M. Daniele, G. Quaranta, P. Masarati, A. Zanoni
Pilot in the Loop Simulation of Helicopter-Ship Operations Using Virtual Reality
Aerotecnica, Missili e Spazio, In press - Published online 02/04/2020
doi:10.1007/s42496-020-00037-3

Pilot in the loop simulation of helicopter-ship operations using Virtual Reality [*]

M. Daniele, G. Quaranta, P. Masarati, A. Zanoni
Politecnico di Milano,
Dipartimento di Scienze e Tecnologie Aerospaziali, Milano, Italy

April 2, 2020

Abstract

Since the beginning of the history of flight, simulation has gained increasing importance in all the life stages of a rotorcraft project. Modern flight simulation systems can be used, among other activities, to study the possible occurrence of adverse rotorcraft pilot couplings originating from the interaction between the pilot (who participates to the human-machine system with the dynamics of her/his body), automatic control systems and rotorcraft dynamics, to replace or complement real flight testing for what concerns machine certification requirements in those situations where testing itself presents high risks of damage for people and things and as powerful conceptual design tools to explore both conventional and non conventional configurations. The work carried out describes how it is possible to build a real-time flight simulator designed to investigate these research fields, that is based on open-source software, cost effective with respect to the solutions actually available in the aerospace industry and which uses virtual reality to give the pilot full involvement in the testing environment. The experimental campaigns carried out have demonstrated the fulfillment of all the requirements, especially concerning real time performances of the system.

1 Introduction

Since the beginning of aviation history, simulation practice has been steadily increasing its relative importance with respect to air operations. Hand in hand with computer technology, flight simulators have evolved into articulated, yet highly flexible systems. This is due to the undisputed convenience for manufacturers and researchers to have instruments with which to study the aircraft behavior in particular conditions without being forced to fly a real machine. Over the years, dedicated software have been developed both for commercial and leisure application; often, products initially conceived for the latter have been demonstrated to be powerful instruments that eventually ended to be used by professionals. The diffusion of the open source philosophy has led to the

---

[*] Based on [28], best paper award at the XXV Congresso Nazionale AIDAA, Settembre 2019 Roma, Italia

development of fully customizable content. Today's flight simulators allow the user to set up ad hoc mission scenarios and even work as visualization tools for external flight dynamics models created with the instruments one prefers, whether they are proprietary or free. In particular, the use of Virtual Reality (VR) headsets are subject of increasing interest in flight simulation research, for their potential in offering access to highly immersive environments in a cost effective way. This paper describes a successful experience in developing a VR-based platform for real time helicopter flight simulation based on first-principles approach offered by the free multibody dynamics software MBDyn [1, 10] (source code repository accessible at [11]). The objective of this paper is to demonstrate that a helicopter flight simulator largely based on open-source and free software can be designed to be cost effective and versatile. The work is intended to further improve the FRAME-Sim (Fixed and Rotary-wing Aircraft Multidisciplinary Engineering Simulator [2]) project of FRAME-Lab at the Aerospace Science and Technology Department, Politecnico di Milano, aimed at creating a facility suitable to meet the requirements set by aviation authorities to qualify as certified virtual training and research platform. The experiments carried out to test the simulator consist in approach and landing maneuvers of a medium-weight helicopter on the moving deck of a ship. This mission has been selected since it represents a typical case in which the enhanced field of view, especially towards the lower elevation angles, and the added *immersiveness* provided by virtual reality can offer significant advantages with respect to conventional, projector-screen simulator setups.

1.1 Virtual Reality approach

The most widespread solution for flight simulation is collimated screen projection. This is also the case of FRAME-Sim [Fig. 1]. Instruments panels are located on a hardware cockpit in front of the pilot, while the environment is projected on a spherical screen. The screen spans 240° in azimuth, and +25°/-70° in elevation. The virtual reality approach was driven from informal inputs given by test pilots involved in conventional simulations [27] who frequently took issue on the lack of visual cues during simulation, especially concerning deck landings and all the operations where the possibility to maintain the view with the portion of space under the rotorcraft is a must. Nevertheless, an experimental campaign is envisioned at DAER to compare VR with conventional visual cue methods. With respect to screen projection, VR headsets allows full immersion in the virtual environment. VR headset are able to make up for some of the shortcomings of collimated screens: they are much less expensive, lightweight and, most important, allow the wearer a 360° field of view without distortion, thanks to the fact that the point of view can be changed by turning the head; however, they present some disadvantages too. Above all there is the so-called ”motion sickness”: this phenomenon happens when human brain visually perceives motion in discordance with the vestibular system; symptoms are nausea and disorientation. A second disadvantage is represented by the limited resolution offered by the displays, further problematic since they are situated very close to the pilot eyes focal planes. This limitation is particularly felt on the visualization of flight instruments. Furthermore, VR headsets do not make provisions, in most cases, for the visualization in the projected environments of the pilot body and in particular of the pilot's arms and hands. The VR headset chosen for this project is the HTC Vive.

All the simulations presented in this paper have been carried out using a desktop control system composed of a classic configuration for helicopters: collective and cyclic sticks for main rotor and pedals for tail rotor control. As soon as FRAME-Sim is completed,

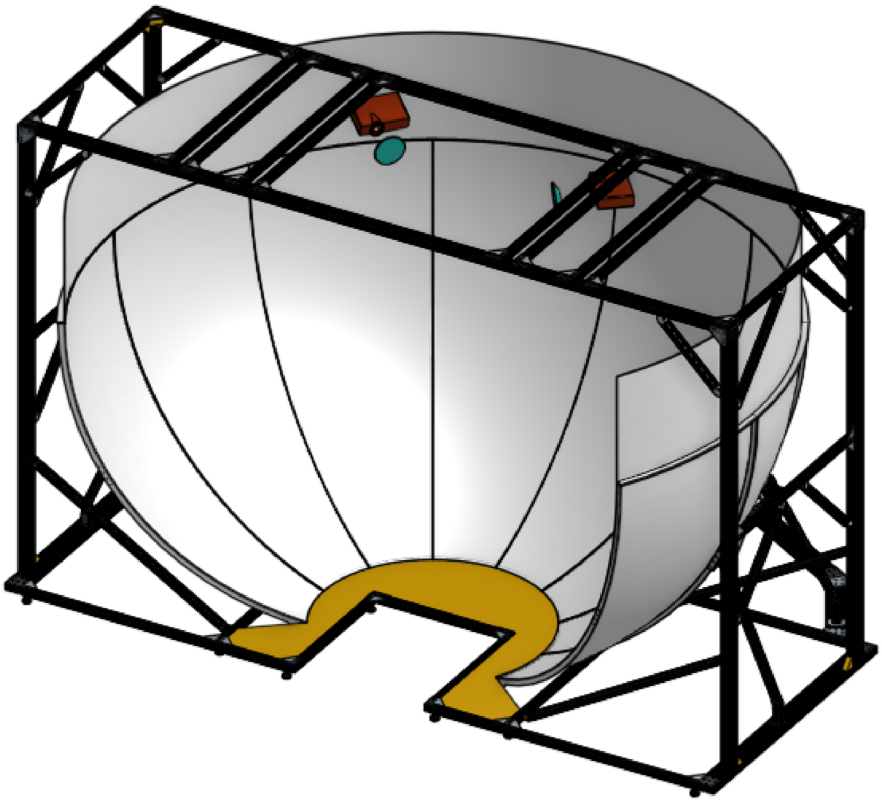

Figure 1: FRAME-Sim projection system.

the same experiments will carried out by using a complete cockpit environment.

2 Simulation test case

The task that has been selected to test VR simulation consists in an approach and landing maneuver of a medium-size multirole helicopter, similar to a Leonardo Helicopters AW101, on the moving deck of a European Multi-purpose Frigate (FREMM, FRegata Europea Multi-Missione). The scenario and the Mission Task Elements (MTEs) have been selected both to provide a challenging scenario for the different components of the flight simulation framework, stimulating its development, and for the reason it represents an ideal test case for VR environment: visual cues are limited and accurate cueing is paramount to the mission success. This type of operation contains some of the critical aspects:

- ship deck landing requires high levels of concentration and experience;
- the moving deck induces the pilot to a series of corrections, mainly to the collective control, which, coupled with the dynamics of the rotorcraft and the Auto-

matic Flight Control System (AFCS), may induce to RPC events that are potentially catastrophic;

- this type of trials performed in flight tests requires a very demanding effort in terms of infrastructures, materials, vehicles, and man-power;
- it very difficult to achieve repeatability to properly study the occurrence of some important events: for example, a flight configuration leading to pilot-rotorcraft interactions both in terms of Pilot Induced Oscillations (PIO) and Pilot Assisted Oscillations (PAO).

Performing simulator tests allows to:

- train the pilot;
- the study of the events leading to situations in which the pilot increases its control action to respond to high gain task requirements, as happens during RPC events;
- to have costs that are much lower than the the ones of the real scenario, both economic and in terms of employed assets;
- the full repeatability of all the phases of the mission;
- to work in perfect safety for people and things.

