COLOSSUS EU Project – Collaborative SoS Exploration of Aviation Products, Services and Business Models: Overview and Approach

Prajwal Shiva Prakasha[1], Nabih Naeem[1], Kristian Amadori[2], Giuseppa Donelli[1], Jasamin Akbari[1], Fabrizio Nicolosi[3], Ludvig Knöös Franzén[4], Manuela Ruocco[5], Thierry Lefebvre[6], and Björn Nagel[6]
[1]Institute of Systems Architectures in Aeronautics, German Aerospace Center, Germany
[2]SAAB Aeronautics, Sweden
[3]University of Naples UNINA, Italy
[4]Linköping University, Sweden
[5]SmartUp Engineering, Italy
[6]French National Aerospace Research Centre ONERA, France

Abstract

The COLOSSUS EU Project is developing a System of Systems design framework and methodology which for the first time will enable the combined optimization of aircraft, fleet, operations and business models. To develop and test the framework two contrasting use cases are considered

1) Sustainable Intermodal Mobility (denoted “ADAM”) and
2) Aerial Wildfire-Fighting (denoted “EVE”)

The two-use cases pose diverse and unique requirements on the framework and thereby aim to rekindle the research approach. This paper provides an overview of the COLOSSUS EU project including its objectives, approach, developed tools and the enabling framework. The project takes a holistic view on aviation product design considering the overall System of Systems to derive insights into the vehicle design and their operations and business models. The multi-level approach employed spans the Subsystem (e.g. propulsion system), Constituent System (e.g. aircraft), System of Systems (e.g. fleet, ConOps), and Business Model levels (Stakeholder Models, Value Functions). Agent-Based Simulations are used to model and evaluate the System of Systems, interlinking with the other levels. Business Models, Life Cycle Costing, and Life Cycle Analysis methods are used to enable a holistic assessment of the SoS considering various stakeholder interests. The project also uses Reinforcement Learning approaches to explore the SoS operational design space.

The developed methods and tools will be openly published in order to foster exploitation for System of Systems approaches in research and industry. This research paper will highlight the framework development, and will present exemplary results and insights from the coupled framework bringing together Aviation Products – Operations – Digitalization and Business Models.

Keywords: System of Systems, Collaborative Aircraft Design, Sustainability, Intermodal Mobility, Aerial Wildfire, Agent Based Simulations

Abbreviations

ABS:	Agent-Based Simulation	**ConOps:**	Concept of Operations
CPACS:	Common Parametric Aircraft Configuration Schema	**CS:**	Constituent System
DE4SoS:	Digital Environment for SoS	**eVTOL:**	electric Vertical Take-Off Landing
MDA:	Multi-disciplinary Design Analysis	**SoS:**	System of Systems
SoSID:	SoS Inverse Design	**SUMO:**	Simulation of Urban MObility
TDCF:	Transformative Digital Collaborative Framework	**TLAR:**	Top-Level Aircraft Requirements

1. Project Overview & Objectives

1.1 Overall Concept

System of Systems (SoS) are an interoperating collection of component systems, denoted as constituent systems, that produce results which cannot be produced by the individual systems alone. Product design with a holistic SoS approach faces two challenges which go beyond the usual and well-understood practices of systems engineering:

1. Additional domains have to be investigated and understood to benchmark their impact and capabilities concerning the SoS design space exploration.
2. Important design parameters are representing non-technical design criteria and are “softer”, or “fuzzier”, and they possess a higher level of uncertainty.

COLOSSUS employs a multi-level approach to identify needs, capabilities, and system requirements in the initial phases of the holistic product development process and to cover them appropriately throughout the development circle. The levels are: (I) business models, (II) SoS design space, (III), constituent systems design space, and (V) subsystem design space (Figure 1).

The expected outcomes of COLOSSUS will provide the aviation sector with a platform to develop new and breakthrough products and technologies in a holistic SoS approach. The approach is developed through the direct application on two distinct use cases with contrasting requirements: Aerial Wildfire Fighting (denoted EVE) and Sustainable Intermodal Mobility (denoted ADAM).

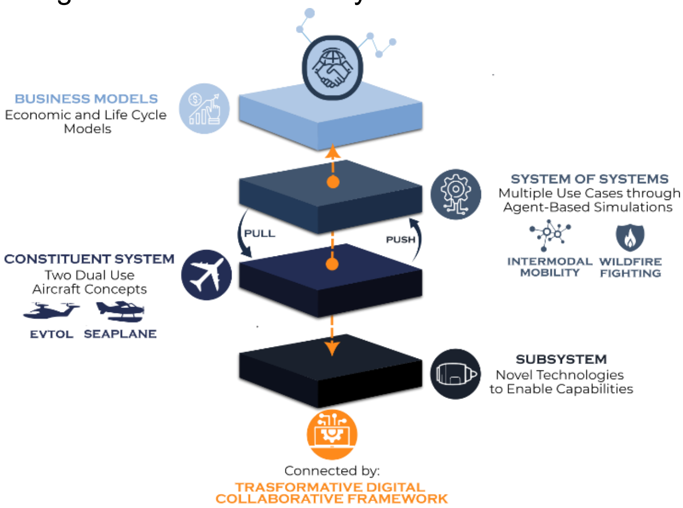

Figure 1 Multi-level approach for the holistic design of aviation systems

The approach employed in COLOSSUS is depicted in Figure 2. Firstly, the problem setup is defined consisting of the identification of stakeholders, needs, and the stakeholder key value indicators. In addition, a set of scenarios frame the problem ensuring the goals of the use case are represented. With the problem and evaluation criteria defined through the scenarios and key value indicators respectively, the solution space can be explored. The aircraft design capabilities are developed considering aircraft type, architecture, subsystems and technology. The ability to rapidly explore broad design spaces are emphasized in this respect.

