UNIVERSITA' DI PISA

Facoltà di Ingegneria

Corso di Laurea Specialistica in Ingegneria Aerospaziale

Tesi di Laurea

Hydro-mechanical Modelling of the Airbus A380 Nose Landing Gear Extension/Retraction Systems

Relatori:

Prof. Ing. Eugenio Denti

Ing. Terence Frost

Ing. Franco Calvanese

Candidato:

Gabriele Bernardini

ANNO ACCADEMICO 2008 – 2009

To my family

Abstract

Modelling and Simulation is a branch of engineering in continuous development across the industry, and follows the evolution of computer technology and simulation tools.

By means of simulation, it is possible to explore and test several design solutions in a virtual environment, and to obtain performance predictions before the physical devices are actually produced. This capability is used throughout the aerospace industry, where the development of a project requires huge investments, and the need of accurate predictions is part of the design process from its early stage, to minimise risks and wastes.

The work described in this dissertation was developed within the Simulation and Modelling Group of the Airbus UK Landing Gear Department. It is focused on the description of the approach, the techniques and the tools used to perform a hydro-mechanical simulation of the Airbus A380 Nose Landing Gear Extension/Retraction Systems.

The creation of the hydraulic model using the AMESim modelling tool is described, as well as the development of a mechanical model of the Nose Landing Gear with the ADAMS modelling tool. The mechanical model already existed, but major rework was necessary in order to couple it with the hydraulic model by means of co-simulation.

The setup of the co-simulation platform is explained, and the results of the validation process for the integrated models are presented, showing the process followed to tune the hydro-mechanical model, to match its dynamic behaviour with reference data.

Finally, the method adopted to extract the Pressure-Flow characteristics of the hydraulic model is described.

Acknowledgements

I would like to thank Prof. Eugenio Denti for giving me the opportunity to apply for the internship, and therefore to “catch the train” that passes by only once in a lifetime. Moreover, I thank him for his support during the internship, and for the help that he gave me when I had to come back to Italy to take my last exams.

I would like to thank Airbus for giving me the chance to work in such an exciting environment, and to experience something that is very difficult to find in other companies. Being passionate for aviation since when I was a kid, I can say that working for Airbus was a dream come true, and I’m really glad to have had the opportunity to stay in the company for the next years.

Special thanks go to my manager, Terry Frost, for his trust, his support and his esteem during the whole internship, and also for his involvement and for being a guide in the decisions concerning my future.

I want to thank all the other members of the Simulation and Modelling Group, including those who have now left: Luke Bagnall, Luke Spencer, Franco Calvanese, Simon Hancock, Ryan Davies, Andrew Bird, Leonardo Vivarelli, Steven O’Brien, Anna Blumel, James Morris, Cécile Garaygay, Bjoern Kirchhoff, Abdi Yusuf, Benjamin Leppier, Steven Brown, Joe Power, Antonio Colosimo, Mark Healey. They all made me feel part of the team since the beginning, and their support made things a lot easier for me during my internship.

Special thanks go to Franco Calvanese and Leonardo Viv/\relli: from a professional point of view for their precious help, support and trust that they gave me; from a personal point of view for the friendship that they demonstrated to me since the beginning.

I am infinitely grateful to Bjoern Kirchhoff. From the first day of my internship I asked him thousands of questions about AMESim and hydraulics, and everytime he was always there to give me an answer. Not only he was able to teach me a lot from a purely technical point of view, but most importantly he was also able to show me the best way to approach engineering problems and to interact with other people during teamwork.

I am also very grateful to my “bros” Antonio Colosimo (aka Tony, Antani and Ant) and Cécile Garaygay (aka Cicci). I shared with them my entire experience in England, and now my life would not be the same without them for so many reasons.

Many thanks to Serena Simoni, I really appreciated her valuable help when I was getting ready for the Assessment Centre.

Many thanks also to all the Italian community in Bristol for having welcomed me as soon as I arrived.

I want to thank all the people that I had the pleasure to meet, coming from all over Europe and beyond, because getting to know them has enriched me from a personal and human point of view. I hope to see again as soon as possible all of them who left England to go back to their own countries. Special thanks to those who came to Italy for my graduation, Sascha Rudel and Jennafer Duerden, as well as Cécile, who I have already mentioned. I am really happy to have you there with me for an important moment of my life.

I am thankful to my family, for giving me all their support for my experience abroad, even if they miss me a lot when I’m not in Italy.

Finally, I thank all my old mates, because even if now we don’t see each other often, they make me feel like I’ve never left when I go back to Italy.

List of Contents

List of Figures

Acronyms

ADAMS	Automatic Dynamic Analysis of Mechanical Systems
AMESim	Advanced Modelling Environment for Simulation
ATA	Air Transport Association
BSCS	Braking and Steering Control System
GDO	Ground Door Opening
LGERS	Landing Gear Extension/Retraction System
LGMS	Landing Gear Monitoring System
MBE	Model Based Engineering
NLG	Nose Landing Gear

Introduction

The present thesis describes part of the activities performed during a 1-year internship in Airbus UK, at the Filton site in Bristol. The placement was within the Simulation and Modelling Group, which is part of the Landing Gear Department.

Modelling and Simulation is a branch of engineering whose importance has greatly increased over the last few years. The advantages of a model driven design are well documented, both in standard textbooks as well as engineering journals.
Through modelling, it is possible to analyse different design solutions, to predict performances, to discover anomalies and resolve them in a virtual environment, before the real objects are realised.

The purpose of this thesis is to describe the process and the methods used to develop, verify and validate a hydro-mechanical model of the Airbus A380 Nose Landing Gear Extension/Retraction Systems.

The first chapter describes the Simulation and Modelling Group and in particular the Physical Systems Team, in which the activities described in this work took place. Then the Model Based Engineering process (MBE) is summarised, which is the official process followed in the Simulation and Modelling Group in order to build, verify and validate all types of models. An overview of the Airbus A380 aircraft is provided, focussing on the Nose Landing Gear and on the Extension/Retraction System.
The second chapter describes the process followed to build and verify the hydro-mechanical model using the LMS AMESim and MSC.ADAMS software.
The third chapter describes the validation process, to show how the model has been tuned in order to match the real behaviour of the system, using data coming from the A380 Landing Gear Test Rig facility.
The fourth chapter highlights the modifications to the hydraulic model needed to extract the Pressure-Flow characteristics of the system, and the process followed for that scope.
In the final chapter conclusion remarks and future developments are outlined.

Chapter 1 – Research Context

1.1 – Simulation and Modelling Group Organisation

The Simulation and Modelling Group within the Landing Gear Department works side by side with several design teams in order to support them with the design of models to define and verify technical requirements, to validate and verify the Landing Gear Systems design, using a wide range of simulation techniques and analysis methods.

The group is divided in the following four teams:

- ATA32[1] Aircraft Modelling;
- Avionic Systems Modelling;
- Physical Systems Modelling;
- Simulation and Software Integration.

1.1.1 – Physical Systems Modelling Team

The Physical Systems Modelling Team is responsible for the hydraulic, mechanical, electrical and thermal modelling of the Landing Gear Systems. The team provides models for two main areas: Landing Gear Extension/Retraction Modelling and Landing Gear Braking and Steering Modelling.

The models are currently used for the following purposes:

- Performances evaluation;
- New design solutions trade off studies and sensitivity studies;
- Failure cases analysis;
- Support validation of requirements;
- ATA32 Desktop Simulation;
- ATA32 Virtual Validation and Verification Rig;
- Support validation activities at aircraft level;

---

[1] ATA32 is one of the sections in which all the systems of an aircraft have been conventionally organised by the Air Transport Association (ATA Chapters): ATA32 is composed by all the Landing Gear Systems.

- Support ATA32 System rig;
- Provide ATA32 models for Aircraft Integration Flight Simulators.

This work is developed within the Physical Systems team and deals with Landing Gear Extension/Retraction Modelling.

1.2 – Model Based Engineering (MBE)

1.2.1 – Overview

The Model Based Engineering is the process followed in the Simulation and Modelling Group to model requirements, in order to produce a design using a traceable consistent approach. This process provides requirements and methods to support model development and model delivery, to enable integration of models into different platforms.

1.2.2 – Main Steps

The generic top-level processes and the subsequent sub-processes can be described in six main steps as follows.

- **Capturing Model Requirements**
  This phase of the process is concerned with the generation of a list of functions on which to base the Model Specification, validation and review of the model requirements.

- **Producing Model Specification**
  The creation and the issue of a Model Specification show how model functions are then implemented in the model. Additional requirements will need to be captured and agreed to allow formal planning of the task to be performed. The deliverables out of this stage are the Model Specification and the Model Verification Plan.

- **Building Model**
  This phase of the process consists in the actual development of the models. There are two defined routes through this phase, one for the initial development or major upgrade of a

simulation model and the second for rectifying anomalies in the simulation model or minor upgrades to the functionality of the model.

- **Delivering Model**

  The validated model is gathered together with supporting documentation in a single entity, which is subsequently provided to the model customer.

  The model and associated documentation, when they reach a certain level of faithfulness, are recorded within the model Versioning Control System: the model can therefore be shared between users to be reused and integrated onto different platforms.

- **Integrating Model**

  The simulation model is integrated into the required test environment and tests are performed to prove a successful integration. These activities do not cover the tests or developments that will be undertaken on the environment itself to fulfill project needs. The deliverable from this phase is the released Model Test Platform. This document summarises the results of integration and validation tests performed by model integrators.

- **Performing Tests**

  Tests are performed to identify non-compliances against the expected system function or performance.

The previous steps are represented in Figure 1. 1.

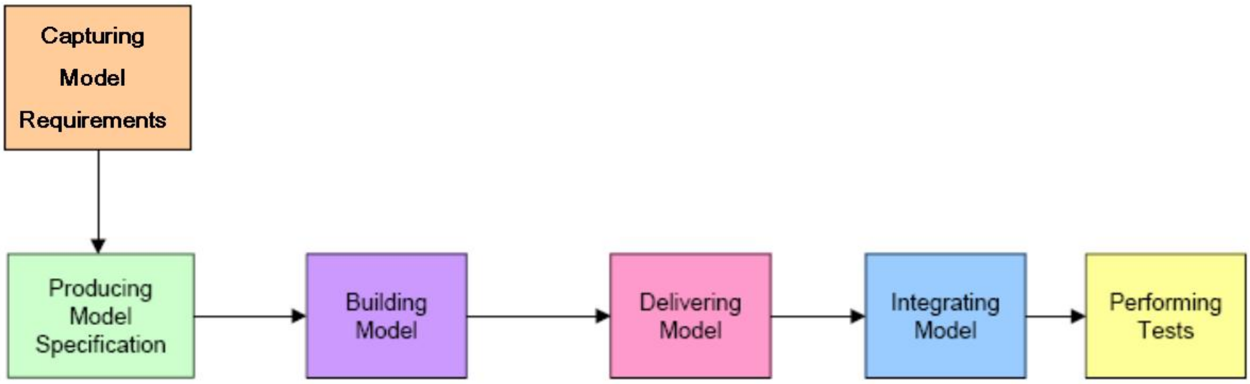


Figure 1. 1 – Model Based Engineering Process

1.3 – Airbus A380

1.3.1 – Aircraft overview

The Airbus A380 is a double-deck, wide-body, four-engine airliner manufactured by the European corporation Airbus. The largest passenger airliner in the world, the A380 made its maiden flight on 27th April 2005 from Toulouse, France, and its first commercial flight on 25th October 2007 from Singapore to Sydney with Singapore Airlines. The aircraft was known as the Airbus A3XX before the official launch of the programme, but the nickname Superjumbo has since become associated with it. The name A380 was chosen because the "8" reminds the aircraft's fuselage cross section, and it is considered a lucky number in the oriental culture.

The A380's upper deck extends along almost the entire length of the fuselage, and its width is equivalent to that of a widebody aircraft. This allows for a cabin with 50% more floor space than the next-largest airliner, the Boeing 747-400, and provides seating for 525 people in standard three-class configuration or up to 853 people in all economy class configurations. The A380-800 is the largest passenger airliner in the world, but has a shorter fuselage than the Airbus A340-600, which is Airbus' next-biggest passenger airplane. The A380-800 has a design range of 15,200 km (8,200 nm), sufficient to fly from Boston to Hong Kong for example, and a cruising speed of Mach 0.85. It is the first commercial jet capable of using GTL-based fuel (fuel obtained from natural gas or other gaseous hydrocarbons using the *Gas to Liquids* refinery process).

Figure 1. 2 – Airbus A380 (© AIRBUS S.A.S 2006 – Photo by e^{x}m Company/H. Goussé)

1.3.2 – A380 Landing Gear overview

The A380 is supported by five landing gears, which allow an optimal load distribution on the runway: one Nose Landing Gear, two Wing Landing Gears and two Body Landing Gears. The configuration is shown in Figure 1. 3.

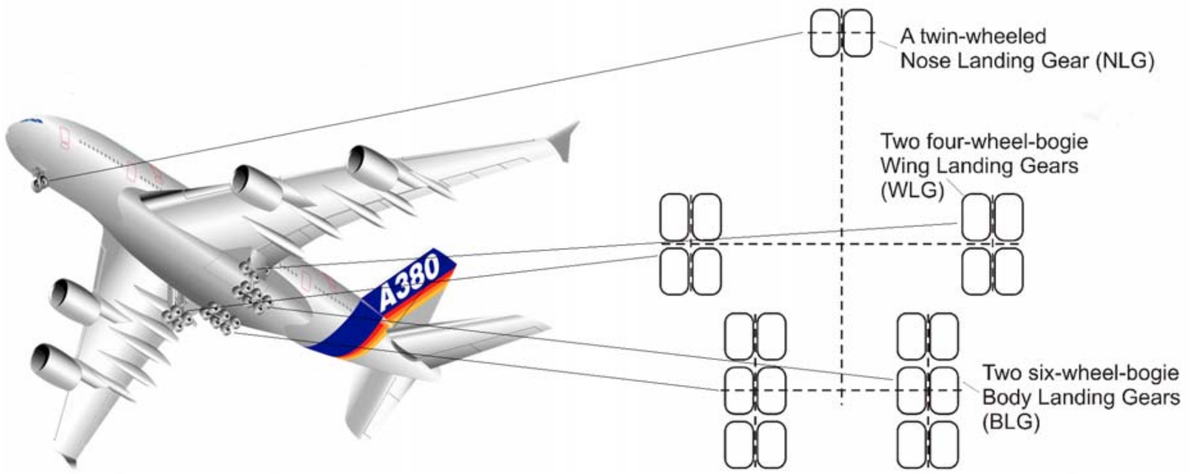


Figure 1. 3 – A380 Undercarriage

The A380 nose landing gear is supplied by Messier-Dowty. Fitted with two wheels, it has a height of 4.8 meters when it is fully extended, and supports more than 150 tons of the aircraft's weight. The nose landing gear assembly is shown in Figure 1. 4.

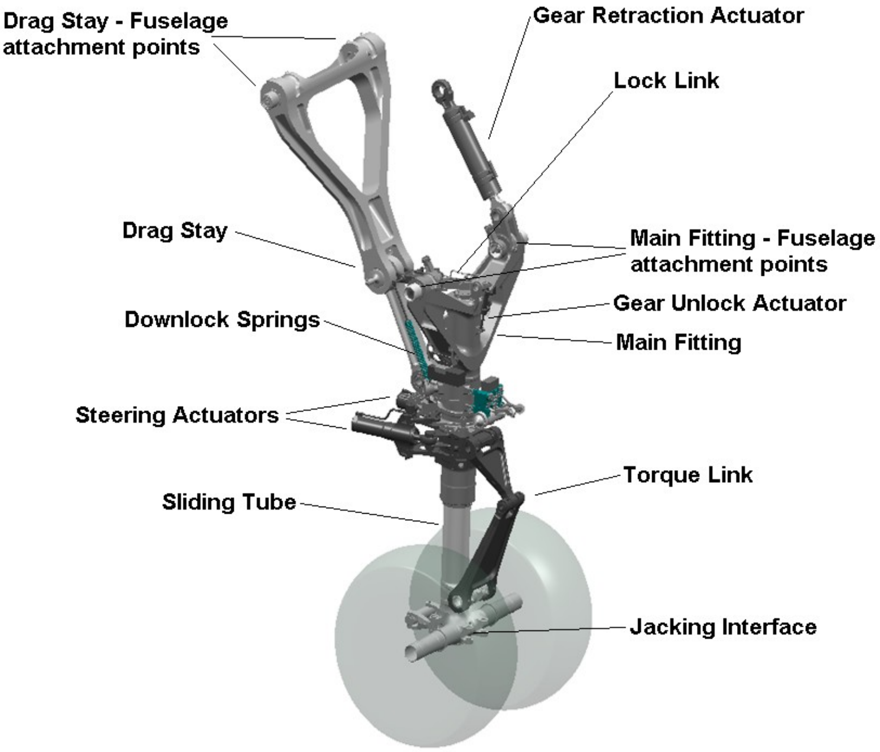


Figure 1. 4 – A380 Nose Landing Gear

The gear is connected to the airfame via a pair of attachment points on the main fitting, which define a hinge line to allow the gear rotation. A drag stay system constrains the rotational degree of freedom of gear main fitting when it is fully extended. It consists of an upper arm, connected to the fuselage, and a lower arm hinged to the gear main fitting; both elements are then linked together via an additional joint. The upper and the lower lock links connect said joint with the main fitting, creating a chain which locks the kinematics after extension. The two downlock springs keep this configuration in position.

The steering system of the gear allows the aircraft to turn by means of two steering actuators in push-pull configuration.

To extend and retract the nose landing gear, a hydraulic actuator is attached to the main fitting and the fuselage. By extending the actuator the landing gear is retracted. Before retraction, the downlock system has to be unlocked by the hydraulic unlock actuator.

In order to absorb the energy induced by vertical impacts, the nose landing gear is equipped with a single stage oleo-pneumatic shock absorber. The inner piston of the shock absorber consists of a sliding tube, and the main fitting acts as the outer cylinder.

The nose landing gear also provides an interface to tow the aircraft, an interface for jacking.

1.3.3 – Landing Gear Extension/Retraction System

LGERS Overview

The Landing Gear Extension/Retraction System (LGERS) is a system in charge of providing hydraulic power for the extension and retraction control and operation of A380 Nose, Wing and Body Landing Gear.

LGERS is a subsystem of ATA32, which includes all the Landing Gear Systems of the aircraft:

- Braking and Steering Control System (BSCS);
- Landing Gear Extension/Retraction System (LGERS);
- Landing Gear Monitoring System (LGMS).

ATA32 is interfaced with ATA29, which is the core of the aircraft’s hydraulic system and is responsible of power generation and distribution. The main sources of hydraulic power are engine driven pumps, therefore the system has a mechanical interface with engines, by means of gearboxes that connect the main pumps to the engine shafts. For safety reasons, ATA29 is split in two subsystems, Green System and Yellow System, to guarantee the redundancy of the power supply infrastructure and complete independence to avoid common failure causes.

LGERS Functions

The LGERS’ basic functions are:

- Door actuation;
- Gear actuation;
- Landing Gear Bogie positioning;
- Emergency lowering of the landing gear;
- Access to the landing gear bays on ground operation of the doors for maintenance;
- Interface to the operating crew for system command and indication;
- Interface to the maintenance crew for test and data retrieval;
- Provide configuration status (e.g. Weight on Wheels) to other systems;
- System monitoring and Built-in Test.

LGERS Operational Modes

The LGERS has the following modes of operation:

- Normal Operation;
- Emergency Operation (in case normal mode is unavailable);
- Ground Door Operation (maintenance access to landing gear bays).

Respectively, these operating modes are executed by:

- The Normal System;
- An independent Freefall System for Landing Gear Gravity Extensions;
- An auxiliary system for Ground Door Operation.

Under normal conditions, the LGERS operates in normal mode, i.e. the Normal System executes powered Landing Gear extension and retraction.

The Normal System is electrically controlled, with hydraulically actuated landing gear and doors and electrically actuated uplocks. It is powered hydraulically by the Green and the Yellow ATA29 hydraulic systems.

If a powered landing gear extension by the Normal System is not possible, the flight crew can initiate a gravity assisted landing gear extension through an Emergency System, called the Freefall System. The Freefall System provides a time-sequenced operation of valves, followed by a release of all door and gear uplocks allowing the gear to extend under gravity.

For maintenance purposes the LGERS system can be used to retract and extend the landing gear with the aircraft on jacks. Moreover, the Ground Door Opening (GDO) System allows opening of the gear doors, using dedicated valves, when the aircraft is on the ground. To close the doors after opening, the GDO System is reset, returning control of the doors to the Normal System. The landing gear doors can be locked in the open position with a suitable gag to prevent inadvertent door closure with the full door actuation force applied.

Chapter 2 – Model Development

2.1 – Sources of information

Before starting the development of the model, it was necessary to identify the sources for system layout and characteristics.

An overview of the system is provided by the A380 LGERS System Description Document, which contains a detailed description of the system architecture (including the hydraulic diagram), of all the equipment and its operation.

All the numeric data was derived from a previous hydro-mechanical model of the A380 NLG LGERS, and all the characteristics of the hydraulic components in said model were validated by the LGERS design team, to guarantee the consistency with the real A380 LGERS equipment. The reference model was created using ADAMS/Hydraulics, which is a toolbox of MSC.ADAMS that allows building a hydraulic power system and coupling it to an existing mechanical model within the same tool. However, ADAMS/Hydraulics was replaced, as AMESim was adopted for hydraulics modelling and simulation; therefore an update of the A380 LGERS model was required using the new software.

