



HAL
open science

Applying a Model-Based Systems Engineering Approach to Model an Unmanned Aerial Vehicle Mission

Badra Anthony Abi, Ombeline Aiello, Jean-Charles Chaudemar

► To cite this version:

Badra Anthony Abi, Ombeline Aiello, Jean-Charles Chaudemar. Applying a Model-Based Systems Engineering Approach to Model an Unmanned Aerial Vehicle Mission. 2023 IEEE International Systems Conference (SysCon), Apr 2023, Vancouver, France. <hal-04968937>

HAL Id: hal-04968937

<https://hal.science/hal-04968937v1>

Submitted on 27 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Applying a Model-Based Systems Engineering Approach to Model an Unmanned Aerial Vehicle Mission

Anthony Abi Badra
*Department of Complex
Systems Engineering (DISC)*
ISAE-SUPAERO
Toulouse, France

anthony.abi-badra@student.isae-supaero.fr

Ombeline Aiello
*Department of Complex
Systems Engineering (DISC)*
ISAE-SUPAERO
Toulouse, France

ombeline.aiello@isae-supaero.fr

Jean-Charles Chaudemar
*Department of Complex
Systems Engineering (DISC)*
ISAE-SUPAERO
Toulouse, France

jean-charles.chaudemar@isae-supaero.fr

Abstract—Unmanned Aerial Vehicles (UAVs) are complex systems whose complexity lies mainly in the plurality of their missions. Among aerospace vehicles with various missions (aircraft, satellite, etc.), UAVs have the advantage of being easily scalable due to their small size. Thus, the missions developed should be relatively easily implementable, raising the need for genericity. For this reason, this paper discusses multiple tools and methods used to model a UAV mission. Starting from a Simulink State flow model, a top-level mission architecture was created, and operational, functional, and logical analyses were conducted using an ARCADIA/Capella based approach.

Index Terms—Unmanned Aerial Vehicle (UAV), Unmanned Aerial System (UAS), Model-Based Systems Engineering (MBSE), mission, operational analysis, functional analysis, logical analysis

I. INTRODUCTION

Nowadays, the applications of Unmanned Aerial Vehicles (UAVs) are wide-ranging. They are used in infrastructure inspection, sensing, construction sites, maintenance, security, precision agriculture, etc. [1]. In these scenarios, the mission is complex since it exhibits a multiscale emergent phenomena, that is a result of the interaction between the UAV's components and their surroundings. To handle complexity, new techniques and tools are needed to define and perform a UAV flight. In fact, UAVs are gaining more success and are being adopted in more domains due to their increased Level of Automation (LoA) [2]. The vehicle should not only follow its pre-defined route, but should also react to external stimuli (target, threat, failure) and adapt accordingly. According to INCOSE [3], Model-Based Systems Engineering (MBSE) was introduced to gain a better understanding of a complex system. In fact, MBSE not only enhances the possibility of discovering non-obvious issues within the system, but also strengthens the solutions of design problems and provides traceability and transparency to all involved parties as the requirements and design parameters are constantly documented [4].

In this paper, we apply an MBSE approach to simulate

and model a UAV mission. Section II investigates Systems Engineering (SE) methods published in literature. Section III describes the process of UAV mission. Section IV implements an ARCADIA methodology to model a UAV mission. Section V concludes the paper and proposes future work.

II. RELATED WORK

A system, by definition, is a set of elements (people, products, and processes) in continuous dynamic interaction, organized to achieve an intended objective [5]. One of the aims of SE is to enhance the system's capabilities, understand its interdependencies, and identify and resolve the real issues that have the most significant impact on the considered system [5].

Various research works have applied a SE approach to address design challenges of systems subject to changing environments. Paper [6] proposes the use of the B language, a formal modeling language introduced by J.R. Abrial, to eliminate the ambiguities related to different interpretations of the system's architecture. The authors rely on the example of the successful implementation of Event-B on RATP Meteor subway line in Paris to prove that the B language allows an incremental development of specifications and their refinements down to an implementation. While [6] proposes a top-down approach, paper [7] adopts a bottom-up approach. The article starts by a joint SysML model based on a mechatronics systems, coupled to discipline specific models. One advantage of this approach is having all the required information in one model, ensuring consistency and the right abstraction level. However, the article does not precise the concise information required in the SysML model. Paper [8] proposes a design method that consists of a hierarchy of models, describing the system at different abstraction levels. A more generic approach is provided in paper [9], which proposes an integrated approach to management of design and engineering of complex socio-technical systems considering the system's design and engineering (technical management) and governance (organizational management). The model is composed