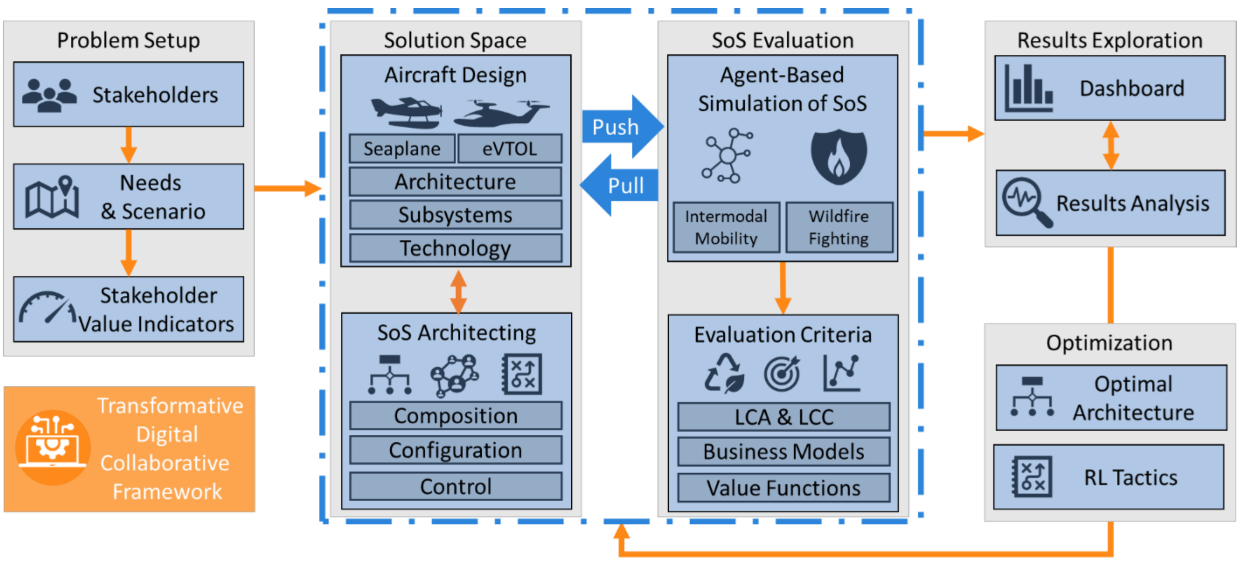

Figure 2: COLOSSUS Project Overview

The designed aircraft, and other constituent systems can then be composed into a SoS Architecture. In addition to defining the constituent systems in the SoS, architecting also defines how the systems are used and how they are controlled. The evaluation of the SoS constellations are achieved through the use of the Agent-Based Simulations developed within the COLOSSUS for each use case. Based on **the Product Push** or **Product Pull paradigm**, the Seaplane and AAM Aircraft are designed with novel technologies for best performance driven by a simulation-embedded SoS approach. Post design, the aircraft performance parameters are modelled as agents in the simulation, which mimics the real-world Concept of Operations (ConOps) of aircraft considering mission performance. Furthermore, multiple agents (aircraft) are combined as fleet operations considering airport/vertiport, Air Traffic Management, trajectory operations and tactics. Thus, the Agent-Based Simulation provide an efficient way to evaluate not just the aircraft level, but also the fleet level operations. As analytical methods cannot provide correct evaluation of aggregated fleeting and combinatorial effects, the framework embeds an agent-based simulation to evaluate. For the mobility use case, intermodal operations are considered with road, rail and long-distance civil aviation models. For this purpose, existing models and data are used. For the wildfire use case, the individual performance of ground-based firefighting units as well as fire-fighting aircraft are modelled in the simulation. Various combinations, using a Design of Experiments (DoE) approach with varied performance of individual vehicles, are analysed for an optimised combination and interaction of firefighting resources (air & ground). Cost and lifecycle assessments are performed for the fleet under consideration of performance metrics of aircraft, other vehicles, aggregated fleet performance and other infrastructures. These serve as additional evaluation modules as post processing blocks after the Agent-Based Simulation to compute the Key Value Indicators defined in the problem setup phase. These characteristics and parameters are then processed in the development and evaluation of the business model.

A SoS assessment & exploration dashboard for all stakeholders will communicate technical information and support the comprehension of complex contexts and their inherent interplay. A dedicated dashboard will support exploration of the available design space, intuitive understanding and multi-criteria decision making. A version of the dashboard will be made publicly available. Finally, the evaluation loop is closed by the introduction of optimization approaches to identify the optimal SoS Architecture. In addition, reinforcement learning approaches are exploited in the EVE use case to identify optimal tactics for the firefighting aircraft.

The aforementioned approach is enabled by an MBSE-driven Transformative Digital Collaborative Framework (TDCF) for the complex multilevel vehicle design and simulation-driven SoS analysis. The TDCF seamlessly connects the individual models in the COLOSSUS project, in addition to adding requirements tracking of the stakeholders and proposed solutions, and data visualization and decision making.

Main technical objectives of COLOSSUS

The main technical objectives of the COLOSSUS project

1. To create a Transformative Digital Collaborative framework (TDCF) that allows European aviation to perform research, technology development and innovation in a holistic SoS approach. The TDCF shall support modelling, analysis, optimisation and evaluation of complex products and services under consideration of real-world conditions.
2. To expand and test the capabilities and performance of the TDCF with two different Use Cases presenting two different SoS problems, both of which address needs identified by the European Commission and thus possess a value of their own:
    - ***Sustainable Intermodal Mobility*** (**ADAM)**: Creating a business model for sustainable 4D-intermodal mobility and evaluating the concept for performance, competitiveness, environmental impact and life cycle footprint.
    - ***Aerial Widlfire Fighting (*** **EVE)**: Developing an integrated fast-response approach for fighting wildfires by combining latest developments in the fields of aircraft design and Subsystems, Concepts of Operations, AI and digitalisation.
3. To perform conceptual studies for two products which could transverse technology enablers for

multi-modal mobility and wildfire fighting: a multi-role seaplane with hybrid propulsion and a product for eVTOL-based advanced air mobility of passengers and goods.

These objectives at a project level can be further broken down into technical objectives at a domain level. As the COLOSSUS project spans numerous domains due to the nature of SoS, The technical objectives can also relate to the integration of several domains to achieve a specific objective. The following list refines the aforementioned objectives, into the domain level technical objectives:

- TO1. Formulate an approach to quantify SoS effectiveness considering Stakeholder Key Value Indicators
- TO2. Architect SoS constellations and their behaviours
- TO3. Explore the ConOps of the SoS using Reinforcement Learning
- TO4. Develop Agent-Based Simulation capabilities that enables the evaluation of different SoS architectures against the mission success and stakeholder satisfaction criteria
- TO5. Develop Aircraft Design capabilities for SoS driven design approach
- TO6. Develop an interactive dashboard to enable effective and interactive result communication with stakeholders and decision makers
- TO7. Develop fast and broad Life Cycle Analysis and Life Cycle Cost models for SoS
- TO8. Develop a digital environment to aid in the formulation and solution of SoS problems while ensuring traceability

2. SoS Approach in COLOSSUS

This chapter focuses on the methodology employed in the COLOSSUS project to achieve the technical objectives highlighted in Chapter 1.

