

VERSATILE OFFLINE SIMULATION TOOL FOR SYSTEMS DESIGN

Laurent Binet, Laurent.Binet@onera.fr, ONERA (France)

Abstract

For dual pilot configuration, Active Side-Stick Units (ASSU) technology can provide intuitive tactile cueing in the cockpit. Through ASSU, haptic cueing is expected to be an efficient and intuitive communication mode with the crew, especially enabling to electronically couple two side-sticks.

Considering new generation of active inceptors providing such functionalities, the EFAICTS (Ergonomic impact and new Functions induced by Active Inceptor integration in CockpitS) project started in December 2018. This project has received funding from the Clean Sky 2 Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement N° 820884, in which Safran Electronics & Defense is the Topic Leader and ONERA the project coordinator. The overall EFAICTS concept is to define, develop and validate new ergonomic information and new functionalities provided by ASSU to enhance crew coordination and autopilot understanding through coupling functions between active sticks, and between active sticks and the flight control system, and haptic feedbacks on sticks.

In order to design and ease the integration of these developments in a real time simulator, an offline simulation tool has been set-up, suited to meet the goals of the EFAICTS project, but also enabling the development of any system that could be implemented in the ONERA's PycsHel simulation bench.

The paper focuses on the development of this offline simulation tool and the possibilities it offers. A short description of the upgrades performed on the simulation bench are proposed before detailing the models and functionalities built in the offline loop to simulate the real environment and enabling the design and pre-validation of all features needed in the framework of the EFAICTS project. Some comparisons between offline computations and real time simulations are shown, and the benefits of this offline tool highlighted. Finally, some future expected or already planned developments are described, demonstrating its high versatility and showing how this tool is fitted to system development and integration.

LIST OF ACRONYMS

AC	AirCRAFT mode
ACAH	Attitude Command Attitude Hold
AcVH	Acceleration Command Velocity Hold
AFCS	Automatic Flight Control System
AIS	Active Inceptor System
AP	Autopilot
ASSU	Active Side-Stick Unit
CTR	Civil Tilt Rotor
DGA-EV	French Flight Test Centre
FCS	Flight Control System
FEP	Flight Envelope Protection
FMC	Flight Mechanics Code

GS	Glide Slope mode
HC	HeliCopter mode
HDG	Heading mode
P/B	Push Button
PF	Pilot Flying
PFD	Primary Flight Display
PM	Pilot Monitoring
QF	Linear force gradient of the force/displacement curve applied on the ASSU (QF=1 represents 1N/°)
RCAH	Rate Command Attitude Hold
SAS	Stability Augmentation System
SFD	Secondary Flight Display
TRC	Translational Rate Command
TU	Transition-Up
VS	Vertical Speed mode

Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper and recorded presentations as part of the ERF proceedings or as individual offprints from the proceedings and for inclusion in a freely accessible web-based repository.

LIST OF SYMBOLS

IAS	Indicated Airspeed (knots)
U	Forward speed (knots)
Vz	Vertical speed (ft/min)
P	Roll rate (deg/s)
Q	Pitch rate (deg/s)
R	Yaw rate (deg/s)
ϕ , PHI	Roll Angle (deg)
Θ , TETA	Pitch Angle (deg)
ψ , PSI	Heading (deg)
RADALT	Radiosonde Altitude (ft)

1. INTRODUCTION

For dual pilot configuration, Active Side-Stick Units (ASSU) technology can provide intuitive tactile cueing in the cockpit. Through ASSU, haptic cueing is expected to be an efficient and intuitive communication mode with the crew, especially enabling to electronically couple two side-sticks. Coupling logics and dedicated functions could ensure the pilot monitoring to be aware of the pilot flying control inputs (Ref [1], [2]) but also allow the crew to follow the actions of the Automatic Flight Control System (AFCS).

Considering new generation of active inceptors providing such functionalities, the EFAICTS (Ergonomic impact and new Functions induced by Active Inceptor integration in CockpiTS) project started in December 2018. Within this project, Safran Electronics & Defense is the Topic Leader and ONERA the project coordinator, and more specifically the Research Unit "Rotorcraft Flight Dynamics and System".

The overall EFAICTS concept is to define, develop and validate new ergonomic information and new functionalities provided by active inceptors to enhance crew coordination and autopilot understanding through:

- Coupling functions between active sticks (collective/collective and cyclic/cyclic), and between active sticks and the flight control system;
- Haptic feedbacks on sticks.

The development of the coupling logics and functions will ensure the PM (Pilot Monitoring) to perceive the control inputs of the pilot flying (PF). They will also allow the pilots to follow the actions of the AFCS. Active inceptor technology enables coupling electronically two side-sticks, and emulating a mechanical linkage. Different logics requiring different coupling functions will be developed and tested depending on the control law engaged the current flight mode or the nature of the commands (collective or cyclic sticks).

EFAICTS project proposes a Human-centred design approach where the solutions are matured through an iterative activity combining "experimental" activities on the ONERA simulation bench (PycsHel) with experienced pilots from DGA-EV, with offline modelling and simulations as depicted in Figure 1.

Within the project, haptic feedbacks are developed for different situations and for different purposes: transfer of authority between both pilots and AFCS, AFCS operating mode transition, failure modes or specific aircraft failures.

Depending on the purpose, the selection of the type of haptic feedback and its defining parameters has been done.

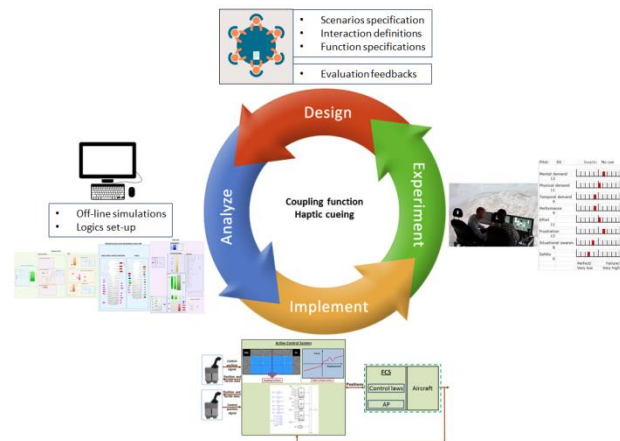


Figure 1: Proposed approach

EFAICTS project focuses on a Civil Tilt-Rotor (CTR) configuration, representing a unique type of aircraft that combines the vertical lift capabilities of helicopters with the speed and range of conventional fixed-wing aircraft. A CTR flight control system must provide adequate control in all three of the flight modes and a smooth blending between them. This requirement renders its design more complex than that of a helicopter, due to the combination of control surfaces and actuators, and naturally leading to the selection of a Fly-By-Wire or Fly-by-Light technology, associated control laws and active inceptors as primary controls.

All these new features are first tuned in an offline simulation environment providing most of the simulator components or functions, such as a tiltrotor flight mechanics model, Flight Control System (FCS) or pilot model. This will enable on one hand, to develop and validate all logics algorithms, and on the other hand, to develop the haptic feedbacks generation modules.

Then, their implementation in ONERA's PycsHel simulation bench will allow evaluations by pilots in a relevant context (in terms of mission task, flight dynamics and systems).

2. ONERA'S PYCSHEL SIMULATION BENCH UPGRADE

ONERA's PycsHel is a prototyping simulation bench dedicated to Vertical Take Off and Landing Systems, located in Salon de Provence. Within the project, it has been upgraded to a dual pilot configuration.