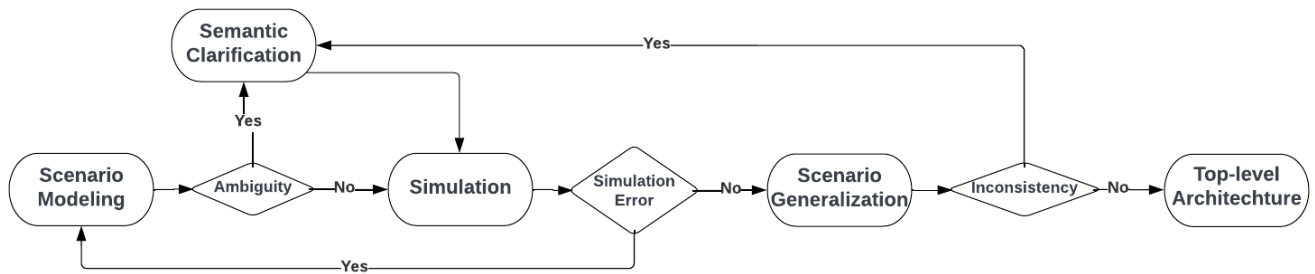


Fig. 1. Methodology followed.

of four main subsystems: requirements design, artefact design, design process coordination, and simulation, experimentation and implementation. This approach encompasses the mission statement. In fact, it captures the needs and the desires of all the involved stakeholders from which requirements will be defined and qualified. Under ARCADIA framework, [10] applies a Capella approach to model and optimize an avionics system. This approach strengthens the understanding of the system by considering it from different viewpoints: operational, functional, logical and physical viewpoints. Paper [11] also applies a viewpoint approach to address the design challenges of an inspection drone.

III. PRELIMINARY APPROACH

Based on the literature, this paper presents a bottom-up systems engineering approach, as it starts by considering a specific example of a mission then abstracts this model to reach a high-level architecture. Figure 1 depicts the methodology followed.

Starting with scenario modeling, a surveillance mission was chosen as a reference and a Simulink Stateflow model¹ was generated based on it. Stateflow is a component of Simulink that provides hierarchical state machines, similar to those of Statecharts. Before running the simulation of the model, the Stateflow was checked for any semantic vagueness that might lead to ambiguity. Then, multiple simulations were carried out to ensure that the model is functional. Having a complete architecture of the surveillance mission, this specific model was generalized in a way that ensures consistency, reusability and applicability to any UAV mission. During this process, semantic clarification was crucial as there were inconsistencies, mainly because the states were too specific to the surveillance scenario. Then, based on the abstract model, a top-level architecture was built. The following sections will present the high-level mission process as well as multiple viewpoints of the system based on the approach proposed in [9].

A. Mission Analysis

The concept of a mission can be considered as a whole, containing all the information needed to reach an intended

target [2]. A mission is carried out by executing actions: movements, interactions with the surroundings, and collection of information. It has a start and an end, between which the actions are executed. A mission also has objective(s) which can be reached in different ways or methods. Each method corresponds to a sequence of actions. The main goal of mission planning is to find the best set of actions that considers the mission environment. Among the possible solutions, the mission planner chooses the one that optimizes criteria, considering the cost and benefits, and respecting time constraints and available resources. In fact, we should define the need(s) that the mission will satisfy, the objectives that it aims to reach, the requirements that should be met and the type of application (military or civil). It is also crucial to determine the operators or the key actors. For an Unmanned Aerial System (UAS), two key actors or planners can be identified:

- external planners which could be the client/user in charge of assessing whether the mission is satisfactory or not [2], [4];
- planners inside the Ground Control Station (GCS) which are the human pilots who have the responsibility to implement the mission depending on the LoA of the system [2].

B. Mission Process

From a mission point of view, systems are intrinsically flexible and functionally overlapping. Based on the Simulink SysML model of a surveillance mission, and following the methodology shown in figure 1, a high-level mission process was created (figure 2).