2.1 Setting up an SoS Problem

The “COLOSSUS methodology” is a capability-focused approach to model a complex, multi-layered SoS. It is basically a stakeholder-/user-centric process to decompose the “real world” so that a corresponding SoS model can be composed.

The SoS approach objectives are:

- To identify the “real questions”: When setting up an SoS, it is easy to get lost in the details. It is the “art of SoS design” to follow the paradigm of Antoine de Saint-Exupéry: “Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away.”
- To get the answers to these questions fast: The purpose of an SoS is not to deliver the one and only “correct result”, but to explore the design space and investigate the respective operational capabilities of the included new technology, product or service.
- To grow the SoS with time and experience: The SoS has to be designed in an agile approach that allows to increase the complexity of the SoS model, e.g. by adding more systems, more detail and more interconnections, as work progresses. In other words, allow for an evolutionary development.
- To be transparent and inclusive: SoS design is teamwork. Every team member, at each hierarchical level and in every disciplinary branch, should be able to understand the basic composition of the SoS and how it interrelates with his or her work.
-

COLOSSUS uses a sequential process that starts from the actual field of application where the new system (e.g., innovation, product or service) has to exercise its specific capabilities: the use cases which define the purpose and the environment in which the new system will operate. For each use case, the relevant stakeholders and their specific needs are then identified. Stakeholders can vary in the nature of their needs or the “stake” they hold in the SoS. As an example, a stakeholder with a financial interest with the SoS would be concerned with profit generation, whereas a traveller may be more interested in time savings they can obtain from the SoS, and a regulator may be more concerned about the environmental and societal impact of the SoS. The needs can be translated into Key Value Indicators (KVIs) of the stakeholders in relation to the SoS, which quantify numerically the stakeholder’s interests in the SoS. In addition, any requirements of the stakeholders that can be imposed on the SOS are also identified, and used to identify viable SoS constellations. In order to be able to evaluate an SoS Constellation's effectiveness in fulfilling the objectives of the SoS, a global function is composed from the identified KVIs, referred to as the Value Function.

In sum, the process of setting up an SoS Problem is constituted of two major steps: defining the stakeholders and scenario, and formulating the Value Function to quantify the effectiveness of the SoS in fulfilling the objectives. The former defines the scope of the problem to solve, and the latter provides the metrics to identify the solution. Interested readers are directed to [1] for a more detailed description of the methodology employed in setting up an SoS Problem.

2.2 The Product Push Paradigm

The Product Push Paradigm is shown in Figure 3 for the Sustainable Mobility use case. The product push paradigm is appropriate for cases where the business model is sought for a known product (e.g. how to enable a market and business sustainably when a sea plane product is introduced to intermodal transportation SoS). In COLOSSUS, the product push paradigm is applied for the ADAM use case where an intermodal business model is sought for the eVTOL and Seaplane aircraft designs. The product push paradigm can also be seen as the bottom-up approach in product development and SoS engineering.

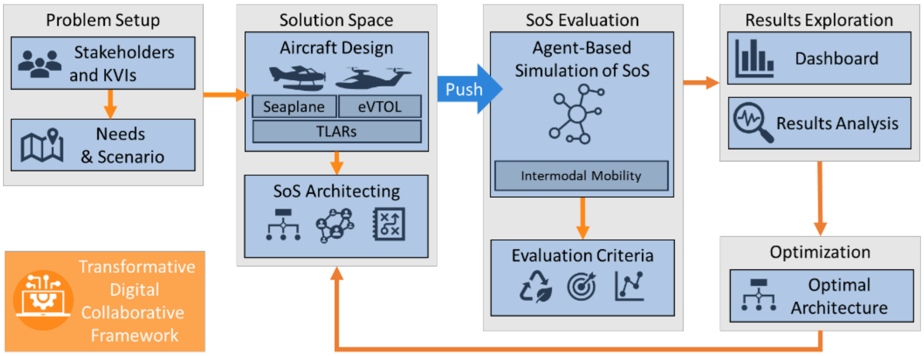

Figure 3 Product Push Paradigm for ADAM Use Case

The focus of the Product Push paradigm, is to identify how to operate the product and maximize the value added by it from an SoS perspective. The quantification of the "goodness" of an SoS is governed by the Value Function as described in Chapter 2.1. Within the Product Push paradigm, the operation of the two products (eVTOL and Seaplane) are in focus, and as such the primary stakeholders and enabling stakeholders are also of crucial importance. The stakeholders are those associated with the System of Interest (Aircraft), namely the Vehicle Operators, and the Vehicle Manufacturers. The enabling stakeholders here are defined as the stakeholders which have an active role in the SoS (i.e. are crucial for the SoS to exist), and are not the stakeholder associated with the System of Interest (Aircraft). Namely, these are the vertiport and seaport operators, air traffic management, passengers, ground/sea transport operators, and Mobility as a Service providers. The parameters associated with the Primary Stakeholders, are of primary interest to investigate and optimize in the product push paradigm. These can be the ticket pricing structure, the fleet size and fleet mix (composition of different vehicle types), routes to operate between, vehicle-passenger assignment strategy, and operational strategy (on-demand vs scheduled).

2.3 The Product Pull Paradigm

The Product Pull Paradigm is shown in Figure 4 for the Wildfire use case. In this Paradigm, the objective is to develop the product according to market needs or given scenario (e.g. Wildfire). In COLOSSUS, the products developed through the Product Push Paradigm are evolved to suit the market needs of the EVE use case. For this the top-level requirements of the vehicle are to be evolved based on the Value Function of the firefighting scenario.

The SoS problem setup following the methodology in Chapter 2.1 are to be solved through leveraging Agent-Based Simulations to aid in the evaluation of the SoS Constellations.