One of the main changes concerned material characteristics with, especially, the integration of two new active side sticks units from Safran Electronics & Defense.

The existent half cave video projection room offering a 270° horizontal field of view plus the ground display was upgraded and PFD (Primary Flight Display) and secondary flight display (SFD)

completely redesigned.

If dedicated models were already included, such as atmospheric conditions, physic engine, vessel, FMS (Flight Management System), specific control laws, AFCS modes and ASSU management system had to be developed.

Safran E&D active side-sticks have been fully installed and implemented into PycsHel. Both mechanical installation and software implementation have been done using a modular approach in order to facilitate the bench reconfiguration. That way, PycsHel can be configured with Safran E&D ASSU or existing ASSU system in every setup (collective or cyclic) as shown in Figure 2. A dedicated mechanical structure has been built in order to provide the ability to switch between a longitudinal and a collective position.

For the completion of the dual cockpit environment, two Brunner rudders with toe brakes have been integrated in the simulator.



Figure 2: PycsHel simulator in dual pilot configuration, integrating Safran E&D ASSU (collectives)

Thus, the upgrade of the bench and its transformation into a dual pilot configuration lead to the integration of:

- 2 new active side-sticks from Safran E&D
- A new rudder pedal system
- New Cockpit visuals and functionalities
- New video projectors and mechanical structure

Most of the simulation performed up to now in the PycsHel environment have been dedicated to conventional rotorcrafts. Therefore, the cockpit layout was adapted to this kind of machine with a “basic” T arrangement flight instrument and some extra instrument dedicated to specific research topics.

In order to endorse the EFAICTS project expectations a new cockpit has been designed specifically for the tiltrotor. The front panel is composed of three tactile displays, duplicated for each pilot and co-pilot. The display in front of the

pilot/co-pilot shows PFD, the display further to the middle showing SFD. As shown in Figure 3 and Figure 4, the PFD, SFD and FMS are not representative of existing on-board systems but all information required to perform elementary missions are present. These displays are built with OpenSceneGraph and can be customized according to the needs.

As shown on Figure 3, all AP modes and flight control laws can be selected on the PFD and specific warning “lights” are displayed for different system failures (AFCS, Engine or ASSU).

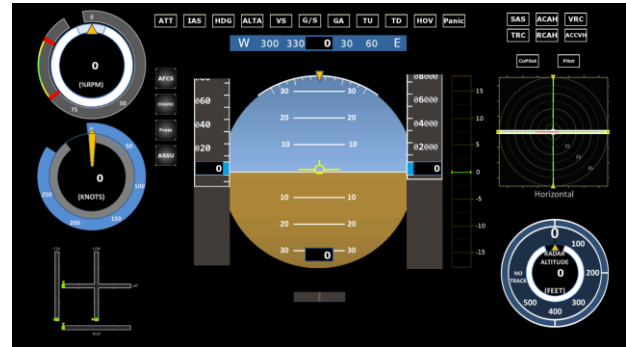


Figure 3: PFD

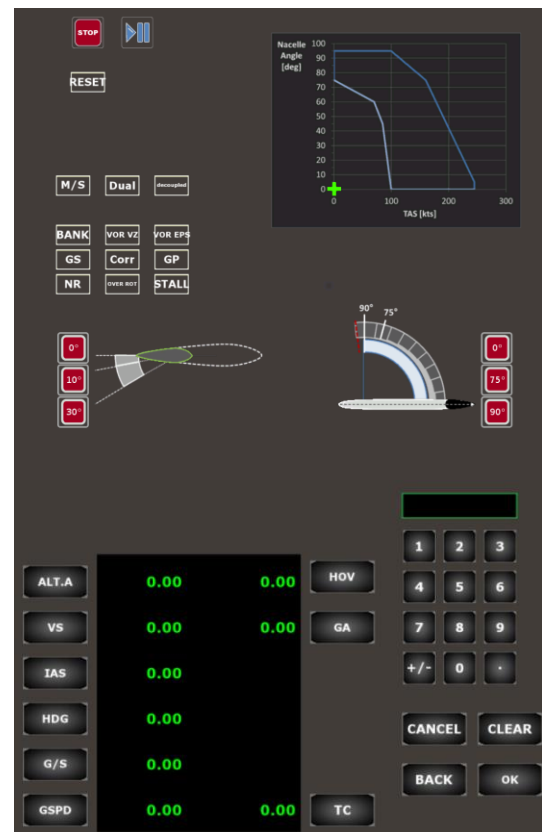


Figure 4: SFD and FMS (lower part)

An indicator light also indicates whose pilot is on command.

All AP modes are duplicated on the SFD and FMS (Figure 4), and target values can be entered through

a panel.

Several Flight Envelope Protection functions can be selected on the SFD, as well as required values for nacelles and flaps angles. The specific Tilt-Rotor transition corridor is also represented.

The coupling modes between sticks are also proposed on the SFD. As explained in the next chapters, Master/Slave, Dual or Decoupled modes can be selected through this panel.

Three 32bits Bitmasks are used to transfer information between PFD/SFD and IAS/FCS modules. These data correspond to the status of AP modes, flight control laws, nacelles and flap angles, failures, coupling modes and FEP functions.

Before EFAICTS project, the PycsHel simulation bench needed ten computers to operate, splitting the different models over a large number of computers. In order to rationalize IT resources and reduce the risk of unavailability, the number of computer has been reduced to six, requiring a big work to reorganize and adapt the different models. They are connected via a 1 GB network for data exchange and a dedicated network for time synchronization using PTP protocol.

Each computer hosts several models, which shares data using a unique distributed partition, managed by a master, insuring the ubiquity of all data stored in the partition.

3. OFF-LINE SIMULATION TOOL

The process proposed by EFAICTS follows a Human-Centred Design approach but the solutions developed (functionalities and ergonomics) are progressively matured through an iterative activity combining “experimental” activities (i.e. real time piloted simulations), with offline modelling and simulations. All features are first implemented on an offline simulation tool, enabling the development of the required algorithms/logics and facilitating their analysis, maturation and integration. Thus, a complete offline simulation loop has been built in MATLAB/SIMULINK, including all models necessary to develop the AIS module (coupling functions, authority transfer logics, haptic functions), as well as the (A)FCS (Control laws, Autopilot modes, etc.). As these modules are transferred on the real time environment, it is necessary to simulate all the components present in the simulator (materials or softwares) such as PFD/FMS to manage pilot/system interactions (e.g. to select or deselect an auto-pilot mode, to change an auto-pilot target), or the aircraft through a Flight Mechanics code.

The goal is not to have a perfect copy of the real time environment, but at least to simulate the functions available in the simulator, and, in order to ease the transfer, to replicate the data exchanges between models, as this is done in the simulator.

3.1. Integrated modules

The modules integrated in the offline tool are those that are used in the simulator. They can be simulator components (PFD, “real inceptor”, etc.) or Simulink blocks that will be then transferred into the real-time loop (AIS, FCS). Integrated modules are described hereafter:

3.1.1. Flight Mechanic Codes (FMC)

A generic Civil Tilt-Rotor has been developed and implemented in FlightLab flight mechanic code.

This includes the description of the overall aircraft structure and elements position, as well as the furniture of the different aerodynamic coefficient tables, mixing and basic control functions. The purpose was not to validate the handling qualities or performances of such an aircraft. Nevertheless, a validation of expected behaviour has been done through trim sweep computations and time function simulation.