In order to define the abstraction, the mission process identifies a list of relevant concepts and a list of relevant relationships between these concepts [13]. Its states are:

- Mission Initialization: This state consists of initializing the parameters that have to be checked before starting the mission. They include but are not limited to: battery level, altitude, longitude, latitude. These parameters will be updated throughout the mission.
- Mission Start: It includes take-off, reaching the first desired waypoints as well as the desired altitude.
- Mission Main Objective: The main objective of a mission depends on the mission statement. The main objective

¹<https://github.com/anthonyabibadra/UAV-Simulink-Stateflow-model.git>

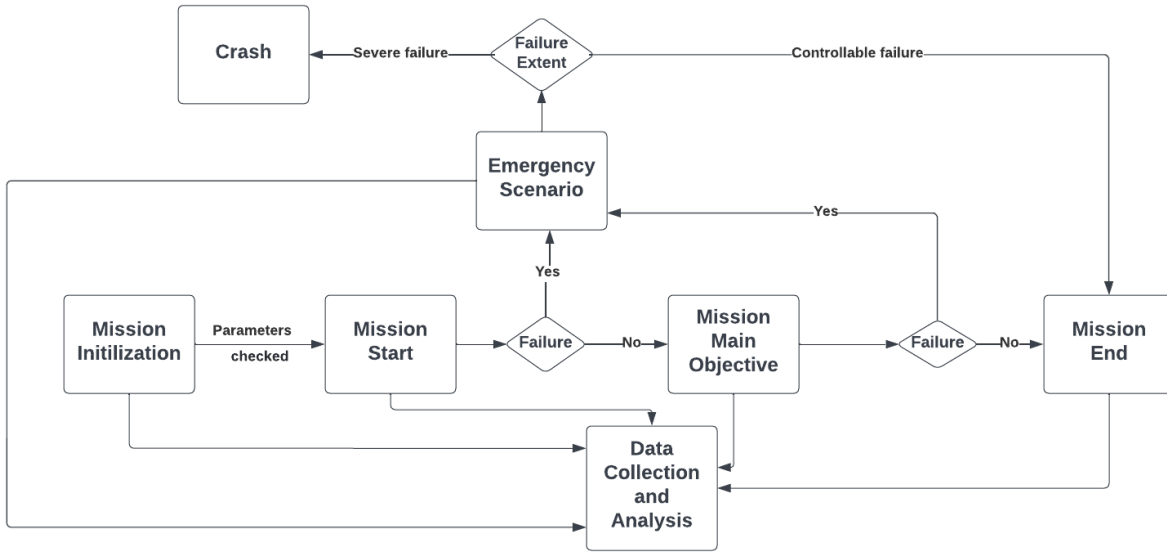


Fig. 2. High-level mission process.

quantifies the mission and is acquired from relevant stakeholders.

- **Data Collection and Analysis:** Throughout the mission, data is collected and analyzed. As the mission parameters are updated, the UAV assesses if the conditions are nominal and if it is performing within the pre-defined limits. This state is highly impacted by the LoA and includes altitude regulation, obstacle avoidance, launching emergency mode, etc.
- **Emergency Scenario:** In any system, there is a probability of failure. In the case of a UAV, there are two possibilities. If the failure is controllable, the vehicle is either flown back to its base or is landed in a site that is assessed as safe. If the failure is severe, the drone has to be crashed, or will crash uncontrollably. The decision whether the drone should be crashed or not highly depends on the LoA of the system.
- **Mission End:** This state is reached if the main mission objective is reached and/or if an emergency scenario is controllable. It includes flying the UAV back to the base, passing through specific waypoints and landing the vehicle.

IV. APPLICATION OF A MODEL-BASED SYSTEMS ENGINEERING APPROACH

In this section, an ARCADIA/Capella based approach is adopted. The Architecture Analysis and Design Integrated Approach (ARCADIA) was developed by Thales [10]. It promotes a viewpoint approach, considering four viewpoints as shown in figure 3.