The simulation requires the interaction between the helicopter and the ship: for both of them mathematical models have been implemented.

2.1 Ship model

The ship model has been developed using Marine System Simulator (MSS [15]), a Matlab/Simulink library and real-time simulator of marine systems developed by Fossen and Perez. MSS allows to simulate vessels for low-speed maneuvering and station-keeping in six degrees of freedom, basing on frequency dependent hydrodynamic data. The ship model is based on hydrodynamic data for a ship with size comparing favourably to that of a FREMM, and is governed by a series of controllers that provide guidance at the desired course and speed in a sea model derived from a directional wave spectrum based on statistical models, which considers significant wave heights, mean wave direction and wave spreading. Aerodynamic interaction between the naval unit and the helicopter and the effect of the helicopter on the dynamics of the naval unit are, at present stage, disregarded.

2.2 Helicopter model

The rotorcraft model is representative of a medium-weight multirole helicopter, akin to a Leonardo Helicopters AW101, frequently employed in offshore activities. The flight dynamics model has been built in MBDyn [1, 13] a free multibody simulation software developed at Politecnico di Milano (POLIMI) Department of Aerospace Science and Technology (DAER). MBDyn is able to perform simulations with real time scheduling based on POSIX primitives or RTAI system calls [12]. The helicopter has been modeled using as reference a typical configuration for a Search And Rescue (SAR) mission: full load and center of gravity aft of the main rotor hub. The fuselage is represented by a single rigid body with inertia properties referred to its center of mass.

Tail vertical and horizontal stabilizer have been modeled as aerodynamic bodies with one integration point each and NACA0012 airfoils. The AW101 five blade main rotor is fully articulated and used BERP IV blades [19]. In the multibody model, the latter have been replaced by rectangular blades. Each blade is represented by a single rigid body, with the center of mass positioned at half the blade span along the feathering axis, positioned at 25% of the blade chord; blade airfoil is NACA0012, with negative twist angle $\theta_{tw} = -5°$ from root to tip. Aerodynamics for each blade are computed in four integration points equally spaced along the span. Each blade is linked to the hub by a spherical hinge; the lag dampers have constant value coefficient $C_{\xi\xi}$ sized starting from the equation of lagging motion for small lag angle ξ [20]. The blades are connected to the rotating swashplate via pitch links modeled as rods with stiffness $K_{pl} = 10^9 N/m$ and damping coefficient $C_{pl} = 10^2 Ns/m$ to avoid possible dynamic instabilities related to the coincidence of aerodynamic center, center of mass and feathering axis. The tail rotor is modeled only through its thrust force, computed in closed form following the model proposed in [21]. Actuator disc theory does not take into account drag and profile power, and the model built does not consider interference effects given by the vertical stabilizer, thus resulting in a overestimation of the performances of the tail rotor; however, these limitations in the aerodynamic model are a compromise accepted to ensure real time performances.

2.3 Preliminary model validation

A preliminary validation of the model has been carried out by comparing the doublet response of the model with the data coming from flight tests performed on a BO105 in [29]. Even if the two helicopters belong to different categories, the comparison is useful to demonstrate the fidelity of the model (Fig.2 to 7). The comparisons reported here refer to a longitudinal and a lateral cyclic doublets performed at 3000ft and 80kt, while the control input has been extracted directly from BO105 flight test data [Fig 2, 5].

2.4 Automatic flight control system model

The control part contains the primary flight controls input manager and the models of the AFCS elements. All these subsystems have been built in Simulink, each one is conceived as an independent module and each can be managed, removed or upgraded separately. All the AFCS elements implement saturation to keep the control/corrective action inside the excursion limits of the actuators. At the current state of the project, actuators dynamics is not modeled, but the system is ready to accommodate future improvements.

The project is here shown at a stage where actuator dynamics was not taken into account because the purpose was to demonstrate the feasibility of a real time simulator with the above mentioned characteristics. The implementation of actuator dynamics and the estimation of transport delays are going to be added in the next iteration of the project.

2.5 Trim

The trim procedure needed to start the simulation in the desired flight state is performed by a chain of PID regulators [Fig.8] acting in closed loop with the flight dynamics model. The controllers scheme is derived from the assumed pilot-vehicle loop structure

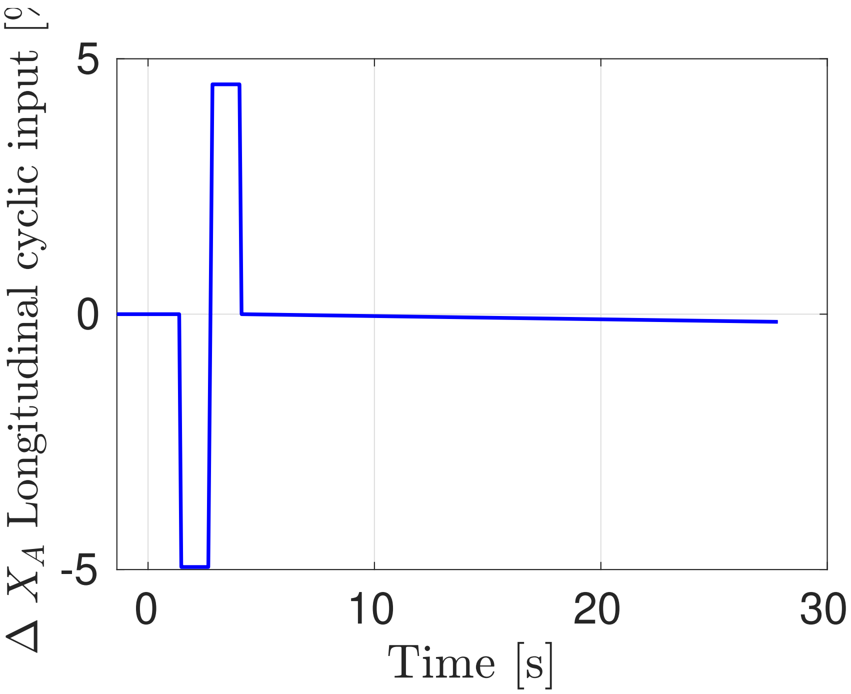

Figure 2: Longitudinal cyclic doublet.

for low speed flight developed by NASA [22]. Implementations of this model have been also developed in other projects at POLIMI DAER [23].

The Stability Augmentation System (SAS) has been developed using initial guess values coming from published data related to the IAR330 Puma helicopter [24], subsequently adapted to the AW101 characteristics.

The AFCS is completed by altitude, heading and cruise speed hold modes, and a simple governor acting on an algebraic model of the engine torque output is in charge of keeping the angular speed of the rotor at the nominal value.

2.6 Contact model

Contacts between the ship deck plane, assumed rigid, and the landing gear wheels, approximated by single contact points, is implemented, together with the associated contact detection module, in a dedicated Simulink block. The inputs are position, orientation and velocities of the deck; position, velocity of the landing gear point (received from MBDyn via UDP socket), and the dimensions of the deck; The three contact forces (normal reaction and friction forces in the deck plane) are computed according to the model presented by Flores et Al [18], and are sent to the MBDyn model via UDP socket [Fig. 9]. The contact scheme proposed in the cited literature, which is also used by the `continuous contact` MBDyn module [17], has been adapted to cope with a finite surface moving in 3D space. The mathematical model of the contacts assumes that the landing deck is a rigid rectangular surface, and the helicopter landing gear wheels are points of application of the reaction forces resulting from the interaction between the deck and the landing gear itself. To preserve modularity, each landing gear is representing the interaction of the single leg and wheel with the deck.

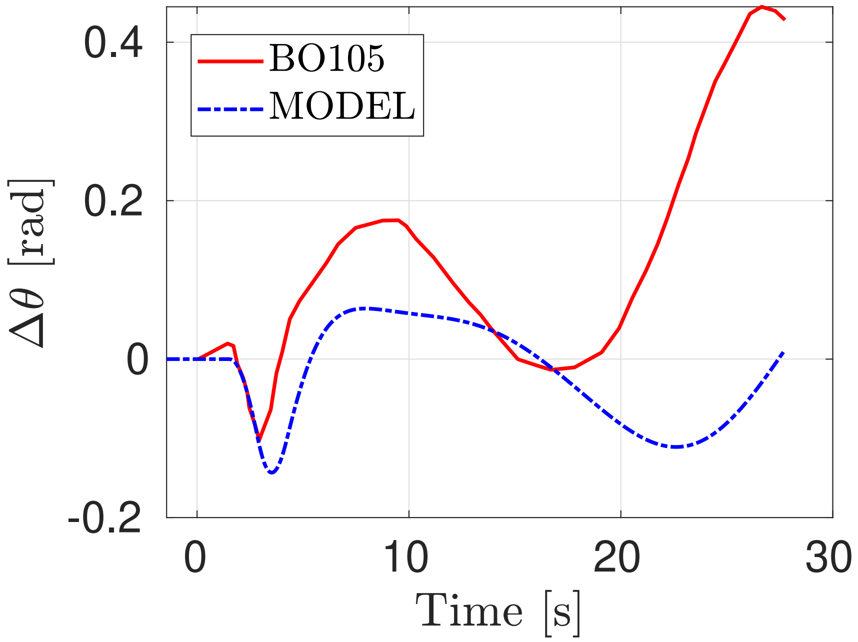

Figure 3: Longitudinal doublet: pitch angle response.