This model has been implemented in the PycsHel real time environment as well as on the offline simulation tool thanks to a mex-function in Simulink. As the model integrated in the flight mechanics code FlightLab is the same on both simulation environments, all developments, modifications and upgrades performed on the aircraft model in the offline environment can be immediately transferred and validated in the real-time environment.

3.1.2. PFD-SFD-FMS

In the simulator, flight control laws and autopilot modes can be selected on the PFD. Once selected, autopilot targets are changeable through the FMS. This is also the case in offline tool, in which signal builders are used to specify the engagement/disengagement logics or to assign new target values for autopilot modes.

It is thus possible to assign target values for airspeed, altitude, vertical speed, slope and heading. These targets parameters are then used as inputs in the different autopilot modes sub-blocks.

3.1.3. Atmospheric environment

This block generates atmospheric turbulence and/or wind variations. It is possible to generate constant wind magnitude, azimuth and vertical component or function of time thanks to signal builder blocks. A band-limited white noise can be added to simulate turbulence. North, East and Vertical components of the wind are then computed and can be used as inputs in FlightLab.

3.1.4. Manoeuvre description block

In order to follow parameters or a more complex trajectory, this block provides orders that could be used by autopilot modes (i.e. airspeed) or the pilot

activity model. Up to now, the settable parameters are at a “relative low level” (airspeed, attitude angles, Vertical speed and heading). Depending of the scenario to be tested, more complex trajectory could be integrated. These parameters are used as targets for the pilot model to act on the ASSU model and finally on the control laws. Hence, for a given manoeuvre, it is possible to estimate the stick positions and forces to be applied by the “real” pilot.

3.1.5. Pilot activity module

Most of the time, control inputs are calibrated inputs (steps, ramps, etc.). They can be generated through signal builder blocks to send calibrated control inputs to the control laws and enabling their design and tuning.

However, in order to generate more realistic control inputs such as a pilot would apply in the simulator, a precision model (based on Mc Ruer pilot model) can be operated, where neuromuscular system is represented with configurable transfer functions and delays. In addition, a PID has been added in order to follow the prescribed target on each axis.

While control inputs for FCS are generally inceptor positions (based on the FMC rules, in percent or degrees), it is also possible to define the pilot input controls in terms of force applied on the sticks. This functionality has been employed to design the force control logic in case of ASSU failure (described in the next chapters).

3.1.6. Inceptors

This block emulates the behaviour of a second order system, with prescribed stiffness and damping. It includes the longitudinal and lateral axis for the cyclic stick, and the longitudinal axis for the collective stick.

The static force/displacement curve and associated haptic cueing functions (such as soft stops) have been implemented in the model. This feature, developed for an optimal haptic design methodology approach [3], can provide the positions and forces the pilot would have to apply to realise a piloting task. This can be helpful to estimate, in advance, the level of force the pilot will have to generate.

3.1.7. FCS-AFCS

This module corresponds to the Flight Control System, used in offline tool and that is also transferred to the real time environment.

The FCS model receives 241 inputs from FMC and AIS. 80 outputs from FCS are sent to FMC, PFD/SFD/FMS and AIS modules. Spares are available, allowing additional inputs and outputs to be taken into account.

3.1.7.1. Nacelle control, Flaps

Nacelle angle is controlled by the pilot. Three pre-defined nacelle angles can be selected: 90°, 75°

and 0°. The nacelle tilt rate is set to 4°/s (up to 8°/s in emergency conditions). Flaps deflection is also controlled manually by the pilot. Three settings can be selected: 30°, 10°, and 0°.

As the nacelle and flap angle are selected through buttons on the SFD in the simulator, it is also possible to set these values through a signal builder in the offline tool.

3.1.7.2. Auto-Pilot modes

Several autopilot (AP) modes have been developed and tuned for this CTR model. Most of them were tuned for the entire flight envelope, with the particularity to be engaged in HC or AC configurations.

- **SAS**

SAS mode provides stability and control augmentation on pitch, roll and yaw axes.

- **ATT**

The ATT mode provides the following functions:

- Longitudinal attitude acquisition and hold
- Lateral attitude acquisition and hold
- Heading hold in hover and low speed

- **Indicated Airspeed Acquisition and Hold mode (IAS)**

This mode enables to acquire or maintain of specific indicated airspeed (IAS). In HC configuration, IAS is maintained constant or modified by the autopilot by means of longitudinal cyclic pitch of rotors. In AC configuration, IAS is maintained constant or modified by the autopilot by means of collective thrust of the rotors.

- **Heading Acquisition and Hold mode / Turn coordinate mode (HDG)**

This mode enables to acquire or maintain a specific heading at low speeds. This heading variation is achieved by a direct control on the yaw axis at low speeds, or a bank angle variation associated with turn coordination for high speeds and AC mode.

- **Altitude Acquisition and Hold mode (ALT/ALT.A)**

This mode enables to acquire or maintain of specific altitude. In AC configuration, all vertical modes (ALT/ALT.A, VS, and GS) are managed by means of elevator while managed through collective thrust of the rotors in HC configuration. The vertical speed is limited to ± 1000 ft/min.

- **Vertical Speed Acquisition and Hold mode (VS)**

This mode enables to acquire or maintain of specific vertical speed. The vertical speed is limited to ± 1000 ft/min.

- **Glide Slope Acquisition and Hold mode (GS)**

This mode enables to acquire or maintain of specific glide slope (taking into account the IAS).

- **Transition-Up mode (TU)**

This mode is used during the climb-out phase from hover. TU mode provides a fully automated ascent combining a prescribe rate of climb of 1000ft/min to 200 ft pre-set radio height and acceleration to 80 KIAS. Roll angle is controlled throughout the ascent profile to maintain the heading on selection of the mode. Once these pre-defined altitude and airspeed are reached, the TU mode automatically switches to IAS, HDG, VS modes.

- **Fly-Through capability**

Most modern helicopter flight assistance functions beyond stability augmentation detect pilot actions, and pilot follow up functions are called “fly through” or “override” modes. Upon detecting pilot action, the AFCS interrupts its long-term hold function to momentarily replace it by pilot inputs. In the framework of the project, it was initially not planned to integrate such functionality. However, the use of this functionality by pilots in recent helicopters, and the questions raised by its integration lead the development and implementation of such functionality.

3.1.7.3. Augmented control laws

It was supposed that the next generation of CTR would be equipped with augmented control laws. Thus, five representative FCS response types had to be available in the simulator:

- Attitude Command Attitude Hold (ACAH)
- Rate Command Attitude Hold (RCAH)
- Translational Rate Command (TRC)
- Acceleration Command Velocity Hold (AcVH)
- Vertical Rate Command (VRC)

Previously designed, the ACAH and RCAH laws were adapted and tuned for a CTR configuration while the other laws were specifically developed for the project.

More adapted to low speed, TRC and AcVH were tuned for hover and low speeds (<20kts), in HC configuration. The other laws were tuned for the entire flight envelope, for HC and AC configurations and even the transition between.

The laws are engaged through PFD in the simulator, through the PFD sub-block (signal generators) in the offline tool.

3.1.7.4. AFCS/Machine Failures emulation

Exchanges with pilots showed that, in normal flight situations, the interactions at sticks level between pilots are very low, or even inexistent. The PF is piloting while the PM is managing the mission

(navigation, radio, etc.). The cases where the two pilots are on controls are tensed or critical situations, for which a potential transfer of the authority could be required. For that reasons, the emulation of some failures have been introduced in the FCS module.

It is possible to generate an AFCS failure, resulting in a loss of AP mode(s).

