System Simulation for Autonomous UAV Design

Application of system simulation to model an octocopter UAV performance and its integration in an autonomous flight co-simulation framework

FEDERICO CAPPUZZO
Siemens Digital Industries Software, France
KENEDY MATIASSO PORTELLA
Federal University of Santa Maria, Brazil
RAPHAEL LALLEMENT
Scalian, France
JEAN-PATRICK MASCOMERE
Total, France
GUILLAUME THALMANN
Altran Technologies, France

This paper presents the use of system simulation techniques to model the performance of an octocopter UAV following specifications shared in the frame of the European research project COMP4DRONES. Batteries, electric motors, propulsion and flight behavior are simulated in the context of a mission including seismic sensors droppings. The performance model was then integrated in a co-simulation framework to include navigation sensors, mission environment, and guidance and control algorithms to simulate the drone's behavior when faced with obstacles avoidance and cluster flight.

CCS CONCEPTS • Computing methodologies • Modeling and simulation • Simulation types and techniques

Additional Keywords and Phrases: System Simulation, Drones, UAVs, Multicopters, Performance Engineering, Model Based Systems Engineering.

ACM Reference Format:

First Author's Name, Initials, and Last Name, Second Author's Name, Initials, and Last Name, and Third Author's Name, Initials, and Last Name. 2018. The Title of the Paper: ACM Conference Proceedings Manuscript Submission Template: This is the subtitle of the paper, this document both explains and embodies the submission format for authors using Word. In Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY. ACM, New York, NY, USA, 10 pages**. NOTE: This block will be automatically generated when manuscripts are processed after acceptance**.

1 INTRODUCTION

1.1 Motivation

The use of drones or Unmanned Aerial Vehicles (UAVs) in commercial applications has the potential to disrupt several industries, including transportation, communication, agriculture, disaster mitigation and environment preservation, dramatically improving the work efficiency, reducing costs and risks related to these activities. [1][2]

To cover effectively such a broad spectrum of applications, UAV integrators require the ability to develop drone platforms that meet the requirements specified for the missions to accomplish. Simulation-based analysis are essential to this extent, as they provide useful means to explore the design space and select the most promising concepts that comply with requirements and specifications. In addition, models and simulation provide an effective complement to more traditional and expensive physical prototyping and testing, contributing to reduce costs and time-to-market. As a result, well-defined, reliable models are essential for the development of new technologies. If integrated into a Model Based System Engineering (MBSE) strategy, models bring the ability to capture, analyze, share, and manage the information related to product design. The benefits associated to MBSE include enhanced communications among the development stakeholders, increased ability to manage system complexity, and improved product quality. [3][4]

UAVs fall into the category of complex multi-physical systems, as their emergent behavior is the result of the integration of sub-systems involving more than one simultaneously occurring physical domains. Due to their ease of both construction and control, multirotor aircraft are frequently used UAV platforms. [5] The most common multirotor is the quadcopter, that relies on four propellers for propulsion and attitude control. These propellers are driven by electric motors controlled by Electric Speed Controllers (ESC) and powered by a battery. Even in the simplest configuration, UAVs behavior is determined by the interaction of electrical, mechanical and control systems, and they are characterized by their dynamics and aerodynamics performance.

Having discussed that UAVs can be considered as multi-physical dynamic systems, this paper shows that system simulation is particularly well suited to analyze their performance. System simulation is a set of techniques dedicated to the modeling, analysis and optimization of multi-physics dynamic systems. This paper describes how system simulation provides the right set of tools to analyze and optimize the performance of drones, showing the intermediate results of an on-going use case related to the European research project COMP4DRONES. In a first step, the software tool Simcenter Amesim is used to model the different subsystems of the UAV and simulate its performance. Then it is coupled with Simcenter Prescan, a tool for environment and sensors modeling, which enables to test advanced control algorithms such as Detect & Avoidance or synchronization with members of a fleet of heterogeneous drones.

1.2 Related work

The increasing interest in UAV applications has led to the modeling of drones primarily with tools such as Matlab Simulink® [6] or ROS [7]. These tools are widely used for the development of control algorithms and robot software development, which are among the most challenging and innovative areas of research for UAV applications. Their use is sometimes extended to build the plant model, which simulates the UAV dynamics and other behavioral performance, with the purpose of testing the control algorithms under development. For this reason, their use for plant modeling activities may require making strong assumptions, (ideal power sources, simplistic propulsion modeling…) resulting into low-fidelity plant models that often are not able to capture the interactions between interconnected systems. Such models can quickly become impractical for the continuous development, verification and validation of GNC algorithms as these become more mature or sophisticated.

The co-simulation framework presented in this paper allows to create higher fidelity models that can simulate the interactions between different systems and their impacts on the drone’s overall dynamics. For this reason, they are more effective in testing, verifying and validating GNC algorithms. To provide a few example, the proposed framework is able to easily model the effect of decreasing batteries voltage on the electric motors response as the battery charge depletes or as its temperature varies; the impact of the sloshing of liquid in a tank on the drones dynamics; more accurate thrust characteristics depending of the propellers geometry used; and more.

1.3 Paper organization

The paper is organized as follows. Section two presents Simcenter Amesim, with a focus on the solutions dedicated to drones modeling. Section three shows the modeling of a drone selected for a use case part of the COMP4DRONES research project, and the validation of the subsystems modeled. Section four describes the benefits of systems simulation applied to UAVs design. Section five illustrates a more comprehensive modeling and simulation framework including the software tool Simcenter Amesim, Simcenter Prescan and Matlab®. Finally, the conclusions and future work are presented.

2 SIMCENTER AMESIM

Siemens' Simcenter Amesim is a software tool dedicated to modelling and simulation of dynamic and multi-physics systems. [8] [9] Multi-physical modeling

In the tool's environment, systems are modelled connecting components available in the libraries, which cover several physical (fluids, mechanical, electrical…) and application (aerospace, automotive, gas turbines…) domains. Figure 1 is a screenshot of the tool's graphical user interface.