In this way asymmetric contacts are possible. The influence of the helicopter on the ship dynamics is discarded.

2.7 Visualization module

All the simulation runs have been carried out using X-Plane [9], with the pilot wearing VR headset. The communication between MBDyn and X-Plane has been established through a C++ X-Plane plugin passing aircraft and ship data via UDP socket. The simulations take place in open sea: a context in which there is a small amount of visual cues except the horizon, the ship itself and some reference navigation points in the form of distant oil rigs and naval traffic. The ship model itself presents markings that can to be used as visual cues. An augmented reality feature was added in the virtual reality environment: a HUD, following pilot's head movements and allowing they to check instruments without losing eye contact with the landing spot. [Fig. 10].

3 Experiments and results

The mission task elements of the simulator runs are summarized in figure 11. The mission variants considered are four and cover standard operating procedures for deck landing: approach with ship at anchor, with ship moving at 12 kt, straight approach from stern and lateral approach from port quarter. The latter is directly related to the position of the pilot, who usually sits on the right-hand seat. This guarantees the best visual cues by maximizing the field of view of the pilot [See Fig.10]. Sea state level has been kept at level 5 [26], representing the upper limit at which landings in rough

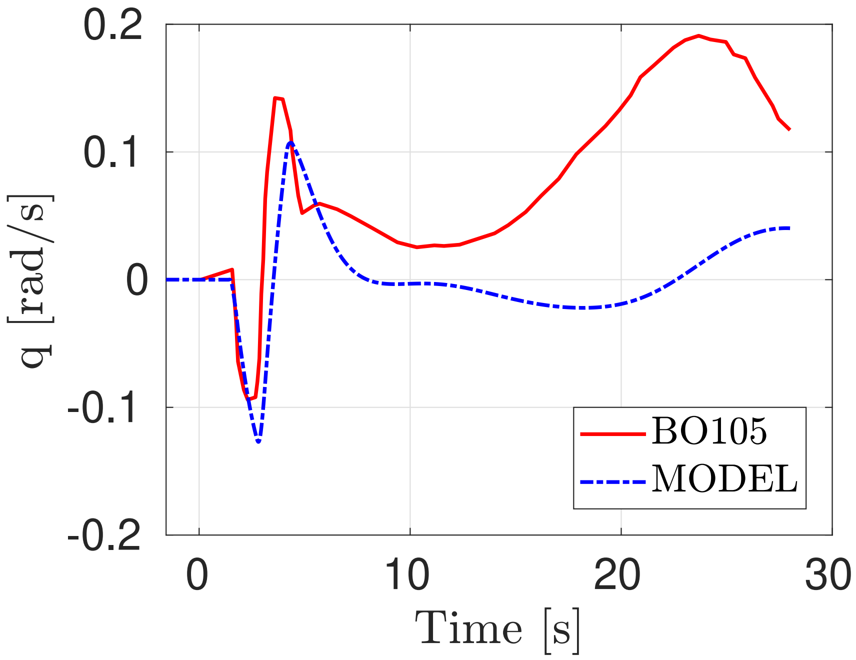

Figure 4: Longitudinal doublet: pitch rate response.

waters are usually carried out with naval units of this size; The limiting factors to deck approach can be identified in the roll, pitch and heave of the naval unit. In real situations landing would be discouraged if not forbidden with the hull undergoing rotations of over 3° along x and y axes. The results reported here are about port-quarter approach with ship at 12 kts: at this velocity the 3° limit is exceeded for roll during short periods, but on average the time history has been considered acceptable for the experiments because, from the point of view of this degree of freedom, the quiescent period remains under the boundaries defined. After touchdown, the rotorcraft and the ship move together. In Fig. [12], [13], [14], [15] are reported the time histories for the ship behavior, respectively ship speed profile, heave motion, roll and pitch angles. Fig. [16], [17], [18], [19], [20], [21] report helicopter behavior during approach and landing. After touchdown, the time histories of the ship are superimposed to show that contact is maintained after touchdown.

3.1 Considerations on real-time performance

One of the objectives was to prove that a simple setup was able to run a real time flight simulator coupled with VR assets. Real time performance requirements of the simulator are ensured by different settings. The first is the choice of the level of detail of the model, which depends on the type of representation that must be studied. The next factor is the integration time step, which depends directly on the nominal value of the angular velocity of the main rotor:

$$dt = \frac{2\pi}{N_{dt}\Omega_N} \tag{1}$$

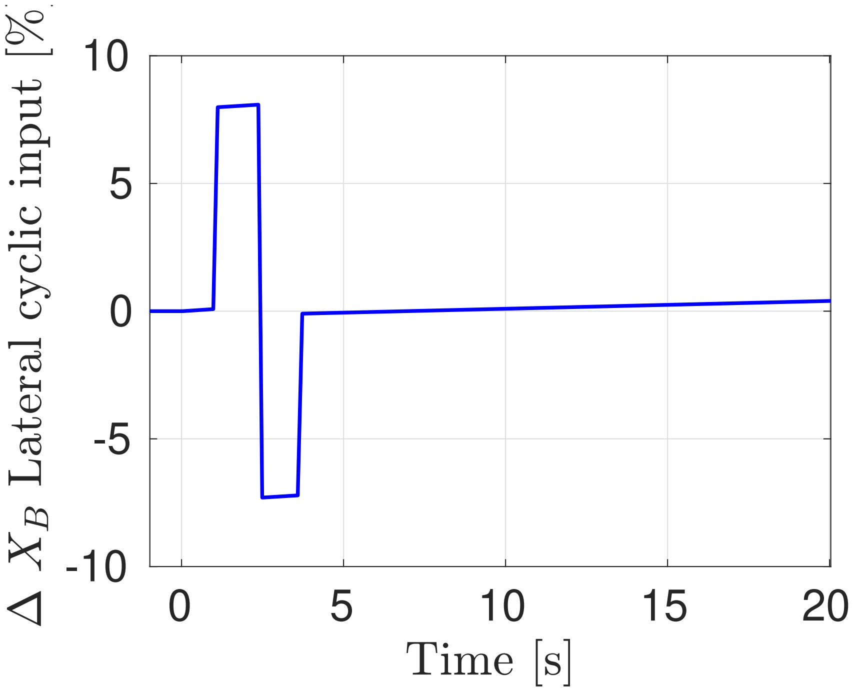

Figure 5: Lateral cyclic doublet.

The factor N_{dt} has been chosen to give the best real time performances without losing numerical stability. The best value for N_{dt} has been found to be 110, but this value can also be increased when faster dynamics need to be investigated. The considerations done brought to a time step value set at about $2.6 \cdot 10^{-3}$ seconds. MBDyn uses real time scheduling based on POSIX standards, which has been used here. Other performance-enhancing factors include the scheduling of the communication with the visualization tools to avoid graphic lags. The scheduling is performed to obtain a reciprocal positive effect: visualization process is CPU-time consuming, but since optimal refresh rate for visual is in general lower than the simulation time step used in the framework of rotorcraft simulation, it is not necessary for the visualization tool to receive information at each time step, because the simulation process would be slowed down in favour of useless rendering. Therefore, it is not mandatory for MBDyn to send information to the visualization tool at the each time step. This aspect is important especially when using the VR headset, for which every condition resulting in user's motion sickness must be avoided. The tuning procedure consists in the choice of the number of steps during which MBDyn does not send information to the visualization layer. The optimal ratio has been found in sending an updated state once every 6 simulation timesteps. Another important aspect of the real time performance is related to transport delays and the latency occurring between the latest available reading of the pilot input on the control sticks and the visualization of the updated aircraft configuration in the VR headset. The control stick input is read by MBDyn at the beginning of each timestep: therefore the input value at timestep t_k is available for computation of the aircraft state at the same timestep. Before the time budget dt has expired, if the output for the current timestep is requested, the new state is sent to X-Plane. Therefore, the maximum latency that