An engine failure has also been introduced, but as a CTR would be a twin-engine machine, this type of failure would be totally managed by the Power-Thrust Management System. Such a complex system was not developed here, so a brief and moderate rotor speed transient only emulates the engine failure. The purpose being only to analyse what kind of haptic feedback could be used to warn the pilot in such a case.

3.1.8. Active Inceptor System (AIS)

This Simulink block embeds all required logics to communicate and manage pilot's ASSU.

The AIS module manages 286 inputs, from FMC, ASSU, FCS and PFD. 462 outputs are generated towards FCS and ASSU models. A large number of these outputs (280) are the parameters needed to define the different haptic feedbacks (vibration, soft stops, detent, QF, etc.) for each ASSU (pilot and co-pilot sticks, cyclics and collectives). As for the FCS module, a large number of spares are available.

The different features managed by the AIS module are described hereafter:

3.1.8.1. ASSU coupling logics

Main objective of the EFAICTS project, the evaluation of coupling functions between sticks requires the capability to select a coupling mode, to switch from one to another if one pilot requires this change, or to automatically manage the coupling mode in case of failure.

Manual selection is performed through a control panel on the SFD (easily reachable by both pilots) and clearly identifiable as shown Figure 5 and through signal builder in the Offline tool.



Figure 5: Selectable coupling mode in the simulator

As the need of decoupling the sticks might be an emergency, it is also possible to switch to this mode through a P/B on the grips.

Then AIS module receives these inputs and sends them back to the ASSU model in the offline tool, to the real ASSUs in the simulator.

The following coupling modes are available:

In **Master/slave** mode, the sticks are linked. The slave follows the master's motions and cannot interact. It is a “one way” interaction, from master to

slave. The master imposes its motions to the slave. Slave feels being linked as with a mechanical linkage with the master stick, with a very high stiffness. Master has no force feedback from the slave stick.

In **Dual** mode, the sticks are linked. This linkage emulates a mechanical linkage between sticks with a very high stiffness. In such a way, sticks have identical motions. There is a mutual interaction between, motions of a stick being felt on the other. It is the classical coupling between mechanically linked controls.

In **Decoupled** mode, the sticks are not linked. They can have different/independent motions and there is no interaction between. It has to be mentioned that the decoupled mode will be available on the simulator but its study is out of the scope of the EFAICTS project. Nevertheless, the possibility to switch to this mode is available.

In addition to these “classical” modes, specific dual modes are currently designed and evaluated, where the force transferred from one stick to the other depends on who is the primary pilot (i.e. pilot on command). The sticks are still linked but the forces exchanged between sticks are potentially not equal. These modes will be evaluated during the EFAICTS project.

3.1.8.2. Buttons management

Each grip offers two pushbuttons, a trigger and a “4-way hat” switch. These different buttons are currently used to activate Trim, Trim Release, Beep Trim functions, transfer of authority between pilot’s sticks and the selection of Decoupled mode. The buttons assignment can be easily changed through modifiable parameters. For another study, the trigger is also used to switch from TRC to ACAH law.

3.1.8.3. Trim management

A force trim release button releases all forces on the stick and pedals when pressed. The controls can be moved around against no stiffness effect, but the point at which the stiffness is restored is the new trim (zero force) point.

By pressing the beep trim switch (standard-shaped four-way switch) the force displacement curve is moved horizontally. For zero grip forces the stick position moves accordingly. The only variable to manage here is the stick displacement speed (i.e. trim speed).

In decoupled mode, beep-trim and trim release functions are independent on both inceptors. A limit in the difference between two inceptor trimmed positions due to both pilots has been implemented and could be analysed.

In dual mode, both inceptor can activate and use trim release function. The trim position remains the same for both inceptors. Logic to manage the beep-

trim function has been set-up, especially in case of dual pilot inputs.

In master/slave mode: only the master inceptor can activate and use trim release function, the trim position set by master being imposed to the slave inceptor.

3.1.8.4. Authority transfer

The transfer of authority from one pilot to the other is done through a P/B or through the selected coupling mode via SFD (only in case of Master/slave mode). If this transfer is requested, the AIS module switches the controls sent to the FCS from one pilot to the other.

3.1.8.5. Haptic functions

Haptic feedbacks can be used for many different applications, bringing information for a better understanding / management of:

- The flight control law engaged
- Authority transfer between pilots
- Authority transfer between FCS and crew
- Flight Envelope Protection functions
- Failure(s) recognition or/and warning

For all these goals, haptic feedback functions have been developed and implemented and will be improved all along the project. Some examples will be given in the next chapters.

3.1.8.6. FEP

Active side-stick technology, through haptic cueing, is very well adapted to provide efficient and understandable Flight Envelope Protection functions to the crew.

The tilt-rotor specificity in this domain is that it combines the FEP needed on Helicopters as well as on fixed-wing aircrafts, with some specific machine limits that have also to be considered.

The following FEP functions have been developed and are currently matured through both offline and real time simulations. Some details are given hereafter.

Vortex Ring State (VRS) protection function

Active in HC mode at low airspeed, this function generates a soft stop on the collective to warn the pilot that the aircraft is descending too fast and is approaching the vortex ring state domain.

Depending on the control law engaged on the vertical axis (VRC or direct control), the function is adapted to provide the corresponding soft stop. The function can use two different parameters, the vertical speed or a VRS proximity criterion developed at ONERA [4].

A vibration on the collective stick can be added to the soft stop cue to better warn the pilot. It is generated when the current collective position is close (a parameter specifies this closeness) and lower than the computed soft stop position.

Corridor/VNE

Active in HC mode, this function generates vibrations, or a soft stop, or an increase of the QF on the longitudinal cyclic to warn the pilot that the aircraft is approaching a maximum predefined forward speed corresponding to the corridor limit.

Bank Angle limitation

Active in HC mode, this function generates soft stops on the lateral cyclic to warn the pilot that the aircraft is approaching a maximum predefined roll angle. Depending on the control law engaged on the lateral axis (SAS, ACAH or RCAH), the function is adapted to provide the corresponding soft stop.

3.1.8.7. ASSU Failures

Safran E&D inceptor internal architecture is based on direct-drive motors, with no gearing, mitigating (and even leading impossible) the possibility of stick jamming. Furthermore, as the system is featuring quadruplex displacement sensors, the probability of a total motors loss is highly improbable. The only impact could be a reduce level of maximum reachable force feedback.

Nevertheless, due to aircraft manufacturer requirements, a back-up mode has been introduced, which could be activated for the worst failure case: In back-up piloting mode, stick is locked in position thanks to brake devices and pilot controls the attitudes of the aircraft by applying force on stick. Force sensors inside the grip collect pilot intentions.

This feature has been implemented on ASSU systems available in PycsHel.

This type of failure and the strategy to keep control have to be further assessed and validated by piloted simulations, to verify if the transients, due to a shift from position to force control (and vice versa) are acceptable. As for the other prescribed failures, the ASSU failure can be generated at prescribed IAS, Altitude or Heading.

3.1.8.8. ASSU Back-drive

In modern helicopters, flight controls are already back-driven by the trim actuators and classical mechanical links, but for example, in Airbus airliners, the side-sticks are not coupled and when upper AP modes are engaged, side-sticks remain fixed to the neutral position. In case of AP disengagement, the pilot has to reach the corresponding stick position manually, potentially leading to transient behaviour of the plane. In the framework of the project, different logics have been developed and implemented to back-drive the ASSU:

1. No back-drive (Corresponding to the current Airbus airliners logic).
2. The ASSU are moved by trim actuators, corresponding to the current situation on

helicopters.