Each component is described by tabulated data and/or nonlinear time-dependent analytical equations. More than one modelling option can be proposed for the same component (i.e. tabulated approach or analytical equation), and the choice is made by selecting the preferred submodel. This mixed-fidelity approach provides the benefit of scalable modelling strategy, where the model accuracy can evolve along with the design cycles as design decisions are made and more information of the product becomes available. Submodels are coded with the programming language C.

The components interface is realized through ports that allow the flow of physical variables. The definition of the interfaces is based on the bond graph theory, which allows a common representation of dynamic systems regardless their physical domain. [11] [12] This approach ensure consistency in the conservation of energy, conservation of mass and units of measurements when connecting different components. Simcenter Amesim is equipped with a solver which automatically adapts the time step and selects the best algorithm for the resolution of the systems of equations. The time step changes according to the frequency content of the simulation, while the numerical scheme depends on the stiffness of the systems of equations to be integrated. This allow faster simulations, and at the same time, it let the user focus on the modelling aspects instead of having to take care of the choice of the solver parameters.

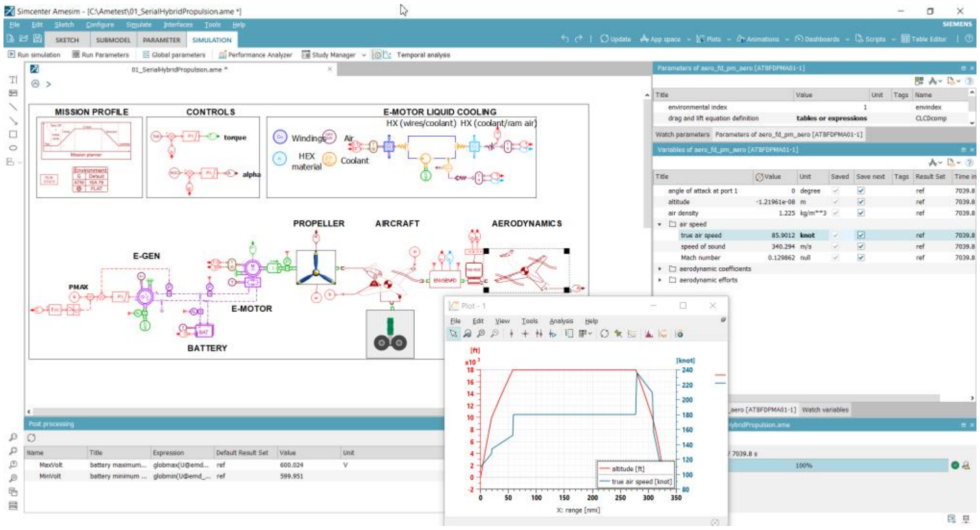
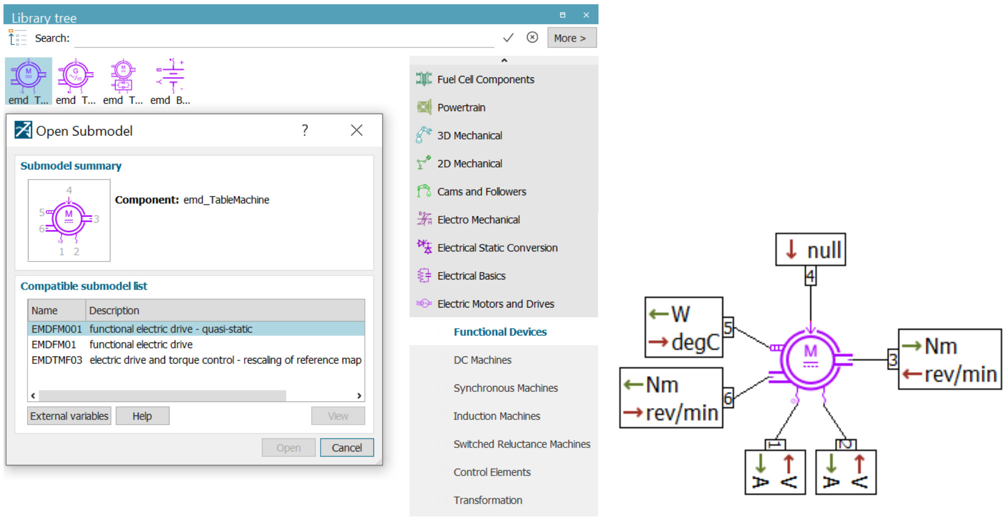


Figure 1: Screenshot of Simcenter Amesim GUI. Flight performance Simulation of a serial hybrid propulsion aircraft.

Figure 2 shows a screenshot of a portion of the library tree, a list of submodels associated to the component of an electrical motor and its interface.



Component: emd_TableMachine

Name	Description
EMDFM001	functional electric drive - quasi-static
EMDFM01	functional electric drive
EMDTMF03	electric drive and torque control - rescaling of reference map

Figure 2: On the left, list of submodels associated to the same component. On the right, an example of component interfaces. The electric motor uses several physical ports: signal (control), rotational mechanical ports (generated and reaction torque), thermal and electrical.

The tool also provides the possibility to use fixed step solvers, which is required for co-simulations with other software tools or Hardware-in-the-loop, Software-in-the-loop, and Real-Time activities.

Simcenter Amesim libraries provide off-the-shelf components to rapidly model the system to analyze. Concerning UAVs, the Electric Motor and Drives library and the Electric Storage library offers submodels with different fidelities of electric motors and batteries respectively. For drones powered by fuels, the Gas Turbine library allows to model the performance of the turboshafts or jet engines. The Aerospace & Marine library provides flight dynamics, aerodynamics and propellers modeling capabilities.