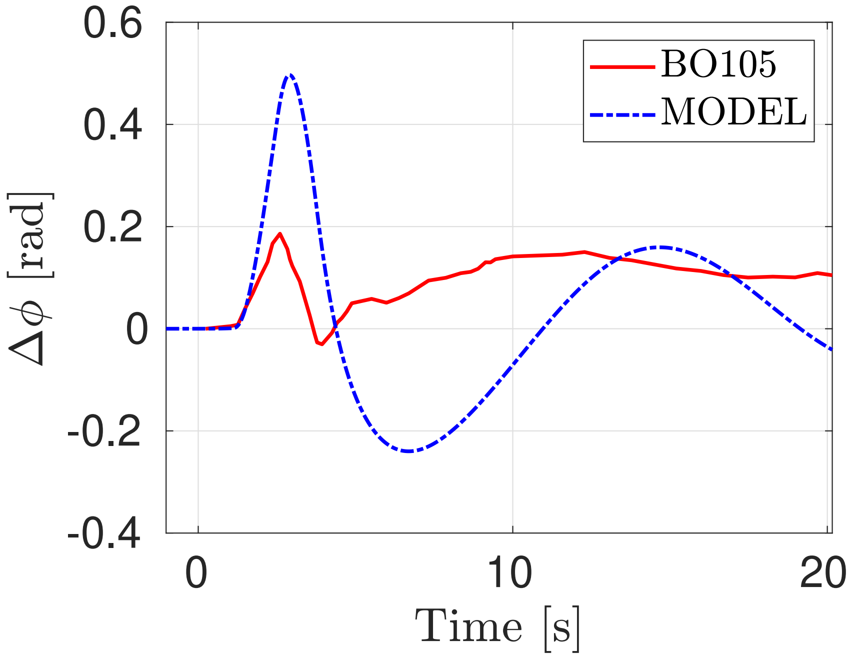

Figure 6: Lateral doublet: roll angle response.

is experienced by the pilot is in any case strictly less than *dt*. The plugin mentioned in Section 2.7 is written to take care of some of these issues as well. To demonstrate real time capabilities of the simulator, after the deactivation of POSIX scheduling, a data collection campaign was conducted. The simulator had been started 100 times with randomly generated initial trim velocity and heading, each run programmed to last 30 seconds. At the end of each run, the CPU time was collected. The data had been statistically analyzed to have an idea of the ratio between the real simulation time and the CPU time. The results of this analysis are summarized in figure 22, 23. All the CPU times collected are well under the 30 seconds of the simulation, the mean CPU time is 18.16 seconds, and the 97.5% of the simulation runs last less than 18.7 seconds. This means that about 37.7% of the CPU time during the simulation is saved and computer computing power can be used for other tasks, such as increasing the complexity of the model.

4 Conclusions

The purpose of this paper was to describe how a helicopter flight simulator based on first-principles approach offered by multibody dynamics, in particular using MBDyn and to validate the approach of equipping the pilot with a VR headset. Particular care has been put on the visualization side of the model, where the standard six instruments and the controls have been recreated to improve the immersion in the simulation. A series of tests, focusing on the high demanding task of approach and landing on the moving deck of a naval unit have been set up to demonstrate the validity of the system. The Ship model, based on the free software MSS, relies on actual hydrodynamics

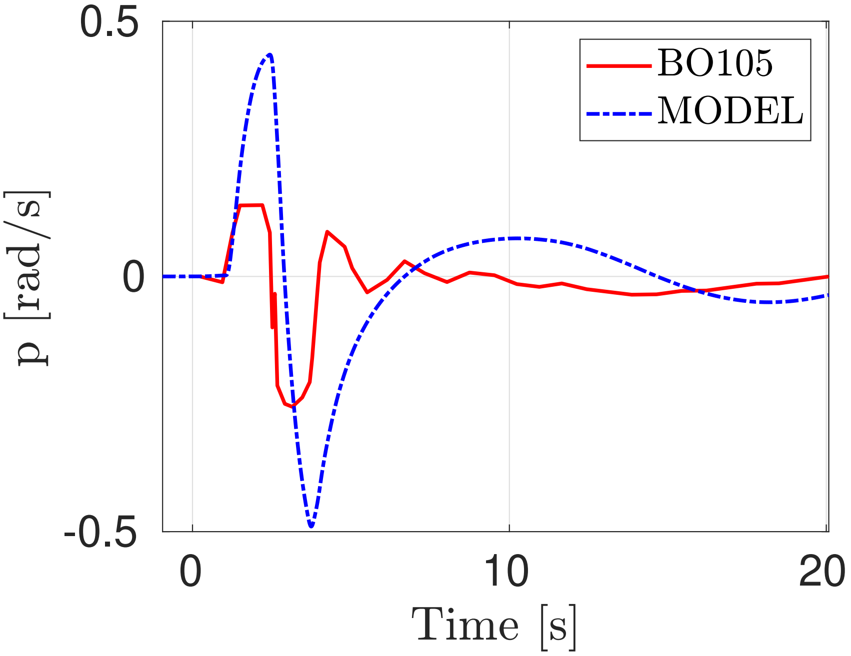

Figure 7: Lateral doublet: roll rate response.

data and statistical wave models based on true sea state to recreate realistic conditions. The helicopter has been modeled after the AW101, widely used to perform offshore missions and operating from naval units, accompanied by a complete AFCS created in Simulink. Four mission layouts recreated standard operating procedures for deck approach and landing. This also proves that it is possible to create low cost, high fidelity simulators, not necessarily built to with the objective to be used only as training devices for pilots, but dedicated primarily to the study of the rotorcraft during early design stages and prototype development. Virtual reality allows fast and easy reconfiguration of the simulation device without having to commit further costs for a new simulator for each new rotorcraft or different design studies. This could anticipate the individuation of problems during the conceptual design phase, leveraging virtual reality simulation in support to the early design stage.

5 Conflict of interest statement

The authors of this paper state that there is no conflict of interest.

References

[1] P. Masarati, M. Morandini, P. Mantegazza, "An Efficient Formulation for General-Purpose Multibody/Multiphysics Analysis", *ASME J. Comput. Nonlinear Dyn., 9(4):041001*, October 2014, doi 10.1115/1.4025628.

[2] A. Zanoni, P. Masarati, G. Quaranta, "Physics-based piloted flight simulation of helicopters using multibody dynamics", *Proceedings of IDETC/CIE 2016*, August 2016

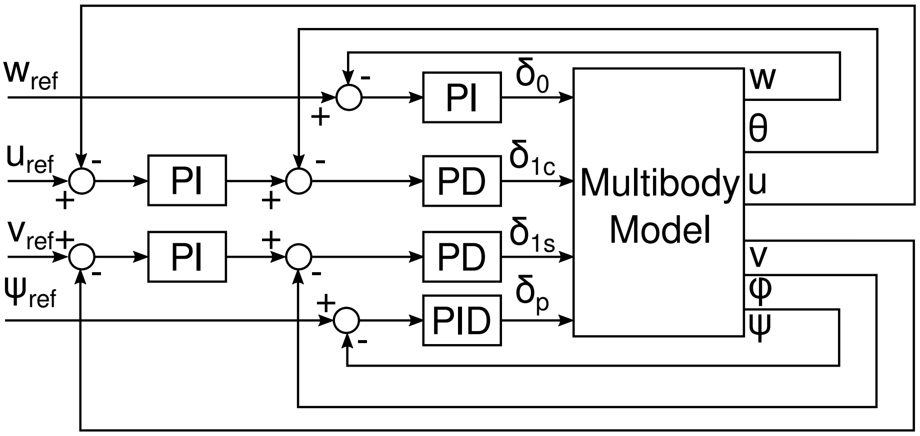

Figure 8: PID controller for the trim procedure.

[3] J. L. Long, C. J. Lloyd, D. A. Beane, "Practical geometry alignment challenges in flight simulation display systems", *IMAGE 2010 Conference* 2010.

[4] H. Franklin, B. Larson, M. Johnson, J. Droessler, W. F. Reinhart, "Panoramic, Large-Screen, 3-D Flight Display System Designs", *NASA Contractor Report 198218*, October 1995, doi NAS 1-20073,

[5] Justin Knaplund, "Large Field of View (FOV) Collimated Displays in Transportable or Relocatable Shelters", *presentation at SIMUniversity*, 2014.

[6] https://www.vive.com/us/comparison/

[7] https://www.oculus.com/

[8] https://www.getfove.com/

[9] https://www.x-plane.com/

[10] https://www.mbdyn.org/

[11] https://public.gitlab.polimi.it/DAER/mbdyn

[12] https://www.rtai.org/

[13] P. Masarati, D. J. Piatak, G. Quaranta, J. D. Singleton, J. Shen, "Soft-Inplane Tiltrotor Aeromechanics Investigation Using Two Comprehensive Multibody Solvers", *Journal of the American Helicopter Society*, Vol. 53, No. 2, pp. 179-192, April 2008, doi 10.4050/JAHS.53.179.

[14] A. R.J.M. Lloyd, "SEAKEEPING: Ship behaviour in rough weather, 2nd Revised edition", May 1998, isbn 978-0953263400.