Each of the layers will be discussed in the following sections. It is to be noted that this paper adopts an ARCADIA-similar approach by considering a multi-viewpoint methodol-

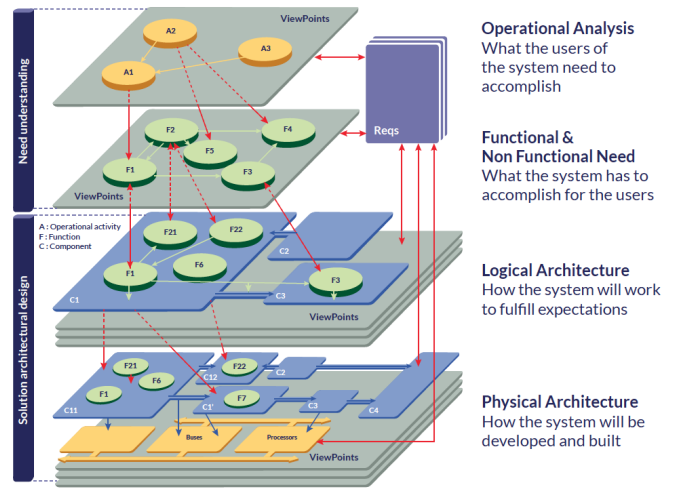


Fig. 3. ARCADIA/Capella Approach [12].

ogy. However, it also links the layers with the Stateflow model that has been created.

A. Operational Analysis

An operational analysis aims to describe the mission actors, mission activities, as well as operational scenarios [10]. The Operational Architecture diagram (figure 4) expresses the stakeholders needs through activities. The operational activity ‘Initialize Mission’ expresses the need for a UAV mission to meet certain objectives. Once defined, the mission plan is sent to the UAV, which executes the mission. As the UAV collects and analyses information, it sends the data to the Ground Station which monitors the mission.

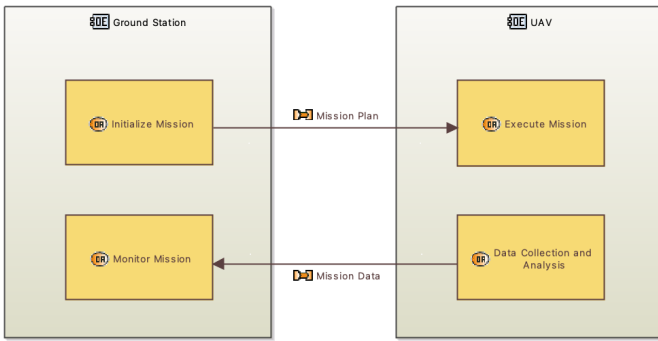


Fig. 4. Operational Architecture diagram.

Outside the scope of the Capella Approach, based on the Simulink Stateflow model and the high-level mission model, eight main operational modes were identified. Table I shows these operational modes, which summarize all the possible behaviors of the system from the mission start to its end.

TABLE I
OPERATIONAL MODES

ID	Operational Mode	Description
OM01	Power ON	The system is powered on
OM02	Power Off	The system is powered off
OM03	Initialization	The system checks the parameters before the mission starts
OM04	Optimal	The system is performing as expected
OM05	Stand-by	The system is waiting for orders to be sent by the Ground Station
OM06	Emergency	The system encountered an emergency that limits one or many of its capabilities
OM07	Fly-to-base	The mission is ended and the drone flies back to the base
OM08	Crash	The drone encountered a severe failure and has to be crashed

To illustrate the capabilities of the UAV in case of emergency, an Operational Capabilities diagram was generated using Capella. It is shown in figure 5. After defining the activities and allocating them to actors, an Operational Entity Scenario was created to further explain the behavior of the UAV in case of emergency (figure 6).

B. Functional Analysis

A system’s functional analysis breaks down a system’s mission into a hierarchy of functions [10]. It articulates how a system operates and what it does to satisfy the stakeholders’ requirements. The target is to study the functions based solely on the needs and the interactions with the different actors. The advantage of this approach is to depict issues that might have been ignored, either in oversight or deliberately [11]. Figure 7 shows the functional analysis of a UAS.