Simcenter Amesim is by nature an integration platform, with dedicated interfaces to other software tools. Siemens is part of the consortium defining and developing the Functional Mock-Up Interface (FMI) standard. This is a protocol that defines a container and an interface to exchange dynamic models using a combination of XML files, binaries and C code zipped into a single file. FMI can be considered as an enabler for scalable and tool neutral integration of simulation models from different technical disciplines, developed by different actors.

2.1 Software architecture

A simplified view of the software architecture is illustrated in Figure 3. [10] The left portion of the figure represents the software operational view. The modeling workflow is divided into three steps. The first is completed in the sketch mode, where components accessible from standard or customized libraries are connected. Then, in the submodel mode, the user selects the level of fidelity of each component when more than one is available. The modeling is completed in the parameter mode, where the user inputs are provided to characterize the model. The simulation and analysis activities are performed in the simulation mode. Here the model is compiled into an executable and the simulation results are analyzed. At this point, using dedicated built-in tools, the model can be used to explore the design space, perform parameters optimization or sensitivity analysis. The right side of Figure 3 summarizes how the software is constructed. The graphical user interface guides the user through the workflow previously described. The code generator enables the export of model for model/ software/ hardware-in-the-loop analysis. The compiler translates the model into an executable, which is coupled to a numerical solver to compute the simulation results. The results manager allows to store and post-process the simulation results.

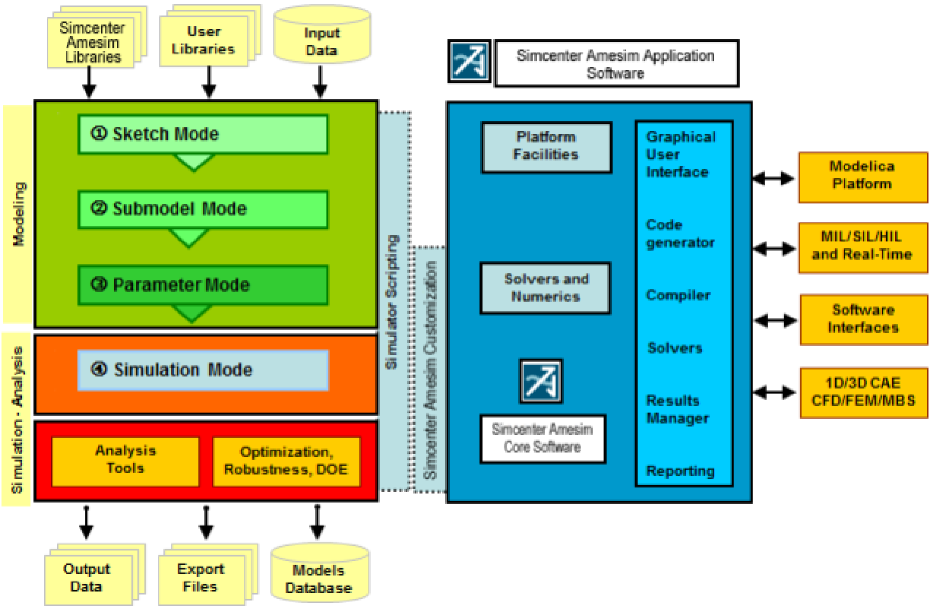


Figure 3: Architecture of Simcenter Amesim (simplified view). Operational (left side) and constructional (right side) views [10]

3 USE CASE: MODELING OF AN OCTOCOPTER FOR OIL AND GAS EXPLORATION

The use case selected for this paper is inspired by the METIS® R&D project led by the oil and gas company Total SE [13], which also represents one of the two demonstrators of the “Logistics” use case of the COMP4DRONES research project. This demonstrator aims at using a fleet of drones to explore hard-to-access onshore areas. UAVs capable of carrying and dropping seismic sensors are working cooperatively, while surveillance UAVs offer an extra level of safety by detecting any intruders. Finally, a ground vehicle will retrieve the sensors once the geophysics acquisition is finished.

Regarding the modelling activities, the first step was to model and validate the main components of the drone, namely the electric motor, the propeller, and the battery, and then verify that the integrated model of the UAV was able to meet the specification provided by the partners of the research project as showed in Table 1.

Table 1: UAV specifications

Component	Characteristic
UAV type	Octocopter with coaxial propellers
Electric motor	T-MOTOR U10 II [14]
Battery	Lipo 6S
Nominal voltage	50V
Propellers	T-MOTOR 30.2x9.9R [15]
MTOW	35 kg
Payload	12 kg
Sensor weight	1 kg
Mission	Take off, 1 km cruise, drop sensor each 100m (6 times), 1 km cruise, landing

Following the main geometrical constraints provided by the partner, a simplified digital mock-up (Figure 4) was created to estimate the inertia matrix of the drone and its variation at each drop event.

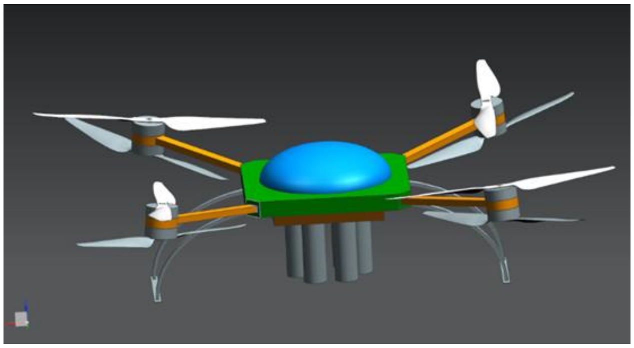


Figure 4: UAV digital mock-up rendering.