[15] T. I. Fossen, T. Perez, *Marine Systems Simulator (MSS)*, https://github.com/cybergalactic/MSS, released under GNU/GPL v3.0 licence

[16] T. I. Fossen, Ø. N. Smogeli, "Nonlinear Time-Domain Strip Theory Formulation for Low-Speed Maneuvering and Station-Keeping", *Modeling, Identification and Control*, Vol. 25, No. 4, pp. 201-202, 2004.

[17] P. Masarati, "MBDyn input file format, Version 1.7.3", chpt B.2.4, October 2017.

[18] P. Flores, M. Machado, M. T. Silva, J. M. Martins, "On the continuous contact force models for soft materials in multibody dynamics", *Multibody Systems Dynamics*, Vol. 25, pp. 357-375, 2011, DOI 10.1007/s11044-010-9237-4.

[19] , K. Robinson, A. Brocklehurst, “BERP IV Aerodynamics, Performance and Flight Envelope”, *34th European Rotorcraft Forum*, Liverpool, September 2008.

[20] A. R. S. Bramwell, G. Done, D. Balmford, “Bramwell’s Helicopter Dynamics, Second edition”, Chpt. 1.3-1.4, 2001, isbn 0750650753.

[21] R. W. Prouty, “Helicopter Performance, Stability, and Control”, Chpt. 3 - Aerodynamics Of Forward Flight, 2002, isbn 1575242095.

[22] R. K. Heffley, W. F. Jewell, J. M. Lehman, R. A. Van Winkle, “A compilation and analysis of Helicopter handling qualities data”, *NASA Contractor Report 3144*, August 1979.

[23] L. Trainelli, A. Croce, C. E. D. Riboldi, R. Possamai, “Trimming a High-Fidelity Multibody Helicopter Model for Performance and control analysis”, *41st European Rotorcraft Forum*, 2015.

[24] V. Muscarello, G. Quaranta, P. Masarati, L. Lu, M. Jones, M. Jump, “Prediction and Simulator Verification of Roll/Lateral Adverse Aeroservoelastic Rotorcraft–Pilot Couplings”, *Journal of Guidance, Control and Dynamics*, Vol. 39, No. 1, January 2016.

[25] V. Muscarello, G. Quaranta, P. Masarati, “The role of rotor coning in helicopter proneness to collective bounce”, *Aerospace Science and Technology*, No. 36, pp. 103–113, 2014.

[26] https://www.nodc.noaa.gov/woce/woce_v3/wocedata_1/woce-uot/document/wmocode.html

[27] A. Zanoni, M. Zago, R. Paolini, G. Quaranta, P. Masarati, M. Galli, G. Maisano, L. Frigerio, M. Murawa, “Flight Simulator Testing to Enhance Comprehension and Modeling of Rotorcraft Pilot Couplings”, *45th European Rotorcraft Forum*, Warsaw, September 2019.

[28] M. Daniele, A. Zanoni, P. Masarati, G. Quaranta, “Development of a Virtual Reality Real Time Helicopter Flight Simulator”, *XXV Congress of the Italian Association of Aeronautics and Astronautics*, Rome, September 2019.

[29] Rohlfs, M. and Grünhagen, Wolfgang, Kaletka, J., “Nonlinear Rotorcraft Modeling and Identification”, January 1999, isbn 92-837-0006-6.

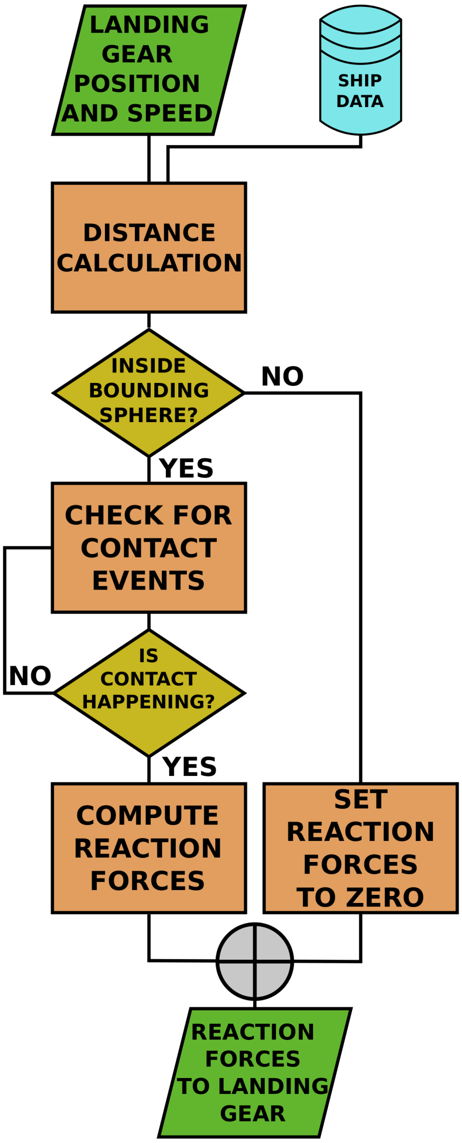

Figure 9: Reaction force calculation flowchart.

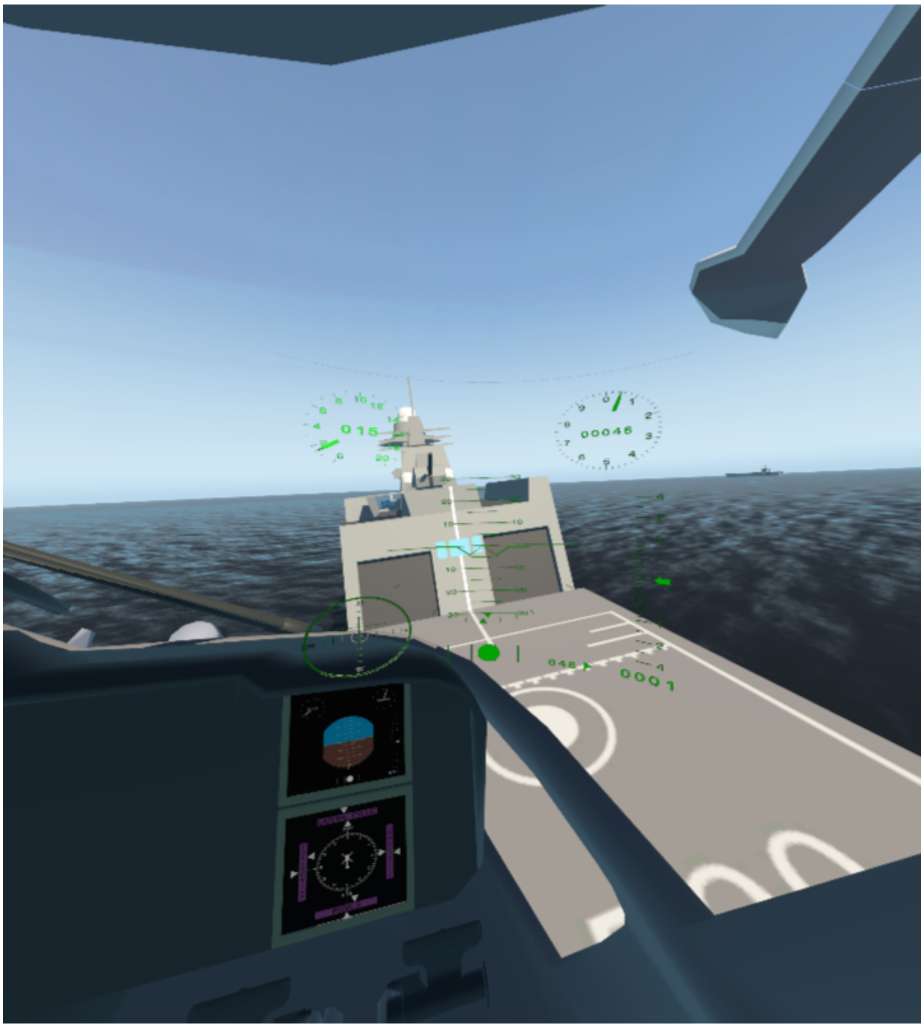

Figure 10: Pilot's view during approach: glass cockpit and HUD are visible.