The high-level functions are the following:

Ground Station functions:

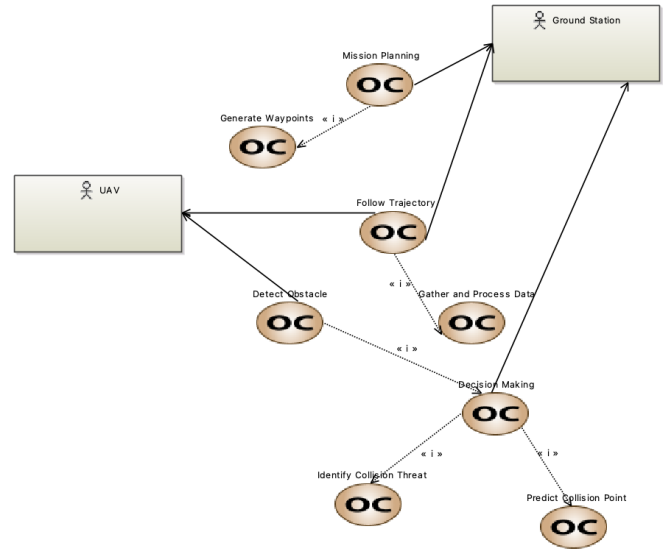


Fig. 5. Operational Capabilities diagram for emergency scenario.

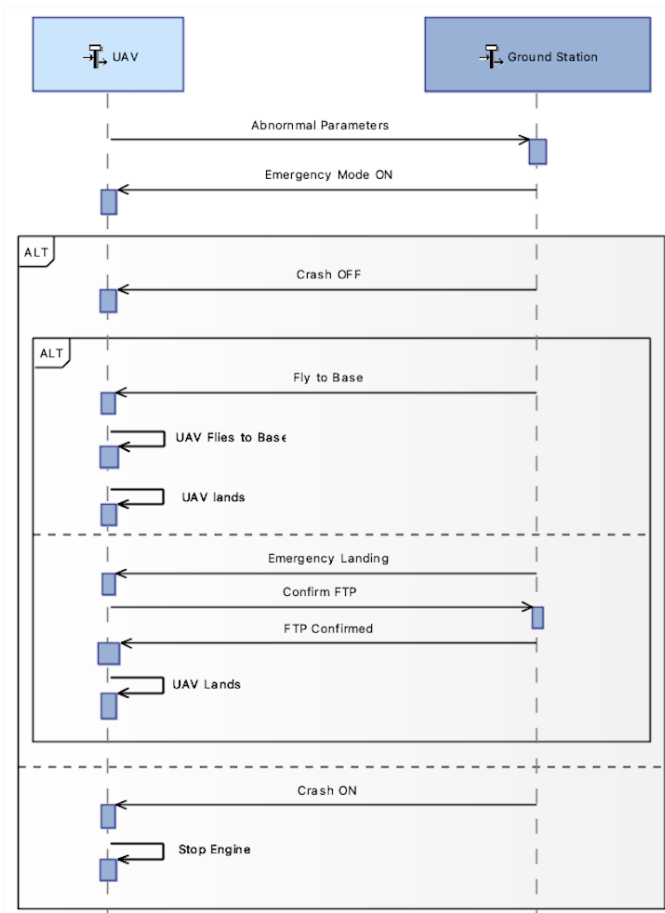


Fig. 6. Operational Entity Scenario diagram for emergency scenario.

- Define Mission: Prior to the mission start, certain parameters such as pre-defined waypoints and the expected limits within which the UAV is expected to operate are

