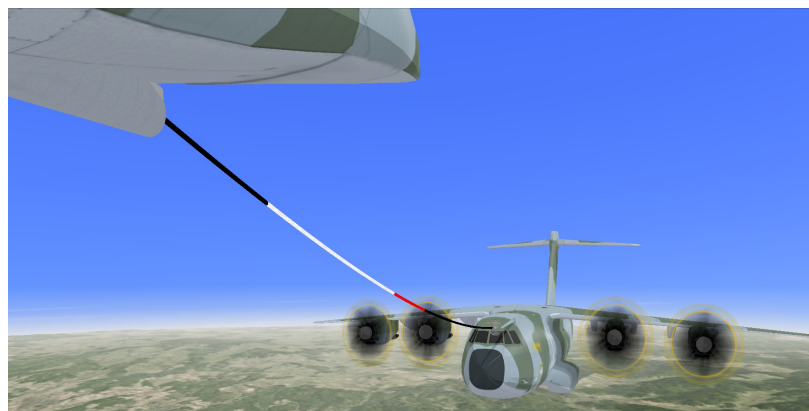


Figure 11. Structure of the Aerial Refueling Simulation Model



(a) View of the Tanker, HDU, Hose, and Drogue From Behind



(b) View of the Receiver, its Probe, the Hose and the Drogue

Figure 12. 3D Visualization for the Air-to-Air Refueling Simulation



For the air-to-air refueling scenario, models of the articulated hose, of the probe, of the drogue, and the hose-drum unit (including lights) were integrated in the visualization. The hose and drogue consists of 51 elements (50 hose segments + the drogue) and this regardless of the actual physical model of the hose and drogue assembly used behind the scenes. The position and orientation of each of these elements shall be determined based on the “real physical” states of the dynamical model that is simulated. This approach permits to use a very simplified physical model in some cases without having any odd visual impression provided to the pilots.

The current state of the visualization can be seen in figure 12. The signal lights at the HDU can be seen in figure 12a (two yellow dots just above the location where the hose goes through the tanker ramp door). In order to reproduce the visual impression correctly, these lights had to be made larger than they are in reality. The different lights (green, yellow, red) are located along the top side of the round HDU hose exit. The hose markings correspond to the different zones as defined in section II.E.

### III.D. Flight Control Laws and Autoflight

#### III.D.1. Overview of the Modules When Simulating the Nominal Aircraft and Regular Control Functions

The regular flight control laws (Normal / Alternate 1 / Alternate 2 / Direct) and the usual Airbus autopilot modes are implemented in the simulation environment of each aircraft. These are not the original Airbus control laws but they follow the same logic and should behave very similarly to the original ones. The various protections and limits used are currently a mix between the protections of the A400M and those of the A320 aircraft.

For each aircraft, the individual controller and autopilot modes are implemented within the Simulink model as well as the required dynamic models for the aircraft sensors. However, the complete logical conditions for the transition between the flight control modes are implemented in a separate program (one instance per aircraft) that sends the information on the currently active mode(s) to the model. The combination of both parts (in Simulink and in the separated executable) contains all Guidance Navigation and Control (GNC) functions and can be seen as one large GNC Module (GNCM).

The main motivation for integrating some parts of the GNCM directly with the aircraft dynamic model is that these parts are directly involved in the closed-loop dynamics: even when not simulating in an environment with strong real time guarantees (e.g. using the “AVES-on-the-Desktop” of section III.E), it remains crucial to ensure that the aircraft and these parts are being simulated in a consistent and synchronous manner. The remaining parts perform all treatments from the usual flight management and guidance computers and the flight control computers (except the pure automatic control tasks). Some of these operations (e.g. management of flight plans) are not easy to implement and to maintain using a block-diagram representation and consequently C++ code was preferred. These parts currently constitute a standalone program but could in principle also be integrated in the Simulink model as an s-function. The quite complex state machines that manage the state of the flight controller / autopilot modes runs within this program. These state machines are somewhat comparable to the ones shown in Ref. 37 but are more complex and very close to the complete original Airbus flight control/autoflight logic. This program also performs many different computations that are needed by the controllers or for the switching conditions between them (e.g. all the usual reference speeds  $V_{S1g}$ ,  $V_{\alpha\text{-prot}}$ ,  $V_{\alpha\text{-max}}$ ,  $V_{SPDFLOOR}$ , the various F-Speeds, etc.).