3. The ASSU motions are based on the control law on which the system would come back when the autopilot will be disengaged. This logic should limit the transient during the change phase between autopilot flight and manual flight. Indeed, the stick positions when the autopilot is controlling the aircraft would be almost the same than those applied by the pilot to perform the same piloting task.

4. Other logics could be envisaged and implemented until the end of the project. For example, inceptor back-driven positions could reflect aerodynamic element deflections (ailerons, fin and empennages) in AC mode.

Independently to the logic used to back-drive the sticks, the possibility to only back-drive the "PF" stick, while keeping fixed the PM's stick, is offered and will be analysed.

3.2. Examples of systems/functions developments

As previously outlined, the possibilities offered by this offline tool in the development and study of some systems/functions are numerous and some examples are discussed hereafter.

3.2.1. ASSU coupling logics

In addition to the implementation of the logics and corresponding algorithms needed for ASSU couplings, the offline tool is very well adapted to perform parametric studies, such as the study of the impact of the transfer of authority between pilots when sticks not at the same positions. Indeed, Figure 6 shows two simulations, where a transfer of authority from one stick to the other is performed at $t=6s$.

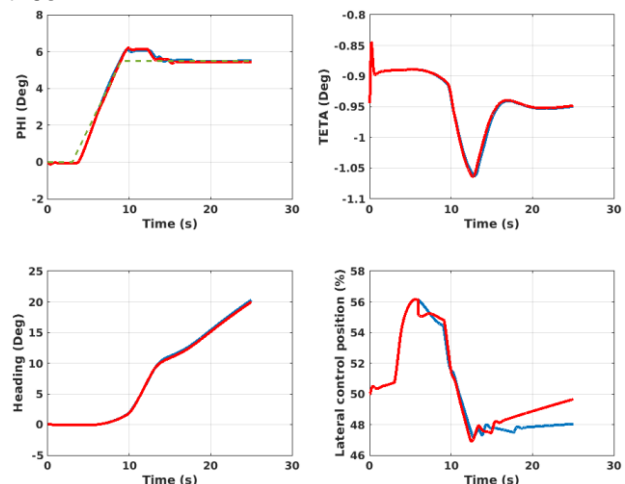


Figure 6: Impact of ASSU position discrepancies at authority transfer

The controls received by the FCS are switched from the pilot stick to the co-pilot stick. Here, the co-pilot ASSU is following the pilot stick position with 1%

discrepancy (i.e. 0.35°) and 0.2s time delay on the lateral axis (see “Lateral control position” plot) . The control law used here is the RCAH. Thus, a sudden variation on the lateral command results in a sudden variation on the roll rate command P. The discrepancy is relatively small, and the pilot model handles (same model for both pilot and co-pilot) the impact on the roll rate to follow the prescribed roll angle PHI.

Although ASSU position are supposed to be identical when they are coupled, this type of analysis can provide system requirements for stick positions or highlight the potential issues of authority transfer.

The use of this functionality can also help the development of adequate solutions such as the integration of fading functions, dedicated to smooth the command transfer.

3.2.2. ASSU-AFCS back-drive logics

As mentioned previously, different logics have been developed and implemented to back-drive the ASSU.

In the following Figure 7, ASSU positions are based on the flight control law on which the system/crew would come back when the autopilot will be disengaged. A take-off is performed with TU mode engaged. On the upper part of Figure 7, the flight parameters during the manoeuvre are shown, while the ASSU position are presented in the lower part. The manoeuvre is performed with three different “back-up” laws. If the flight parameters are always the same (resulting of the autopilot controls), the ASSU positions are dependent on the underlying law.

The first lower column shows the ASSU position (Longitudinal, Lateral, collective and pedal positions) when based on trim actuator positions.

In ACAH, ASSU positions generate aircraft attitude targets to be reached/followed by the law. Therefore, it is possible to inverse current aircraft attitudes to ASSU positions.

In RCAH, ASSU positions generate aircraft angular speed targets to be reached/followed by the law. Thus, ASSU positions are reflecting current aircraft angular speeds as outlined on the upper plots.

In ACAH or RCAH, the VRC law is engaged. As the collective directly controls the vertical speed, the collective position is then based on the current aircraft climb rate.

As there is no action on the roll axis, the position of the lateral cyclic remains almost constant. However, on the longitudinal axis, as the aircraft is accelerating, this figure clearly shows the impact of the control law on ASSU positions. When based on trim actuator, the longitudinal cyclic is pushed forward as the speed is increasing (longitudinal static stability).

In ACAH, as it is based on pitch angle, the stick moves longer than in RCAH where it is dependent on the pitch rate. Hence, in ACAH, the stick will be moved as long as the pitch angle is changed from its reference, while in RCAH, the stick positions reflect the pitch rate variations that are null once stabilized manoeuvre.

Simulator trials will be performed to analyse how the pilots understand these different back-drive logics. As they are used to the current stick motions (trim actuators), the logic based on the control laws could be less well understood/accepted. However, imposing an ASSU position different from the one corresponding to the control law would require the implementation of fading functions, in order to smooth the transition between autopilot and law controls.

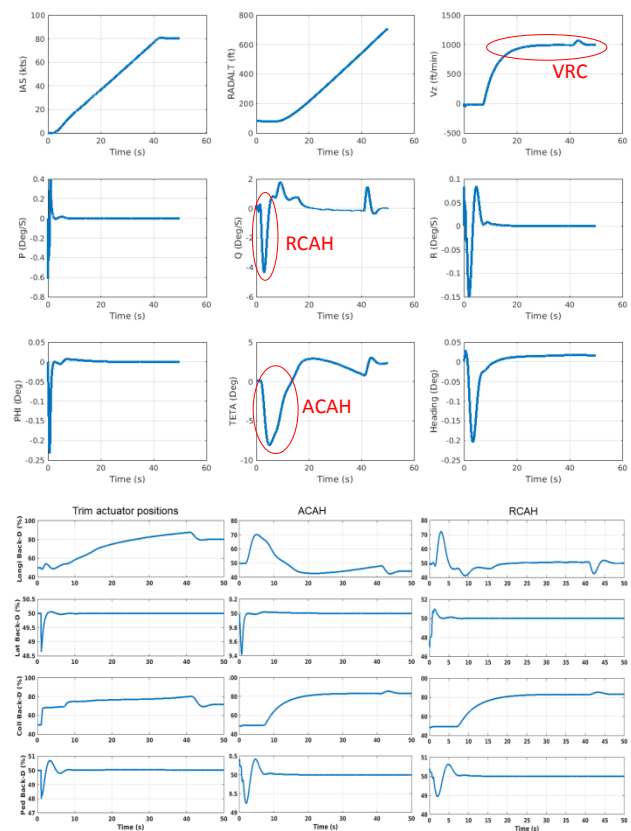


Figure 7: ASSU position during AP backdrive, depending on back-up control law

3.2.3. Haptic function design

- The flight control law used

In addition to possible adjustments of the basic force-feel characteristics of the sticks depending on pilot desires, some specific features have been introduced. Thus, depending on the flight control law engaged, different haptic feedbacks are proposed and can be generated to increase the use and understanding of the laws. For example, when using the VRC law, a detent placed at the neutral

inceptor position provides a clear inceptor reference for level flight. Pilots considered this feature very efficient, comprehensible and useful. Additional detents, providing other vertical speeds could be envisaged. These kind of haptic feedbacks have been also introduced for other flight control laws, and will be soon assessed.