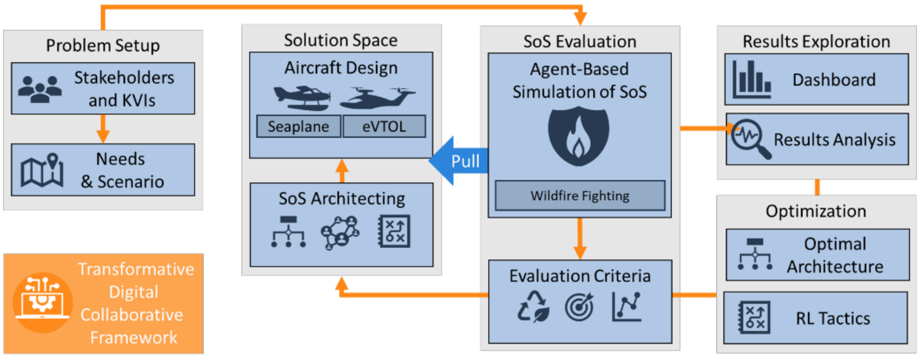

Figure 4 Product Push Paradigm for EVE Use Case

The desired outcome of the product pull paradigm is to derive the optimal Top-Level Aircraft Requirements (TLARs) for the aerial assets in the firefighting fleet from an SoS context. A crucial aspect of any SoS are the ConOps of that SoS; the firefighting effectiveness of the SoS is highly related to how the available assets are utilized. Therefore, in this paradigm the ConOps and the Assets themselves are evolved to find the optimal constellation. As aforementioned, optimality of the constellation is measured with respect to the Value Function defined for the firefighting scenario. In order to be able to measure the effectiveness of the aircraft designs with respect to the Value Function, the aircraft designs are modelled in the Agent-Based Simulation using their performance characteristics. This allows for the SoS effectiveness of the aircraft designs to be captured considering the fleeting and ConOps decisions. The ConOps exploration, as considered within the project are detailed further in Chapter 12.5.

To explore the aircraft design space and evaluate it holistically at an SoS level, several aspects must be dealt with care. Firstly, the variables to investigate at an SoS level must be chosen. Those variables of interest must then be dealt with as a fixed input to the aircraft design workflow so that the design space exploration approaches (Chapter 2.5) can be used with those variables. In COLOSSUS the primary variables of interest were chosen to be the TLAR such as payload, design cruise speed and design range. Additional variables were also chosen for each architecture, such as the hybridization factor for Seaplanes and number of rotors for the eVTOL. The remaining variables can be optimized at the aircraft design level, to obtain the optimal designs corresponding to the desired TLARs. Through this approach a set of designs can be obtained with TLAR sweeps, which can then be evaluated holistically at the SoS level through the Agent-Based Simulations. In order to embed the designs into the Agent-Based Simulation, several coupling parameters are identified. These are primarily the aircraft performance parameters and additionally use case specific parameters such as the field length required for a water scooping manoeuvre. Once the parameters have been identified, the coupling with the ABS can occur in several ways such as a database of aircraft parameters, or surrogate models. A critical question in relation to the coupling of aircraft design tools and Agent-Based Simulations is the that of the “correct” fidelity level. While in the short term, an educated “guess” of the correct modelling fidelity of the aircraft are used considering that of the entire ABS, it is of interest to perform investigations of the effect of different modelling fidelities on the SoS evaluations.

Through the exploration of the SoS design space, optimal TLARs of the CS can be derived based on the objectives of the SoS as quantified by the Value Function.

2.4 Simulating SoS with Agent-Based Simulations

The Agent-Based Simulation of the SoS serves as a testbed to measure the effectiveness of an SoS Constellation in fulfilling the scenario objectives in the scenario conditions as defined in the scenario definition. Agent-Based Simulations allow for the investigation of the components in an SoS Architecture, from the number and composition of constituent systems, the concept of operations, to the degree of control over the constituent systems (i.e. who takes decisions and to what extent). In addition, Agent-Based Simulations are also a suitable match for Reinforcement Learning methodologies allowing for the agents to learn the optimal operational tactics. The aforementioned factors make Agent-Based Simulations a very suitable tool for the SoS evaluations.

Simulating a SoS in an ABS consists of modelling the key actors of the SoS in the ABS while considering the ConOps of the actors especially the interactions between the actors, and the processes performed by each actor. The role and interaction of each actor must be carefully considered and modelled in the Agent-Based Simulation as it would be in the SoS. Within the boundaries prescribed by an agent’s role, their processes can be prescribed by chaining together tasks and conditionals. Moreover, the environment of the SoS also has to be recreated with sufficient fidelity to be able to capture any stakeholder-environment interactions.

For both the Aerial Wildfire Fighting (EVE) and Sustainable Intermodal Mobility (ADAM) use cases, this approach is taken and further details are provided in Chapter 4.2. The Agent-Based Simulation environment called the SoSID Toolkit developed by DLR [2–4] is extended in COLOSSUS to model the scope of both the use cases.

2.5 Design Space Exploration and Optimization

SoS design space exploration involves the exploration of the constituent systems, the concept of operations (ConOps), and its governance. In COLOSSUS the primary focus is placed on the constituent systems and the ConOps. The design space exploration can be accomplished in a number of ways. This includes design of experiments, architecture generation, and machine learning algorithms. Within the context of the project all of the aforementioned methodologies are employed for different purposes. The design of experiment offers the simplest path to understand the model through capturing its overall behaviour, although at the cost of runtime. Architectural generation allows for enumeration through complex, inter-linked and multi-layered design spaces and are suitable for multi-level design spaces as in COLOSSUS. The ConOps design space is considered in two parallel branches within the project: ConOps enumeration and ConOps exploration. ConOps enumeration refers to the enumeration through a set of pre-defined ConOps which sample the ConOps design space. This involves prescribing different sets of logic to the actors involved in the SoS and modelling them in the SoS Simulation. The ConOps exploration refers to the unbounded exploration of the ConOps design space using Reinforcement Learning methods. This amounts to having the agents in the SoS Simulation *learn* the best ConOps to use. As SoS missions are highly cooperative in nature involving multiple Constituent Systems, the behaviour of the CS plays an important role in the overall success. As such, to find the best overall SoS, both the CS and the ConOps must evolve together. The aerial wildfire fighting (EVE) use case presents an ideal platform for such investigations In the COLOSSUS project.

SoS optimization can be formalized with an MDAO [5] (Multi-disciplinary Design Analysis and Optimization) approach with some specificities that make it more complex than usual system optimization. For instance, the problem will be multi-level with strong coupling between SoS level and System level (agent design), of intermediate dimension in terms of variables (up to 100), with mixed variables (including categorical choices) and with several objectives (linked to stakeholder KPIs) and constraints. In addition, the solvers involved (SoS simulation for instance) will hardly provide derivative information and will be costly to operated. To increase the complexity, architectural choices as well as CONOPS parameters (for enumeration branch) are bound to be part of optimisation variables.