3.1 Electric motor modeling and validation

Thanks to the electric motor datasheet available on the manufacturer website [14], it was possible to build an Amesim model, then conduct a series of test to measure the performance and compare the results obtained from the simulation with those of the datasheet. The voltage, rotary speed and the resistive torque generated by the propeller are the inputs to the model. The motor power, efficiency and current were compared with those provided by the manufacturer. The comparisons are shown in Figure 5. The velocity input has a linear profile that covers all the speed range assessed in the test. The voltage is constant at 48.5 V. The simulated motor efficiency and power are close to the experimental results provided by the manufacturer. On the other hand, the results for the current shows discrepancies that were not understood by the author. In fact, given that the power profile of the test and simulation matches, the same should be true also for the current given the linear relation of power, current and constant voltage ($P = VI$). However, this was not considered as a blocking point: as the power absorbed by the motor was computed correctly by the model, this was deemed as sufficient for the purpose of the analysis.

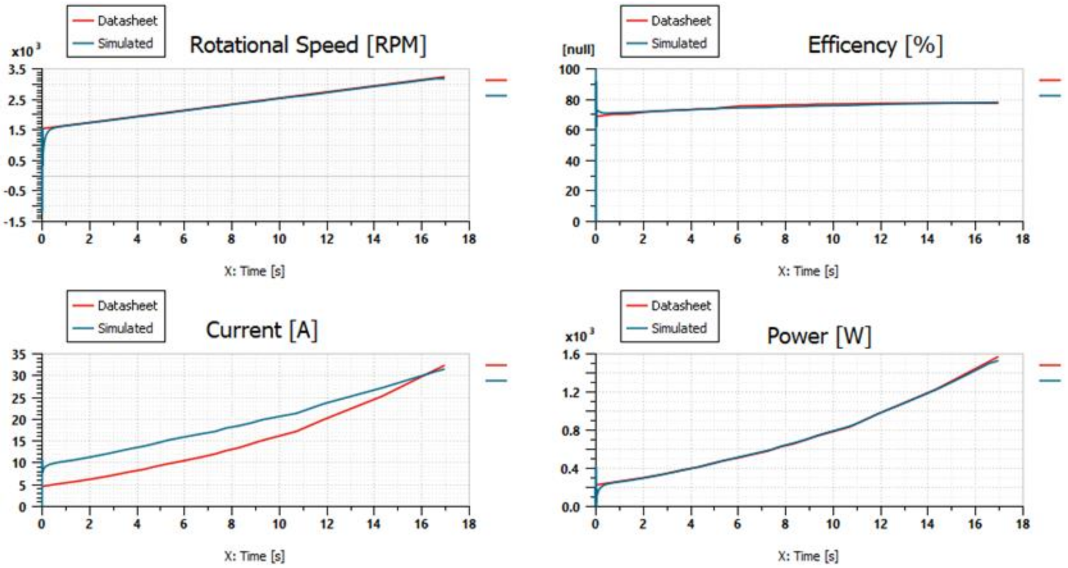


Figure 5: Comparison of experimental (red lines) versus test (blue lines) data.

3.2 Propeller modeling and validation

On the topic of the propeller’s modeling, the Aerospace & Marine library of Simcenter Amesim provides a set of components and tools based on the Blade Element Momentum Theory [16], a low-order model for the computation of the local forces on a propeller, effectively and extensively used to address aircraft, marine, and wind turbine propellers. Kolaei et al. [17] published the experimental results of a test performed on propeller T-motor G30X10.5, which has a slightly different diameter and pitch with respect to that selected by the partner (see Table 1). It was decided to model the T-motor G30X10.5 and validate the simulation results against the test data found in Kolaei et al. before proceeding to the modeling of the propeller type used for the UAV analyzed.

Figure 6 presents the comparison between simulated and experimental data. The results obtained by simulation have a mean error 4.24% for the torque and 7.92% for the thrust compared to the experimental results. Given the complexity of the geometry and the errors not associated with the computing method (like measurements errors, extraction of the results from a graph, etc.), the accuracy of the blade element momentum theory implemented in Amesim was deemed as acceptable and used to model the propeller mentioned in Table 1.

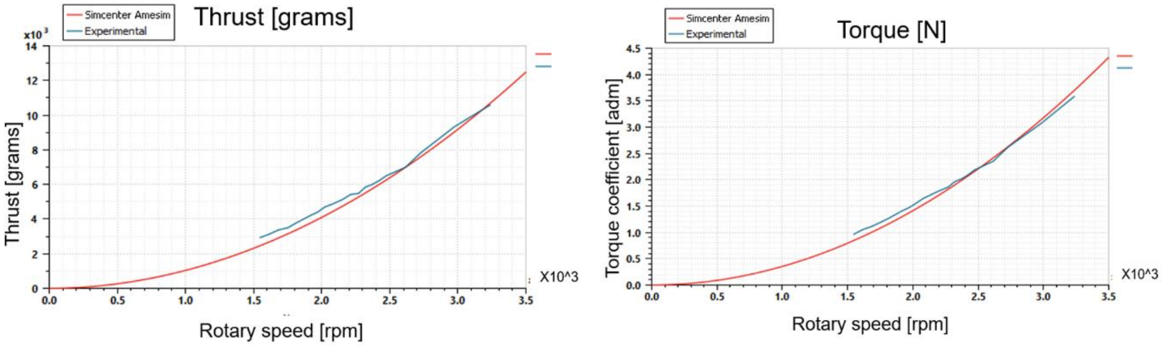


Figure 6: Comparison between simulation and experimental results of thrust and torque

The UAV selected for the project is an octocopter with four pairs of propellers in a coaxial configuration. To address this configuration, the original blade element momentum theory had to be enhanced. Leishman et al. [18] [19] proposed a modification of the momentum theory, this makes the method capable of addressing identical coaxial propellers turning at the same velocity (applicable to coaxial helicopters, but not to multi-copters). A model that considers propellers with generic geometries and any operational conditions is proposed by Rand et al. [20]. This model, however, is restricted to axial flight. The forward flight condition for coaxial propellers is finally addressed in Enconniere et al. [21] using a modification of the model proposed by Rand et al. [20]. This last one was implemented in Amesim to address multirotor UAVs with contrarotating propellers capable of rotating at different speeds and in hover, vertical and forward flight.