- Authority transfer between pilots

Figure 8 shows a feature dedicated to authority transfer between pilots that can be designed and pre-evaluated through offline simulation.

The co-pilot presses the dedicated P/B to get the authority and to become the primary pilot. The coupling mode is Master/slave. The co-pilot becomes Master at 2s while the pilot becomes slave.

This action leads to an inversion of the master and slave, and consequently to the modification of the Force/displacement curve parameters applied on both pilot and co-pilot sticks. When the stick is "Master", the QF is set to 1, while it is set to 12 when the stick is in "slave" mode. It means that when in slave mode; the force gradient applied on the stick is very high, preventing the stick to be moved. The need of imposing a high value for the QF is only needed with previous ONERA's system, as the ASSU developed by Safran E&D is integrating this feature by just changing the coupling mode.

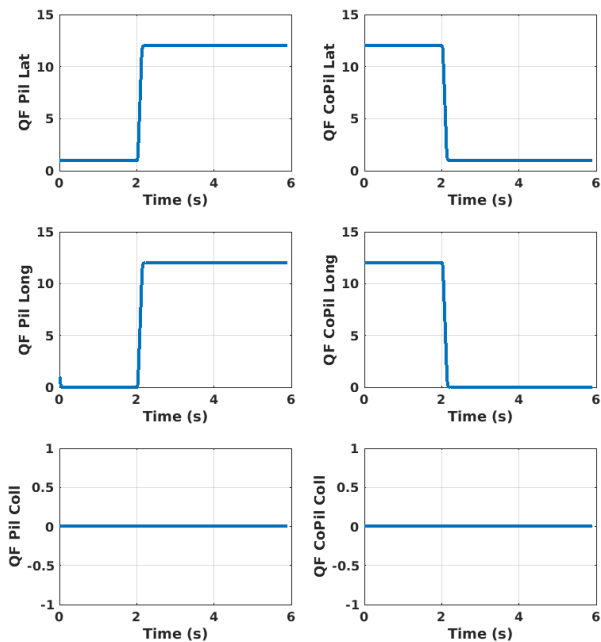


Figure 8: QF modification at authority transfer (Master ↔ Slave)

A specific haptic feedback, consisting in a vibration, warns the pilots that an authority transfer is done. This feedback is activated when the P/B is pressed. Here, one vibration pulse of 1,5s is generated. The

vibration amplitude is set to 4N with a constant frequency of 50Hz.

Of course, the offline tool is perfectly adapted to design and validate this kind of function before real time integration, but some trials on the simulator are still necessary to find the best values for these parameters.

- Authority transfer between FCS and crew, Failure(s) recognition or/and warning

Other vibrations are generated to inform both pilots that a transfer of authority between them and AFCS is performed, or for some types of failures (AFCS or Engine). Depending on the purpose, these vibrations are generated on cyclics or collectives sticks.

For all potential vibrations, the parameters that can be adapted are:

- Vibration Duration
- Vibration Amplitude
 - o Starting amplitude
 - o Ending amplitude
- Vibration Frequency
 - o Starting Frequency
 - o Ending Frequency
- Definition of pulses:
 - o Time between Pulses
 - o Number of Pulses

These parameters will have now to be specified through piloted simulations, one of the questions being: is it necessary to have different types of vibration for each purpose, or is it useless, or even confusing?

3.2.4. Failure emulation and analysis

As mentioned before, the capability to generate failures on different systems is mandatory for the EFAICTS project.

AFCS Failure

The following Figure 9 shows the failure of the IAS mode. Figure 10 shows the pilot's longitudinal cyclic positions.

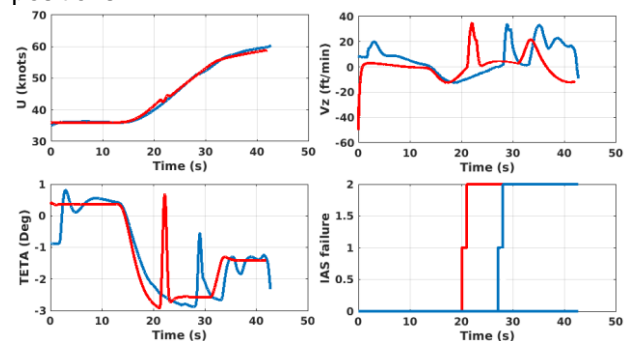


Figure 9: IAS mode failure

At $t=1.6s$, IAS mode is engaged. At $t=12.84s$, an IAS target of 60 kts is entered through the FMS. It can be seen a comparison between real time simulation results in blue, and results from offline computations in red. As the initial conditions are not exactly the same, and in order to have the mostly the same cyclic position at failure in both simulations, the occurrence of the failure is not prescribed at the same time in offline simulation.

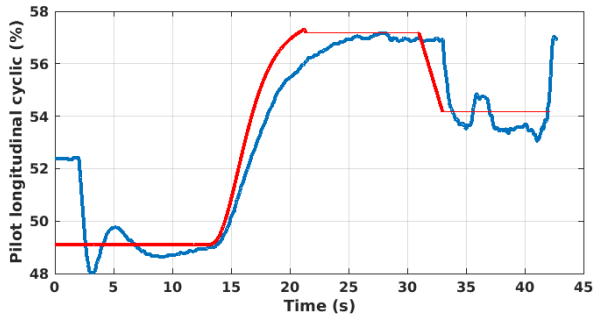


Figure 10: Control during IAS mode failure

The failure is prescribed to happen at IAS=50 kts ($t=28.1s$), with 1s delay in real time simulator, while it has been generated at IAS=42 kts ($t=21.1s$) in the offline tool.

The parameter "IAS failure" takes two values: 1 corresponds to a warning light/initiation of the failure – 2 corresponds to the failure "activation".

The longitudinal cyclic is back driven by the AFCS when the IAS mode is engaged. At the failure occurrence, the stick is released to its current position. In the real time simulation, the pilot slightly pulls the stick. In order to reproduce it, a comparable action is performed in the offline computation at $t=31s$.

Figure 9 shows that the results between both simulations are very similar, only shifted due to the difference of failure time. This demonstrates the capability and benefits of using the offline tool to be able to generate failures, to develop and implement back-drive logics, to analyse transfer of authority between AFCS and crew and the associated potential transients when the pilot gets back the controls after AFCS failure.

ASSU Failure

In order to emulate the ASSU lock in case of failure (feature not available on ONERA's systems), the QF is set to 30 on both lateral and longitudinal axes. In the simulator, pilot actions are then collected by force sensors inside the grip, switching from position based logic to force control logic.

In the following figures, a level flight acceleration is performed in RCAH and ACAH. A comparison is done between offline (red lines) and real time simulations (blue lines), both involving an ASSU failure of the pilot's cyclic ASSU, prescribed at IAS=20 kts. Once the ASSU is failed, the AIS

module transforms the forces into positions and send them to FCS. During the EFACTS project, different formulation to transform forces to positions will be investigated.

In Figure 11, RCAH law is engaged. As controlling the pitch rate with the longitudinal cyclic, the pilot releases the stick to its neutral position to maintain the pitch angle and let the CTR accelerate. At the ASSU failure ($t=14,6s$), the force applied on the ASSU is null. Once transformed into position, this corresponds to the neutral position (which is the current position of the stick) and corresponding to a pitch rate target of 0. In this case, there is no transient and the FCS maintains the pitch angle. At $t=24,9s$, the pilot pulls the ASSU back (by applying a force) which generates a pitch-up command. When releasing the ASSU at $t=29.7s$, the force being null again, the pitch is then maintain to its new value.