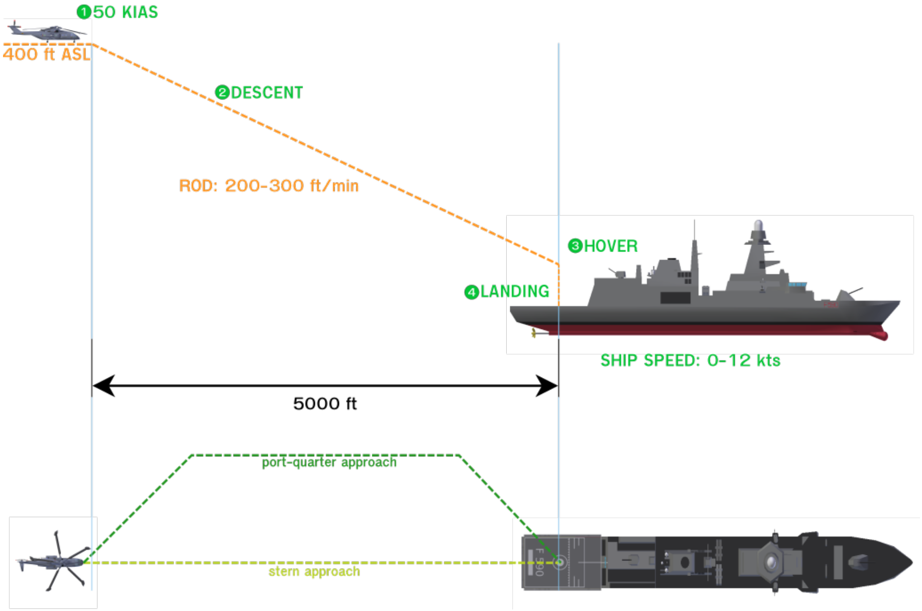

Figure 11: Mission profiles.

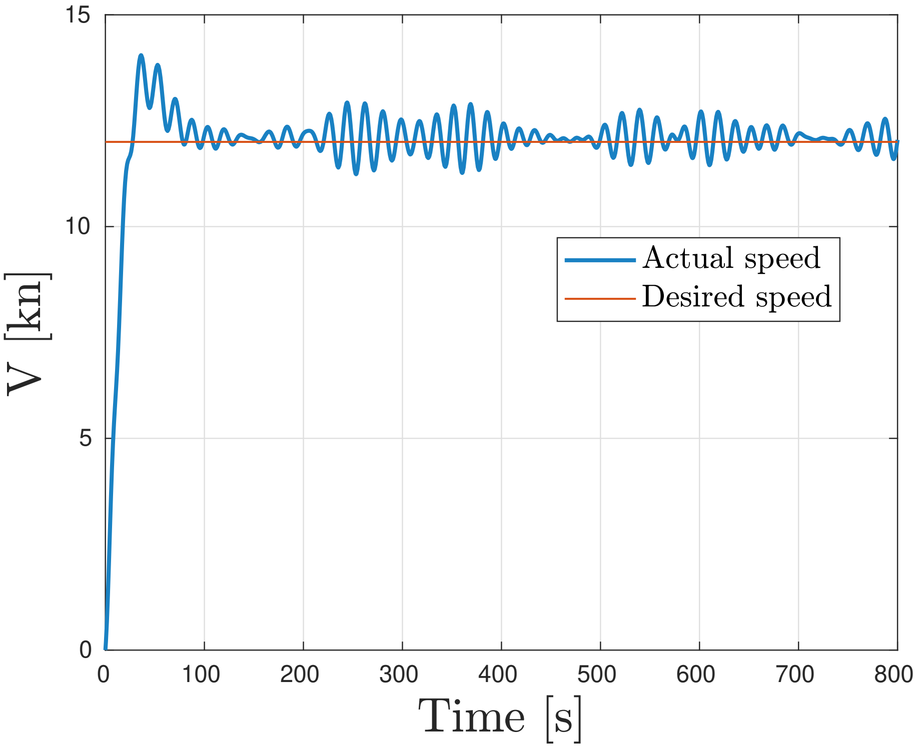

Figure 12: Ship at 12 kt: speed profile.

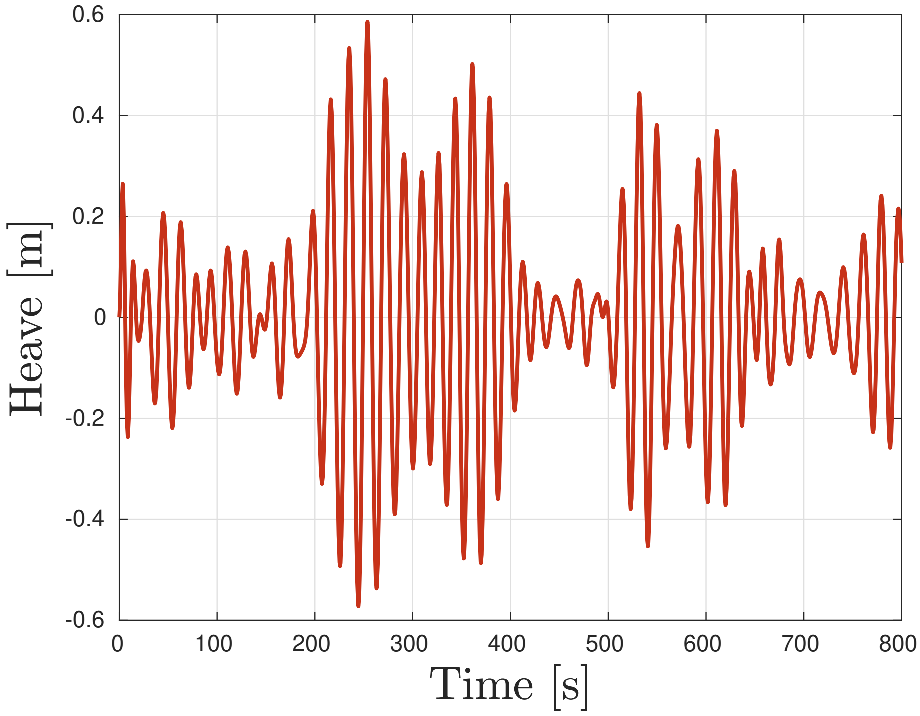

Figure 13: Ship at 12 kt: heave motion.

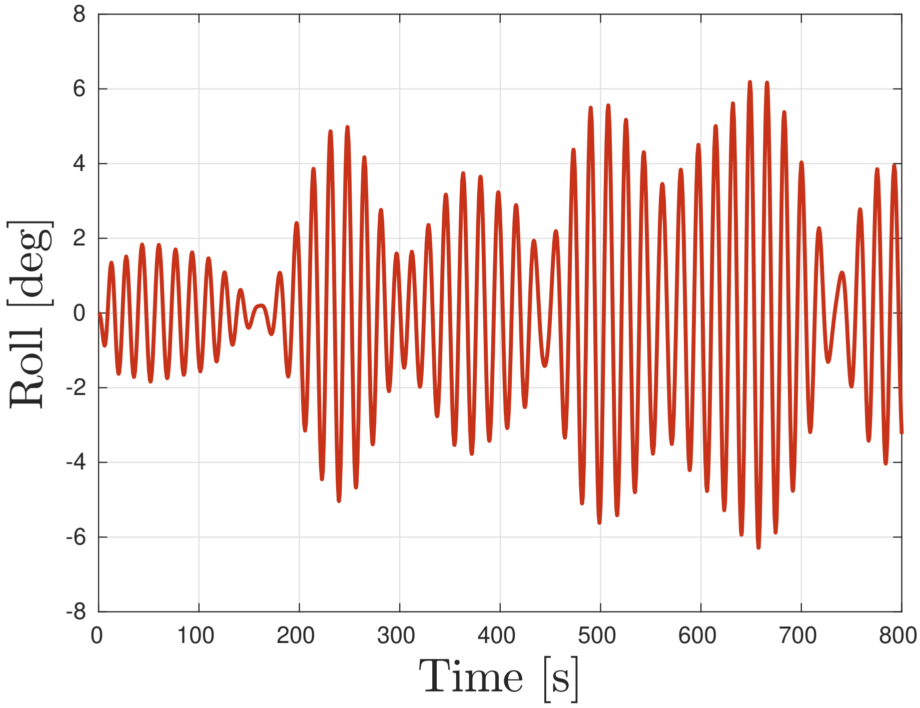

Figure 14: Ship at 12 kt: roll angle.

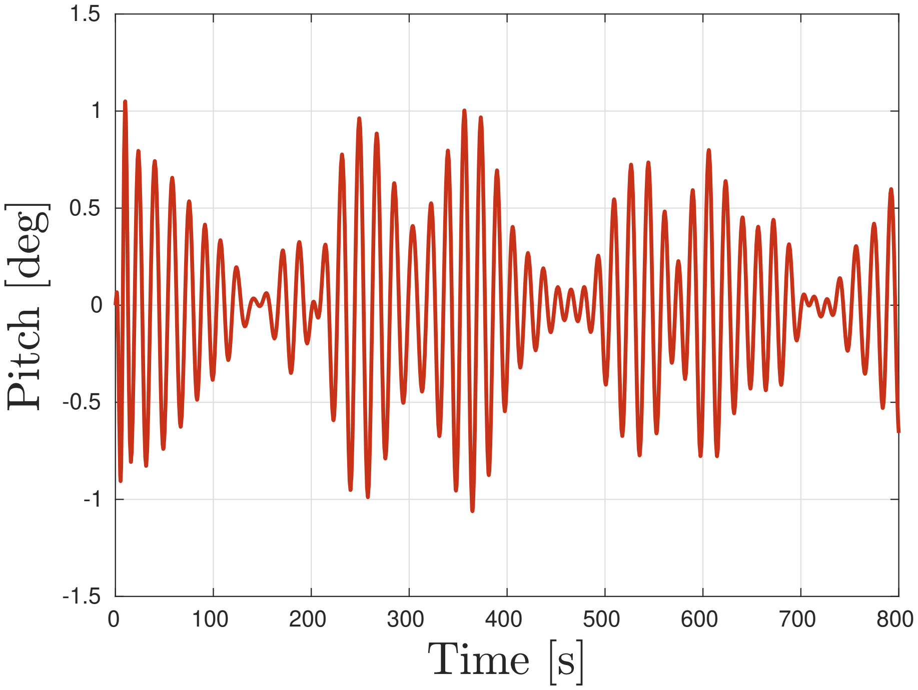

Figure 15: Ship at 12 kt: pitch angle.