3.3 Batteries

Table 1 provides the type and the nominal voltage of the battery. Simulating the mission with the complete model of the UAV replacing the battery model with an ideal energy source, gives an approximation of the energy needed to accomplish the mission. This information can be used to generate the performance maps (namely open circuit voltage, entropic coefficient, and charge resistance as function of the battery state of charge and temperature) using the dedicated *“Battery pre-sizing tool”* of Amesim’s Electrical Storage library.

3.4 Controllers

To control the position and attitude of the UAV, a set of PID controllers was implemented. These adapt the rotational speed of each of the eight propellers to reach the commanded position and attitude.

3.5 Flight dynamics

The dynamics of the UAV was represented by a six degrees of freedom body assumed to be rigid with variable mass and inertia. The six degrees of freedom allow to describe the position and attitude of the UAV in space. The assumption of rigid body, i.e. no aeroelastic phenomena considered, is acceptable given the expected range of loads applied to the UAV’s frame. The capability to consider a variable mass and inertia is fundamental to capture the effect of the sensor dropping on the flight dynamics.

3.6 Simulation and results

The reference mission specified in Table 1 is plotted in Figure 7. Each star on the plot corresponds to a drop event.

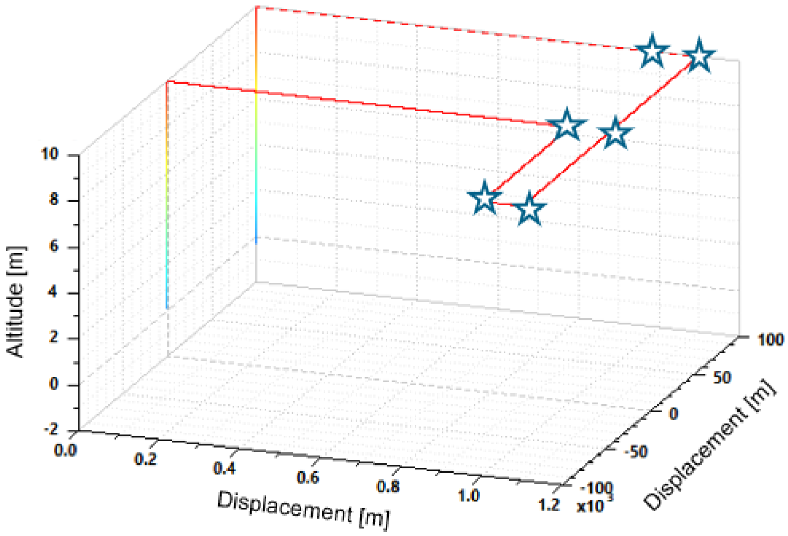


Figure 7: UAV target mission.

Figure 8 shows the target and the simulated trajectory in blue and red respectively. The trajectory is defined in terms of altitude, longitudinal position, lateral position, and heading (yaw angle). The octocopter increases its pitch or roll attitude to move longitudinally or laterally respectively. This is accomplished varying symmetrically the rotational speed of two pairs of propellers, while vertical motion is obtained modulating simultaneously the rotational speed of all propellers. Comparing the target and simulated results of Figure 8, it can be concluded that the drone is able to follow the trajectory required, compensating the sudden variation of the drone's inertia due to the sensors drop, plotted in Figure 9. As the simulation takes approximatively 10 seconds to run on a regular laptop (Intel® Core™ i5 processor, 16 Gb or RAM), the model supports design space exploration activities in case the drone performance requirements change, and the design must be reviewed.

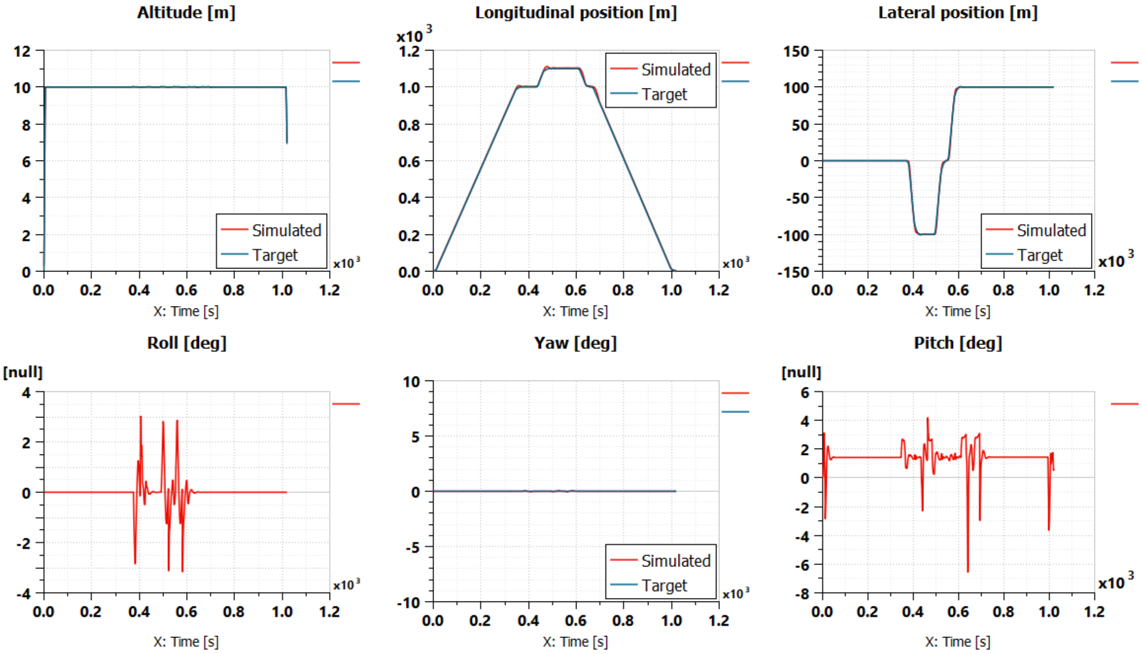


Figure 8: UAV target versus simulated trajectory plotted against the mission time. The UAV changes its attitude (roll and pitch angles) to follow the target trajectory.