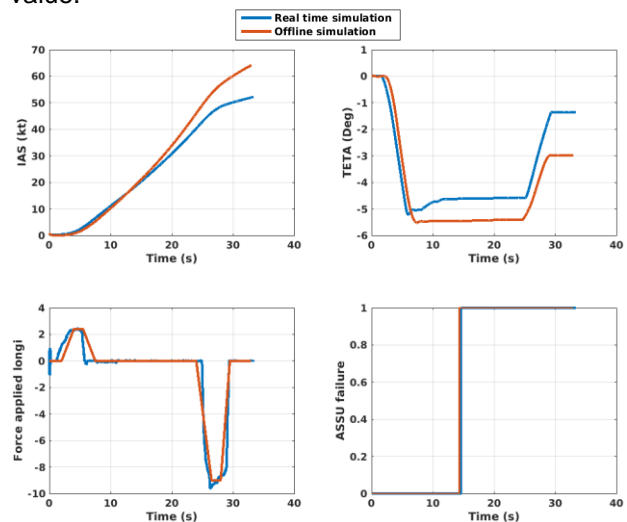


Figure 11: ASSU failure in RCAH law

It has been tried to reproduce in the simulator the actions prescribed through signal builders in the offline tool, but the main purpose here is to show the capability to generate the manoeuvre in offline simulations, the ASSU failure and the possible assessment of the logics used for force control logic.

In Figure 12, the ACAH is used. As directly controlling the pitch angle with the longitudinal cyclic, the pilot has to maintain the stick to maintain the pitch angle (if he/she does not trim the stick).

Thus, the pilot has to apply a force based on the force/displacement curve assigned to the ASSU. At ASSU failure, the pilot increases the force during 1,5s and releases the stick. Once no more force is applied, the targeted pitch angle is zero, leading to a pitch-up motion, up to $\theta=0^\circ$.

Once again, the offline and real time simulations provide almost same results/behaviours.

In the offline tool, the pilot model is based on positions or forces applied on the sticks. It is

possible to switch from one logic to the other depending on the occurrence of an ASSU failure, or at a prescribe time or event.

While the logic to switch from position to force control when the ASSU fails is the same in both cases, these examples highlight the impact of the law and the possible developments to reduce transients.

They also demonstrate the interest of using the offline tool to develop and test force control logics or algorithms. Besides, the capability to generate failure on ASSUs is a very important feature to study its impact on the coupling between pilot sticks, and more especially to analyse if an automatic transfer of authority from the failed ASSU to an operative one is mandatory or not.

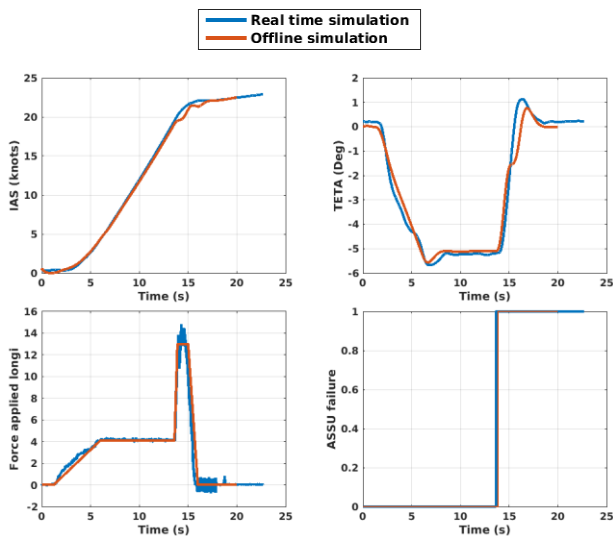


Figure 12: ASSU failure in ACAH law

It has to be mentioned that, for dual pilot configuration, this force control strategy is not supposed to be maintained and used for the remaining time of flight and that a transfer of authority to the other pilot is supposed to be done rather quickly after the failure detection. Nonetheless, for single pilot operations, this strategy would be necessary to enable a quick emergency landing.

3.3. General features and possibility offered

The AIS and FCS Simulink blocks are extracted from the offline environment and Real-Time Workshop® is then used to generate a C++ code from these models. The generated source codes are then adapted to our simulator real time core by an “in-house” software. The entire process duration is around 20 minutes, from their extraction from the offline environment, to their implementation and possible use in the simulator.

All the data used in the models (law gains, prescribed values for failure, options, etc.) are defined in a “xml” file at the simulation start, but can

be modified during the simulation, enabling a direct and instantaneous tuning. 570 parameters are currently used in AIS and FCS blocks that can be changed in real-time during a piloted simulation. These parameters are values used in PID gains for example, or options to switch between different algorithms.

4. BENEFITS AND OTHER FEATURES

4.1. Offline simulation tool benefits

Up to EFAICTS project, any study in PycsHel bench required a specific real time simulation loop. Depending on the studies for which they were designed, they integrated different models and/or architectures, making these loops difficult to backup and sustain. In addition, several days were needed to built-up a loop, as well as their modification.

One of the major benefits is to have now a common architecture for real time simulations, incorporating most of the main necessary modules for any study and enabling the realisation of simulation environment backups.

Another great benefit consists in the large reduction of the time needed to generate and implement a modification in the simulator. Indeed, any development performed in the AIS or FCS modules can be tested in less than 20 minutes, time required to extract these modules from the offline Simulink model, to generate a C++ code and to integrate it in the real time simulation loop.

As shown in the paper, this environment is very modular in terms of models and functionalities. It can be used to develop and upgrade systems, to add new features in these systems, or to perform parametric studies.

All modules provide many options and modifiable parameters, enabling real time tuning in the simulator, or the possibility to switch from one option to another. This large number of options allows selecting and defining failures, different algorithms for a same function or different coupling logics.

More than a simulation loop where calibrated inputs can be generated to design and tune flight controls (control laws or AP modes), this offline tool gives the possibility to develop, integrate and analyse the required modes and commands and to mimic Human Machine Interfaces (HMI) necessary to manage the FCS, or to handle the interactions which have to be taken into account by the AIS and active side-sticks. The data transferred through the simulator HMI (cockpit screens, buttons on grips, etc.) are the same and thus can be applied, in advance, through offline simulations such as button activations.

It is also possible to use data from real time simulation as inputs in the offline tool. “Real” pilot

controls can be treated in offline simulation as shown in Figure 13, where the recorded pilot control positions from a real time simulation are used as inputs in the offline simulation. One can see some very slight discrepancies, mainly due to the difference in the initial conditions, but the actions on the stick provide almost identical effects on pitch and roll angles (ACAH law engaged). A brief variation can be noticed on the vertical speed in offline simulation, due to a transient behaviour of the FlightLab code at the simulation start, in relation to an inconsistency at the simulation start ($t=0$) between FlightLab and Simulink. While not being a big issue in the process, a solution is investigated to resolve it.