In order to efficiently solve the global SoS problem, several characteristics need to be specified, namely, the variables and their nature (local, global and coupling ones), the objectives functions and the constraints to handle. In the frame of the project, the retained approach has been to start focusing on the SoS level and increase progressively the complexity of the variables to handle and to the coupling process with the System level. For instance, for the EVE use case, from SoS level, the variables could be the number of each agent, their distribution on each base, the choice of agents among a catalogue provided by the System level and the choice of fire attack strategy. Regarding the objectives functions, they would be selected among the stakeholder's key value indicators and a possible constraint could be a limit on the burnt area. The output to System level would be the best agent and its updated TLAR. To efficiently tackle this problem, the selected optimization process is Super-Efficient Global Optimization coupled with Mixture Of Experts (SEGOMOE) [6]. The algorithm is based on a sequential enrichment approach, typically the Efficient Global Optimization algorithm [7] or Super EGO an evolution of EGO to handle constraints [8]. Some recent developments have been made to consider highly nonlinear constraints [9], mixed integer variables [10] and multi-objective applications [11].

3. Transformative Digital Collaborative Framework

The COLOSSUS project targets the development of new Transformative Digital Collaborative Framework (TDCF) that aims at accelerating, improving and streamlining the development of complex SoS. The Framework consists of:

- *Models*: abstraction of an SoS, CS
- *Processes*: logical sequence of tasks performed to achieve a particular objective
- *Methods*: techniques for performing a task
- *Tools*: mean implementing the method to perform the tasks of a process
- *Glossary*: reference material that provides definitions or explanations of terms or words

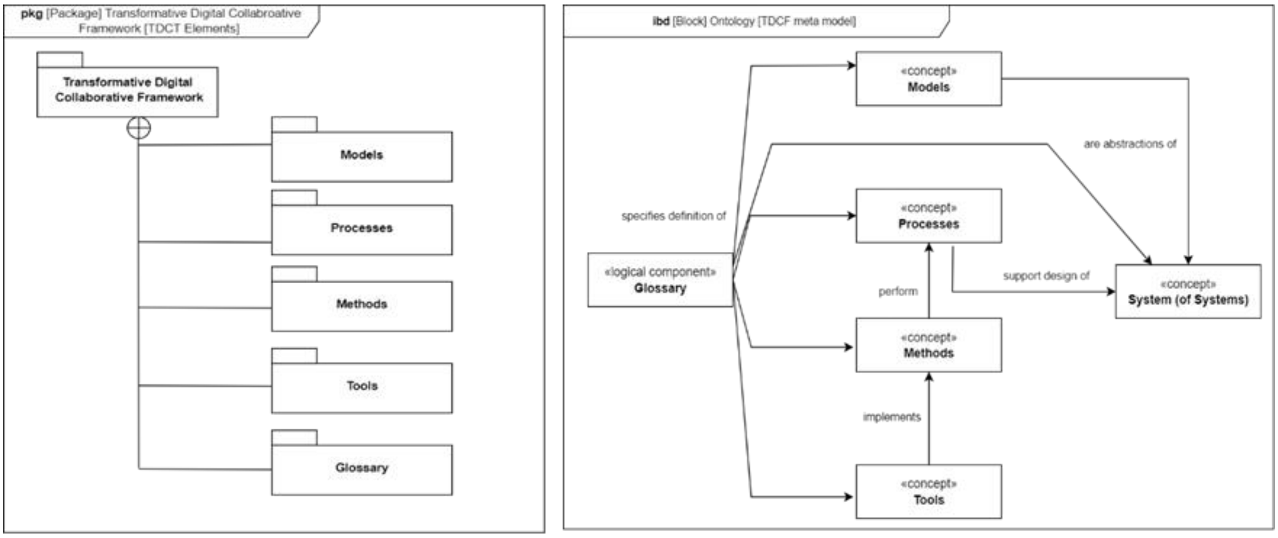

Figure 5 Transformative Digital Collaborative Framework Elements and Their Connections

As shown in Figure 5 *Glossary* specifies the definition of *Models*, *Processes*, *Methods*, and *Tools* as well as the definition of SoS according to standards. It is essential to define a common language among all the experts involved in the SoS design. *Models*, instead, are the abstraction of the SoS and define the level of details of interest for each level of the SoS. The abstraction layers and the decomposition layers are shown in Figure 6. For instance, the CS can be modelled as a black box if detailed information at this level are not needed.

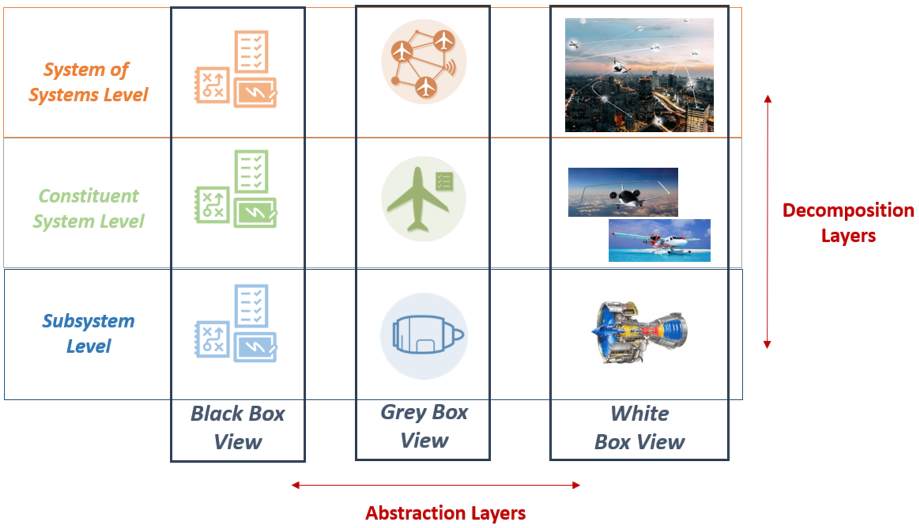

Figure 6 Abstraction and Decomposition Layers

Finally, *Processes* performed by *Methods* implemented in *Tools* support the design of SoS. Four main Processes, including system engineering and multidisciplinary activities, cover the complete design of the SoS:

- **Problem Definition** addressing the description of the problem to solve (e.g. firefighting) through the specifications of use cases, scenarios, success criteria and identification of all Stakeholders involved and their needs
- **System Specification** focusing on the correct and unambiguous transformation of stakeholders 'needs into SoS, Constituent Systems Subsystem-level technical requirements
- **System Architecting** generating all the combinations obtained combining different types of constituent systems and sub-constituent systems and their way of operating
- **System Design and Exploration** evaluating all the architectures to identify the best one according to decision-makers 'expectations.