2.2 – Hydraulic model overview

The creation of the A380 Nose Landing Gear LGERS hydraulic model in AMESim is based on the general architecture of the system, represented in Figure 2. 1.

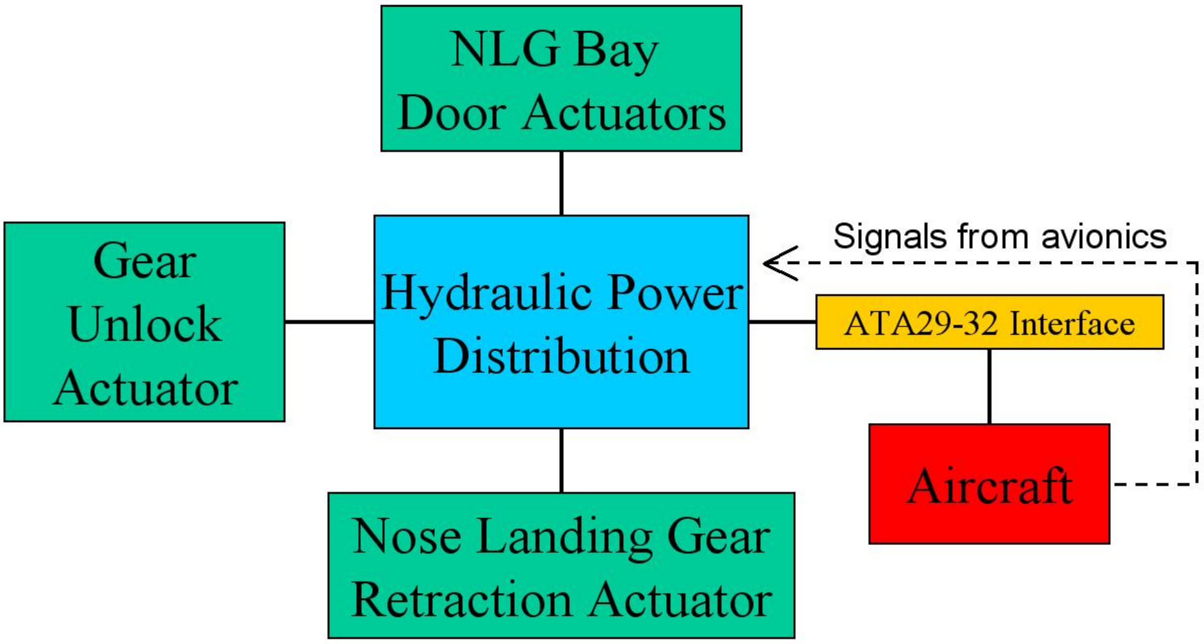


Figure 2. 1 – General Architecture of A380 NLG LGERS

2.1.1 – ATA29-32 Interface

At the ATA29-32 interface, the system is connected to power generation and distribution through a supply and return line. The supply line was modelled using a constant pressure source component, which acts as a perfect pressure compensated pump. With this assumption, the pressure supply to the hydraulic system is always kept constant throughout the simulations, without any dependence on the flow rate supplied to the system. Similarly, the return line was represented by a constant pressure hydraulic reservoir, which can accept any return flow rate while maintaining a constant return pressure.

2.1.2 – Hydraulic Power Distribution Subsystem

The hydraulic power distribution group consists in a subsystem, which is connected to hydraulic supply and return lines and receives control signals from the aircraft avionics, in order to route the hydraulic power to the door and gear actuators.

The subsystem is composed by a number of valves, which open and close connections between the actuators and the supply/return lines, according to the operational mode that has been selected. The door and gear selector valves are the components that mainly fulfil that function, but additional valves are present, in order to modify the configuration of the system, and change the system operational mode (Normal Operation, Freefall, Ground Door Operation).

The selector valves were modelled using standard AMESim components, but user created components were adopted for the other valves.

A set of input signals was created to command the valves according to the selected simulation scenario, as shown in Figure 2. 2.

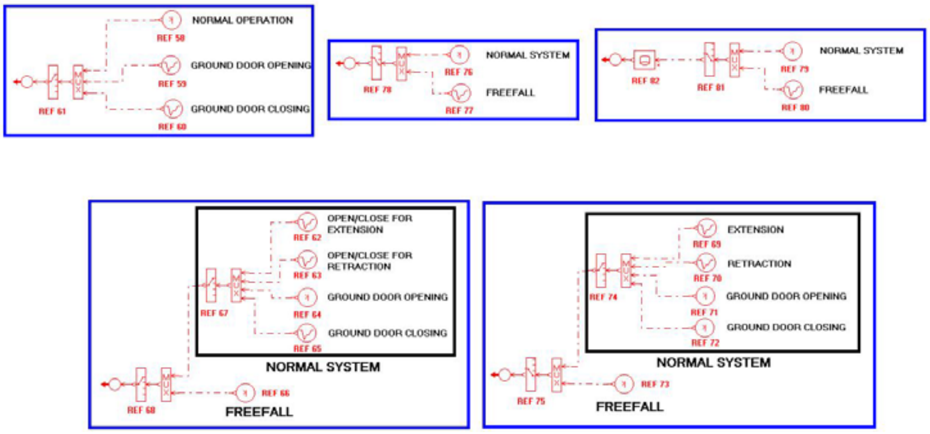


Figure 2. 2 – Set of inputs to valves

As can be seen in the picture, each command signal sent to the valves is selected among different possible signals that converge into a switch. Each switch is characterised by a parameter, which identifies the number of the input port that has to be passed to the output.

It was necessary to parameterise the hydraulic model, to be able to select the desired simulation scenario by changing a limited number of parameters. In order to pilot the simulation, the three following parameters were identified, and for each simulation case an appropriate combination of their values was chosen.

	SIMULATION SCENARIOS				
PARAMETER NAME	Normal Extension	Normal Retraction	Freefall	Ground Door Opening	Ground Door Closing
PHASE	1	2	1	3	4
SYSTEM	1	1	2	1	1
GROUND_DOOR_OPER	1	1	1	2	3

Table 2. 1

The switches for valve inputs were created accordingly, so that the correct valve signals for each simulation scenario could be automatically selected by setting the parameters previously shown.

For this scope, an AMESim feature called *global parameter setup* was used. The global parameters are parameters that can be defined in an appropriate window; each one has a name and an associated value, and said value can be set once for all in the Global Parameters window. Then, each parameter can be used in the definition of any component's characteristic property, and can be even included in mathematical expressions. For the selection of the correct input signal for each valve, the global parameters were assigned to the switches, as highlighted in Figure 2. 3.

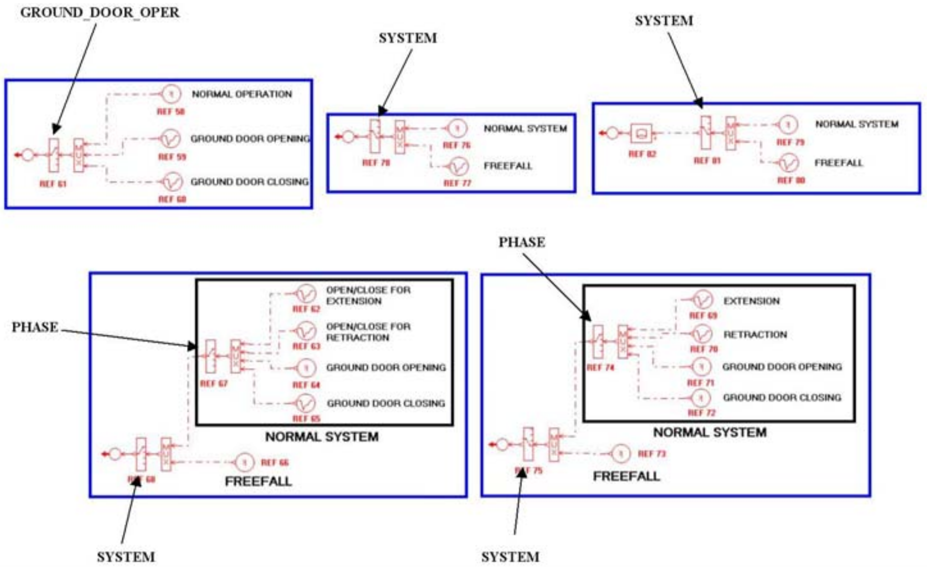


Figure 2. 3 – Parameters associated to switches

Regarding the input signals to the switches, each of them was defined as a constant value or as a specific time sequence; the first choice was made for valves which are kept in a fixed position during the simulation case, the second for the valves whose spool has to move throughout different positions.

For the definition of the signals, additional global parameters were used to describe and coordinate the time sequences among all the valves.

2.1.3 – Door and gear actuators

The Nose Landing Gear and bay doors are driven by single-rod actuators, connected to the hydraulic power distribution subsystem. More in details, a pair of actuators powers the bay doors, while two other actuators are fitted on the landing gear structure. In fact, as explained in par. 1.3.2, a retraction actuator is responsible for powering the gear main fitting for extension/retraction, while an unlock actuator is mounted on the gear locklinks. When the unlock actuator is pressurised during retraction, the mechanism is unlocked by the application of the hydraulic force on an appropriate leverage.

Door actuators and gear retraction actuator were modelled by means of user-created supercomponents, to take into account the snubbing effect; such an actuator is not present in the AMESim standard library.

A snubbing actuator is a hydraulic actuator containing devices that act only in the final and/or initial travel of the piston, in proximity of its endstops. A system of orifices which open or close in relation to the stroke's displacement is fitted at the inlet ports of the actuator, and affects the evolution of the pressure in the actuator chambers. The result is that when the piston gets close to an endstop, an additional hydraulic resistance is created, and the piston itself is decelerated, to avoid a violent impact on the endstop.

In AMESim, a supercomponent consists of a subsystem created by the user and hidden behind a customised icon. Said icon presents a number of ports, according to the external connections required by the subsystem modelled underneath. Door and gear actuators created for the A380 NLG LGERS hydraulic model are shown in Figure 2. 4.

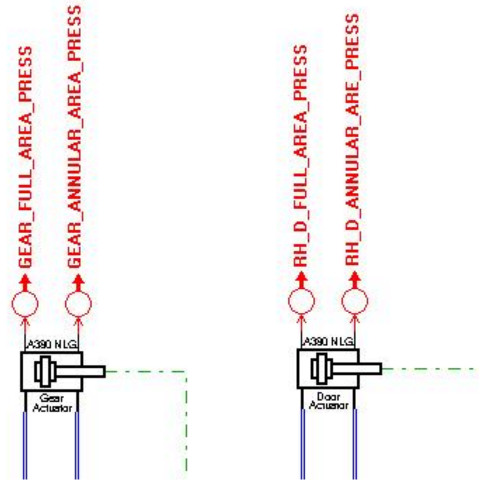


Figure 2. 4 – Actuator supercomponents

The subsystem hidden behind each icon is connected to the rest of the model via two hydraulic ports (blue connections) and a mechanical port (green connection); the latter allows linking an external mechanical load to the actuator rod. Moreover two additional ports (red connections) are present, to supply pressure measurements of the actuator chambers with sensors appropriately placed in the actuator subsystem.

Regarding the unlock actuator, a standard single rod actuator was used since no snubbing effect has to be taken into account.

If supply and return lines were directly connected to the actuator chambers, the force exerted by the actuator rods would be extremely high; doors would then open and close at very high speeds and with big accelerations, as a consequence of very high flow rates through the actuator ports. The same problem would affect the gear, and the high kinetic energy gained by gear and doors during operation would damage the structure when reaching the final phase of the travel. Nevertheless, hydraulic problems would arise in the ATA29 system, because of the very high flow rate required by the LGERS system. To avoid the issue, flow control devices are fitted on the connections between the actuators and the hydraulic power distribution subsystem. Each device consists in a system of hydraulic restrictors, which limits the flow rate supplied to the actuators, by means of creating a pressure drop between the power distribution subsystem and the actuator chambers, according to Bernoulli’s equation. Flow control devices were created in the AMESim model, and placed appropriately in front of the hydraulic ports of each actuator.

2.1.4 – Pipelines

All the hydraulic components introduced in the model were linked with the appropriate pipelines, and a suitable submodel had to be chosen. In AMESim, in fact, a component submodel consists in the mathematical representation associated to the component itself. Even though the icon on the sketch remains the same, by changing the associated submodel it is possible to explore different levels of approximation and therefore obtain a different fidelity to reality. Although it is worth noting that the level of approximation required is in direct relation to the analysis that would be performed. Obviously, increasing the accuracy of a submodel leads to a higher number of parameters associated to the component, and the overall complexity of the model grows, with consequences on CPU time and a potential increase in human error.

The pipe submodel that was adopted is a *compressibility + friction hydraulic pipe/hose* submodel. It consists in a lumped parameter mathematical representation of a pipe, which takes into account capacitance (due to fluid compressibility and pipe wall deformation) as well as friction. The calculation of the density (needed to determine the volumetric flow rate) is based on the pressure of the fluid, and the pressure value used for the scope depends on the connectivity of the model. In fact, in some cases a pressure state is modelled to represent the pressure at one end of the pipe; in other cases a pressure state is needed at both ends, and density is calculated based on the pressure in the middle of the pipe.

Friction is a function of pipe diameter, length and a friction coefficient. The latter is calculated in different ways depending on the flow regime: in case of laminar flow only the Reynolds number is considered, while for turbulent flow also the roughness of the pipe is taken into account.

The present submodel was chosen because it models with sufficient accuracy the impact of pipes on the system dynamics (mainly pressure drops due to friction, and capacitance). A lumped parameter submodel was considered satisfactory for the present application, because the hypothesis of negligible variation of pressure with position along the pipe is consistent. A distributed parameter submodel, in fact, is beneficial only for extremely long pipes, or if wave dynamics are significant, and such conditions are not experienced in the present model. Moreover, fluid accelerations are considered too modest, to justify the use of a submodel which takes into account fluid inertia.

2.1.5 – Hydraulic fluid model

The choice of the mathematical representation of the hydraulic fluid has a very significant impact on simulation results, especially in those situations where very low values of pressure are experienced. Therefore, the choice of an advanced hydraulic fluid submodel is important, as it gives the user the possibility of setting a wide range of parameters, for a detailed definition of fluid properties.

The fluid submodel that was chosen allows the user to supply fluid properties using an ASCII file. Said file contains tables defining the bulk modulus (or the density) and the kinematic viscosity of the liquid at different temperatures and pressures. The fluid properties stored in these tables apply to the fluid in liquid form, in the absence of air, gas or vapour. In order to respect mass conservation, the density and bulk modulus supplied by this submodel are always consistent. The air/gas release and cavitation phenomena are also included: when the *void content* internal variable is greater than zero, the density, bulk modulus and viscosity values are lowered to take into account the presence of the bubbles.

The air/gas can be dissolved in the liquid or can be free, therefore forming bubbles. The user specifies a pressure above which all the air is dissolved, the *saturation pressure*. The value of the saturation pressure has a major influence on the fluid properties at low pressure and must be set carefully. Moreover, the user can set the *polytropic index for the air/gas/vapour content* and the *absolute viscosity of air/gas* parameters, which have less influence on the fluid properties and can usually be left at their default values.

A cavitation model is also included. When the pressure drops below the *high saturated vapour pressure* the liquid begins to vaporize; when the pressure drops below the *low saturated vapour pressure* the cavitation stops and the fluid has become a mixture of air/gas and vapour (there is no liquid anymore).

2.3 – Verification of the model

Once the model was built and populated with the appropriate numerical values, it was necessary to perform a preliminary verification against top-level requirements, to ensure that the model is capable of working under all the necessary simulation scenarios (Normal Extension, Normal Retraction, Freefall, Ground Door Opening, Ground Door Closing).

For this purpose, it was necessary to create within AMESim a model of the A380 bay doors and landing gear, in order to simulate the loads acting on the hydraulic actuators and door/gear dynamics with an acceptable level of approximation.

The schematisation that was adopted allows the use of the same model to represent either a door or the landing gear, and the only load applied is door/gear own weight (i.e. hangar operation only is considered). Said schematisation is presented in Figure 2. 5.

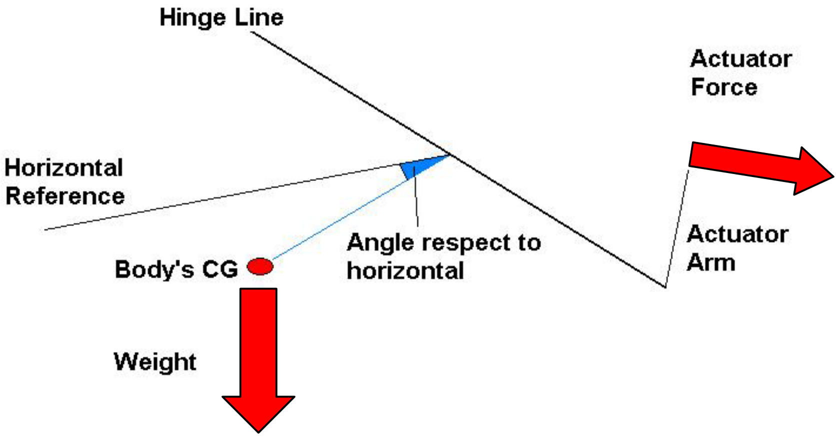


Figure 2. 5 – Door/gear schematisation

The mass of the body is concentrated in its own centre of gravity, and its initial position is taken into account, in terms of initial angle between CG and horizontal reference, around the hinge line. During the rotation of the CG, the force representing door/gear weight is always kept vertical.

The force applied by the actuator is translated into a torque around the hinge line, by considering the appropriate actuator arm, whose value changes during the rotation.

Figure 2. 6 shows how the schematisation of the problem was implemented in an AMESim mechanical model.

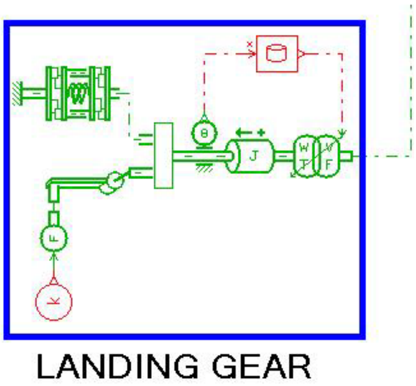


Figure 2. 6 – AMESim model of landing gear or door

The rotation of the CG around the hinge line is modelled using a mechanical arm element. On the left hand side of said element, the weight of the body is applied by supplying a constant force as an input (red block). The force is then translated by the arm into torque acting on the hinge line.

During its rotation, the arm is designed to keep the same direction of the applied force, which is therefore always parallel to itself. This allows a correct modelling of the body's weight, whose direction always remains vertical.

A rotary endstop is used to constrain the rotation of the mechanical arm between two limit positions.

A rotary inertia element is placed on the hinge line, next to a component that translates the torque around the hinge line back into a force. The conversion is made by supplying an input signal representing the actuator arm, which is a function of the angular displacement. Next to the inertia component, an angular sensor measures the angular displacement around the hinge line and sends the value to a lookup table, which gives as output the corresponding value of the actuator arm. Finally, the output load is applied to the actuator.

All the mechanical components, as well as the actuators, are parameterised using the global parameters shown in Table 2.1. In this way, it is possible to define all the initial values of angular displacement and actuator stroke as a function of said parameters, and the correct initial position of the components can be selected by setting the appropriate combination of parameters, according to Table 2.1.

In order to hold doors and gear in their initial closed/retracted position, a model of door/gear uplock must be provided. Therefore the model shown in Figure 2. 7 was created.

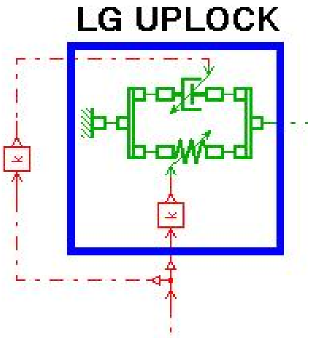


Figure 2. 7 – AMESim model of landing gear uplock or door uplock

The uplock is modelled as a spring damper with variable stiffness and damping coefficients. The force generated by the spring damper is summed to the force exerted by the actuator and applied to the door/gear model shown in Figure 2. 6. The parameters of the spring and the damper are appropriately set (once again following the parameterisation showed in Table 2.1), to make sure that when doors and gear are in their initial position, spring compression/extension is zero. In this way, when the uplock is active, the weight of gear/door can be balanced by the reaction of the spring-damper, and the system can be held in place without any load acting on the hydraulic actuators. Since stiffness and damping can be provided as an external input, it is sufficient to send a high value of stiffness and an appropriate damping in order to activate the uplock. To release the uplock, it is sufficient to set both values to zero.

Using the mechanical models previously described, it was possible to verify the model against top-level requirements. In particular, door and gear angles were monitored, as well as pressure in actuator chambers.

After the successful completion of the test, the mechanical parts were removed from the model, and modifications were made to setup the model for a co-simulation with the ADAMS A380 Nose Landing Gear Bay mechanical model, as described in the following paragraph.

2.4 – AMESim/ADAMS Co-Simulation

2.4.1 – Overview

The possibility of coupling the hydraulic model with an ADAMS mechanical model allows simulating those scenarios that can be recreated only if the geometry of the mechanical system is accurately modelled.

Co-simulation is necessary to simulate a freefall, because in that case a contact between gear and doors is experienced, and the doors are pushed open by the gear falling under its own weight. Moreover, the co-simulation allows the study of failure cases such as tyre missing, shock absorber failure and uplock failure, in which the dynamic behaviour of the system is affected by the contact between components, as well as clash analyses between bodies and the study of clearances between parts.