#### III.D.2. Definition, Integration, and Test of New Control Functions

In order to enable the development and test of new control laws/autopilot modes (one of the next steps in the current research on air-to-air refueling automation), a possibility to integrate the new control functions in the Simulink model and activation flags/conditions must be defined. These flags are commanded by the aforementioned external program based on the current states of the different state machines. Consequently, it must be possible to define some new states for the new control functions in these state machines.

At first, the designers of the new control function need to be able to activate this new function and possibly to let pilots evaluate it in the simulator. A graphical user interface (GUI) was developed specifically for this need and permits to set the state machines in any desired mode, to reset them, or to freeze them all at once or individually (i.e. no transition is fired anymore). This permits to activate modes which are unreachable otherwise (i.e. through the transitions defined in the state machines). This GUI is meant to be used by the simulator operator and not the pilots.

Thanks to this new capability, many new modes can be defined and tested (including with pilots) without having to tackle the complexity of defining all necessary state transitions first. These transitions can however be added progressively to finally obtain a new set of flight control/autoflight logic whose entire behavior (including transient responses and switching conditions) can be tested during representative scenarios by the pilots. This newly developed functionality will certainly be reused in many other works involving airplane simulation with Airbus-like flight control system architectures in the AVES simulator.

### III.E. AVES-on-the-Desktop

Simulators can be seen as more or less representative functional clones of the systems that they simulate. Many internal implementation details differ between a simulator and the original system, however the general behavior (e.g. as seen from the pilot's perspective) must be reproduced correctly. As a consequence, the tremendous complexity of modern aircraft systems and the advanced functions that they include directly impact the complexity of the corresponding simulators. The AVES Software Development Kit (AVESSDK) framework<sup>32</sup> as well as a series of work processes that were defined in addition to the tool itself have been of great help for tackling and managing the simulation of such complex systems.

The development of new functionalities for a complex and distributed simulation environment is usually relatively inefficient when working directly on the simulator. This is especially true, when the considered development consists in improving the model of the flight physics (e.g. adding effect that were neglected before) or for testing the dynamical response obtained under a some particular control inputs. These are typical situations for which flight dynamicists and control scientists typically use desktop simulations, do all kinds of tuning and validation, before checking that the real simulator (in the present case AVES) behave as in the tests on the desktop.

When dealing with the level of complexity attained with the simulation of modern airliners, the model and control designers also need to manage many large interfaces between many modules. Most new function or modification involve several new developments/modifications that must be made in a consistent manner across several of these modules. Having a possibility to test directly on the desktop the way the whole (distributed) simulation behave after such a modification is very useful and drastically increases the productivity of the model and control designers during these developments. Besides, this also enables several developments (e.g. for different projects) to be conducted in parallel (engineers working on their own desktop PCs) and without blocking the simulator. This "AVES-on-the-Desktop" functionality was originally developed for the A320 simulation programs and modifications to the AVESSDK framework were progressively made to enable always more advanced features in the desktop environment. Even though this capability has not been very much put into relief in the publications related to the AVES simulator, it became a standard tool for the airplane simulation developments in AVES and only the latest development stages and those that are very close to the cockpit hardware do require testing in the real simulator. Note that "AVES-on-the-Desktop" is not just the execution of the dynamical model in a Simulink environment, but a complete virtualization of the whole AVES simulation infrastructure on one or several desktop PC(s).

It is important to notice that in order for such a capability to be practical, duplicated developments must be prevented at all costs. The AVES-on-the-Desktop environment is based on the same source code than when working in the real AVES simulator. Only a restricted use of precompiler instructions and additional parameters permits to adapt the whole infrastructure to the used environment, i.e. AVES or one/several regular desktop PC(s). When running on the desktop, hard real time requirements and synchronization between all modules are not ensured as in the real simulator anymore (assuming that no specific synchronization mechanism is added), but the simulation behavior and results are very close to the ones that would be obtained with the real simulator. This lack of task synchronization in the desktop PC environment was also one argument (among many others) for integrating both tanker and receiver aircraft as well as their aerodynamic interaction into a single module (see figure 10). The slightest asynchronicity between two separate aircraft simulation modules would have constituted a showstopper and prevented any use of AVES-on-the-Desktop.

In the architecture shown in section III.A and in particular in figure 10 many modules/programs are duplicated (one instance for the receiver and one instance for the tanker). For most of these modules and when simulating only one aircraft, there was no need to be able to have various instances running simultaneously on the same computer. As a consequence, various modifications to these modules have been necessary to make this possible (conflicts in the access to network sockets or some shared-memory area). For most of them, additional configuration parameters were sufficient to solve these conflicts between the

multiple instances. It is also crucial to ensure that the initialization of the simulation is consistent across the multiple instances, e.g. if multiple copies of a physical parameter exist but should always be equal to each other in order for the flight physics to be accurately simulated.