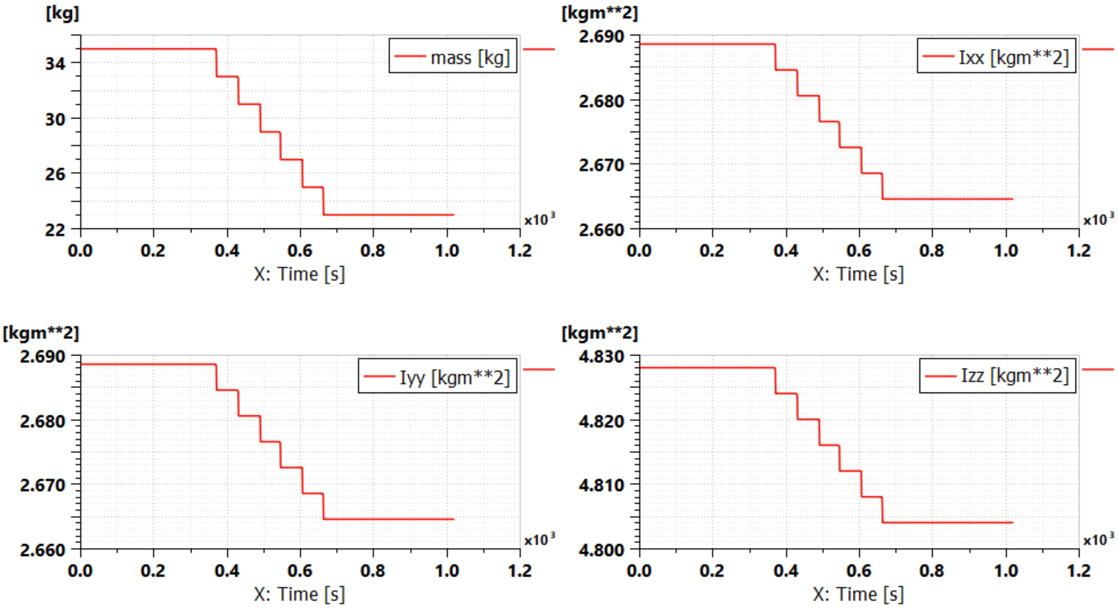


Figure 9: UAV mass and inertia variation with respect to the mission time

4 CO-SIMULATION FRAMEWORK FOR HETEROGENEOUS FLEET MODELING

Having completed the model of the UAV for its performance simulation, the next step includes the modeling of the mission environment, the drone's sensors required for the navigation, and the implementation of Guidance Navigation and Control (GNC) algorithms to perform obstacle avoidance maneuvers. Finally, the implementation of GNC algorithms enabling drones to fly in formation is addressed.

The framework depicted in Figure 10 enables this kind of co-simulation. Based on drone's state variables (accelerations, velocities, position and attitude), Simcenter Prescan computes an output for each sensor. Using this information, the GNC algorithms coded in MATLAB Simulink™ compute the control actions for the drones (in this case the rotary speed of each propeller). These control actions are sent to Simcenter Amesim, which computes the states for the following time step. For the co-simulation of a cluster of drones, the Simcenter Amesim model was exported as a Functional Mockup Unit [22] to allow multiple instantiations, one for each drone. On top of modeling capabilities of sensors like lidars, radars, and cameras, ranging from ideal to physics-based level of detail, Simcenter Prescan provides advanced animations that allow to visualize the behavior of the drone in the environment.

4.1 Obstacle Avoidance

Autonomous systems shall be able to handle a series of tasks in order to accomplish their missions. It implies a capability to make decisions when faced to unexpected events. Algorithms for real-time free of collision path generations are essential. The Optimized Artificial Potential Field Algorithm strategy proposed by Sun et al. [23] was selected to this purpose. The basic principle behind the method is to consider the movement of the drone as a type charged particle moving in a virtual force field. To do that, virtual charges were attached to the drone, to the objective (with opposite sign than the drone) and to the obstacles (with the same sign than the drone). These relations of charges make the drone attracted towards the objective and repelled by obstacles. The force vector resulting by the superposition of potential fields is then normalized to obtain its direction. The magnitude is imposed with a gain that is directly proportional to the desired UAV velocity. Figure 11 shows the algorithm's outcome. The blue dot is the initial position of the drone, the red star is the target destination. The environment was modeled such that an obstacle, a building, intersects the shortest route connecting the drone's starting and final position. The blue line is the trajectory autonomously computed by the drone allowing to avoid the obstacle. The ideal sensor called Actor Information Receiver was selected in the Simcenter Prescan tool. This sensor provides the position information for any object inside the covered area. It was chosen as the main goal of the analysis at this point was to design guidance and control algorithms, without spending too much effort on the selection of the best physic-based sensor for the mission.