2.4.2 – The Co-Simulation technique

Generally speaking, the expression *Co-Simulation* is used to identify a process in which two models are interfaced together and synchronised to run at the same time. Actually, a Co-Simulation is only one of several techniques that can be used for the scope. The following methods can be used to interface ADAMS and AMESim:

- **ADAMS to AMESim Full Export**

  During an ADAMS to AMESim Full Export, the AMESim solver integrates all the sets of equations, and ADAMS is used as a function evaluator. This means that ADAMS calculates derivatives and other outputs, to be passed to AMESim solver at fixed time steps, defined by the *communication interval*. AMESim, moreover, acts as a user interface to the simulation.

  In this case, numerical problems may arise, especially when importing a large ADAMS system. In fact, since AMESim integrator is designed to handle equations typical of hydraulic problems, it may find some difficulties while solving a large number of mechanical equations. The consequences may be a high CPU time, or even the impossibility of completing the integration.

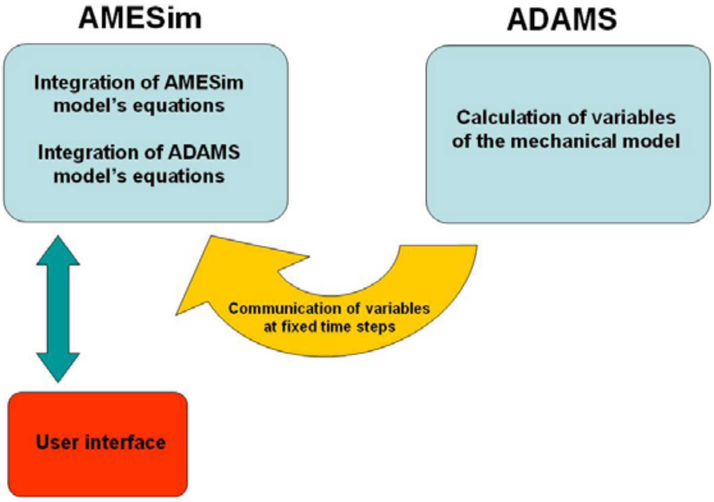


Figure 2. 8 – ADAMS to AMESim Full Export

- **AMESim to ADAMS Full Export**

  The method is similar to the previous one, but the difference is that ADAMS solver integrates all the equations, and AMESim works as a function evaluator; moreover ADAMS acts as a user interface.

  Even in this case numerical problems may arise. In fact, ADAMS integrator is tuned to work well with the equations governing multi-body systems, but the numerical characteristics of a fluid power system are different. Moreover, AMESim integrator has some particular features designed to deal with the typical discontinuities of stiff systems; this means that if ADAMS solver is used to integrate AMESim equations, numerical problems may arise while AMESim integrator, in the same situation, might instead produce a good solution. Therefore, when using ADAMS solver, it is recommended to avoid hard discontinuities (jump changes in the value of state variables).

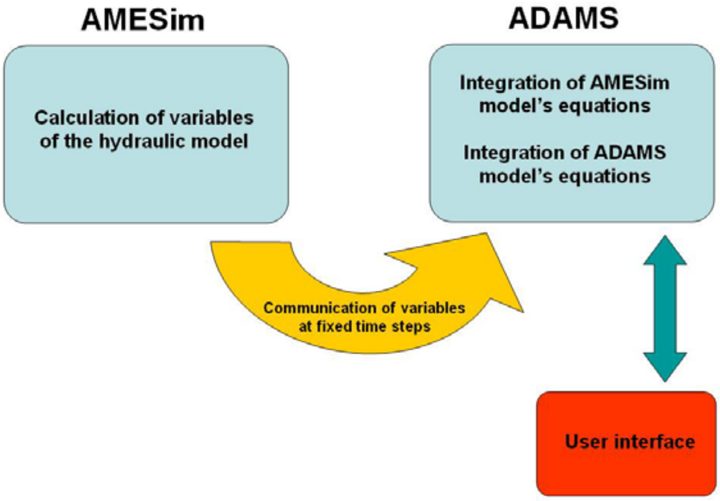


Figure 2. 9 – AMESim to ADAMS Full Export

- **Co-Simulation from AMESim**

  During a Co-Simulation, each programme uses its own solver for the set of equations related to its own model; both solvers exchange information at a fixed interval, called *communication interval*. When ADAMS receives a signal, this will stay constant during the whole communication interval, until a new value is received, and the same thing happens in AMESim. Each solver can run in variable step size mode, therefore optimising CPU time.

  The main problem consists in finding a correct setting for the communication interval. This is very critical, because each solver is basically running independently from the other one, and the exchange of information only occurs according to the communication interval. A large interval leads to a quick run, but also to less accurate results, which in some cases may be totally unrealistic. Even a divergence of some state variables can be experienced, and the result is the crash of the simulation.

  During a Co-Simulation from AMESim, the latter works as a user interface to the model.

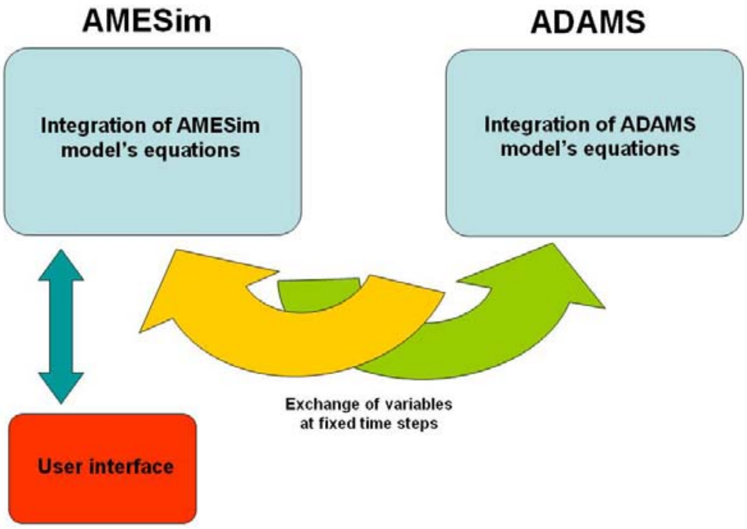


Figure 2. 10 – Co-Simulation from AMESim

- **Co-Simulation from ADAMS**

  The method is similar to the previous one, but ADAMS acts as a user interface for the simulation.

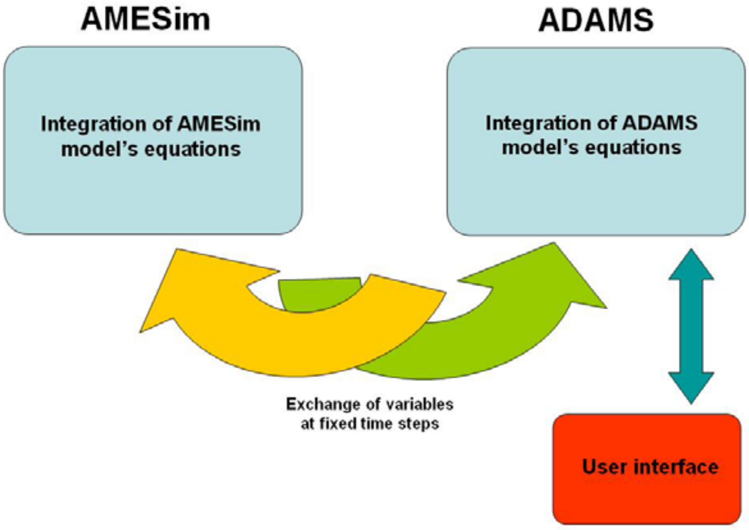


Figure 2. 11 – Co-Simulation from ADAMS

The method that was chosen is the co-simulation from ADAMS. First of all, both the full export techniques were tested, but the integration of all the equations resulted impossible for both solvers. Typically, the integration process could be started, but was proceeding very slowly from the beginning, to reach the point where the CPU time became so big that the simulation appeared not to be proceeding at all to the user.

Instead, both the co-simulations techniques were applied successfully and allowed reasonable CPU times. The co-simulation from ADAMS was preferred, as it allows controlling the simulation by using ADAMS as a user interface. An advantage consists in the possibility of having the animation of the model ready to be post processed at the end of the simulation, as well as the results in ADAMS/Postprocessor tool. This leads to a quick analysis of the simulation results, and a rapid comparison with reference data, which can be loaded in ADAMS/Postprocessor as well. In case of co-simulation from AMESim, results are still available for postprocessing in ADAMS, but each time a simulation is completed, the results file created by the program has to be manually reloaded. Moreover, with minor modifications to the model, it is possible to automatically transfer the plots related to the hydraulic systems to the ADAMS model, to concentrate all the results to be analysed (mechanical and hydraulic) in the ADAMS/Postprocessor tool. The same thing can be done in the opposite way (transfer of mechanical quantities to AMESim) but the AMESim's postprocessing capabilities do not offer the same flexibility and rapidity that can be found using ADAMS/Postprocessor.

Moreover, a valuable advantage obtained by adopting the co-simulation from ADAMS consists in the possibility of using the ADAMS/Insight toolbox to automate simulations. As explained in Chapter 3, it was very important to create a method to launch simulation runs automatically, changing the value of some key parameters before each run. The use of ADAMS/Insight allowed an easy implementation of the process, necessary for the validation of the models.

2.4.3 – Programme settings

In order to perform a co-simulation, the appropriate software must be installed and a configuration process must be followed.

Once an AMESim hydraulic model is created, the programme translates it into a C++ source code, which is needed by the computer to perform the simulation. Through an external software called *Microsoft Visual Studio*, the source code is compiled into dll executable. This will allow the ADAMS model to be able to communicate with the hydraulic model.

After the installation of Visual Studio, it is necessary to setup the Windows environmental variables using the VCVARS32.BAT batch file, which is located in the *Microsoft Visual Studio\VC\bin* folder. This will allow AMESim to link to the appropriate compiler.

In the *interfaces\adams* folder (located in the AMESim main folder), the *amesim_adams_gsec.make* file must be opened using Wordpad and edited. The last line by default looks like the following:

dformd.lib asutility_imp.lib

The first part must be deleted, so that the line looks as follows:

asutility_imp.lib

In Windows it is then necessary to set an environment variable that points to the ADAMS installation directory. Therefore, a new system variable named *AME_ADAMS_HOME* must be created in *Control Panel → System*, and the ADAMS main folder typed in the Variable *Value* field. Then another variable named *$ADAMS_CONTROLS_WTIME* must be created, and its value set to 20.

Finally, within AMESim it is necessary to add in the *category path list* the library *$AME/libadams*, and in the *AMESim preferences* Microsoft Visual C++ must be set as AMESim compiler.

2.4.4 – Preparation of the mechanical model

The mechanical model that was adopted for the co-simulation represents the complete A380 Nose Landing Gear Bay. Figure 2. 12 and Figure 2. 13 show the model respectively in the gear retracted and gear extended positions.

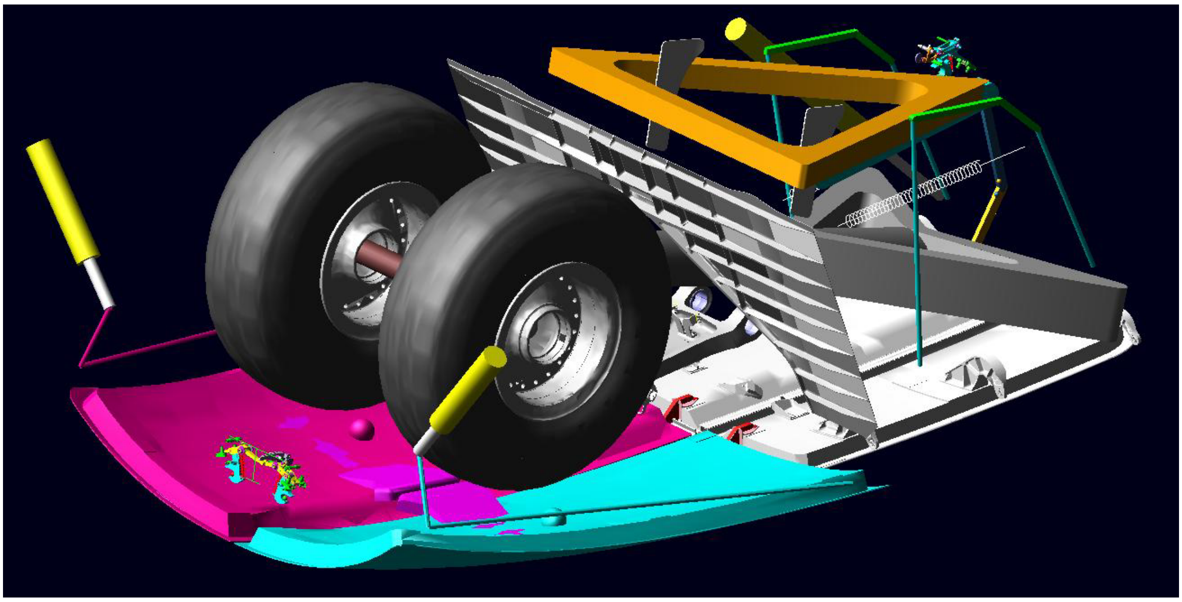


Figure 2. 12 – Nose Landing Gear Bay in gear retracted configuration

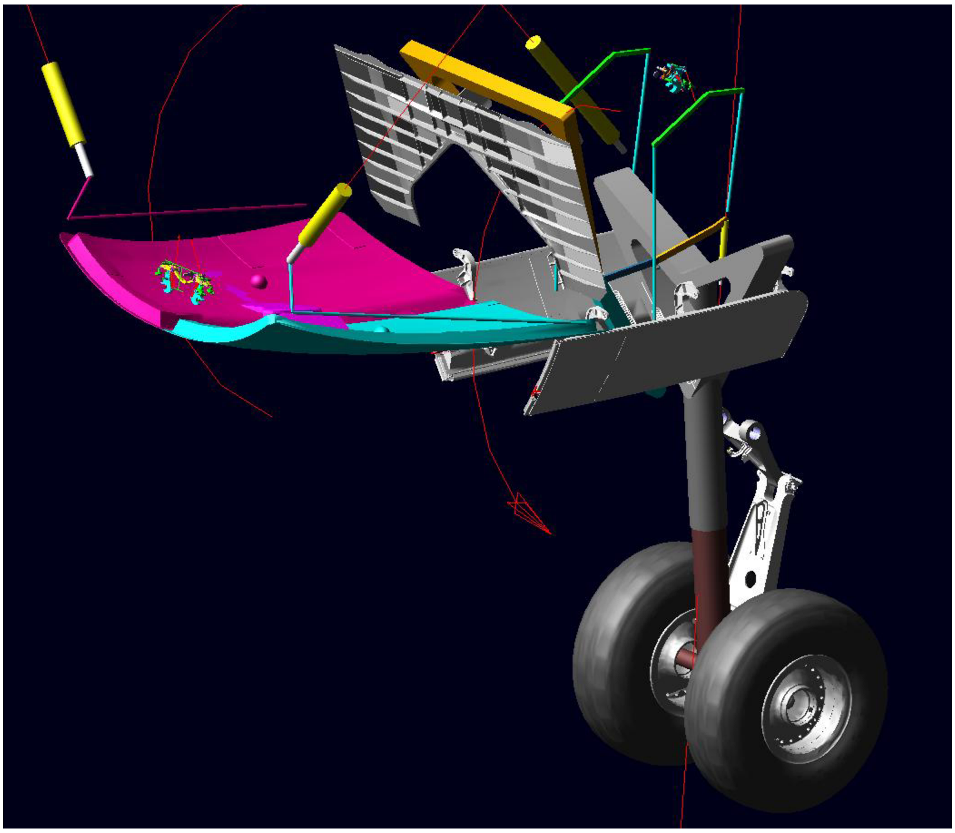


Figure 2. 13 – Nose Landing Gear Bay in gear extended configuration

The shape of the landing gear components is simplified, but all the parts that affect the kinematics of the system are modelled, and all the mass properties are correctly assigned.

In addition, the model contains the gear-driven doors (pair of small doors that move via a mechanical connection to the gear main fitting), the hydraulically actuated bay doors, a support plate, as well as actuators and gear/door uplocks (hooks and all the related mechanisms necessary for their operation).

The complete bay model was already available, built in ADAMS and supplied with an integrated hydraulic model developed in ADAMS/Hydraulics. Unfortunately this represents a problem when trying to disconnect and eliminate the hydraulic part, to extract only the mechanical part in order to use it in standalone. In fact, the mechanical and hydraulic components are modelled in the same environment, and they are closely linked together. Therefore it is very difficult to remove all the hydraulic components and the related settings accordingly made in the model, to be able to run a purely mechanical simulation without experiencing any error messages.

A purely mechanical model of only the landing gear was available, as shown in Figure 2. 14 in retracted and extended position.

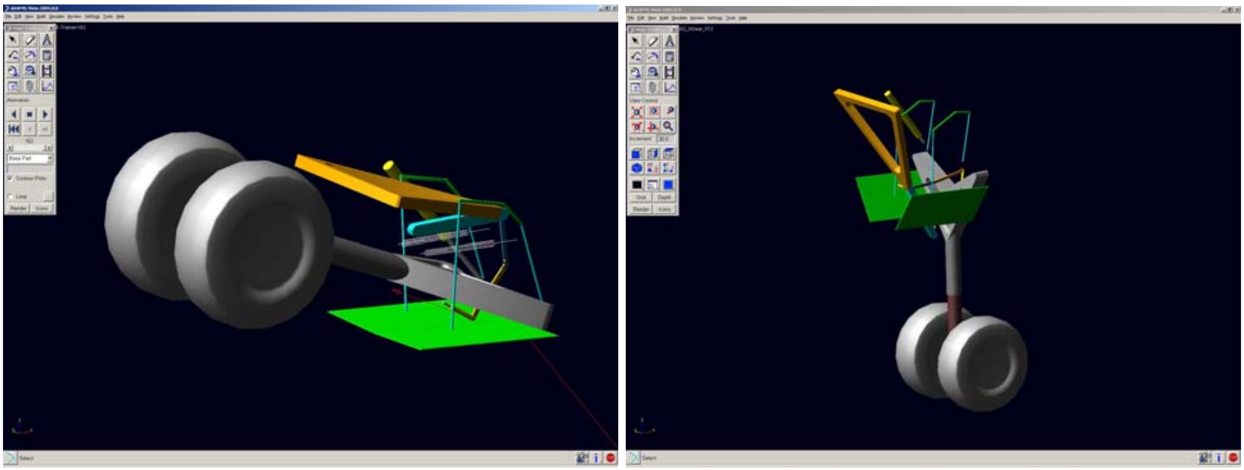


Figure 2. 14 – Original mechanical model of the A380 Nose Landing Gear

As can be seen in the picture, the gear model still provides the main fitting/sliding tube assembly, the drag stay, the locklinks, and both the retraction actuator and the unlock actuator.

Comparing to the Landing Gear Bay model, the gear-driven doors are represented in a much simpler way, and their dimensions are significantly different from reality. Being shorter than the real doors, they need to be replaced to avoid gaps between them and the hydraulically powered doors in gear retracted configuration.

Moreover, it can be noted that wheels and tyres are merged in a single geometric feature. Actually, a good accuracy of tyre shape in the model is required, as it plays an important role during the simulation of a freefall scenario. In fact, in that case contact between tyres and doors occurs, and the gear pushes the doors open while tyres slide on a pair of ramps appropriately fitted on the doors. An inaccurate modelling of the tyres (especially in terms of dimensions and profile shape) could therefore affect simulation results.

The gear-driven doors and the wheels were deleted and replaced with more advanced components, exporting them from the ADAMS/Hydraulic reference model in Parasolid format, and importing them in the new model.

Afterwards, the gear torque link and a support frame were added. Even though they are not necessary to drive the movement of the landing gear, they were included because their presence is important to perform clash analyses in case of failure of some components.

Bay doors were imported in the model, and a representation of door actuators was adopted, similar to the one related to the gear retraction actuator and unlock actuator.

Finally, door and gear uplocks were added, together with the mechanisms that allow their operation.

All the mass properties were assigned to the new components, as well as constraints and contacts.

Once built, the mechanical model was simulated in standalone mode. This means that, starting from the door closed/gear retracted position, the uplocks were released in sequence, to check the functionality of the mechanisms and to make sure that the constraints are correctly assigned. Afterwards, it was necessary to implement in the model some features that allow the communication with AMESim and therefore an exchange of variables between the mechanical model and the hydraulic system.

2.4.5 – Setup of the mechanical model for the co-simulation

In general, the first step for the setup of a model for co-simulation consists in the identification of inputs and outputs. In the present case, AMESim computes actuator forces generated by the hydraulic system, and passes them to ADAMS, which applies them on the mechanical structure of gear and doors on the actuator attachment points.

The definition of valve command signals in AMESim has been described in par. 2.1.2, but additional command signals are necessary for the co-simulation. In fact, door and gear uplocks have to be engaged and disengaged by means of forces and torques acting on the uplock mechanisms, appropriately activated or deactivated in parallel with valve operation. Even though it is possible to define signals for the mechanical model in ADAMS, the decision was made to command the uplocks from AMESim and send the generated signals to the mechanical model through the co-simulation interface. The benefits consist in the possibility of concentrating all the command signals for the integrated models on the AMESim schematic, without splitting them in the two models and complicating their management. Moreover, the definition of the inputs signals is much more flexible in AMESim, as it is possible to use signal blocks and global parameters as shown in par. 2.1.2. Therefore, signals for uplocks were created as done for valve input signals, as shown in Figure 2. 15 (where the assignment of the global parameters to the switches is highlighted).