These Processes are strictly interconnected to each other. The stakeholders identified in the Problem Definition impact the SoS or CS requirements through their needs. Instead, the requirements at SoS or CS level lead to the different architectures since different combinations of CSs, for instance, are needed to fulfill the requirements. Finally, the criteria important for decision-makers (part of the stakeholders) are needed to identify the best solution on the design space including all the generated architectures. Several technologies have been implemented or are still under development to automate the execution of each of the mentioned process and their connections. These technologies are part of the DLR Digital Environment for SoS (DE4SoS), shown in Figure 7. ARMADE, the DLR Requirements Engineering Tool, is used to model the problem definition and system specification processes. Technical requirements addressing the SoS, the CS and the subsystem levels are implemented following standards. ADORE supports the modelling of the system architectures including the different SoS layers and their way of operating (ConOps) [12]. The system design and exploration are then supported by MDAx and RCE [13,14]. These technologies support the formulation and execution of workflows which includes tools evaluating the main performance of the SoS and/or CS. For instance, OpenAD is used to design and evaluate the aircraft performance, and the SoSID Toolkit for the analysis and evaluation of the SoS performance. A workflow connecting tools at different SoS levels can be built to address evaluations at different SoS layers at the same time. Once the design space is explored and data obtained, VALORISE is used to post-process the results and identify the best solution by trading the decision-makers 'expectations, for instance, by prioritizing the main criteria important for taking decisions in different ways [15].

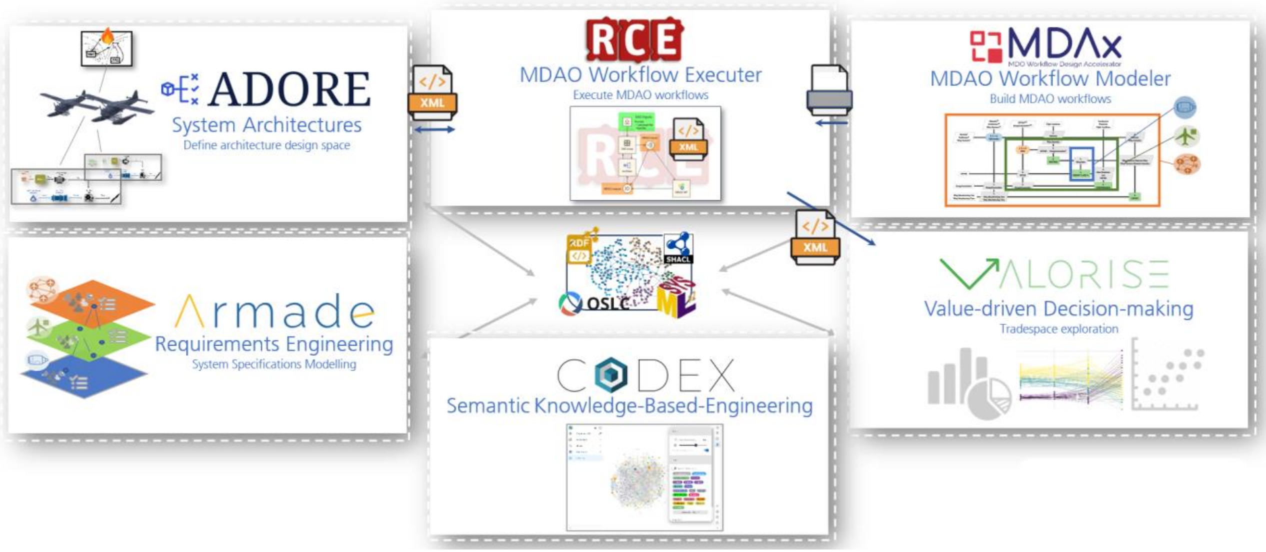

Figure 7 DE4SoS: Digital Environment for SoS [16,17,2]

All the technologies that are part of the DE4SoS can share information through different formats, such as CPACS [16]. However, a common data management store is under development to allow an automatic exchange of data among these technologies without import/export of files. Finally, data generated in this DE4SoS environment can be easily visualized in an Interactive Dashboard which is under development. This Interactive Dashboard provides users also the possibility to play around with the created data allowing for the user to gain insights through a hands-on approach. These technologies enable a holistic approach in which the focus is shifted from the design of individual vehicles to vehicles interconnected and interoperating together in a SoS.

An example of application of the DE4SoS technologies to the EVE case is reported in the following pictures. In Figure 8, requirements implemented in ARMADE are showed. These requirements are related to the SoS but also the CS (i.e. the seaplane).