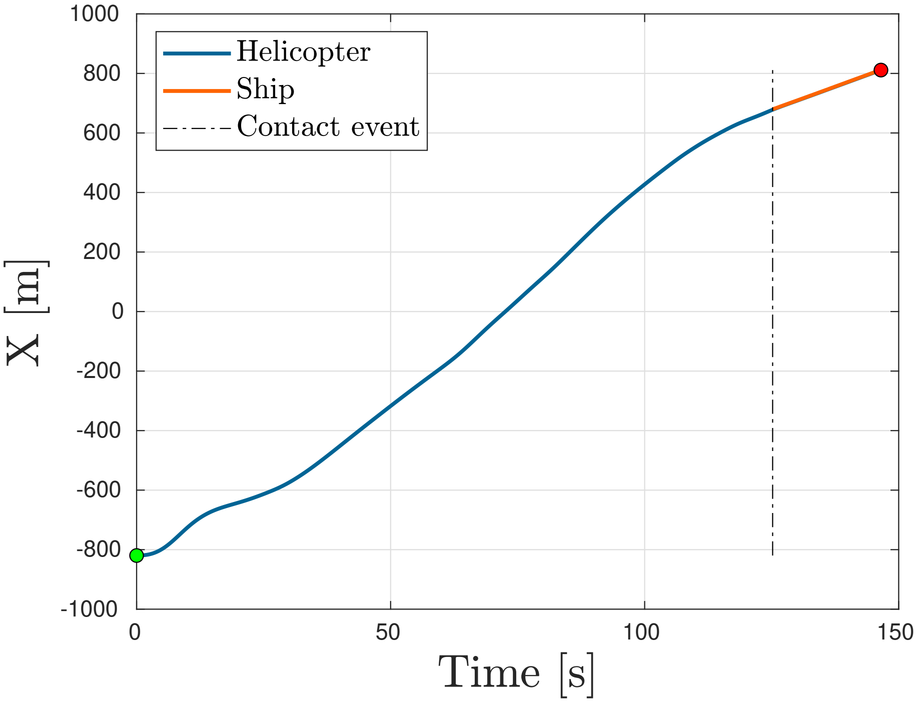

Figure 16: Port quarter approach: X.

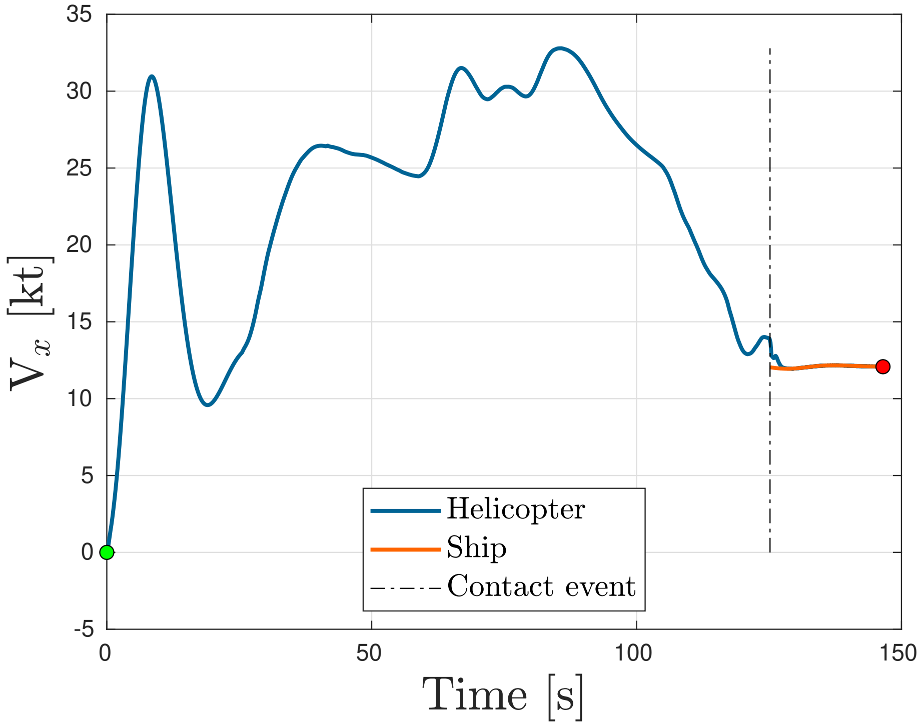

Figure 17: Port quarter approach: V_x.

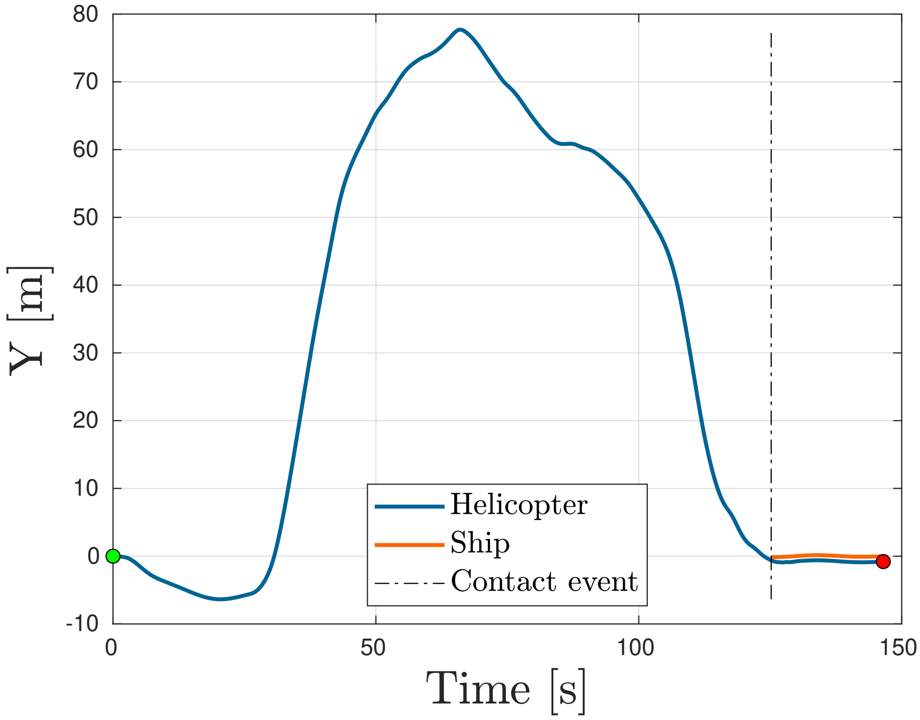

Figure 18: Port quarter approach: Y.

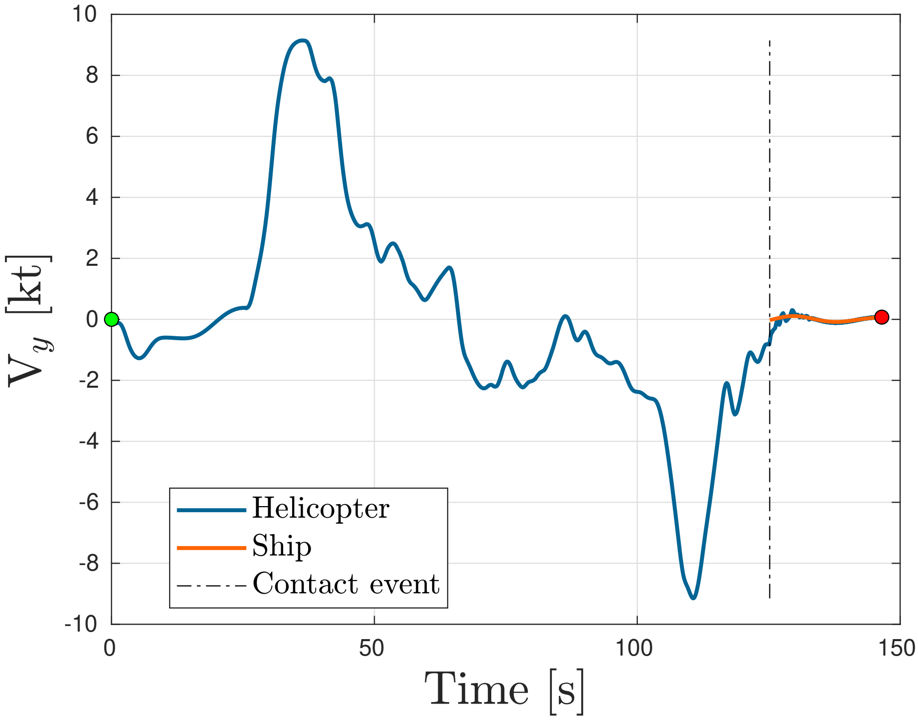

Figure 19: Port quarter approach: V_y.