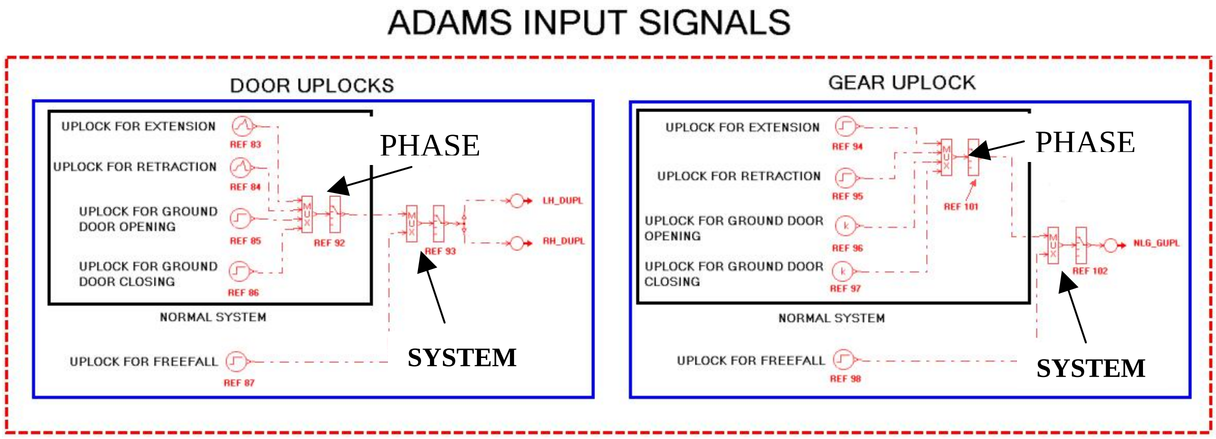


Figure 2. 15 – Input signals to ADAMS model

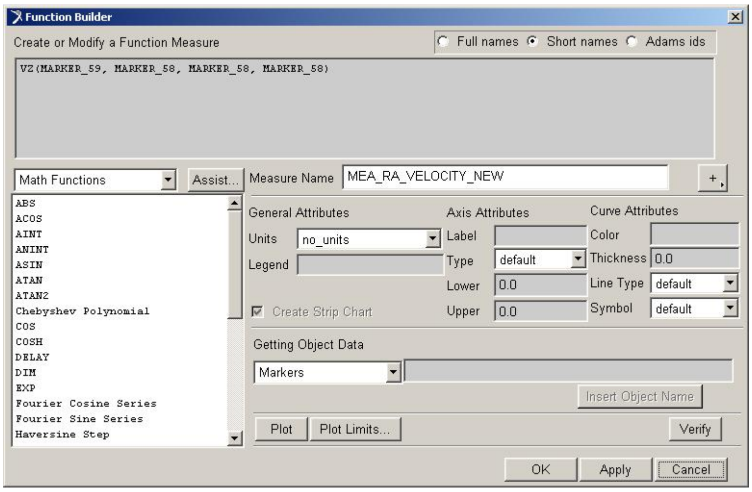
Uplock signals are generated in AMESim as Boolean values (1 for uplock engaged and 0 for disengaged), and they are sent to ADAMS, where they must be converted into appropriate forces and torques that act on the uplock mechanism to engage or disengage it.

Regarding AMESim inputs, ADAMS measures actuator rod velocities and communicates them to AMESim, which needs them to be able to compute actuator positions.

The modifications needed by the mechanical model depend on the co-simulation technique adopted. As previously explained, the co-simulation can be setup in order to be controlled either from ADAMS or AMESim; the appropriate model configurations for both methods are explained in the following paragraphs.

Co-Simulation from ADAMS

The first step consists in the definition of the inputs to the AMESim model, which the ADAMS model has to provide. For this purpose, a velocity measure has been created in the model for each of the four actuators (the two door actuators, the gear retraction actuator and the gear unlock actuator). Said measure calculates the relative speed of the two actuator attachment points and considers its projection along the actuator axial direction. It is very important to make sure that the velocity measure is positive when the actuator extends, i.e. when the distance between the two attachment points increases. The definition of the measure for the gear retraction actuator is shown in Figure 2. 16; the measures for the remaining actuators can be defined in the same way.



Create or Modify a Function Measure

○ Full names ● Short names ○ Adams ids

```
VZ(MARKER_59, MARKER_58, MARKER_58, MARKER_58)
```

Measure Name: MEA_RA_VELOCITY_NEW

Units: no_units

Type: default

Lower: 0.0

Upper: 0.0

Thickness: 0.0

Line Type: default

Symbol: default

☑ Create Strip Chart

Getting Object Data: Markers

Figure 2. 16 – Definition of measure for actuator velocity

Once velocity measures are set, they have to be declared as outputs from ADAMS, and therefore inputs to AMESim. To do so, a feature named U input array must be created, as shown in Figure 2. 17.

Name: ARRAY_U

Solver ID: 1

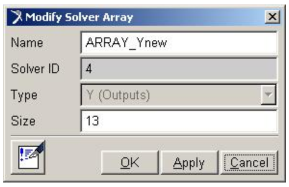
Type: U (Inputs)

Variables: MEA_UA_VELOCITY_NEW,MEA_RA_VELOCITY_NEW,MEA_RH_DOOR_ACT_VELOCITY_NEW,MEA_LH_DOOR_ACT_VELOCITY_NEW

Figure 2. 17 – Definition of the U array

The U array is a list of all the measures that have to be sent to AMESim; the order in which the measures are listed is very important, because it must match the order of AMESim inputs defined in the hydraulic model. Therefore, during the AMESim setup the sequence of the measures in the U array must be taken into account, as explained in par. 2.4.6.

After the setup of the ADAMS outputs, all the inputs must be created. Again, the exchange of variables between the two programs occurs through the definition of an array, called Y array. The definition of the Y array is shown in Figure 2. 18.

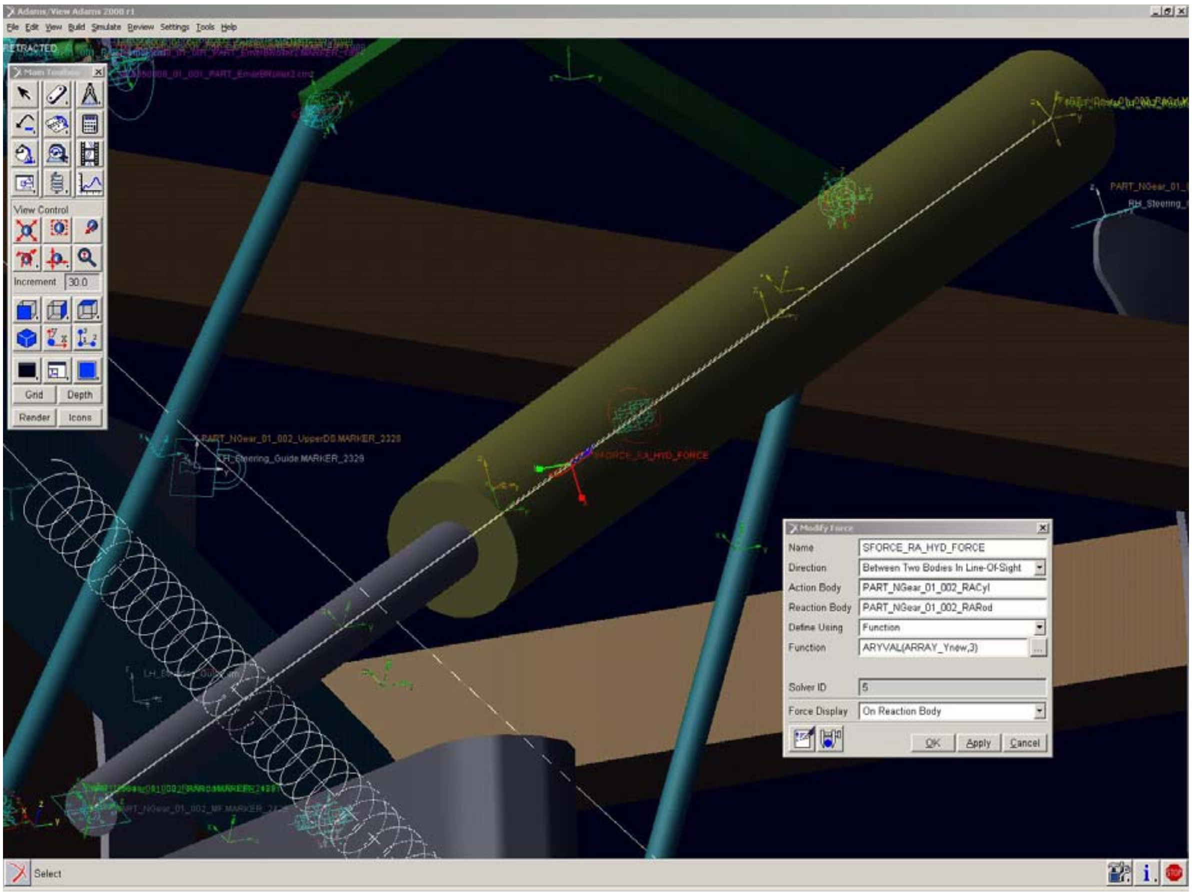


Name: ARRAY_Ynew
Solver ID: 4
Type: Y (Outputs)
Size: 13

Figure 2. 18 – Definition of the Y array

In this case, it is sufficient to specify the number of variables that ADAMS has to receive from AMESim during the simulation.

As previously explained, ADAMS inputs consist in forces to be applied to actuators and uplock mechanisms. Figure 2. 19 shows the definition of the hydraulic actuator force for the gear retraction actuator, applied on the two attachment points of actuator rod and cylinder; the forces for the remaining actuators can be defined in the same way.



Name: SFORCE_RA_HYD_FORCE
Direction: Between Two Bodies In Line-Of-Sight
Action Body: PART_NGear_01_002_RACyl
Reaction Body: PART_NGear_01_002_RARod
Define Using: Function
Function: ARYVAL(ARRAY_Ynew,3)
Solver ID: 5
Force Display: On Reaction Body

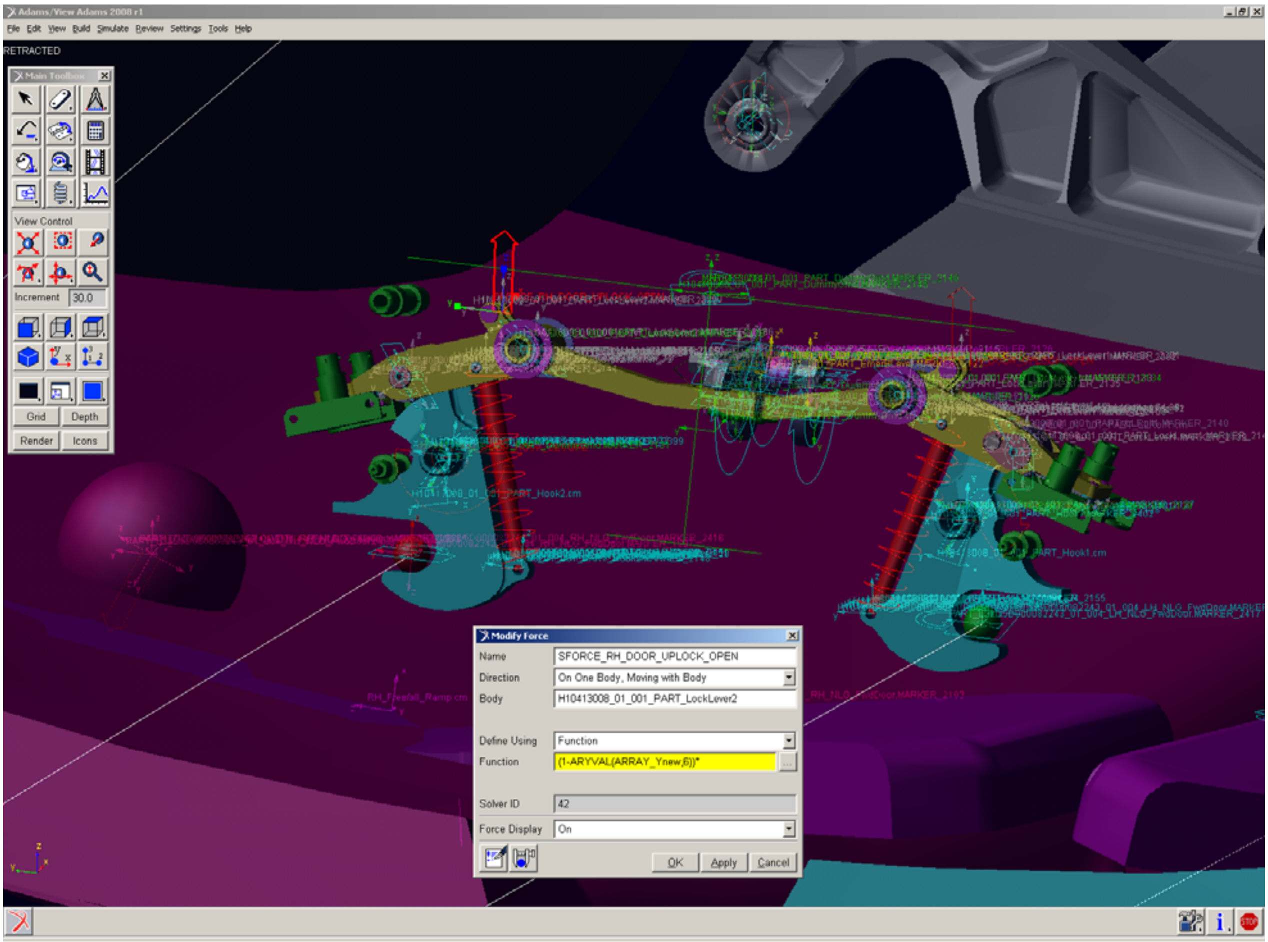
Figure 2. 19 – Definition of the hydraulic force

In the *Function* field, where the magnitude of the force must be specified, the following expression has to be entered:

ARYVAL(ARRAY_Y, *position*)

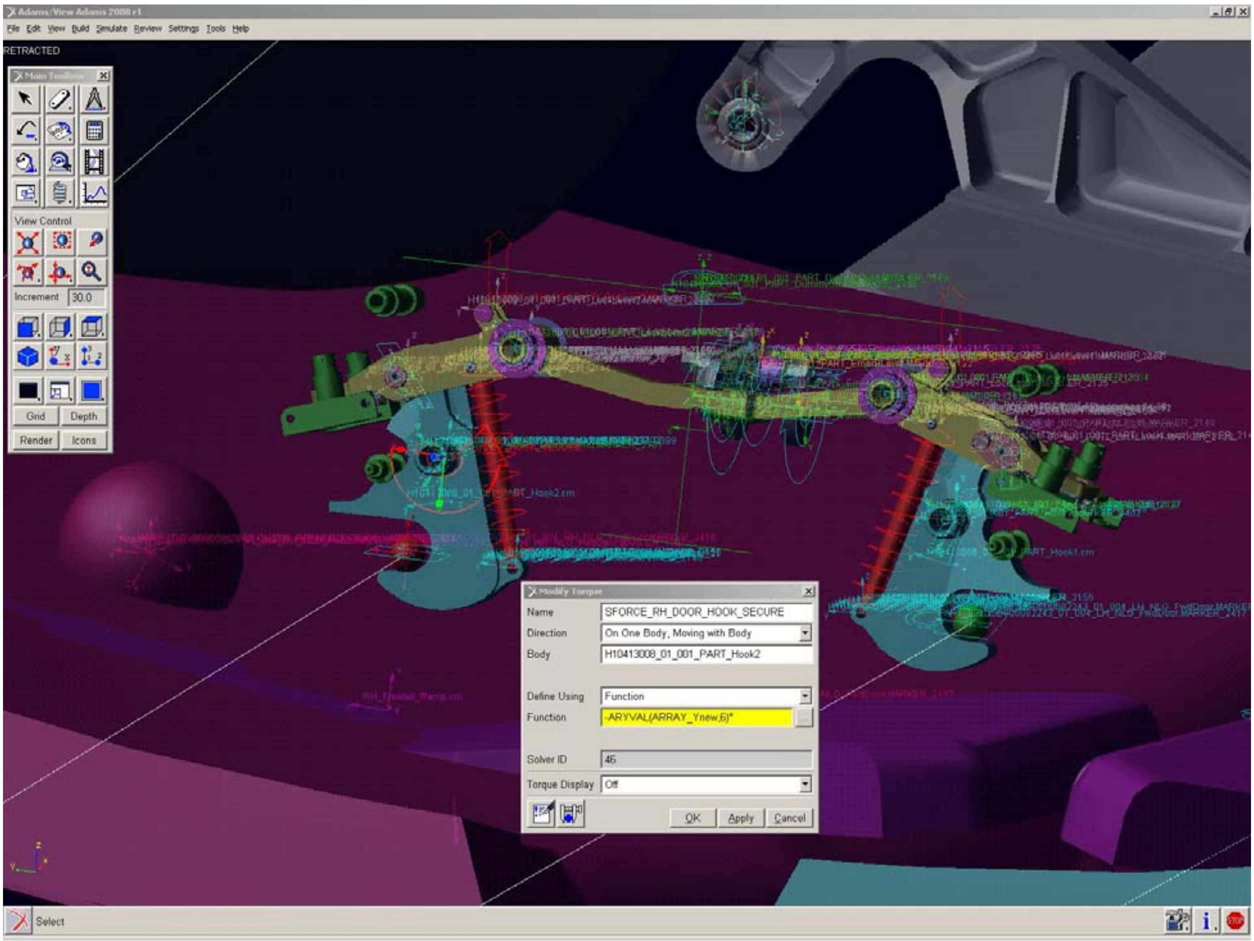
where *position* is a number that identifies the position of the retraction actuator force value in the Y array. Said position depends on how the inputs to ADAMS are defined in AMESim, as explained in par. 2.4.6.

The same procedure can be adopted to create the commands for the uplocks. Figure 2. 20 and Figure 2. 21 show the door uplocks and how the mechanism is commanded.



Name: SFORCE_RH_DOOR_UPLOCK_OPEN
Direction: On One Body, Moving with Body
Body: H10413008_01_001_PART_LockLever2
Define Using: Function
Function: (1-ARYVAL(ARRAY_Ynew,6))*
Solver ID: 42
Force Display: On

Figure 2. 20 – Creation of the force that opens the uplock



Name: SFORCE_RH_DOOR_HOOK_SECURE
Direction: On One Body, Moving with Body
Body: H10413008_01_001_PART_Hook2
Define Using: Function
Function: -ARYVAL(ARRAY_Ynew,6)*
Solver ID: 46
Torque Display: Off

Figure 2. 21 – Creation of the torque that holds the mechanism in closed position

A force is applied on a mechanical latch that secures the hook in closed position. The force is activated when the uplock has to be disengaged, i.e. when the door uplock variable sent from AMESim is equals to 0. The magnitude of the force is therefore defined as follows:

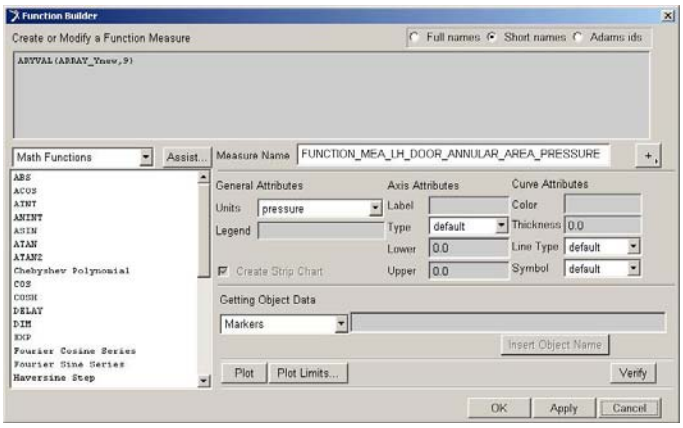
$$(1\text{-ARYVAL}(\text{ARRAY\_Y}, \mathit{position}))*\mathit{force\_magn}$$

where *position* identifies the door uplock Boolean signal in the Y array, and *force_magn* is the appropriate magnitude of the force when fully active.

Moreover, a torque is applied on the hook, to secure it efficiently in closed position when the door is uplocked. The magnitude of the torque is consequently defined as follows:

$$-1*\text{ARYVAL}(\text{ARRAY\_Y}, \mathit{position})*\mathit{torque\_magn}$$

Finally, even if no more variables need to be exchanged between the models to make the co-simulation work, some additional variables can be passed from AMESim to ADAMS. In fact, to make the postprocessing easier, it is possible to transfer pressure measurements in actuator chambers to ADAMS, which can acquire them by means of measures. At the end of the simulation, said measures can be loaded in ADAMS/Postprocessor for an easy postprocessing. Therefore, all the necessary measures (pressures in door actuators and gear retraction actuators) were created, and Figure 2. 22 shows the definition of one of them.