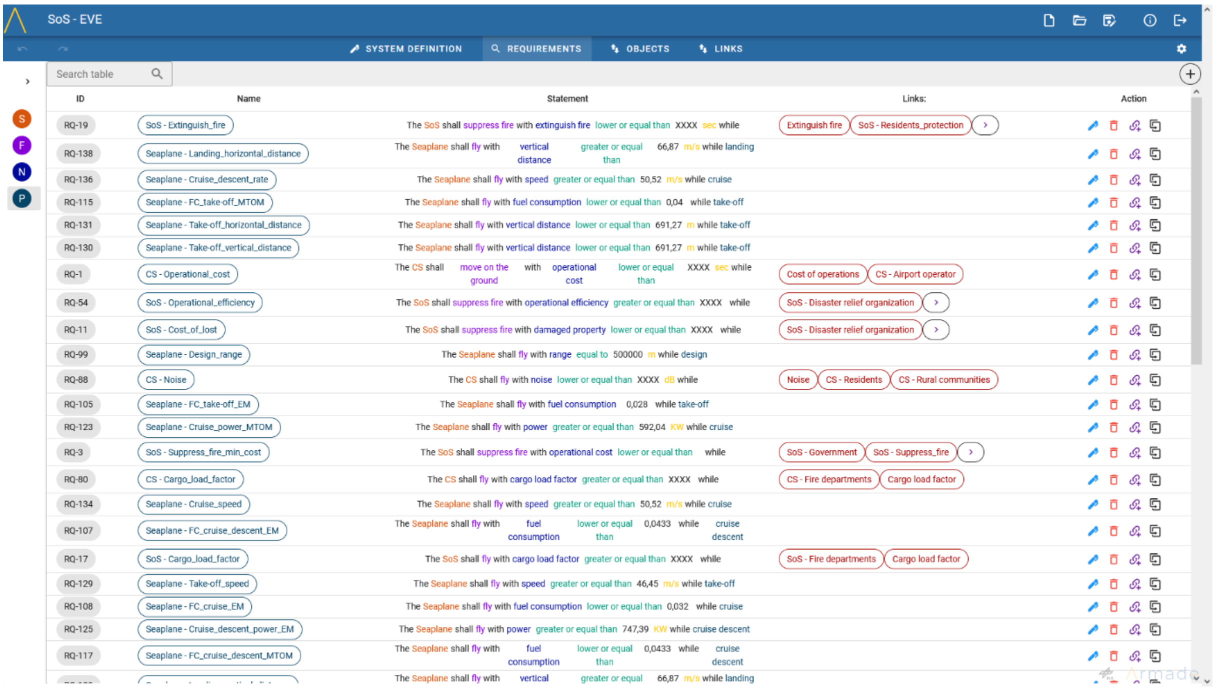

Figure 8 DE4SoS: ARMADE application to EVE

In Figure 9, a simple example of the architectural design space is depicted. It includes choices related to the SoS and CS like the number of aircraft that should be part of the fleet or the type of CS to adopt or which propulsion system characterize each CS. In addition, also choices related to the operations of the SoS can be implemented. For instance, the choice of which tactic to use when fighting the fire.

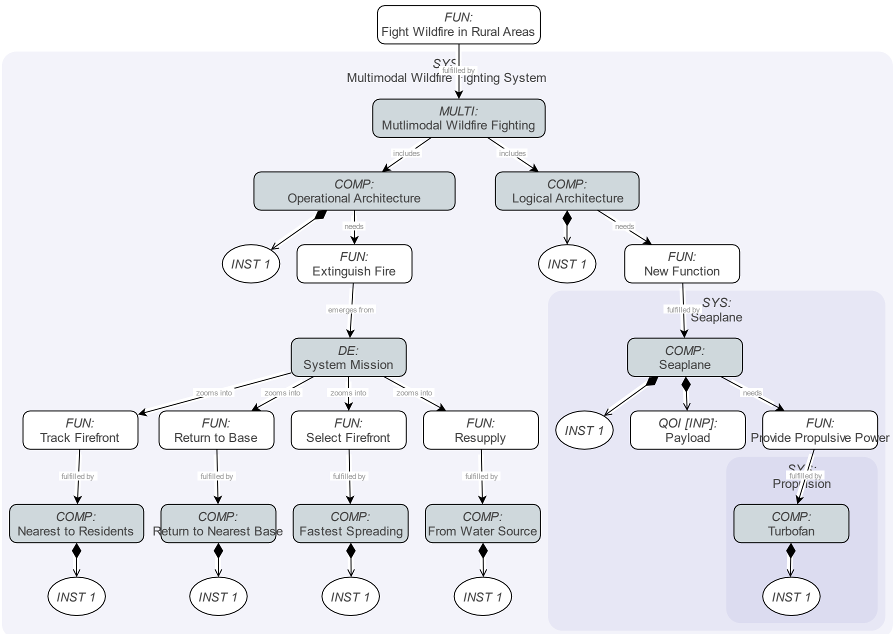

Figure 9 DE4SOS: ADORE Application to EVE

From the architectural design space, several architectures can be generated making decisions on the different choices and later optimized in the RCE environment. As already mentioned, toolkit is used to evaluate the SoS performance. The automatic connection between ADORE and toolkit allows to optimize the selected architecture. The RCE workflow run to execute the optimization problem and the visualization in toolkit is shown in Figure 10. The optimization problem is set in ADORE by introducing the concept of Quantity of Interest (QoI). These play the role of the design variables, objective functions and constraints of the optimization problem.

Figure 10 DE4SoS: Optimization Workflow and Agent-Based Simulation for the EVE use case [14,2]

More complex applications of the DE4SoS technologies will be addressed in the project, also related to the ADAM use case.

4. Aircraft Design and System of Systems implementation

The preceding chapter focused on the methodology employed to achieve the goals of the project. In this chapter, demonstration and exemplary results from the tools will be highlighted to show the progress made towards achieving the goals of the project.

4.1 Initial Results from Aircraft Design

4.1.1 Seaplane Design

According to project's objectives the design activities of the multi-role seaplane will follow the Product-Push paradigm for the ADAM use case and the Product-Pull paradigm for the EVE use case. For each use case, design tasks will be carried out through an iterative, multidisciplinary approach and in three design loops with increasing level of fidelity of involved design tools.

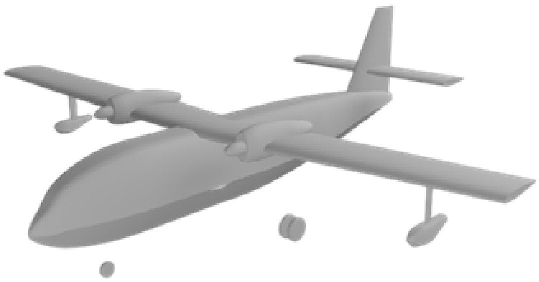

Figure 11 Conceptual Sketch of the Flying Boat Configuration

Following the product push paradigm, and initial set of aircraft are designed considering Seaplane operations in the Baltic, Balearic and Aegean Seas as per the scenarios, and considering stakeholder requirements (such as for certification and travel time), TLAR ranges are formulated as given Table 1. Three architectures are considered for the Seaplane design; Flying boat, floatplane and amphibians. A floatplane is a landplane that has had its landing gear removed and replaced with pontoons that are partially submerged floats. A flying boat is a seaplane configuration with a specially designed fuselage to function as a hull achieving operations on water. An amphibian aircraft is an aircraft that features both a hull-shaped fuselage for operations on water surfaces and retractable landing gear to achieve operations on land. While the floatplane architecture offers advantages through economy of scale through it being able to operate on land with only minor modifications, it has associated drawbacks due to decreased in-flight and on-ground performance due to the additional weight and drag. It also has limited on-water performance as it can only operate on low level of sea states. Amphibian architectures while not offering significant weight savings, perform better in-flight and also on-water. The flying boat architecture was chosen as it performs the best with challenging wave states thereby making it the most robust to weather while being performant in-flight.