## IV. Summary and Outlook

This paper presented an overview of the on-going modeling activities performed at DLR with the aim of supporting research on air-to-air refueling automation for both manned and unmanned receivers. The paper mainly presented the simulation infrastructure being integrated to the existing AVES full flight research simulator. The overall infrastructure as well as some selected components were presented. For the manned receiver application a configuration with a so-called “Future Military Transport Aircraft” FMTA (an Airbus A400M-like configuration) in both tanker and receiver roles is considered. For simplicity reasons, the entire flight physics is included within one single model in this application. Other possibilities have been considered and rejected for this application, but a co-simulation principle (two separated simulation program working in a synchronous manner) has been selected for the application to the refueling of an unmanned aircraft. This latter application is not detailed in the present paper, but the co-simulation mechanism that was developed will be published within the near future. The modeling of the interaction between both aircraft as well as the modeling of control surface nonlinearities is supported by RANS CFD computations and the a quite extensive flight control/autoflight architecture is also integrated to the simulation environment. Whilst this simulation infrastructure has been developed for air-to-air refueling scenario, it is also planed to reuse and extend it for formation flight investigations (aiming at saving fuel by surfing the wake of the leader aircraft).

## Acknowledgments

The authors would like to thank and acknowledge all colleagues working in the LUBETA project and in particular Sven Geisbauer and Patrick Löchert (both from the DLR Institute of Aerodynamics and Flow Technology), who tackle the RANS CFD aspects supporting the high-fidelity modeling of the involved aerodynamics effects, as well as Torsten Gerlach, Jan Hettwer, Umut Durak, Jürgen Gotschlich, Sven Oppermann (now at Hochschule Bremen), and Dominik Niedermeier for their contributions in porting of the FMTA model to the AVES simulator, in the implementation of the new features for AAR, as well as for all the daily maintenance of the AVES simulation center.

## References

- <sup>1</sup>Smith, R. K., *75 years of inflight refueling - Highlights 1923-1998*, Air Force History and Museums Program, 1998, ISBN 13: 9780160497797.
- <sup>2</sup>Jann, T., “Coupled simulation of cargo airdrop from a generic military transport aircraft.” *Proc. of the 21<sup>st</sup> AIAA Aerodynamic Decelerator System Technology Conference*, Dublin, Ireland, 23–26 May 2011, AIAA 2011-2566.
- <sup>3</sup>Geisbauer, S., Schade, N., Enk, S., Schmidt, H., and Arnold, J., “Experimental and Numerical Investigation of the Flow Topology During Airdrop Operations,” *Proc. of the 21<sup>st</sup> AIAA Aerodynamic Decelerator Systems Technology Conference and Seminar*, Dublin, Ireland, 23–26 May 2015, AIAA 2011-2565.
- <sup>4</sup>Geisbauer, S. and Schmidt, H., “Development and Validation of a RANS-based Airdrop Simulation Approach,” *Proc. of the 33<sup>rd</sup> AIAA Applied Aerodynamics Conference*, Dallas, TX, USA, 22–26 June 2015, AIAA 2015-3016.
- <sup>5</sup>Schmidt, H., “Implementierung eines MKS-Fahrwerksmodells in Matlab/Simulink mittels „Response Surfaces“,“ Tech. rep., DLR (German Aerospace Center) - Institute of Aeroelastics, 2012, Internal report - IB 232-2013 C 03.
- <sup>6</sup>Geisbauer, S., “Untersuchung des Einflusses generischer Vorderkantenbeschädigungen auf die aerodynamische Leistungsfähigkeit von Flugzeugprofilen,” Tech. rep., DLR (German Aerospace Center) - Institute of Aerodynamics and Flow Technology, 2012, Internal report - IB 124-2012/903.
- <sup>7</sup>(NSA), N. S. A., “NATO Standard ATP-3.3.4.2 Air-to-Air Refueling (ATP56), Edition C., Version 1.” Tech. rep., NATO, November 2013.
- <sup>8</sup>Kaden, A. and Luckner, R., “Modeling Wake Vortex Roll-Up and Vortex-Induced Forces and Moments for Tight Formation Flight,” *Proc. of the 2013 AIAA Modeling and Simulation Technologies Conference*, Boston, MA, USA, 19-22 August 2013, pp. 1–16, AIAA 2013-5076.
- <sup>9</sup>Gerhold, T., Galle, M., Friedrich, O., and Evans, J., “Calculation of Complex Three-Dimensional Configurations Employing the DLR-Tau Code,” *Proc. of the 35<sup>th</sup> AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, USA, 1997, AIAA 97-0167.
- <sup>10</sup>Gerhold, T., *Overview of the Hybrid RANS Code TAU*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 81–92.
- <sup>11</sup>Schwamborn, D., Gerhold, T., and Heinrich, R., “The DLR TAU-Code: Recent Applications in Research and Industry,”