Function Builder

Create or Modify a Function Measure

○ Full names ● Short names ○ Adams ids

```
ARYVAL(ARRAY_Ynew,9)
```

Math Functions

Measure Name: FUNCTION_MEA_LH_DOOR_ANNULAR_AREA_PRESSURE

General Attributes
Units: pressure
Legend:
☑ Create Strip Chart

Axis Attributes
Label:
Type: default
Lower: 0.0
Upper: 0.0

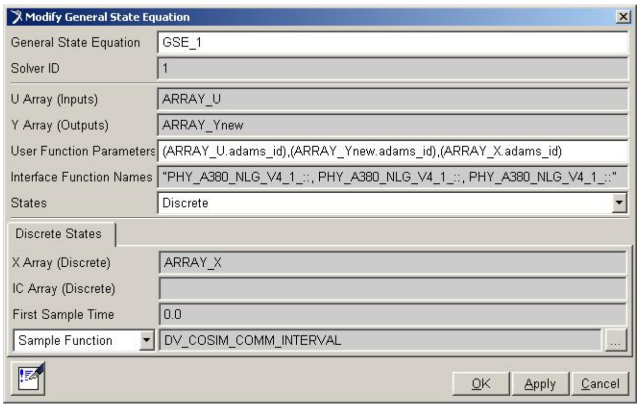
Curve Attributes
Color:
Thickness: 0.0
Line Type: default
Symbol: default

Getting Object Data: Markers

Figure 2. 22 – Creation of a measure to transfer AMESim pressures to ADAMS

An additional array called X states array must be created in ADAMS. To perform a co-simulation, it is necessary to type 1 in the *Size* field of the array.

Finally, the link between the ADAMS model and the AMESim model can be established by creating a General State Equation, as shown in Figure 2. 23.



Modify General State Equation

General State Equation: GSE_1
Solver ID: 1
U Array (Inputs): ARRAY_U
Y Array (Outputs): ARRAY_Ynew
User Function Parameters: (ARRAY_U.adams_id),(ARRAY_Ynew.adams_id),(ARRAY_X.adams_id)
Interface Function Names: "PHY_A380_NLG_V4_1_::, PHY_A380_NLG_V4_1_::, PHY_A380_NLG_V4_1_::"
States: Discrete

Discrete States
X Array (Discrete): ARRAY_X
IC Array (Discrete):
First Sample Time: 0.0
Sample Function: DV_COSIM_COMM_INTERVAL

Figure 2. 23 – Definition of the General State Equation

In the *U array*, *Y array* and *X array* fields, the arrays previously created must be selected.

In the *User Function Parameters* field, the following expression must be typed:

(ARRAY_U.adams_id),(ARRAY_Y.adams_id),(ARRAY_X.adams_id)

The *Interface Functions Names* field allows the indication of the AMESim model to be used. When AMESim is set for the co-simulation, it generates a .dll file, which is the responsible for the communication between the two programs. AMESim assigns to said file the name of the model, and adds an underscore ("_") at the end. This happens because an AMESim model consists in a .ame file, which is actually a package of files that is automatically unpacked when the model is opened. All the files belonging to the package receive the model filename and the final underscore is added, together with the appropriate extension for each of them. Therefore, in order to indicate the .dll file to ADAMS, the *Interface Functions Names* field must be filled in the following way:

"*filename_*::, *filename_*::, *filename_*::"

where *filename* is the name of the AMESim model without extension.

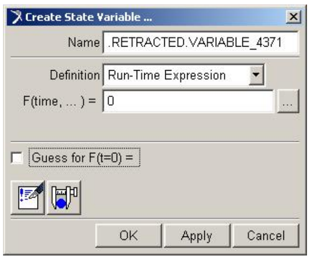
Finally, in the *States* field *Discrete* must be selected, while the *Continuous* option has to be selected for a full export; in the *Sample Function* field the communication interval for the co-simulation must be specified.

Regarding ADAMS/Solver options, the default integrator was selected, and the FORTRAN solver was chosen, and set to External.

Co-Simulation from AMESim

Even though the choice of a co-simulation from ADAMS was made, the setup of a co-simulation from AMESim was explored. The approach to be adopted is equivalent: measures need to be created for ADAMS outputs, and forces/torques for inputs. Regarding the latter, the syntax to be adopted for their magnitude is however different, because for this method the X, Y and U array are not created.

For each input to ADAMS, an empty state variable must be created, as shown in Figure 2. 24; 0 must be typed in the *F(time,…)* field.



Name .RETRACTED.VARIABLE_4371

Definition Run-Time Expression

F(time, ...) = 0

☐ Guess for F(t=0) =

Figure 2. 24 – Definition of an empty state variable

After the setup, each state variable will act as an interface between the models, and will be continuously updated during the simulation, following the evolution of one of the quantities sent to ADAMS by AMESim.

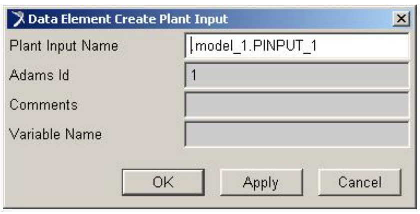
Then, for each force and torque, the following syntax must be adopted to define the magnitude:

VARVAL(*statevariablename*)

where *statevariablename* is the name of the state variable to be associated to the force/torque.

To create the interface between the two models, the *Plant Input* and *Plant Output* features have to be used.

The *Plant Input* window is shown in Figure 2. 25:



Plant Input Name .model_1.PINPUT_1

Adams Id 1

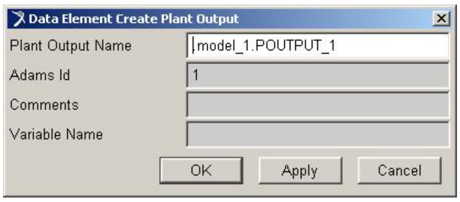
Comments

Variable Name

Figure 2. 25 – Plant input feature

In the *Variable Name* field, all the state variables previously created for ADAMS inputs have to be listed.

Similarly, all the ADAMS outputs (i.e. the measures previously created) have to be listed in the *Plant Output* window, shown in Figure 2. 26:

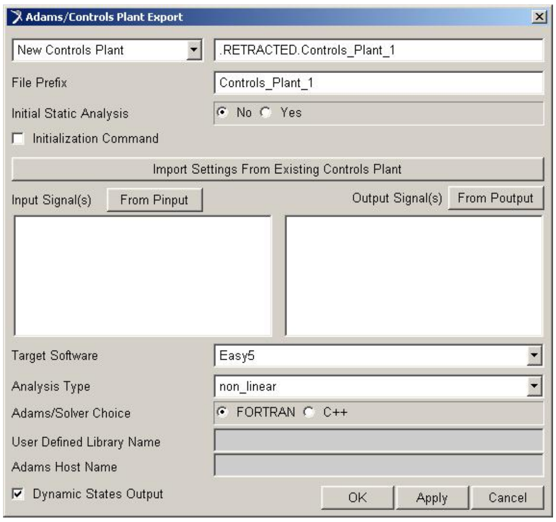


Plant Output Name: model_1.POUTPUT_1
Adams Id: 1
Comments:
Variable Name:

Figure 2. 26 – Plant output feature

Again, the sequence of inputs and outputs listed in the previous windows must be coincident to the order in which inputs and outputs are defined in AMESim, as explained in par. 2.4.6.

Finally, the *ADAMS/Controls* toolbox must be activated, and the *Plant Export* feature has to be used, as shown in Figure 2. 27.



New Controls Plant: .RETRACTED.Controls_Plant_1
File Prefix: Controls_Plant_1
Initial Static Analysis: ● No ○ Yes
☐ Initialization Command
Input Signal(s)
Output Signal(s)
Target Software: Easy5
Analysis Type: non_linear
Adams/Solver Choice: ● FORTRAN ○ C++
User Defined Library Name:
Adams Host Name:
☑ Dynamic States Output

Figure 2. 27 – Plant Export feature

The inputs and outputs previously set using the *Plant Input* and *Plant Output* have to be retrieved by clicking on the *From Pinput* and *From Poutput* buttons, and then selecting them. All the input and output signals then appear in the two white boxes of the *Plant Export* window.

The *Initial Static Analysis* option has to be set on *Yes* if a static analysis of the mechanical model is desired before the beginning of the co-simulation (which is then started from the equilibrium position of the mechanical model).

In the *Target Software* field, *Easy5* must be selected. The *Analysis Type* must be set on *non-linear*, and in the *ADAMS/Solver Choice* field *FORTRAN* can be selected.

Finally, if the mechanical model needs a special .dll library file in order to run, the filename has to be written in the *User Defined Library Name* field.

2.4.6 – Modifications to the hydraulic model

Modifications to the hydraulic sketch were necessary in order to enable the communication between the two programs.

In general, an AMESim actuator requires, at the interface between the rod and the external load, the exchange of the following variables:

- actuator displacement and velocity as inputs;
- force generated by actuator rod as output.

So, for the co-simulation to work correctly, said exchange of variable between the two programs must occur for every actuator. Therefore, the assembly shown in Figure 2. 28 was placed in front of each actuator.

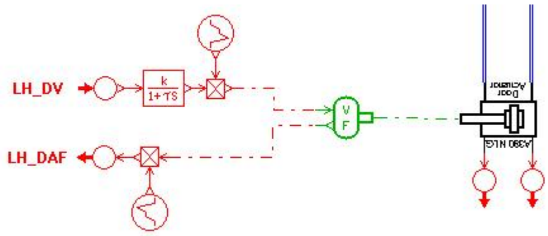


Figure 2. 28 – Exchange of variables for each AMESim actuator

The green block accepts in input a signal corresponding to the actuator velocity and passes it to the actuator, calculating in addition the actuator displacement by integration of the velocity. Therefore, an initial condition for the displacement must be supplied, and that can be done by using the global parameters previously defined and shown in Table 2.1. The actuator velocity is sent to the block by ADAMS, where a velocity measure is set between the two actuator attachment points, as explained in par. 2.4.5. The green block then receives the hydraulic force from the actuator and transforms it into a signal, which can be sent to ADAMS and assigned as the magnitude of the actuator force created between the two actuator attachment points. The AMESim actuator submodel automatically reverses the sign of the velocity signal that receives as an input. So a –1 gain has to be put in place to multiply the velocity input signal to correct the discrepancy. Moreover, another gain must be included to provide a unit conversion for the velocity (in fact ADAMS uses mm/s while AMESim requires m/s). Therefore it is sufficient to consider only one gain and assign to it the value of –0.001.

In Figure 2. 28 the gain is included as part of a first order lag component which multiplies the velocity input signal. The reason for the use of the first order lag is related to the problems that can occur when the communication interval between the two models is too big. In that case, discontinuities can be generated in the velocity signal coming from ADAMS, and vibration phenomena can appear in the system dynamics. Figure 2. 29 shows high frequency fluctuations in an actuator pressure measurement, due to the use of a too large communication interval.

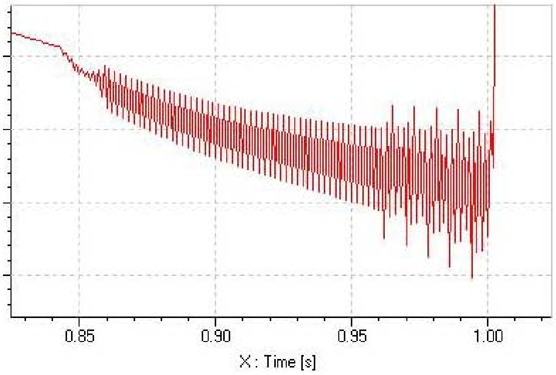


Figure 2. 29 – Vibrations in actuator chamber pressure measurement

Therefore, the communication interval must be reduced until vibrations disappear from the dynamic response of the system. On the other hand, an excessive reduction of said parameter leads to huge CPU times, which are often incompatible with the accuracy/efficiency compromise required for the model.

The use of the first order lag helps in reducing the need of a very small communication interval: vibrations can disappear using higher values for the communication interval, respect to what would be needed without the use of the lag. The first order lag, in fact, is able to "filter" the velocity signal and reduce the discontinuities coming from ADAMS, which cause the vibration phenomena. The time constant for the lag must be small enough, to keep the desired accuracy of the results. The value that was adopted is *communication_interval/5*.

In addition, forces and velocities are multiplied by a signal, which is kept constantly 0 during the initial phase of the simulation, then changes rapidly from 0 to 1 and finally is kept constantly equals to 1 throughout the simulation. A plot of said signal is represented in Figure 2. 30.

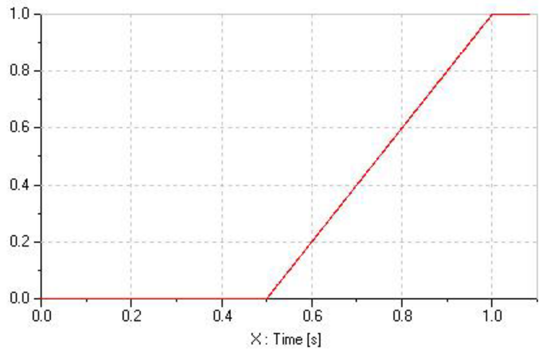


Figure 2. 30 – Signal that multiplies velocity and force

This method artificially suppresses the communication between the two programs at the beginning of the simulation, and then restores it gradually to a full value. This allows a completely independent initial stabilisation of the two models: AMESim can reach an initial equilibrium of pressures, and the ADAMS model can reach a mechanical equilibrium. The fact that the interface is initially suppressed avoids an influence between the two models during the stabilisation.

The communication between AMESim and ADAMS occurs via an interface block that has to be created in the sketch, as shown in Figure 2. 31.

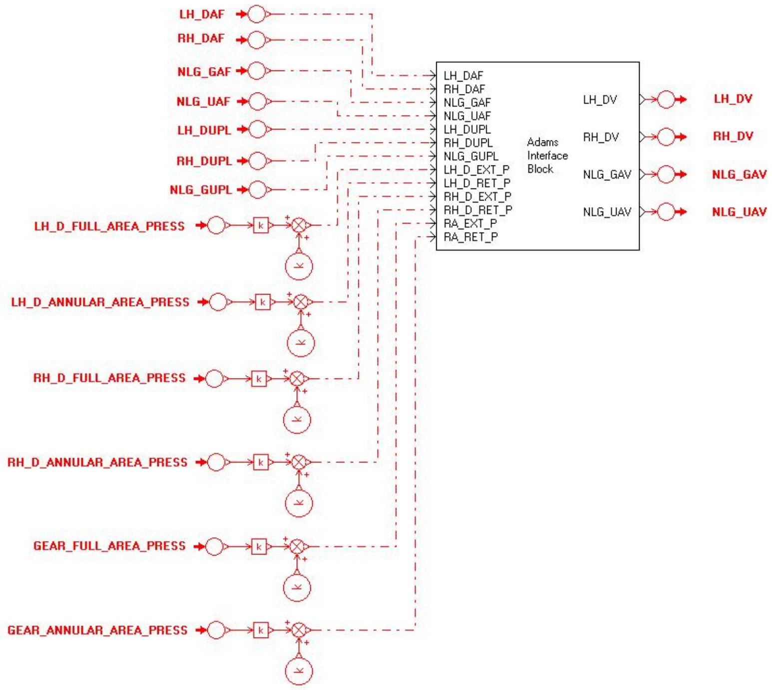


Figure 2. 31 – AMESim/ADAMS interface block

On the left hand side of the block all the inputs to ADAMS coming from AMESim are listed, while on the right hand side all the output variables coming from ADAMS and passed to AMESim are shown. The red elements with a circle and an arrow are transmitters and receivers, and each transmitter is coupled with its own receiver, in order to create a connection between two ports on the sketch without visualizing a line (to keep the sketch clean). The receivers on the left hand side receive all the forces transmitted by the actuators (as previously described) and doors and gear uplock commands. The transmitters on the right hand side send actuator velocity measures coming from ADAMS to AMESim actuators. In addition, pressure sensors are set in the actuator chambers, and their signals are sent to the interface block together with the forces, and so additional pressure inputs can be seen on the left hand side of the block (with gains and sums to perform some unit conversions). This allows the transfer of pressure measurements to ADAMS, which can be loaded in ADAMS/Postprocessor as standard measures, as previously explained. When a run is launched and completed, pressure measures are automatically updated, together with other mechanical variables that need to be monitored. This method can significantly speed up the validation process, as explained in the next chapter.

It is important that the order of inputs and outputs on the interface block matches the order of the variables in the U and Y arrays in ADAMS. Therefore, the output ports of the block must be ordered as the measures in the U array, proceeding from the bottom to the top of the block. Similarly, the order of the input ports from the top to the bottom of the block determines the position of each variable in the Y array, to be assigned to forces and torques in ADAMS, as previously explained.

2.4.7 – Additional comments

After the correct setup of the programmes and the two models, the co-simulation can be launched from ADAMS as if running a classic ADAMS mechanical simulation. AMESim needs to be kept open, and runs in background while in the ADAMS message window all the messages displayed by both solvers are shown.

It is always possible to make modifications to both models, and launch the co-simulation afterwards, without the need of any reconfiguration of the interface (unless a modification of the variables exchanged by the two models is made).

2.5 – Verification of the interface

Once the co-simulation interface was created and configured as shown, the two models (at this point fully integrated and acting as single model) were verified against top-level requirements. In particular, an initial tuning of the communication interval was performed for each simulation case, by checking the quality of the plot of the reference pressures and door/gear angles.

After these preliminary tests, the model had to be tuned in order to validate it against test rig data. Temporary modifications were made to the model in order to speed up the validation process, as explained in the following chapter.

Chapter 3 – Model Validation

3.1 – Overview

The validation process requires a tuning of some key parameters, both in the mechanical and in the hydraulic model, in order to match its behaviour with reference data coming from the A380 Landing Gear Test Rig.

Test Rig Data were acquired for extension, retraction and freefall cases (hangar case only); among all the variables monitored, the following were selected as representative of the system's behaviour:

- pressures in door actuator chambers;
- pressures in gear actuator chambers;
- door retraction angle;
- gear retraction angle.

Therefore the tuning of the model was performed in order to match those quantities. The parameters that were modified during the validation process are:

- door actuator seal friction;
- door actuator flow control restrictors (at the inlet of full and annular chamber);
- door actuator snubbing orifices;
- gear actuator seal friction;
- gear actuator flow control restrictors (at the inlet of full and annular chamber);
- gear actuator snubbing orifices;
- door uplock roller diameter;
- gear uplock roller diameter;
- stiffness and damping of gear actuator – fuselage attachment point.

3.2 – Modifications to the models

3.2.1 – Overview

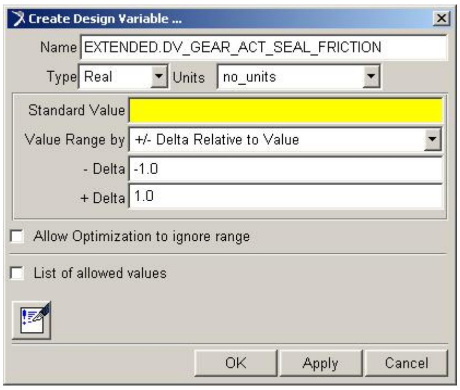
Both the AMESim and the ADAMS models were modified to be able to run several simulations, changing the values of key parameters automatically. The tuning of the model, in fact, is a long process that requires a high number of simulation runs to be performed. Several parameters need to be tuned at the same time, and many iteration cycles have to be started, to let the model dynamics converge on the real system's behaviour.

The use of ADAMS/Insight has been adopted to perform a Design of Experiment. ADAMS/Insight is an ADAMS toolbox that allows launching a sequence of batch runs from ADAMS/View, and during each run a set of parameters can be varied, to explore the consequent change in the system's response. Said parameters have to be defined as Design Variables in ADAMS, and then set in Insight as quantities to be varied during the Design of Experiment.
For each simulation, the values of the design variables are updated and the results automatically stored in a separate file, containing the model animation and the results set. Each results file can be then loaded in ADAMS/View, and results can be examined using ADAMS/Postprocessor.

3.2.2 – Modifications to the hydraulic model

Since ADAMS/Insight is designed to operate on mechanical parameters in the ADAMS model, some modifications to the AMESim model were necessary in order to control hydraulic parameters with the ADAMS toolbox. The procedure that has been created will be explained referring to seal friction tuning, but can be adopted for the tuning of all the other hydraulic parameters (i.e. restrictor flow areas).

The first step consists in the definition of a design variable in ADAMS, containing the nominal value of the seal friction, as shown in Figure 3. 1.



Name EXTENDED.DV_GEAR_ACT_SEAL_FRICTION

Type Real

Units no_units

Standard Value

Value Range by +/- Delta Relative to Value

- Delta -1.0

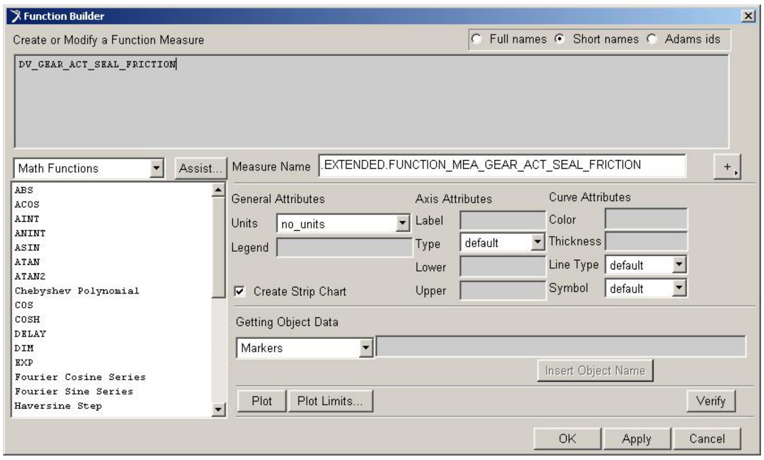
+ Delta 1.0

☐ Allow Optimization to ignore range

☐ List of allowed values

Figure 3. 1 – Design variable for seal friction

The value of the design variable has to be passed to AMESim through the co-simulation interface, so it has to be added to the ADAMS outputs and therefore listed in the U array. But since only measures can be accepted by the U array, the design variable must be assigned to an appropriate measure, which will remain coincident with the design variable throughout the simulation. Figure 3. 2 shows the creation of the measure.