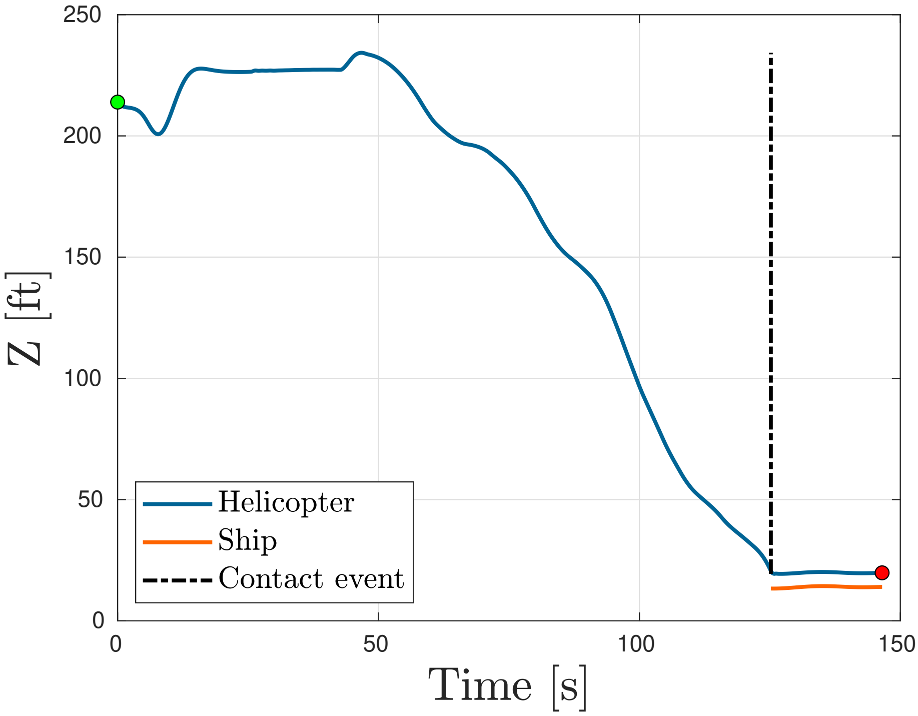

Figure 20: Port quarter approach: Z.

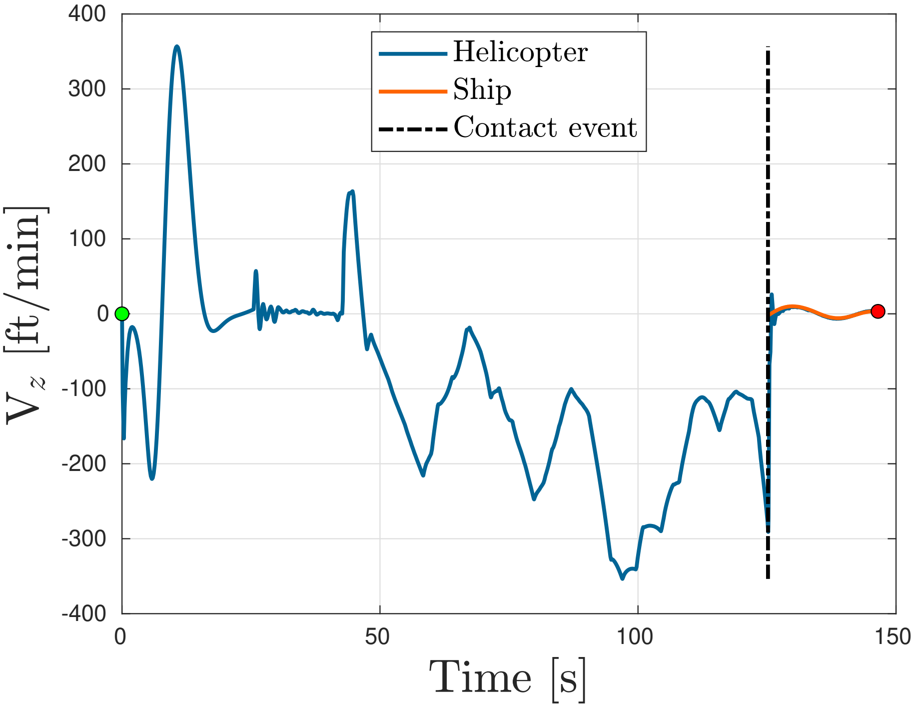

Figure 21: Port quarter approach: V_z.

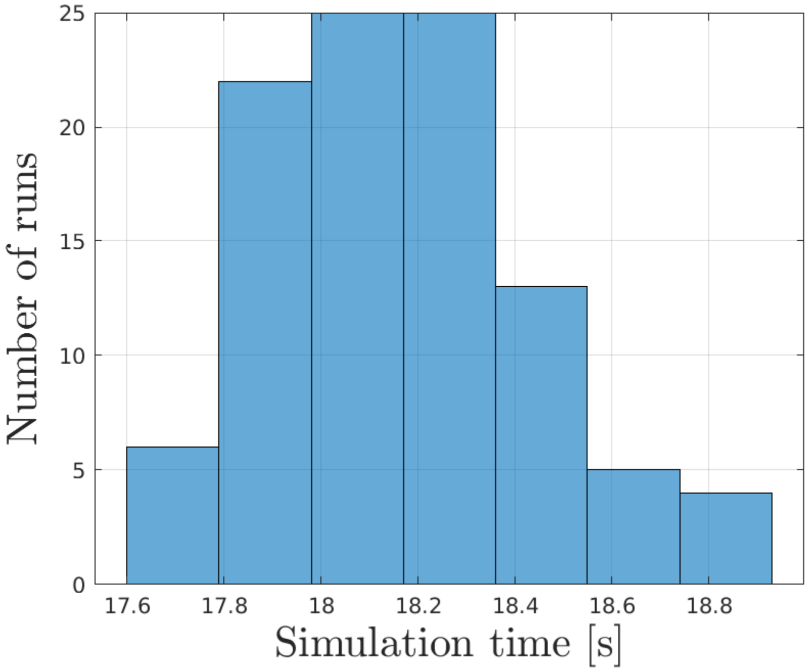

Figure 22: CPU times histogram.

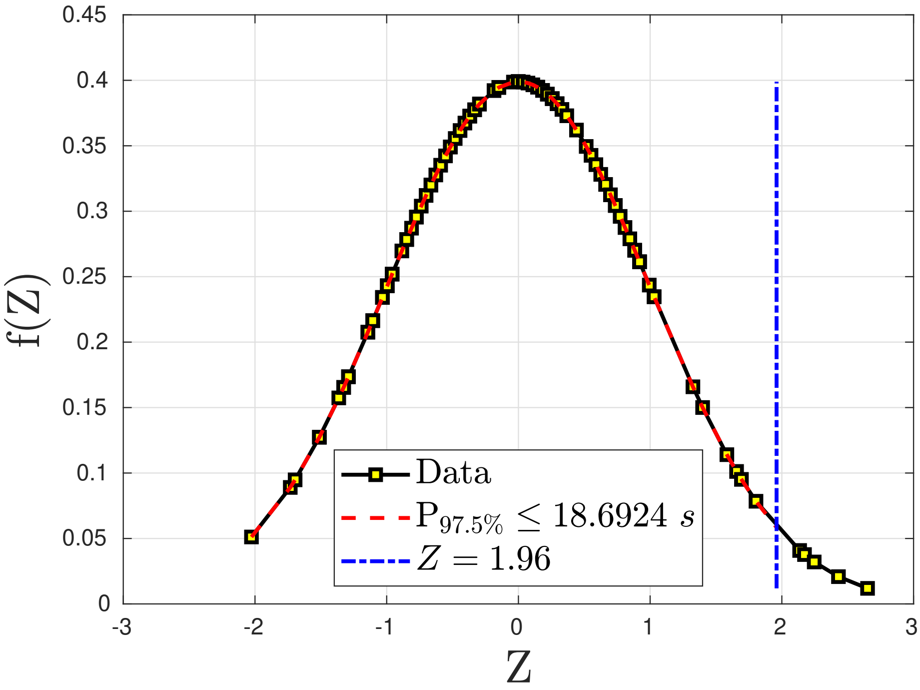

Figure 23: CPU times normal distribution.