- Proc. of the 2006 European Conference on Computational Fluid Dynamics (ECCOMAS CFD)*, edited by P. Wesseling, E. Oñate, and J. Périaux, TU Delft, Delft, The Netherlands, 2006.
- <sup>12</sup>Dogan, A. and Blake, W., "Modeling of Bow Wave Effect in Aerial Refueling," *Proc. of the 2010 AIAA Atmospheric Flight Mechanics Conference*, Toronto, ON, Canada, 2-5 August 2010, pp. 1-17, AIAA 2010-7926.
- <sup>13</sup>Dogan, A., Blake, W., and Haag, C., "Bow Wave Effect in Aerial Refueling: Computational Analysis and Modeling," *Journal of Aircraft*, Vol. 50, No. 6, November-December 2013, pp. 1856-1868.
- <sup>14</sup>Ro, K., Basaran, E., and Kamman, J. W., "Aerodynamic Characteristics of Paratrogue Assembly in an Aerial Refueling System," *Journal of Aircraft*, Vol. 44, No. 3, May-June 2007, pp. 963-970.
- <sup>15</sup>Haag, C., Schwaab, M., and Blake, W., "Computational Analysis of the Bow Wave Effect in Air-to-Air Refueling," *Proc. of the 2010 AIAA Atmospheric Flight Mechanics Conference*, Toronto, ON, Canada, 2-5 August 2010, pp. 1-11, AIAA 2010-7925.
- <sup>16</sup>Khan, O. and Masud, J., "Trajectory Analysis of Basket Engagement during Aerial Refueling," *Proc. of the AIAA SciTech 2014 - Atmospheric Flight Mechanics Conference*, National Harbor, MD, USA, 13-17 January 2014, pp. 1-15, AIAA 2014-0190.
- <sup>17</sup>Bhandari, U., Thomas, P. R., Bullock, S., Richardson, T. S., and du Bois, J. L., "Bow Wave Effect in Probe and Drogue Aerial Refuelling," *Proc. of the 2013 AIAA Guidance, Navigation and Control Conference*, Boston, MA, USA, 19-22 August 2013, pp. 1-21, AIAA 2013-4695.
- <sup>18</sup>Luckner, R., "Modeling and Simulation of Wake Vortex Encounters: State-of-the-Art and Challenges," *Proceedings of the AIAA Modeling and Simulation Technologies Conference*, Minneapolis, MN, USA, aug 2012, AIAA-2012-4633.
- <sup>19</sup>Fischenberg, D., "A Method to Validate Wake Vortex Encounters Models From Flight Test Data," *Proceedings of the 27th International Congress of the Aeronautical Sciences*, Nice, France, September 2010.
- <sup>20</sup>Schwarz, C. W., Hahn, K.-U., and Fischenberg, D., "Wake Encounter Severity Assessment Based on Validated Aerodynamic Interaction Models," *Proceedings of the AIAA Atmospheric and Space Environments Conference*, Toronto, ON, Canada, August 2010, AIAA-2010-7679.
- <sup>21</sup>Ro, K. and Kamman, J. W., "Modeling and Simulation of Hose-Paratrogue Aerial Refueling Systems," *Journal of Guidance, Control and Dynamics*, Vol. 33, No. 1, January-February 2010, pp. 53-63.
- <sup>22</sup>Ro, K., Ahmad, H., and Kamman, J. W., "Dynamic Modeling and Simulation of Hose-Paratrogue Assembly for Mid-Air Operations," *Proc. of the 2009 AIAA IntoechAerospace Conference*, Seattle, WA, USA, 6-9 April 2009, AIAA 2009-1849.
- <sup>23</sup>Styuart, A. V., Yamashiro, H., Stirling, R., Mor, M., and Gaston, R., "Numerical Simulation of Hose Whip Phenomenon in Aerial Refueling," *Proc. of the 2011 AIAA Atmospheric Flight Mechanics Conference*, Portland, OR, USA, 8-11 August 2011, pp. 1-10, AIAA 2011-6211.
- <sup>24</sup>Vassberg, J. C., Yeh, D. T., Blair, A. J., and Evert, J. M., "Numerical Simulations of KC-10 Centerline Aerial Refueling Hose-Droque Dynamics With A Reel Take-Up System," *Proc. of the 2004 AIAA Applied Aerodynamics Conference and Exhibit*, Providence, RI, USA, 16-19 August 2004, pp. 1-19, AIAA 2004-4719.