Create or Modify a Function Measure

○ Full names ● Short names ○ Adams ids

```
DV_GEAR_ACT_SEAL_FRICTION
```

Math Functions

Measure Name .EXTENDED.FUNCTION_MEA_GEAR_ACT_SEAL_FRICTION

Units no_units

Type default

Line Type default

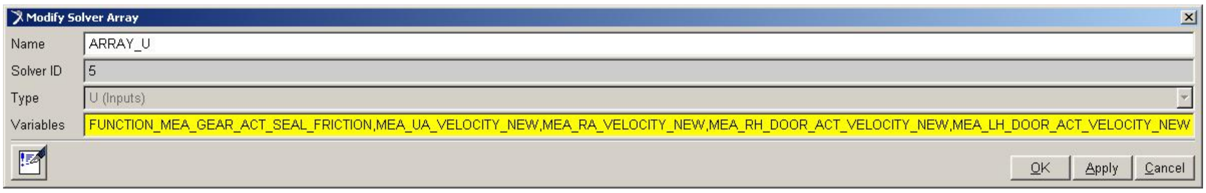
Symbol default

☑ Create Strip Chart

Getting Object Data: Markers

Figure 3. 2 – Creation of a measure coincident with the seal friction design variable

Once the measure is created, it can be added to the existing U array, as highlighted in Figure 3. 3.



Name ARRAY_U

Solver ID 5

Type U (Inputs)

Variables FUNCTION_MEA_GEAR_ACT_SEAL_FRICTION,MEA_UA_VELOCITY_NEW,MEA_RA_VELOCITY_NEW,MEA_RH_DOOR_ACT_VELOCITY_NEW,MEA_LH_DOOR_ACT_VELOCITY_NEW

Figure 3. 3 – Modification to the U array

Then, modifications to the AMESim model are necessary, to add the seal friction to the outputs of the interface block. The latter has to be deleted and created again, adding a new variable to the list of outputs. It is very important to match the order of the outputs with the order of the measures listed in the U array, so if the friction is added at the beginning of the array, it has to be placed at the first output port of the interface block, as shown in Figure 3. 4.

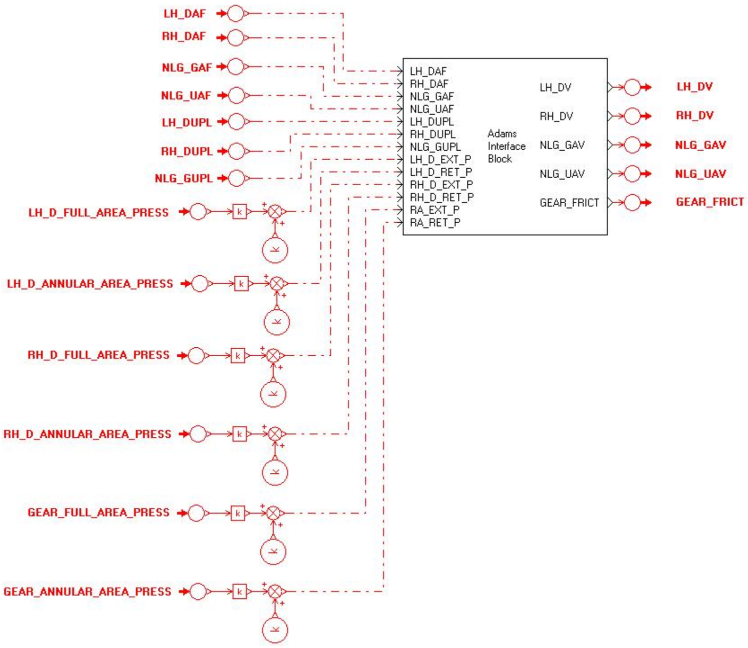


Figure 3. 4 – Modification to the interface block

Finally, using a transmitter/receiver pair, the seal friction variable can be sent to the friction generator block, which accepts the value of friction as an input and sums it to the force exerted by the actuator rod, as shown in Figure 3. 5.

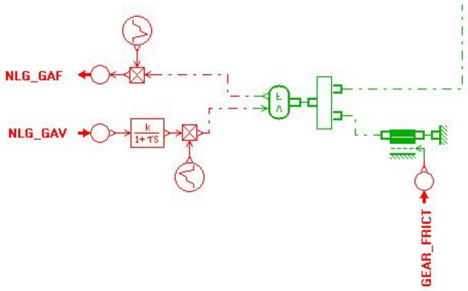


Figure 3. 5 – Input to friction generator block

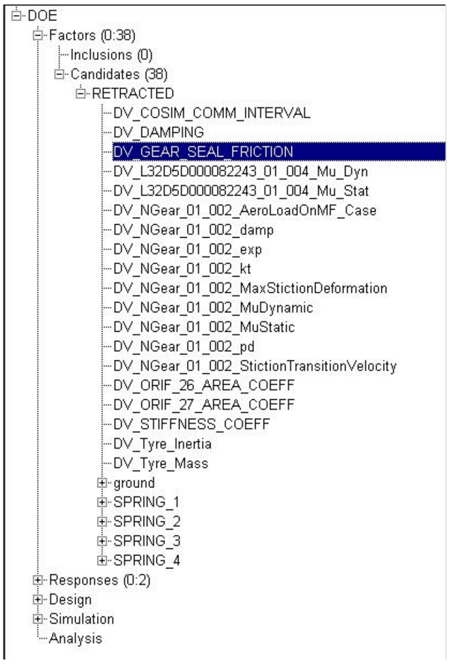
The same procedure can be repeated for each orifice to be tuned; in this case the design variable created in ADAMS consists in a correction factor used to multiply the orifice nominal area. For the purpose, the orifice needs to be replaced with a variable orifice, as shown in Figure 3. 6.

Figure 3. 6 – Variable orifice with the correction factor as input signal

The properties of the orifice must be set so that the flow area is the orifice nominal area when the input signal is set to 1. By changing the input signal, the nominal area can be multiplied by a correction factor in order to tune the orifice. The procedure to transform the ADAMS design variable into an input signal for the orifice is identical to what explained for seal friction.

3.2.3 – Setup of ADAMS/Insight for model tuning

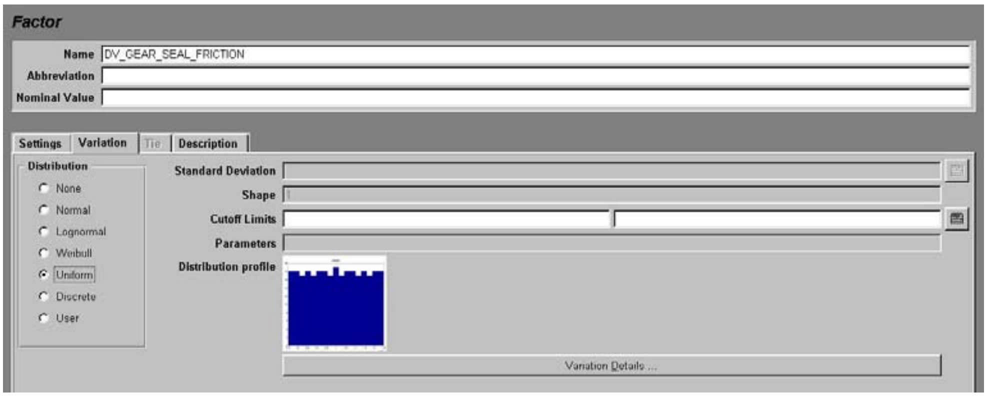
The first step to setup a sequence of batch runs in ADAMS/Insight consists in the selection of the parameters to be varied during the process. If the design variables for said parameters are correctly defined, they should appear in the *Candidates* list of the *Factors* menu, as shown in Figure 3. 7.



- DOE
  - Factors (0:38)
    - Inclusions (0)
    - Candidates (38)
      - RETRACTED
        - DV_COSIM_COMM_INTERVAL
        - DV_DAMPING
        - DV_GEAR_SEAL_FRICTION
        - DV_L32D5D000082243_01_004_Mu_Dyn
        - DV_L32D5D000082243_01_004_Mu_Stat
        - DV_NGear_01_002_AeroLoadOnMF_Case
        - DV_NGear_01_002_damp
        - DV_NGear_01_002_exp
        - DV_NGear_01_002_kt
        - DV_NGear_01_002_MaxStictionDeformation
        - DV_NGear_01_002_MuDynamic
        - DV_NGear_01_002_MuStatic
        - DV_NGear_01_002_pd
        - DV_NGear_01_002_StictionTransitionVelocity
        - DV_ORIF_26_AREA_COEFF
        - DV_ORIF_27_AREA_COEFF
        - DV_STIFFNESS_COEFF
        - DV_Tyre_Inertia
        - DV_Tyre_Mass
        - ground
        - SPRING_1
        - SPRING_2
        - SPRING_3
        - SPRING_4
  - Responses (0:2)
  - Design
  - Simulation
  - Analysis

Figure 3. 7 – Selection of parameters to be tuned

For each chosen parameter, its range of variation has to be defined, as shown in Figure 3. 8.



Name DV_GEAR_SEAL_FRICTION

Distribution ● Uniform

Figure 3. 8 – Definition of the range of variation of a parameter

Several methods are available to define the distribution of the tentative values of the parameter around its nominal value: normal, lognormal and Weibull distributions, uniform distribution between two extreme values, user defined distribution.

The measures to be monitored during the Design of Experiment, in order to analyse the impact of the tuning on the response of the system, can be selected among the *Candidates* of the *Responses* menu, as shown in Figure 3. 9.

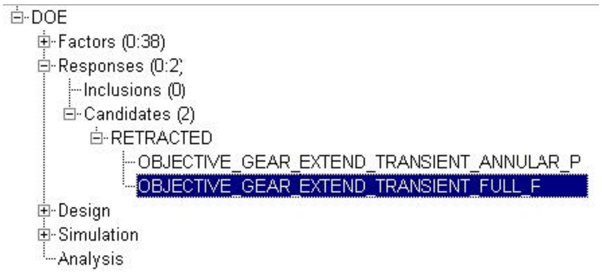
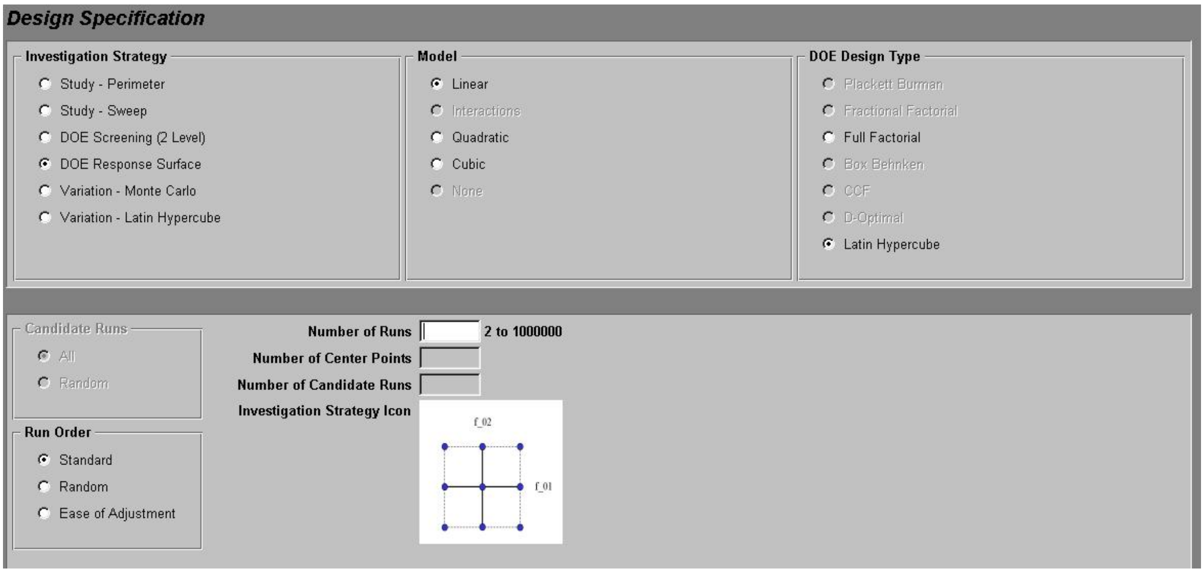


Figure 3. 9 – Selection of responses to be monitored

Finally, the type of analysis to be performed must be selected. By choosing a DOE Response Surface and Latin Hypercube in the window shown in Figure 3. 10, an editable workspace is automatically created, depending on the total number of runs chosen and on the definition of the range of variation adopted.



Design Specification

Investigation Strategy
○ Study - Perimeter
○ Study - Sweep
○ DOE Screening (2 Level)
● DOE Response Surface
○ Variation - Monte Carlo
○ Variation - Latin Hypercube

Model
● Linear
○ Interactions
○ Quadratic
○ Cubic
○ None

DOE Design Type
○ Plackett Burman
○ Fractional Factorial
○ Full Factorial
○ Box Behnken
○ CCF
○ D-Optimal
● Latin Hypercube

Candidate Runs
● All
○ Random

Run Order
● Standard
○ Random
○ Ease of Adjustment

Number of Runs ☐ 2 to 1000000
Number of Center Points ☐
Number of Candidate Runs ☐
Investigation Strategy Icon

Figure 3. 10 – Definition of the type of analysis

The workspace can be edited, and therefore the simulation cases can be fully modified.

For each simulation run, a results file is generated. To analyse the history of a certain measure, the results file corresponding to the desired simulation run can be loaded in ADAMS/View, and then studied in ADAMS/Postprocessor, together with the animation of the model.

3.3 – Validation results and process followed for tuning

3.3.1 – Validation results

The following pictures show the results of the validation process, in terms of pressures in actuator chambers and door/gear retraction angles, for the normal extension, normal retraction and freefall hangar cases. Since both doors behave very similarly, only the plots referred to the LH door are shown.

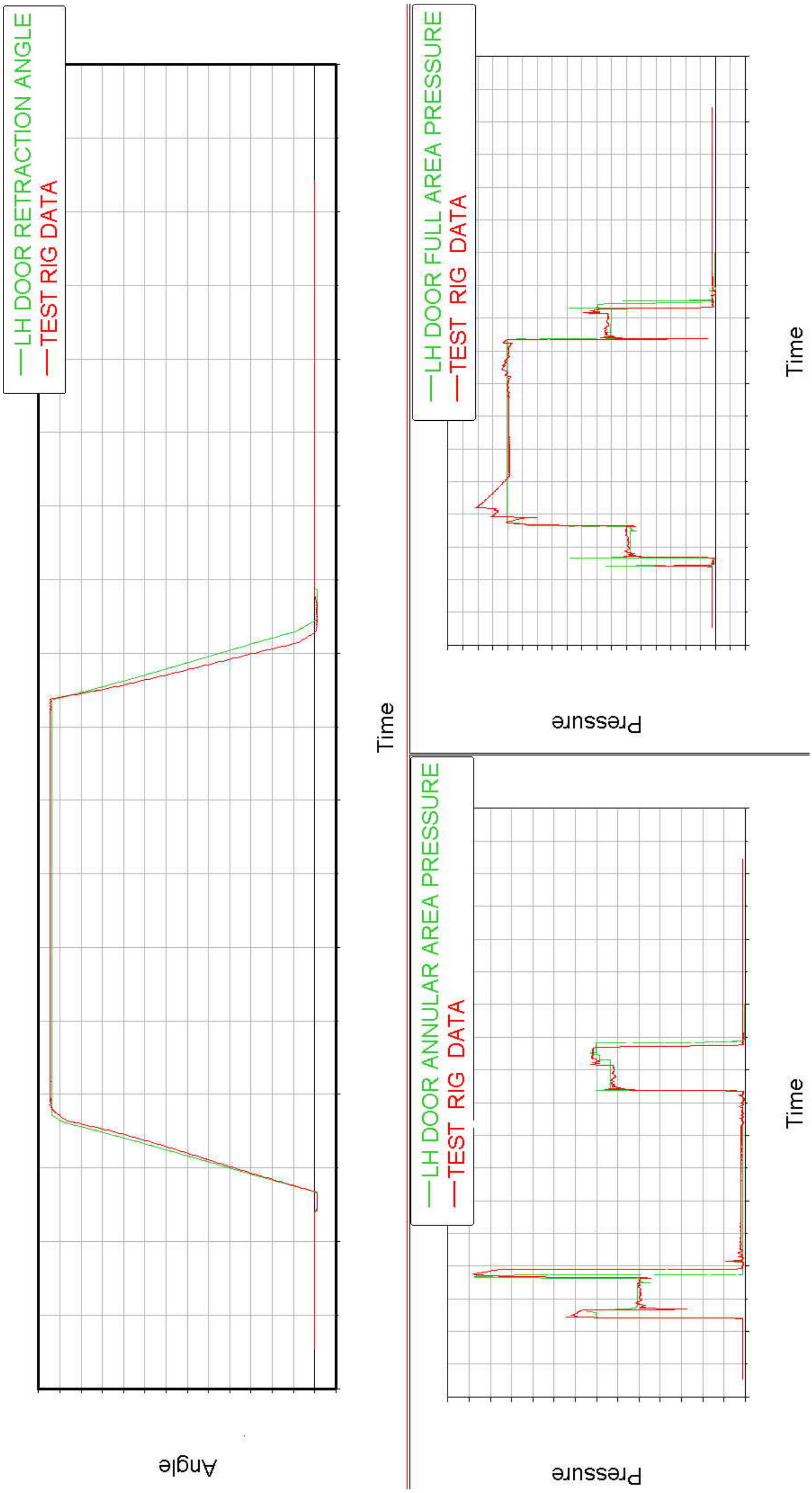


Figure 3. 11 – LH door retraction angle and pressures in door actuator chambers for normal extension

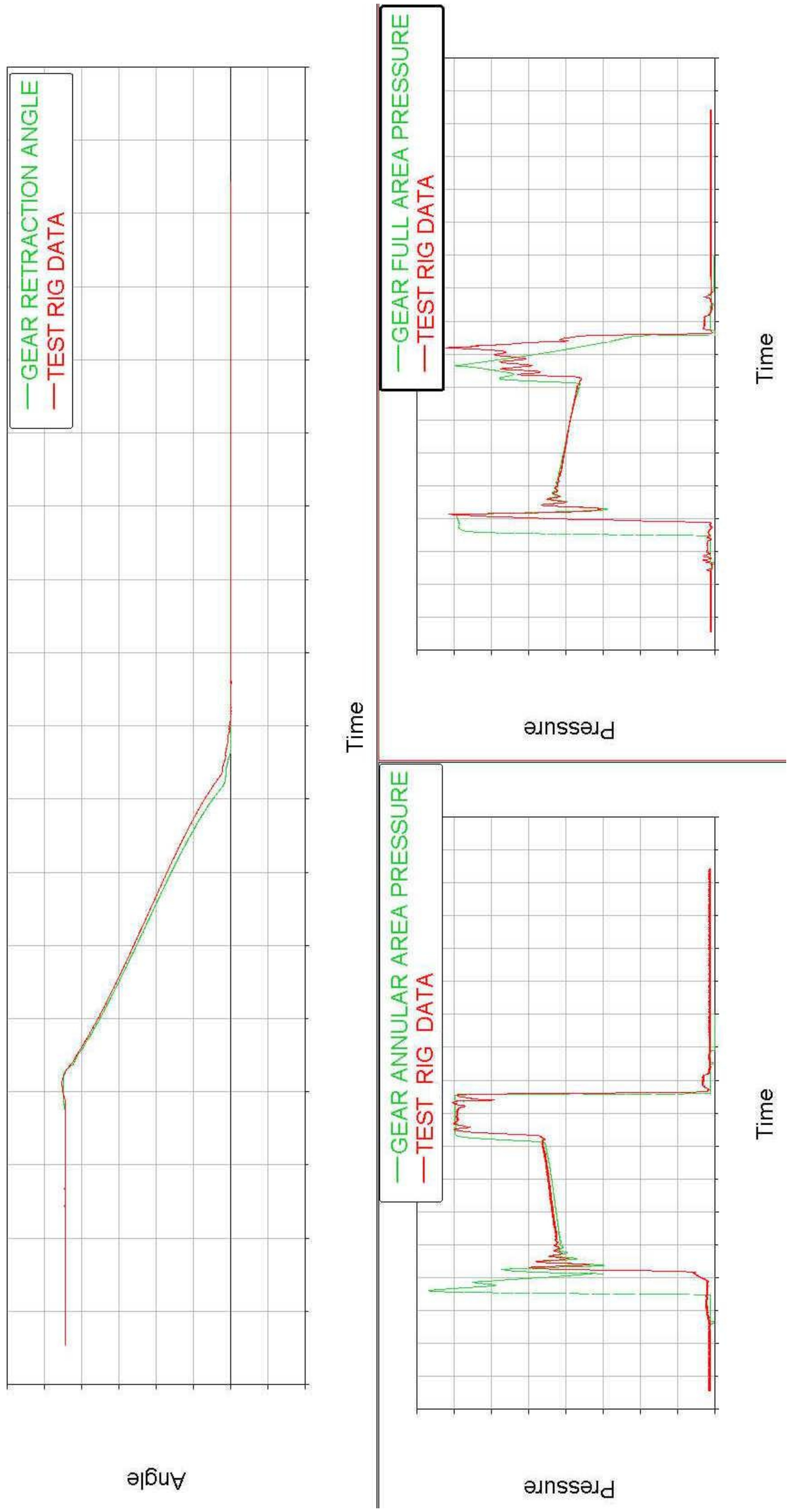


Figure 3. 12 – Gear retraction angle and pressures in gear retraction actuator chambers for normal extension