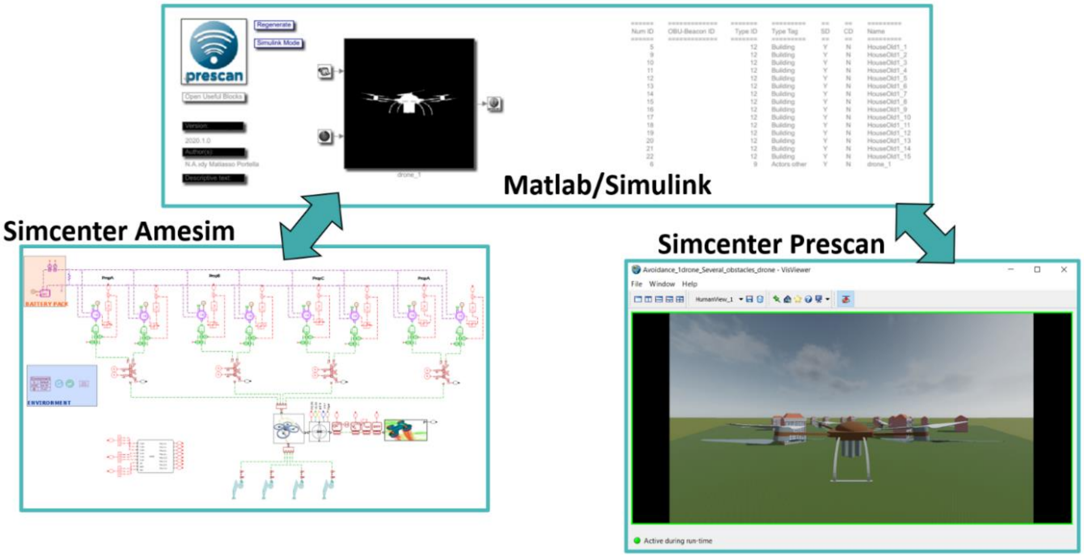


Figure 10: Autonomous drone co-simulation framework.

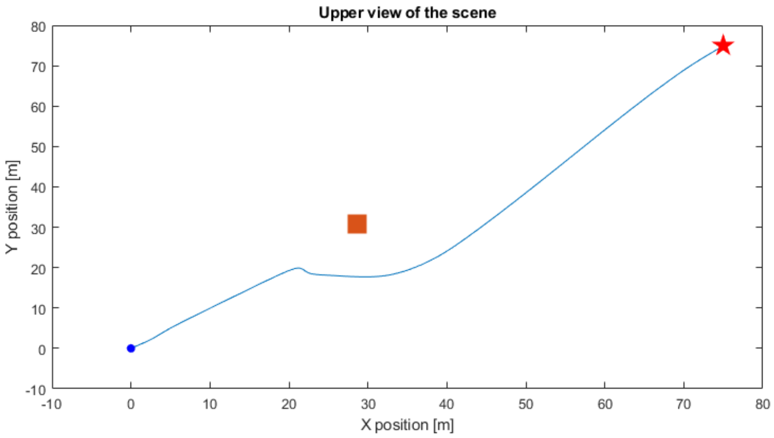


Figure 11: Avoidance maneuver. The blue point is the UAV’s initial position. The red star the destination. The square represents the obstacle. The blue line is the path followed by the UAV.

4.2 Cluster flight

The final step of this study regards the synchronization of a fleet of drones flying in cluster formation. This aspect is still an on-going activity and partial results are presented in this paper. The navigation algorithm Artificial Potential Field proposed by Galvez et al. [24]. In this case, in addition to the potential field described in the previous section, a second mimicking the gravitational field was added. While the first generates a repulsive force to avoid collisions, the latter attract drones in flight. The effect of the superposition of the two fields creates local minima represented by equilibrium distances between drones. This mechanism allows implementing autonomous formation flights. The test scenario involves four identical UAVs departing at different locations with the objective to reach the same destination keeping a safe distance among them. Figure 12 represents an upper view of the scene. The trajectory of all drones converges towards the common destination. Drones do not follow straight lines as they concurrent objective is to form a close group.

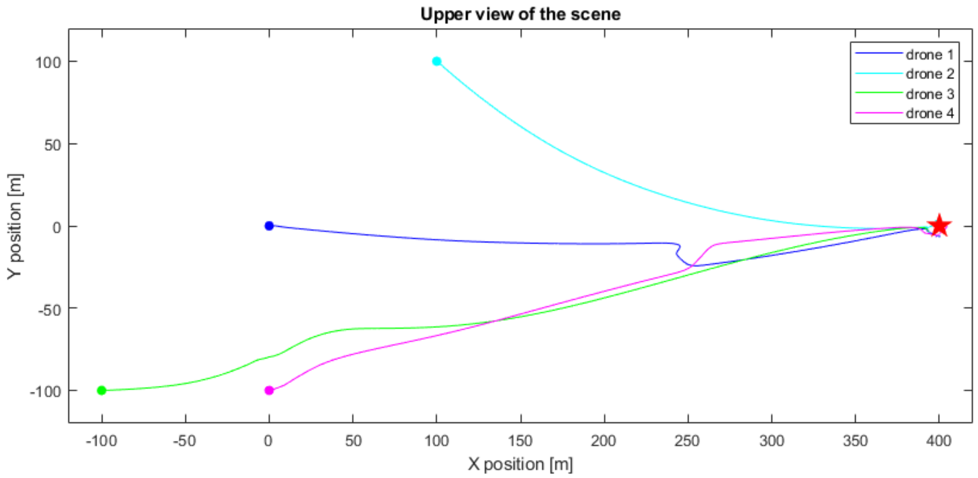


Figure 12: Cluster flight scenario. Upper view of the scene. Four UAVs departs from different locations and reach a common destination while creating a formation and avoiding collisions.

5 CONCLUSIONS AND FUTURE WORKS

This paper addressed the modeling of the performance, including flight dynamics, of an octocopter UAV, with four coaxial contra-rotating propellers and conventional full electric propulsion. The electric motors and the propellers performance were favorably compared against experimental data provided by the manufacturer. UAV integrators or designers can use this model to verify that the design meets the mission specification before any physical prototype is available.