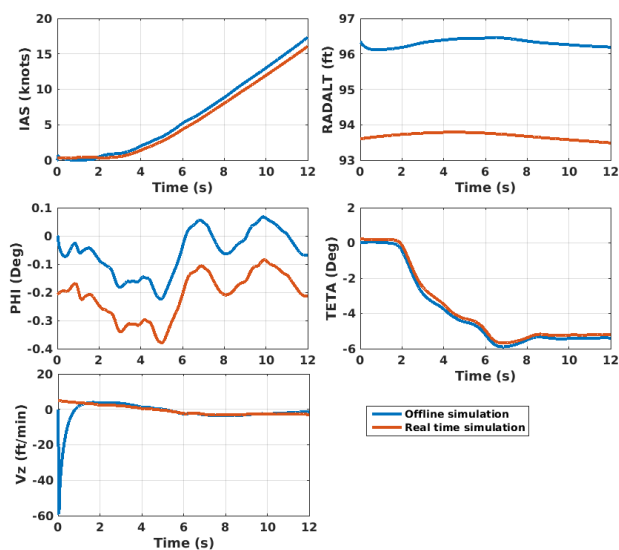


Figure 13: Real time results used as inputs in offline simulation: pilot controls

Finally yet importantly, during the different lockdown periods we faced in the last months, this tool offered the capability to perform developments through teleworking, by connecting to ONERA's computational servers. Of course, the final integration in the simulator was not possible, but a lot of development and associated debugging were done through offline computations.

4.1.1. "Extra" features developed and integrated

In the framework of a cooperation between US Army and DGA (French Defence procurement agency), ONERA is involved in a task dealing with the autorotation management of UAV and OPV helicopters [5], [6]. This study is using an OH-6A helicopter model in FlightLab, for which specific piloting aid functions have been developed and integrated in the FCS module. For example, a direct control of the rotor speed through the collective and an automatic engagement of the IAS autopilot mode

at the engine failure have been developed and implemented on both offline and real time environments. In addition, a specific controller capable of performing an automatic autorotation manoeuvre is currently under integration.

A specific feature of the pilot activity model was not detailed in this paper. Presented in previous publication [3], the main idea is to provide to a pilot model the capability to take into account force feedbacks (such as soft stops) during the realization of a manoeuvre. The definition of criteria then enables the tuning of haptic function parameters through an optimization process. For that purpose, optimization algorithms were coupled to the offline tool enabling the realisation of batch computations.

Finally, a Joint Team between DLR and ONERA is studying the benefits of using augmented control laws for ship deck landing operations. In this project, the HOST flight mechanic code (from AirbusHelicopters) simulating a H225 helicopter models is used on both simulation environments. HOST has been also embedded as a mex-function and coupled with Simulink in the offline tool. The previously mentioned control laws (ACHA, RCAH, TRC, AcVH and VRC.) were tuned for this model and specific new flight control laws developed, such as a relative TRC where the reference velocity is the ship speed.

5. NEXT STEPS AND CONTINUING IMPROVEMENTS

Discrepancies in the AIS or FCS "behaviour" between offline and real time simulations still exist. Some differences at VS mode engagement can be observed or during the use of the HDG mode. During the successive integrations of AIS or FCS modules, some coding errors in the C++ generated code were sometime noticed. If issues in the C++ translation remain possible, other explanations have to be further investigated.

While a relatively large number of auto-pilot modes have been already integrated, it is expected to continue to develop other upper modes which can be found on last generation of rotorcraft such as "Transition Down", "HOVER", "Go Around" or "Wings Level" (WLVL) mode.

EFAICTS project requires the implementation of many haptic feedbacks, dealing with different goals such as Flight Envelope Protections, system situational awareness (failures, authority transfer, flight control law switch, etc.). In order to generate understandable haptic feedbacks, it is necessary to prioritize their generation on the inceptors. This work has been initiated and preliminary logics have been integrated. Thus, haptic feedbacks used for flight envelope protection or failure warning have priority over those used for normal use of the

system (e.g. authority transfer). Nevertheless, the increase of haptic feedbacks potentially used at the same time would require further work on this prioritization.

If it is possible to switch from different control laws, the transition robustness has not been finalised and some improvements are still required to switch from one control law to another (transient behaviour, stick positions, etc.).

PycsHel simulator is a very well adapted device to study the new urban air mobility concepts, future VTOL configurations or remotely controlled vectors. Corresponding flight mechanic models, Flight Control Systems and all the systems necessary to fly these machines could be developed through this offline tool and transferred on the simulator. Hence, it is planned to implement an "in house" flight mechanic model of a new VTOL configuration in these environments, to be prepared for future projects in this domain.

For that goal, a navigation system would be certainly mandatory. A reflexion is ongoing for such integration. In the same way, an AFCS system featuring realistic architecture (e.g. with two different channels, modules such as Air Data Computer (ADC) or Attitude and Heading Reference System (AHRS), etc.) could be developed. If not necessary for our current studies, this might be interesting to develop such architecture, closer to flightworthy systems and in which realistic failures could be analysed.

6. CONCLUSIONS

A versatile offline simulation tool has been developed enabling system design and direct integration into ONERA's simulation bench.

Most of the Simulink models constituting this offline tool are dedicated to replicate simulator components and functionalities, but (A)FCS and AIS modules can be extracted from this tool and directly implemented in the real time simulator.

This offline loop now plays a key role in ONERA's simulation bench and the implementation of systems in this device. It reduces the number of iterations, in reducing the extensive use of pilots' feedbacks and, consequently, reducing the cost of the approach.

Hence, in the framework of EFAICTS project, all the developments are achieved through offline simulation tool, enabling:

- The development of the logics of the different coupling functions between sticks, and between sticks and AFCS with the analysis of the impact of the control law engaged, the current flight mode and the control axis;
- The logics to employ during the transfer of authority;
- The development of fading functions required in the case of different sticks positions during transfer

of authority phases (the validation that there is no consequence to the flight being mainly possible through piloted simulations);

- Haptic feedback developments in different situations and for different purposes: transfer of authority between both pilots and AFCS, AFCS operating mode transition, ASSU failure modes or specific aircraft failures and Flight Envelope Protection functions.

In the future, it is expected to integrate all the required systems thanks to this tool, by developing and validating them first in offline simulations, and implementing them in PycsHel via the dedicated modules.

7. ACKNOWLEDGEMENTS

This project has received funding from the Clean Sky 2 Joint Undertaking under the European Union's Horizon 2020 research and innovation programme under grant agreement No 820884.

The JU receives support from the European Union's Horizon 2020 research and innovation programme and the Clean Sky 2 JU members other than the Union.

This publication reflects only the author's view and the JU is not responsible for any use that may be made of the information it contains.

8. REFERENCES

- [1] Uehara, A. F. and Niedermeier, D., "Limited Evaluation of the Influence of Coupled Sidesticks on the Pilot Monitoring's Awareness during Stall in Cruise," Proceedings of the AIAA Modeling and Simulation Technologies Conference, Vol. 1, 2013.
- [2] Sampaio, R.S., "Electronic coupled active sidesticks in dual pilot helicopters for instructional flights" Proceedings of the AIAA Modeling and Simulation Technologies Conference, Vol. 1, 2013.
- [3] Gemma Prieto-Aguilar G., Binet L., Rakotomamonjy T., "Design Methodology of Force Feedback Laws through Helicopter Control Loop Simulation", 45th European Rotorcraft Forum, Warsaw, Poland, 17-20 September 2019.
- [4] A. Taghizad, J. Jimenez, L. Binet, D. Heuzé, "Experimental and Theoretical Investigations to develop a model of rotor aerodynamics adapted to steep descents", 58th AHS American Helicopter Forum, Montréal, Canada, 2002
- [5] Binet L., Brackbill C., Quinn D., "Simulation tools for UAV/OPV autorotation performance metrics evaluation". 44th European Rotorcraft Forum, Delft, The Netherlands, 19-20 September 2018.
- [6] Binet L., Martin J.N., Brackbill C., "Autorotation manoeuvre analysis of main rotor and aircraft flight from engine failure to ground contact", 42nd European Rotorcraft Forum, Lille, 2016.