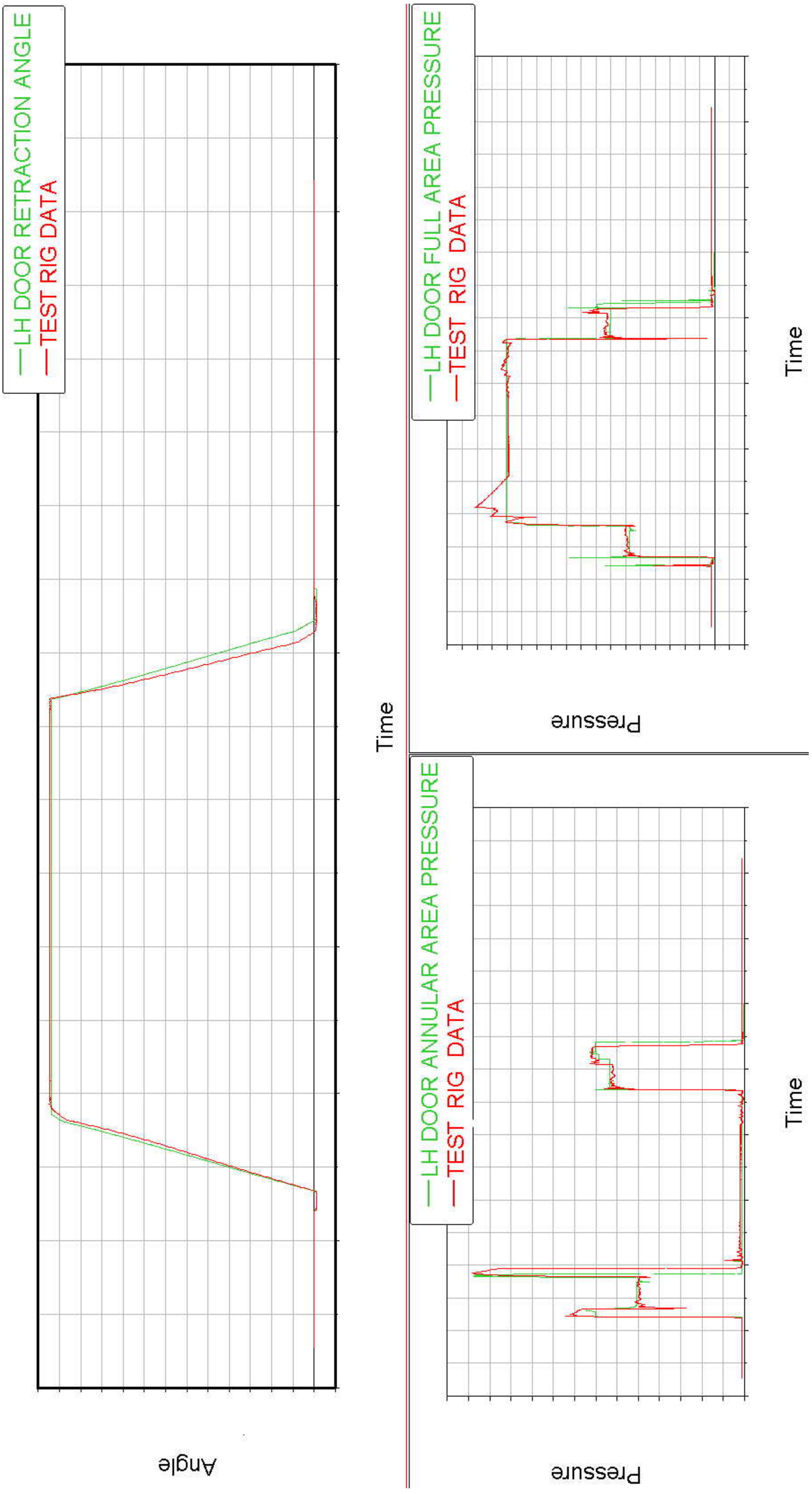


Figure 3. 13 – LH door retraction angle and pressures in door actuator chambers for normal retraction

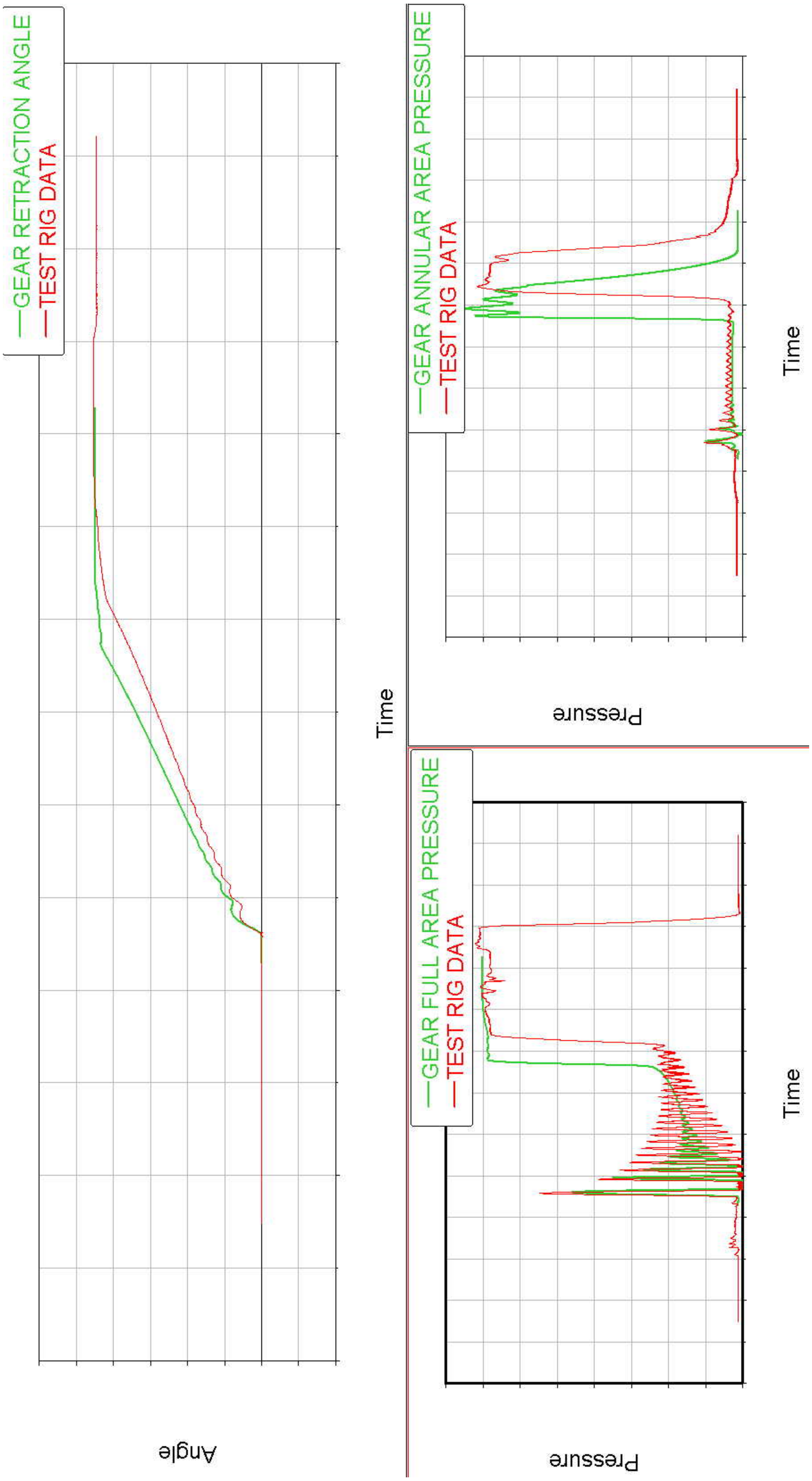


Figure 3. 14 – Gear retraction angle and pressures in gear retraction actuator chambers for normal retraction

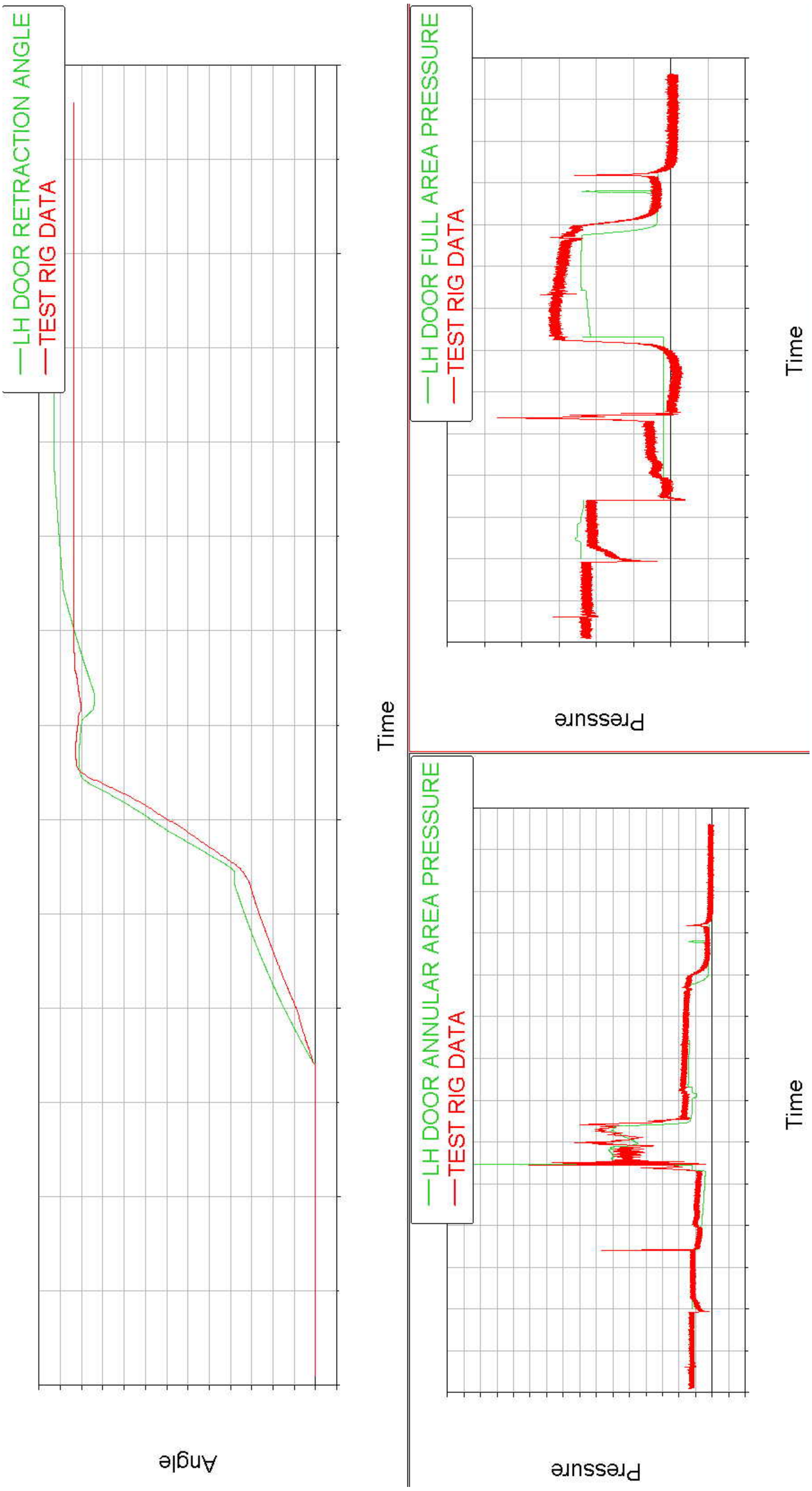


Figure 3. 15 – LH door retraction angle and pressures in door actuator chambers for freefall

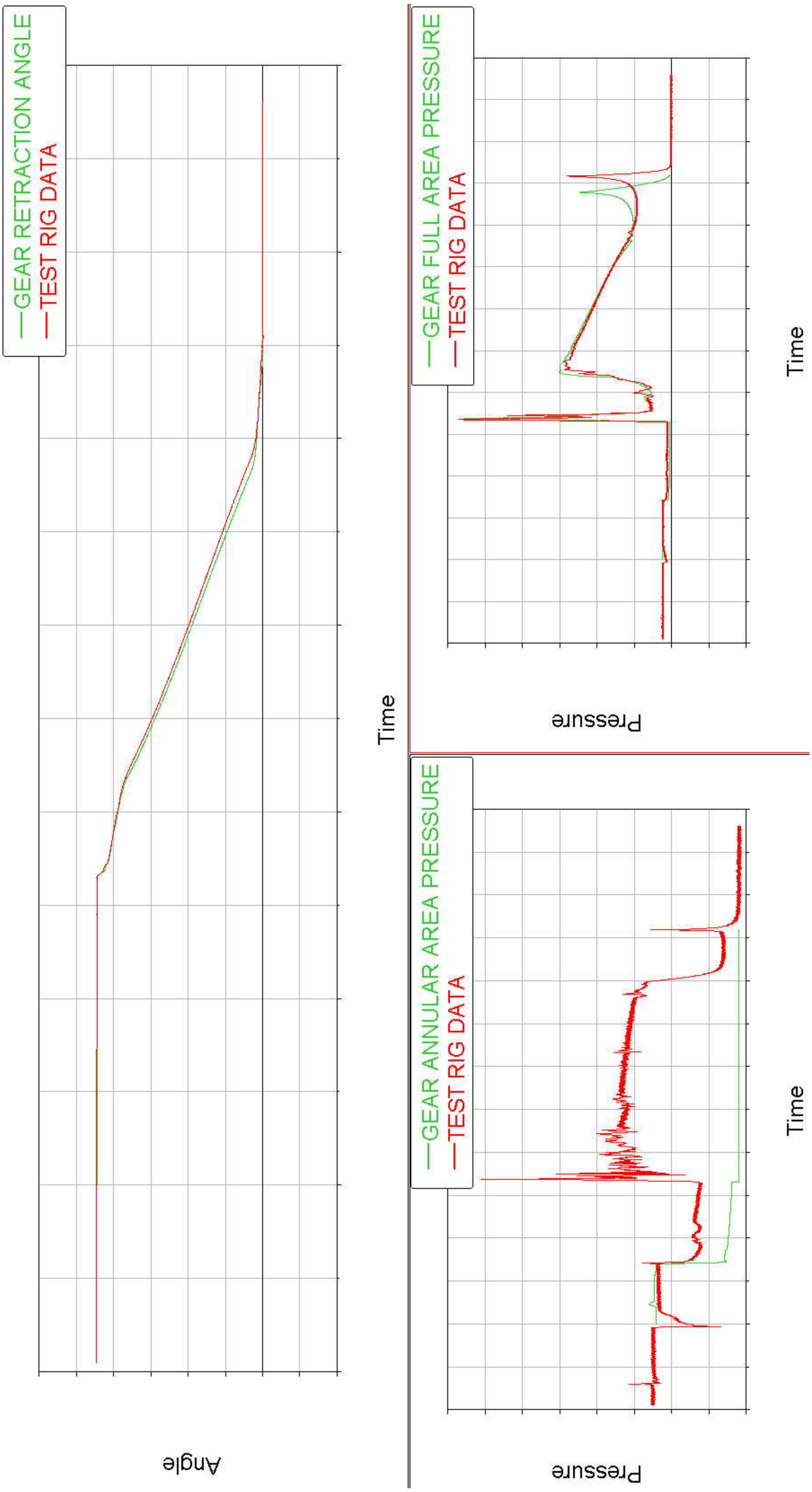


Figure 3. 16 – Gear retraction angle and pressures in gear retraction actuator chambers for freefall

The tuning of the model was performed by modifying the key parameters in a specific sequence. For each simulation scenario, in fact, it is possible to identify portions of the dynamic response that are influenced only by a limited number of parameters. Once a satisfactorily match of the variables was achieved in those areas, additional parameters were tuned to modify the remaining parts of the response. The following sequence was adopted.

3.3.2 – Doors in normal operation

Door actuator flow control restrictors for door opening

The first parameters to be tuned were door actuator flow control orifices for both chambers. The following portions of pressure plots were examined.

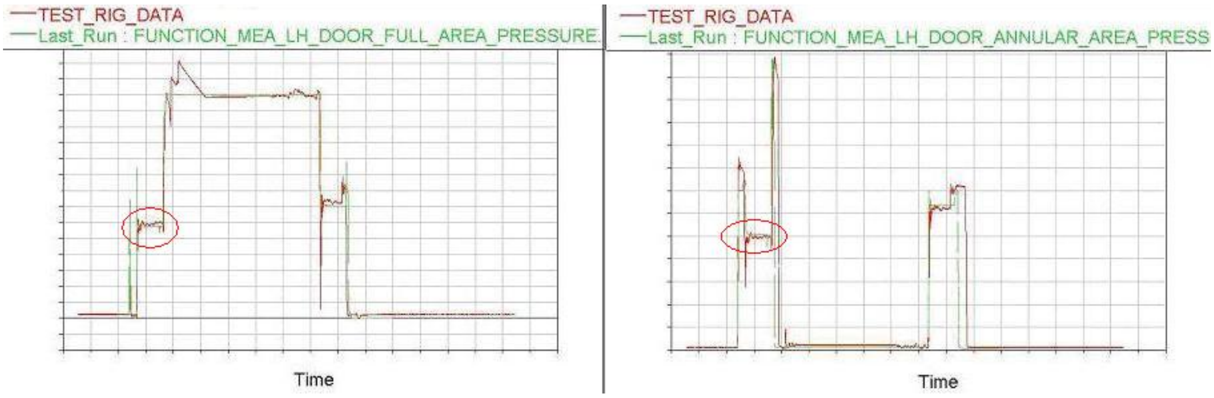


Figure 3. 17 – Tuning of flow control devices for door opening

The variation of diameter of each flow control restrictor has a direct effect on the pressure of the chamber in front of which it is placed. Nevertheless, any modification of the flow control devices changes the value of actuator velocity at which the system stabilises during this phase of the door opening (the velocity, in fact, is almost constant). Said effect causes a variation of the flow rate through both actuator ports and, according to Bernoulli’s equation, changes the pressure drop across flow control orifices. Therefore, the pressure in the opposite chamber is also impacted. Some iterations were necessary to be able to identify the diameters of flow control devices of both chambers, which guarantee an appropriate match of both pressures.

Door opening – final snubbing

In this case the tuning was focused on the flow control devices that act in the last part of the actuator stroke, to slow down the doors before they hit their endstops. The pressure peak on the annular chamber of the door actuator was matched with reference data, as shown in Figure 3. 18.

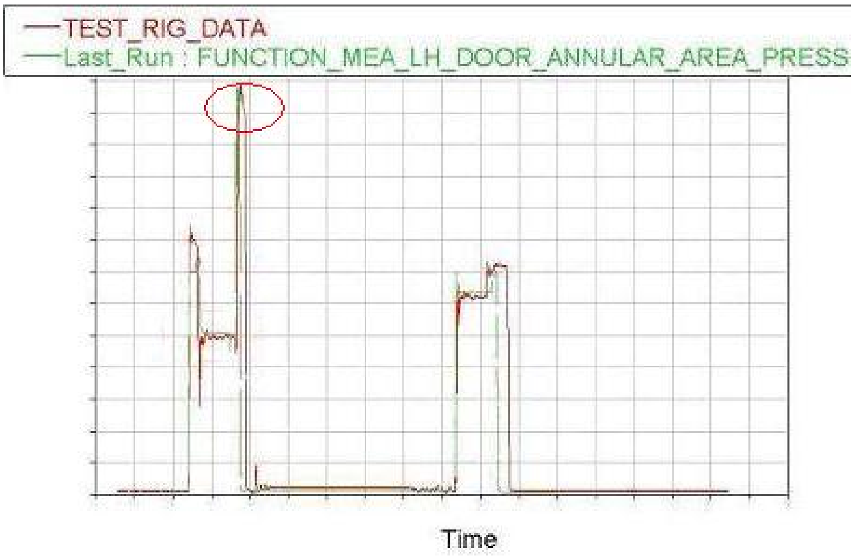


Figure 3. 18 – Tuning of snubbing for door opening

Door opening – initial travel

The diameter of the roller that is coupled with the uplock hook to keep the door closed has been slightly changed. This allows the tuning of the initial part of door angular displacement, when the door is lifted before the uplock release, to alleviate the load acting on the uplock itself. Figure 3. 19 shows the portion of the plot of door retraction angle affected by the modification, while Figure 3. 20 shows a detail of the door uplock rollers that were modified.