The performance model was then integrated in a co-simulation framework capable of modeling drone's navigation sensors (lidars, cameras, etc…), the mission environment and GNC algorithms. This framework enables UAV integrators to perform trades on the UAV's sensing capabilities to accomplish the mission, as well as test different GNC algorithms.

Further improvements include the validation of the octocopter performance model with respect to experimental data to assess its fidelity and determine its credibility to support future design modifications decisions. Furthermore, physics-based sensors can be used instead of ideal sensors in the Simcenter Prescan tool. This would result into more accurate results of the drone navigation. The cluster flight formation algorithm could be tested with more complex scenarios, including more obstacles, drones and missions. Modeling of communication links between UAV and ground station, the fleet, and the traffic control system could add interesting insights, especially for certification activities and for regulation authorities.

Finally, it could be interesting to run the models on real-time machines to allow software-in-the-loop and/or hardware-in-the-loop simulations to verify and validate on virtual test benches the behavior of control algorithms and/or embedded processors.

ACKNOWLEDGMENTS

This project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 826610. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Spain, Austria, Belgium, Czech Republic, France, Italy, Latvia, Netherlands.

ECSEL Joint Undertaking
Electronic Components and Systems for European Leadership

REFERENCES

[1] Floreano, D., Wood, R. Science, technology and the future of small autonomous drones. Nature 521, 460–466 (2015). https://doi.org/10.1038/nature14542

[2] Aurambout, J., Gkoumas, K. & Ciuffo, B. Last mile delivery by drones: an estimation of viable market potential and access to citizens across European cities. Eur. Transp. Res. Rev. 11, 30 (2019). https://doi.org/10.1186/s12544-019-0368-2

[3] Friedenthal, Sanford & Griego, Regina & Sampson, Mark. (2009). INCOSE Model Based Systems Engineering (MBSE) Initiative.

[4] Madni, Azad M., and Michael Sievers. "Model-based systems engineering: Motivation, current status, and research opportunities." Systems Engineering 21, no. 3 (2018): 172-190. https://doi.org/10.1002/sys.21438

[5] Wikipedia. https://en.wikipedia.org/wiki/Multirotor. Visited on October 23, 2020.

[6] Cuauhtémoc Acosta Lúa, Claudia Carolina Vaca García, Stefano Di Gennaro, B. Castillo-Toledo, María Eugenia Sánchez Morales, "Real-Time Hovering Control of Unmanned Aerial Vehicles", Mathematical Problems in Engineering, vol. 2020, Article ID 2314356, 8 pages, 2020. https://doi.org/10.1155/2020/2314356

[7] Meyer J., Sendobry A., Kohlbrecher S., Klingauf U., von Stryk O. (2012) Comprehensive Simulation of Quadrotor UAVs Using ROS and Gazebo. In: Noda I., Ando N., Brugali D., Kuffner J.J. (eds) Simulation, Modeling, and Programming for Autonomous Robots. SIMPAR 2012. Lecture Notes in Computer Science, vol 7628. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-34327-8_36

[8] Wikipedia. https://en.wikipedia.org/wiki/Simcenter_Amesim. Visited on October 23, 2020.

[9] Simcenter Amesim. https://www.plm.automation.siemens.com/global/en/products/simcenter/simcenter-amesim.html. Visited on October 23, 2020.

[10] Simcenter Amesim 2020.2 Reference guide.

[11] H. M. Paynter. Analysis and design of engineering systems. MIT Press, Cambridge, MA, 1961

[12] Wikipedia. https://en.wikipedia.org/wiki/Bond_graph. Visited on October 23, 2020

[13] Lys, P.-O., Elder, K., & Archer, J. (2018, July 11). METIS, A Disruptive R&D Project to Revolutionize Land Seismic Acquisition. Society of Exploration Geophysicists.

[14] T-MOTOR U10 II KV100. https://store-en.tmotor.com/goods.php?id=700. Visited on October 23, 2020.

[15] T-MOTOR FA30.2×9.9 Prop-2PCS/PAIR. https://store-en.tmotor.com/goods.php?id=401. Visited on October 23, 2020.

[16] F.B. Gustafson and A. Gessow. Effect of Rotor-tip Speed on Helicopter Hovering Performance and Maximum Forward Speed. National dvisory Committee for Aeronautics - NACA Report, 1946

[17] A. Kolaei, D. Barcelos and G. Bramesfeld, Experimental Analysis of a small-scale rotor at various Inflow angles, International Journal of Aerospace Engineering, 2018.

[18] J.G. Leishman, Principles of Helicopter Aerodynamics, Second edition, 2006.

[19] J.G. Leishman and S. Ananthan. An Optimum Coaxial Rotor System for Axial Flight. Journal of the American Helicopter Society, 2008.

[20] O. Rand, V. Khromov, Aerodynamic Optimization of Coaxial Rotor in Hover and Axial flight, proceedings 27th International Congress of the Aeronautical Sciences, 2010.

[21] J. Enconniere, J. Ortiz-Carretero, V. Pachidis, Mission Performance Analysis of a Coaxial Rotorcraft for Air Taxi Applications, Aerospace Science and Technology, vol 69, 2017.

[22] Functional Mock-up Interface. https://fmi-standard.org/. Visited on October 23, 2020.

[23] J. Sun, J. Tang and S. Lao, Collision avoidance for cooperative UAVs with Optimized Artificial Potential Field Algorithm, IEEE Access, 2017.

[24] R.L.Galvez, G.E.U.Faelden, J.M.Z.Maningo, R.C.S.Nakano, E.P. Dadios, A.A.Bandala, R.R.P.Vicerra, A.H.Fernando, Obstacle Avoidance Algorithm for swarm of aerial Quadrotor Unmanned Aerial Vehicle Using Artificial Potential Fields, Proceedings 2017 IEEE Region 10 Conference, 2017.