- <sup>25</sup>Leitner, R. M. and Estrugo, R., "Numeric Simulation of Aerial Refueling Coupling Dynamics in Case of Hose Reel Malfunction," *Proc. of the 2011 AIAA Modeling and Simulation Technologies Conference*, Boston, MA, USA, 19-22 August 2011, pp. 1-5, AIAA 2013-4840.
- <sup>26</sup>Hansen, J. L., Murray, J. E., and Campos, N. V., "The NASA Dryden AAR Project: A Flight Test Approach to an Aerial Refueling System," *Proc. of the 2004 AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Providence, RI, USA, 16-19 August 2004, AIAA 2004-4939.
- <sup>27</sup>Zhu, Z. and S.A., M., "Modeling and simulation of aerial refueling by finite element method," *International Journal of Solids and Structures*, Vol. 44, 2007, pp. 8057-8073, doi:10.1016/j.jisolsr.2007.05.026.
- <sup>28</sup>Ro, K., Kuk, T., and Kamman, J. W., "Dynamics and Control of Hose-Droque Refueling Systems During Coupling," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 6, November-December 2011, pp. 1694-1708, DOI: 10.2514/1.53205.
- <sup>29</sup>Dibley, R. P., Allen, M. J., and Nabaa, N., "Autonomous Airborne Refueling Demonstration Phase I Flight-Test Results," *Proc. of the 2007 AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Hilton Head, SC, USA, 20-23 August 2007, pp. 1-19, AIAA 2007-6639.
- <sup>30</sup>Dibley, R. P., Allen, M. J., and Nabaa, N., "Autonomous Airborne Refueling Demonstration Phase I Flight-Test Results," Tech. rep., Dryden Flight Research Center, Edwards, CA, USA, 2007, NASA/TM-2007-214632.
- <sup>31</sup>Duda, H., Gerlach, T., Advani, S., and Potter, M., "Design of the DLR AVES Research Flight Simulator," *Proc. of the 2013 AIAA Modeling and Simulation Technologies Conference*, Boston, MA, USA, 19-22 August 2013, pp. 1-14, AIAA 2013-4737, DOI: 10.2514/6.2013-4737.
- <sup>32</sup>Gerlach, T. and Durak, U., "AVES SDK: Bridging the Gap between Simulator and Flight Systems Designer," *Proc. of the AIAA AVIATION 2015 Modeling and Simulation Technologies Conference*, Dallas, TX, USA, 22-26 June 2015, pp. 1-10, AIAA 2015-2947.
- <sup>33</sup>Gerlach, T., Durak, U., Knüppel, A., and Rambau, T., "Running High Level Architecture in Real-Time for Flight Simulator Integration," *Proc. of the AIAA AVIATION 2016 Modeling and Simulation Technologies Conference*, Washington, D.C., USA, 13-17 June 2016, pp. 1-10, AIAA 2016-4130.
- <sup>34</sup>Burns, R. S. and Clark, C. S., "The Automated Aerial Refueling Simulation at the AVTAS Laboratory," *Proc. of the 2005 AIAA Modeling and Simulation Technologies Conference and Exhibit*, San Francisco, CA, USA, 15-18 August 2005, AIAA 2005-6008.
- <sup>35</sup>Williams, R. D., Feitshans, G. L., and Rowe, A. J., "A Prototype UAV Control Station Interface for Automated Aerial Refueling," *Proc. of the 2005 AIAA Modeling and Simulation Technologies Conference and Exhibit*, San Francisco, CA, USA, 15-18 August 2005, AIAA 2005-6009.
- <sup>36</sup>Gerlach, T., "Visualisation of the brownout phenomenon, integration and test on a helicopter flight simulator," *RAeS The Aeronautical Journal*, Vol. 114, No. 1163, January 2011, pp. 57-63, DOI: <https://doi.org/10.1017/S0001924000005364>.
- <sup>37</sup>Gunetti, P., Cassaro, M., Battipede, M., and Gili, P., "Modeling Autopilot Suites For A Multi-Aircraft Simulator," *Proc. of the 2013 AIAA Modeling and Simulation Technologies Conference*, Boston, MA, USA, 19-22 August 2013, pp. 1-15, AIAA 2013-4736.