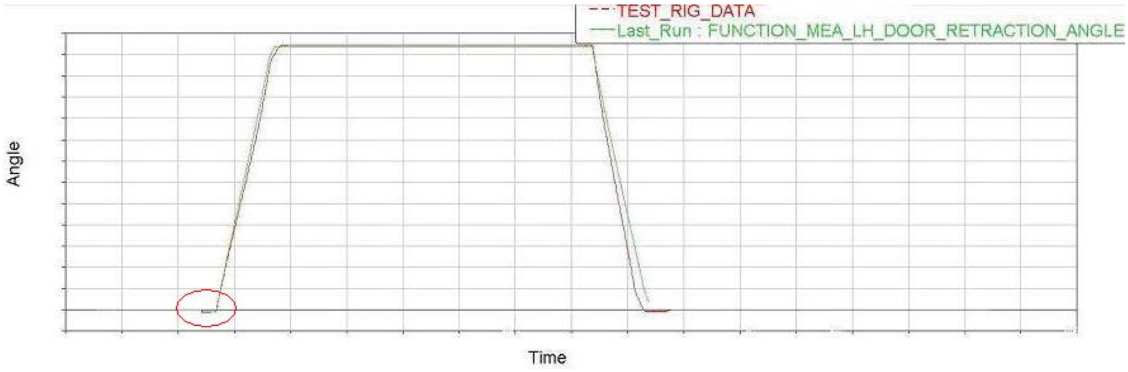


Figure 3. 19 – Tuning of initial angular displacement for door opening

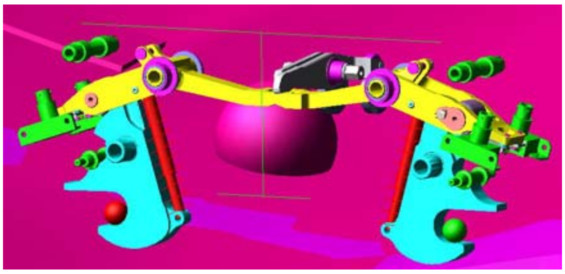


Figure 3. 20 – Door uplock rollers

Door actuator flow control restrictors for door closing and final snubbing

The tuning followed for door opening was repeated for door closing. Again, flow control devices were modified first, then snubbing was tuned. Figure 3. 21 shows the details of the pressure plots that were considered.

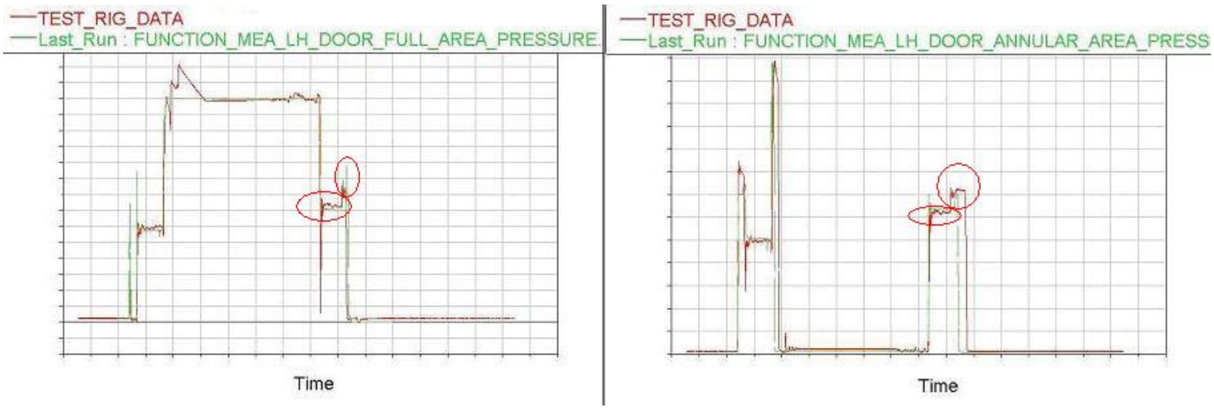


Figure 3. 21 – Tuning of flow control devices and snubbing for door closing

3.3.3 – Gear in normal operation

The tuning of gear behaviour followed the same approached adopted for doors.

Gear actuator flow control devices have been tuned to match pressures in actuator chambers, in the areas highlighted in Figure 3. 22.

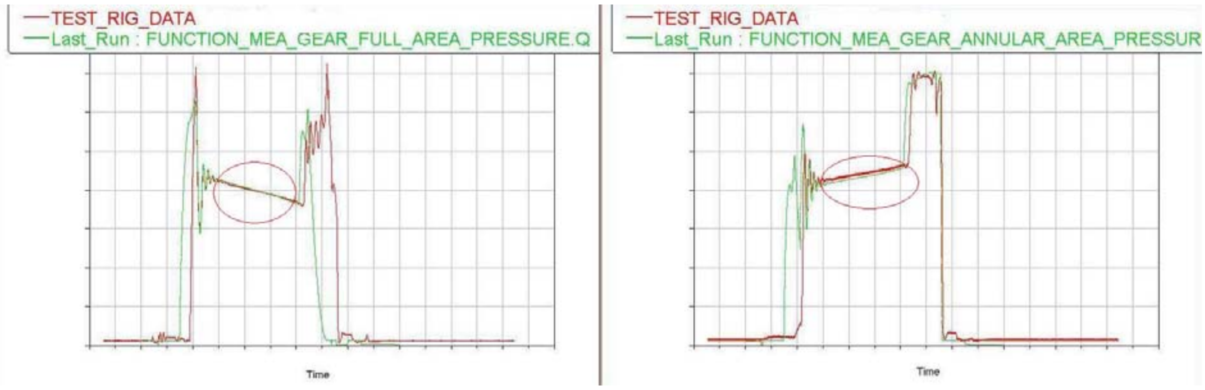


Figure 3. 22 – Tuning of flow control devices for gear extension

Initial pressure oscillations were considered, and the stiffness of gear actuator attachment point was adapted, to achieve the correct frequency of the oscillations. Damping was tuned as well, to match the amplitude of the oscillations with test rig data. Figure 3. 23 shows the related areas of pressure plots.

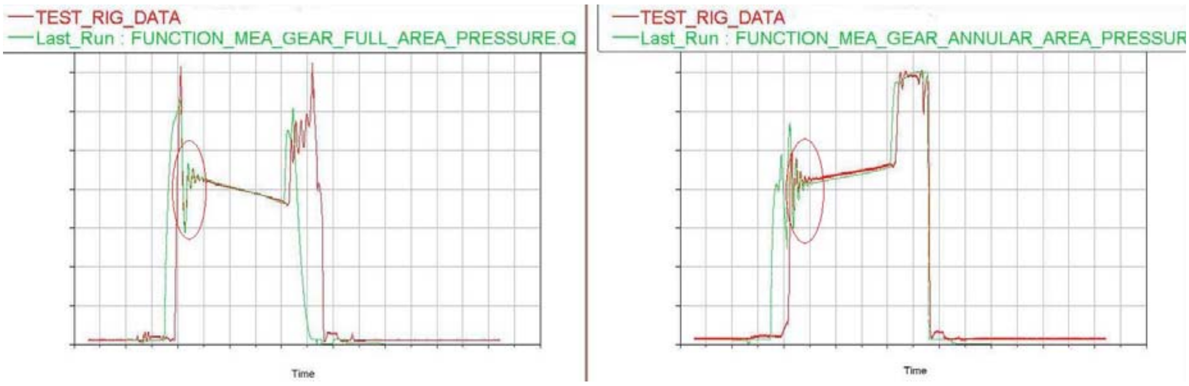


Figure 3. 23 – Tuning of stiffness and damping of gear actuator attachment points

Snubbing flow control devices were then modified, to affect the dynamics of the system in the areas shown in Figure 3. 24.

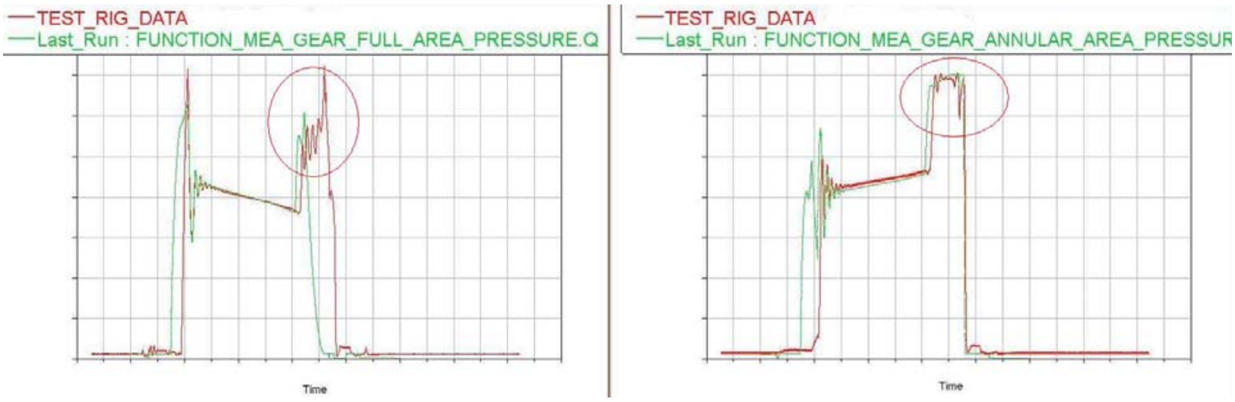


Figure 3. 24 – Tuning of snubbing for gear extension

Finally, the diameter of gear uplock roller was reduced, as for doors. Figure 3. 25 shows the initial travel of the gear achieved after the tuning.

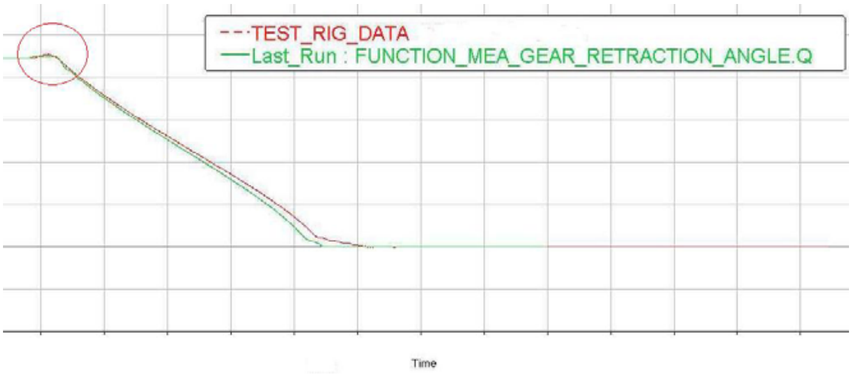


Figure 3. 25 – Tuning of initial angular displacement for gear extension

Gear retraction has been approached accordingly to gear extension. Flow control devices and then snubbing have been tuned, and a modification to stiffness and damping of gear actuator attachment point was required to try to reproduce test rig data oscillations. Stiffness and damping then had to be defined as a function of the retraction angle, in order to have an acceptable behaviour both in extension and retraction phases.

3.3.4 – Door and gear freefall

Friction in actuator seals has a significant impact on door and gear freefall: since no hydraulic power is supplied, friction takes part in determining the extraction velocity of actuators, together with hydraulic resistance. Therefore friction values were tuned in order to adapt door and gear angle behaviours to test rig data for freefall. The results are shown in Figure 3. 26 and Figure 3. 27.

Figure 3. 26 – Tuning of door actuator seal friction

Figure 3. 27 – Tuning of gear actuator seal friction

Seal frictions have a small impact on actuators behaviour in normal operation, so the tuning already performed for normal extension and normal retraction is not significantly affected by the last modifications.

3.4 – Comments about validation results

The tuning of the models that has been performed produced satisfactory results from a point of view of pressures, for all simulation scenarios except gear retraction, where the overall damping of the system is higher than expected. In some cases the behaviours of door and gear retraction angles show some differences with test rig data, even if pressures in actuator chambers match. It is understood that said differences with reference data are mainly due to the assumption of constant pressure source in the hydraulic model. In the reality, supply pressure drops quite significantly when supply flow rate from ATA29 increases. This phenomenon is not included in the model, therefore even if the model was tuned in order to match pressures, the corresponding flow rate through actuator ports has some differences respect to the real system. Consequently, a difference in flow rates leads to a difference in door and gear retraction angles, and can alter the hydraulic damping of the system.

This limitation could be removed by introducing in the model supply P-Q curves, which provide supply pressure at the interface between the two systems as a function of the flow supplied to ATA32.

In terms of modelling, this could be achieved by measuring the flow at the supply line, using that signal as an input to a lookup table containing the supply P-Q curve, and sending the output to a pressure generator block to be connected to the system. A first order lag has to be placed between the flow sensor and the P-Q lookup table to break the implicit loop that would be otherwise generated. Figure 3. 28 shows a possible implementation of a P-Q curve in AMESim for that purpose.

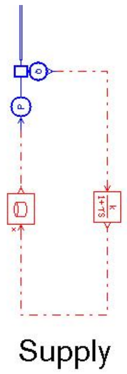


Figure 3. 28 – P-Q curves for supply line

It was not possible to find validated P-Q curves for A380 Nose Landing Gear LGERS, so the assumption of constant supply pressure was necessary.

A lack of accuracy is experienced at very low pressures during freefall, because the simulation tends to be stabilised on a constant value of pressure in the actuator chambers, while in the reality fluctuations are experienced. The phenomenon is related to the accuracy of the submodel that is used for the representation of fluid characteristics at very low pressures. Cavitation is experienced during freefall, and the behaviour of the fluid changes significantly, therefore accurate predictions are very difficult to be obtained. However, the difference in pressure between simulation and test rig, in this case, does not have a significant impact on the overall response of the system.

Another limitation of the model consists in the rigidity of gear uplock attachments to fuselage. In the real system, several tons are applied on the hook, and the surrounding structure presents visible deformations, affecting the initial angular position of the landing gear. When the gear is lifted to alleviate the load on the hook, prior to uplock disengage, said deformation is relieved. This leads to an initial travel of the gear actuator that differs from the prevision of the rigid model. In the latter, in fact, the initial travel of the actuator can be tuned only by reducing the diameter of the gear uplock roller, but the amount of travel that is experienced in reality cannot be achieved, even with a dramatic reduction of roller diameter. This has an impact also on pressure behaviour when the gear is lifted, because flows generated through the actuator ports cannot match the real system.

The need for an elastic attachment point of landing gear retraction actuator to fuselage leads to a big increase of CPU time. When the spring-damper is introduced in the model, the value of the communication interval that is suitable for a fully rigid model has to be further reduced. This is a limitation for the efficiency of the model, but it is necessary to achieve a good accuracy of results, as pressure oscillations would totally disappear with a rigid attachment point.

Chapter 4 – P vs Q Characteristics

4.1 – Overview

Once the co-simulation model was validated, Pressure-Flow characteristics were calculated in some representative points of the hydraulic circuit.

P-Q curves consist in plots where the pressure at a specific point of a hydraulic circuit is represented versus the flow rate measured at the same point. This means that if the flow rate in a point of the circuit is known, the pressure of the fluid can be predicted if the related P-Q curve is supplied.

P-Q curves are calculated in a steady state condition, i.e. when all the transient effects have disappeared from the system's response, and all the variables have stabilised on a constant value. Therefore, during transients such as a sudden valve opening, the response of the system does not match the predictions provided by the P-Q curves.

For the A380 LGERS NLG model, it was necessary to calculate the P-Q curves at the inlet ports of door actuators and gear retraction actuator, between each flow control device and the actuator port to which it is connected. The curves were extracted for four different simulation cases in normal operation, commanding the system valves accordingly: door opening and door closing (with P-Q measurements at door actuator ports), gear extension and gear retraction (with P-Q measurements at gear actuator ports).

In order to extract the P-Q curves, the hydraulic model validated with the co-simulation was modified, as explained in the following paragraph.

4.2 – Method used and modifications to the model

For P-Q curves calculation, the AMESim hydraulic model was separated from the ADAMS model, and the interface block for co-simulation was deleted. In this way the AMESim model can run again in standalone mode.

The actuator supercomponents were replaced with AMESim standard single-rod actuators. The geometry of the actuator does not affect the calculation of the P-Q curves, because what needs to be done for the purpose is to command the valves to the appropriate position, and then force the desired flow rate through the control points of the circuit. So, whatever the actuator used is, it is sufficient to set the valves correctly (for door opening, for example), and then force artificially the actuator rod to move at a defined constant speed. Depending on the actuator piston areas, this will establish a flow rate through the actuator ports, equals to the piston velocity multiplied by the related actuator piston area. By tuning appropriately piston areas and velocity, the desired value of flow rate can be achieved, and therefore a point of the P-Q curve can be determined by measuring the pressure corresponding to the flow rate that is created. By repeating the simulation with different piston velocities, the whole P-Q curve can be calculated by adding other points and then interpolating.

Figure 4. 1 shows the modifications to the actuators that were implemented in the model.

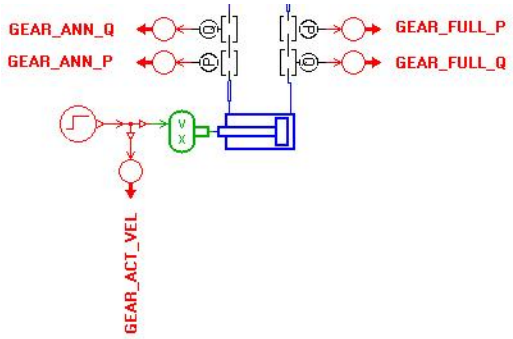


Figure 4. 1 – Modification to actuators for P-Q curve calculation

A step input signal is supplied for actuator velocity and actuator displacement is then calculated by the green block, by integration of the velocity. At each actuator port, pressure and flow are measured and sent to an appropriate block using transmitters and receivers, in order to save the variables into an ASCII results file, together with the value of actuator velocity that is selected. The time interval between two consecutive printouts of the results was set equals to the entire simulation time. In this way, AMESim outputs the results to the ASCII file only twice during the simulation, i.e. at the beginning and the end, avoiding having a large file in which all the intermediate measurements would not be necessary. Figure 4. 2 shows how the selected variables were saved in the ASCII file.

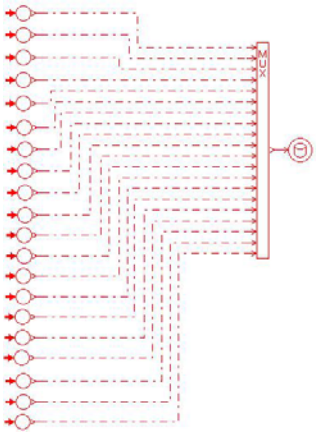


Figure 4. 2 – Collection of variables to be saved in a results file

In order to bring the system into a steady state condition, the stroke of the actuators must be very big, typically set to a few meters, and the simulation time very long, to make sure that all the transients are extinguished. The P-Q pair must be then measured at the end of the long run.

Valve input signals were simplified respect to the co-simulation, because in this case only a constant or step value is necessary to command each valve. Said inputs signals are saved in the ASCII file as well.

A set of batch runs was configured for each simulation scenario: the actuator velocity is varied in each run, in order to automatically obtain all the desired points of the P-Q curve in the ASCII results file.

Conclusions and Future Developments

AMESim demonstrated to be a user-friendly programme, offering a great flexibility in terms of modelling capabilities. The high number of components available in the AMESim library allows the modelling of complex systems through the creation of supercomponents. For each hydraulic component a wide range of submodels is available, therefore the mathematical description of the system can be very accurate, if needed. A lack of accuracy was experienced at very low pressures, where cavitation phenomena occur. However, these differences with reference data do not have a significant impact on the overall behaviour of the system.

The implementation of co-simulation for LGERS modelling and simulation revealed a good potential in terms of the accuracy of results that can be achieved. The introduction of the AMESim/ADAMS co-simulation is an important step forward respect to the use of the ADAMS/Hydraulics toolbox. The capabilities of the latter in terms of hydraulic modelling are more superficial comparing to AMESim, which is a software specifically designed for the purpose. The AMESim/ADAMS co-simulation offers the possibility to make the most of both programmes, putting together the benefits coming from an accurate modelling of the hydraulic behaviour, and reliable description of the dynamics of a mechanical model. On the other hand, the simulation is performed by two separate programmes, which exchange informations only at fixed time steps. Therefore small communication intervals are needed to prevent excessive discontinuities in the exchanged variables, which can lead to vibrations in the dynamics of the system. The CPU time needed to achieve good results can be quite high in some cases.

Future developments of the models described in this work could consist in the introduction of supply P-Q curves, as previously explained. In that case, a new tuning of the model would be probably needed.

Elasticity could be added to the landing gear uplock to fuselage attachment point, to have a better representation of actuator initial travel in gear extension.

The applicability of the model could be extended to extension/retraction in flight, while in the present status of the model only hangar cases were considered. The upgrade could be done by adding aerodynamic loads on doors and gear, in terms of torques applied around the hinge line as a function of door and gear retraction angle, and flight parameters.

The AMESim/ADAMS co-simulation could also be adopted to update LGERS models of A380 Wing Landing Gear and Body Landing Gear extension/retraction systems.

References

Landing Gear Simulation and Modelling LGSIM:Wiki (Airbus internal page)

Airbus A380, Wikipedia, *http://en.wikipedia.org/wiki/Airbus_A380*

A380-800 Landing Gear Extension and Retraction System (LGERS) System Description Document (Airbus UK internal document)

Snubbing Analysis for A320 MLG Door Actuator (Airbus UK internal document)

A380 LGERS Nose Landing Gear Bay Version Description Document (Airbus UK internal document)

Bjoern Kirchhoff, "A study of the effects of aircraft flexibility and shock absorber characteristics during ground manoeuvres by simulation", University of Technology Hamburg-Harburg, Department of Aircraft System Engineering, 2006.

AMESim 7.0b reference manual.