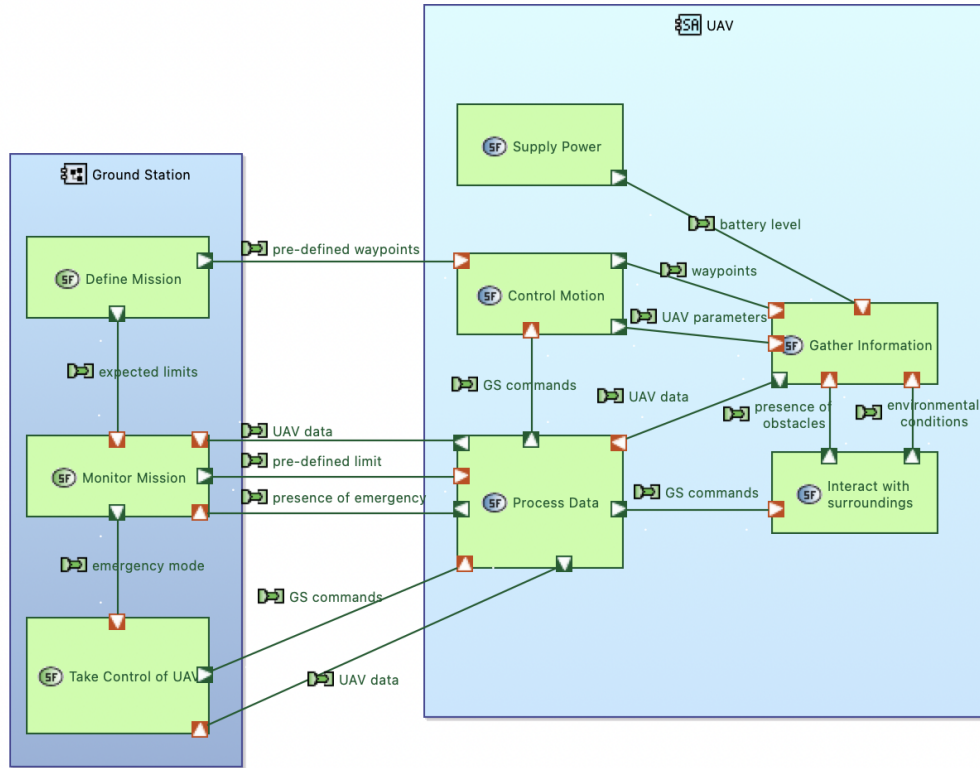


Fig. 7. Functional analysis of a UAV mission.

defined by the Ground Station.

- **Monitor Mission:** Throughout the mission, the Ground Station constantly receives data from the UAV, and ensures that the mission is running as expected. This data includes the current waypoints, the battery level, the speed, the altitude, the environmental conditions, the presence of obstacles, etc. In addition to that, the Ground Station constantly shares with the UAV the pre-defined limits or the expected behavior. If the UAV faces an emergency, it reports it back to the Ground Station.
- **Take Control of UAV:** In case of emergency, the Ground Station takes full control of the UAV. The mission is directly run by an operator, and all the autonomous functions of the UAV are paused, as the UAV is controlled by Ground Station commands (GS commands in figure 7).

UAV functions:

- **Supply Power:** the power supply provides power to the system.
- **Gather information:** the UAS gathers information from its surroundings and constantly tracks its own parameters (current waypoints, the speed, the altitude, the environmental conditions, the presence of obstacles, etc.) as well

as its battery level.

- **Process Data:** the system analyses the collected information and compares it with the pre-defined nominal limits set by the Ground Station in order to depict any problem that might arise. Even for highly automated or autonomous UAV, the system sends data to the Ground Station and receives commands from it.
- **Control motion:** the system controls the thrust, roll, yaw, pitch, speed, etc. to allow movement and to enable a successful implementation of the mission.
- **Interact with surroundings:** the UAV adapts accordingly to its operational environment and considers weather conditions and obstacles that it might face.

C. Logical Analysis

The logical analysis breaks down the system into logical elements that form a system architecture [11]. The functional diagram and the logical diagram are interrelated, as each function is associated to one or many high-level logical elements. Figure 8 depicts the logical architecture of a UAV. In this paper, we define the logical components or logical sub-systems as the following:

UAV logical components:

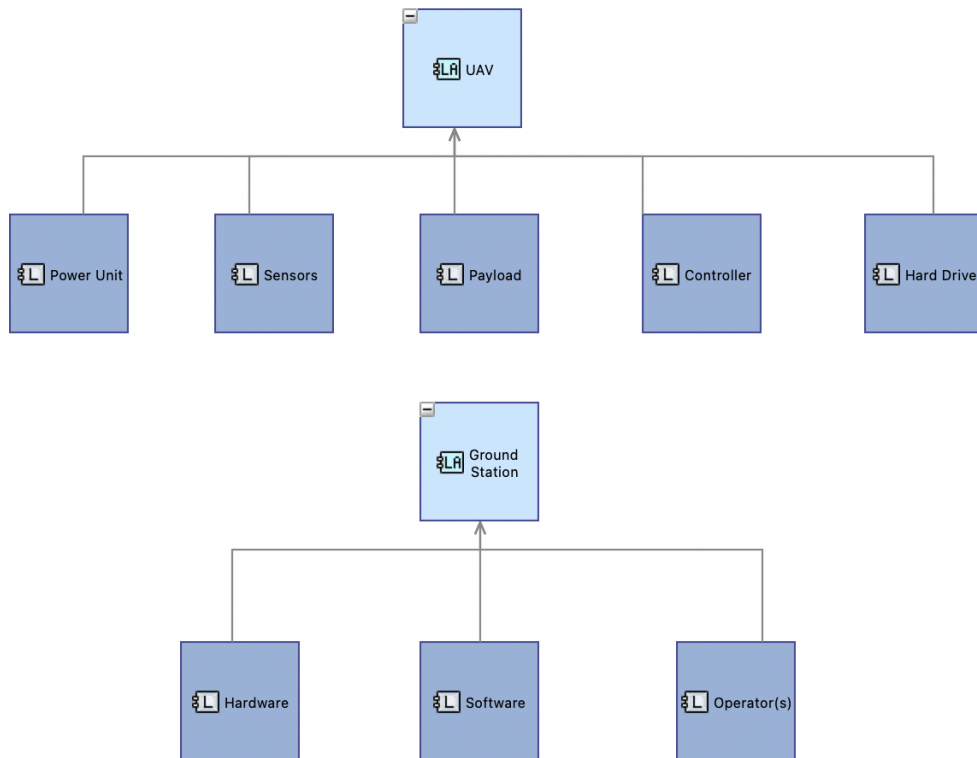


Fig. 8. Logical Component Breakdown diagram of a UAV.

- Power unit: it supplies the system with power.
- Sensors: sensors are used to measure altitude, vertical distances, linear acceleration, angular rate, speed, horizontal motion, etc.
- Payload: it provides the space and mounting location for the other UAV components.
- Controller: it is the ‘brain’ of the vehicle and controls its dynamics.
- Hard drive: during a normal flight, a significantly large amount of data is collected. The hard drive stores and communicates this data to the Ground Station.

Ground Station logical components:

- Hardware: the Ground Station’s hardware controls the UAV. It includes but is not limited to tracking and communication antennas as well as a Human-Machine Interface.
- Software: used for guidance, the creation and execution of the flight plan, the generation of waypoints, and the adding of new data without repeating the entire mission.
- Operator: an operator is an individual present in the Ground Station. The role of the operator starts at the early stage of the mission. In case of emergency, it is the operator that has the decision-making role.

D. Physical Analysis

From a high-level design view, logical components are stable in time. Based on these components, multiple physical architectures can be considered using different technologies. The aim of this paper is to propose a general approach to model a UAV mission. Hence, specific technologies for UAVs will not be discussed, and the physical architecture will not be considered.

V. CONCLUSION & FUTURE WORK

With the increasing complexity and heterogeneity of complex systems, model-based design approaches tend to derive new and unnoticed information within the system. This paper presents an integrated bottom-up approach to simplify and gain a better understanding of a UAV mission. Starting from a concrete Simulink Stateflow model, a mission model was created and both global and specific views of the UAV mission were investigated. Through the presentation of all the relevant information, this approach fosters reusability as it can be integrated to new project-model on the respective level. The MBSE approach presented in this paper is successful in modeling a UAV mission. Throughout an early-design phase, the main systems engineering processes for this modeling involve structuring steps from the identification of operational modes and scenarios towards the definition of a logical architecture. However, it is crucial to recognize that the full

picture of the UAV mission is not complete without taking into account safety and security aspects. Safety involves ensuring the UAV system operates without causing harm to people, property, or the environment. Security, on the other hand, involves protecting the UAV and its data from unauthorized access, theft, or damage. Both of these aspects are critical for the successful implementation and operation of UAV missions and require further attention in future studies [15], [16].

Future work can also include an operational formal semantics verification of the Stateflow model. One possible tool is CoCoSim which is an automated analysis and compilation framework for Simulink/Stateflow [14]. Model-based Ontology Coverage Analysis (MOCA) can be used to assess the overlap between the system's architecture model and a given ontology, which will result in a standardized mission architecture with less ambiguity, enhanced models, and smoother stakeholders' engagement and communication. In addition to that, the modes of the UAS can be investigated. As for the functional analysis, further work can investigate which function or functions need to be active in each state of the mission. A functional hierarchy can be built, specifying the order of execution of the functions.

ACKNOWLEDGMENT

This work was supported by the Defense Innovation Agency (AID) of the French Ministry of Defense (research project CONCORDE N° 2019 65 0090004707501).

REFERENCES

- [1] J. A. Besada, L. Bergesio, I. Campaña, D. Vaquero-Melchor, J. López-Araquistain, A. M. Bernardos, and J. R. Casar, "Drone mission definition and implementation for automated infrastructure inspection using airborne sensors," MDPI, 11-Apr-2018. [Online]. Available: <https://doi.org/10.3390/s18041170>.
- [2] L. Damilano, G. Guglieri, F. Quagliotti, I. Sale, and A. Lunghi, "Ground control station embedded mission planning for UAS - Journal of Intelligent & Robotic Systems," SpringerLink, 04-Aug-2012. [Online]. Available: <https://link.springer.com/article/10.1007/s10846-012-9697-2>.
- [3] "Model-based systems engineering (MBSE)," Guide to the System Engineering Body of Knowledge. [Online]. Available: [https://www.sebokwiki.org/wiki/Model-Based_Systems_Engineering_\(MBSE\)](https://www.sebokwiki.org/wiki/Model-Based_Systems_Engineering_(MBSE)).
- [4] Q. Do, S. Cook, P. Campbell, W. Scott, K. Robinson, W. Power, and D. Tramoundanis, "Requirements for a metamodel to facilitate knowledge sharing between project stakeholders," Procedia Computer Science, 06-Mar-2012. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1877050912000609>.
- [5] J. de Rosnay, The macroscope: A new world scientific system. New York: Harper & Row, 1979.
- [6] J.-P. Bodeveix, A. Dieumegard, and M. Filali, "Pattern-based requirements development," HAL Open Science, 14-Feb-2018. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01708993>.
- [7] A. A. Shah, A. A. Kerzhner, D. Schaefer, and C. J. J. Paredis, "Multi-view modeling to support Embedded Systems Engineering in SysML," SpringerLink, Jan-2010. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-642-17322-6_25.
- [8] L. Bassi, C. Secchi, M. Bonfe and C. Fantuzzi, "A SysML-Based Methodology for Manufacturing Machinery Modeling and Design," in IEEE/ASME Transactions on Mechatronics, vol. 16, no. 6, pp. 1049-1062, Dec-2011, doi: 10.1109/TMECH.2010.2073480.
- [9] F. Brazier, P. van Langen, S. Lukosch, and R. Vingerhoeds, "Complex Systems: Design, engineering, governance," TU Delft Research Portal, 2018. [Online]. Available: <https://research.tudelft.nl/en/publications/complex-systems-design-engineering-governance>.
- [10] L. Batista and O. Hammami, "Capella based system engineering modelling and multi-objective optimization of avionics systems," 2016 IEEE International Symposium on Systems Engineering (ISSE), 2016, pp. 1-8, doi: 10.1109/SysEng.2016.7753127.
- [11] E. Razafimahazo, P. De Saqui-Sannes, R. A. Vingerhoeds, C. Baron, J. Soullax and R. Mège, "Mastering Complexity for Indoor Inspection Drone Development," 2021 IEEE International Symposium on Systems Engineering (ISSE), 2021, pp. 1-8, doi: 10.1109/ISSE51541.2021.9582483.
- [12] Capella MBSE Tool - Arcadia. [Online]. Available: <https://www.eclipse.org/capella/arcadia.html>.
- [13] J.-M. Favre, "A story of words: From MED to MDE via model in five millenniums: Semantic scholar," 2006. [Online]. Available: <https://www.semanticscholar.org/paper/A-story-of-Words>
- [14] "CoCoSim," CoCoSim - Automated analysis framework for Simulink/Stateflow. [Online]. Available: <https://cocoteam.github.io/cocosim/>.
- [15] J.-C. Chaudemar and P. De Saqui-Sannes, "MBSE and MDAO for early validation of design decisions: a bibliography survey," 2021 IEEE International Systems Conference (SysCon), 2021, pp. 7-8, doi: 10.1109/SysCon48628.2021.9447140.
- [16] O. Aiello, O. Poitou, J.-C. Chaudemar and P. De Saqui-Sannes, "Sizing a Drone Battery by coupling MBSE and MDAO," 11th European Congress Embedded Real Time Systems (ERTS) 2022, 2022. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-03740634>.