The propulsion systems considered for the seaplane design are driven by the project goal of reducing emissions of the aircraft with respect to aircraft operating in 2020. As such, three powertrain configurations are considered: a conventional internal combustion engine, a hybrid-electric concept with batteries and a thermal engine, and a hybrid-electric concept with fuel cells and a thermal engine.

The Multidisciplinary Design Analysis (MDA) formulation and design methodologies are employed to perform design, optimization and assessment of the aforementioned air vehicle and brings together disciplines from various project partners through the TDCF. Figure 12 depicts the MDA formulation for the product-push paradigm of the Seaplane in the ADAM use case.

Table 1 TLAR Range considered for Seaplane for the Product Push paradigm

Requirements	Value
Number of Passengers	12-19
Range	200-600 km
Cruise Speed	250 - 400 km/h

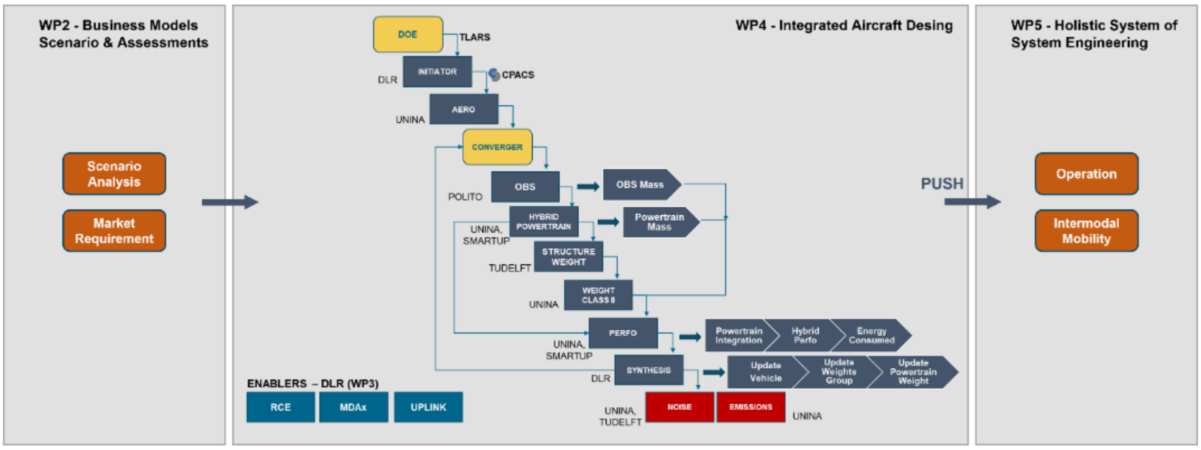

Figure 12 Seaplane concept MDA Formulation for Product-Push Paradigm - ADAM Use Case

4.1.2 eVTOL Design

eVTOL aircraft are characterized by its fully-electric powertrain and its vertical takeoff and landing capabilities. Several different architectures fall within the catergory of eVTOL, primarily the Multicopter, Tiltrotor, Lift + Cruise, and Ducted Vector Thrust architectures. Based on the minimum performance requirements imposed by the scenarios, the Multicopter architecture is the only excluded architecture due to the limited range associated with it. Between the other possible architectures, the Tiltrotor is deemed to be the most flexible for the framework and is the preferred design architecture for the vehicle.

In summary, the vehicle has a conventional airframe architecture, with a main wing and V-tail, following concepts such as the Joby Aviation S4, the Vertical Aerospace X4, and the Vahana Alpha 2. The vehicle is a propeller-driven aircraft, where all the open rotors can be tilted to allow for vertical takeoff and landing operation. Two propellers are located in the tail-end while an even number of propellers is distributed over the wing, following the schematics shown in Figure 13.

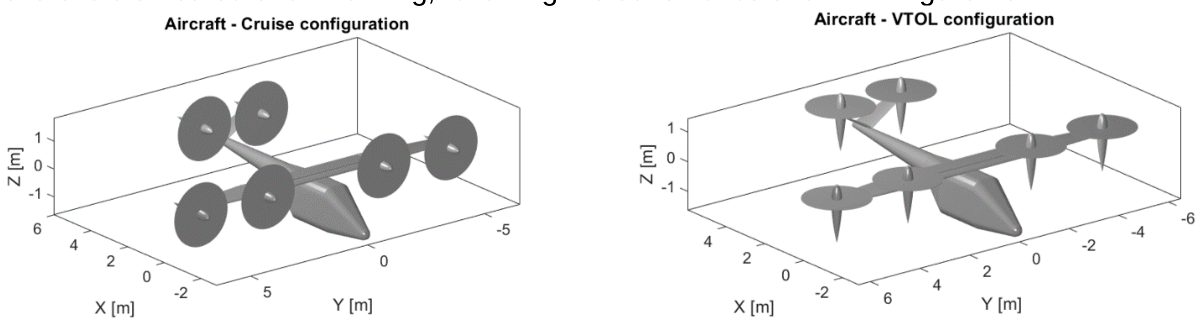

Figure 13 Conceptual Sketch of eVTOL vehicle design in cruise (left) and vertical flight (right)

The vehicle must be fully electrically powered, meaning batteries are the choice of energy storage and electric motors are used to drive the propulsors, i.e., the open propellers. Battery energy density is a major concern for the realization of eVTOL vehicles. The direct impact on range and Maximum Takeoff Weight (MTOW) can make the difference between a realistic design and more importantly, a profitable or non-profitable operation. Current applications of electric aircraft have demonstrated battery densities in the order of 200 Wh/kg, potentially fulfilling aeronautical certification requirements. According to [18], the theoretical limit for lithium-ion batteries for aircraft is in the order of 250 Wh/kg. Developments in solid-state batteries can increase this number by as much as 400 Wh/kg. According to [19], the value of 400 Wh/kg is an enabling value, above which the battery weight does not represent a significant concern. [20] shows that the current development of solid- state batteries can lead to mass production by 2025. This technology would allow densities in the order of 400-500 Wh/kg to be achievable for the application of eVTOL vehicles. The latter value (400 Wh/kg) is chosen as a current "best-case-scenario" for the design of the eVTOL aircraft. This vehicle must meet the preliminary targeted performance parameters shown in Table 2. This performance is chosen to fulfil both Urban Air Mobility (UAM) and Regional Air Mobility (RAM) operations. The MDA formulation of the eVTOL concept for the product-push paradigm (ADAM use case) is given in Figure 14.

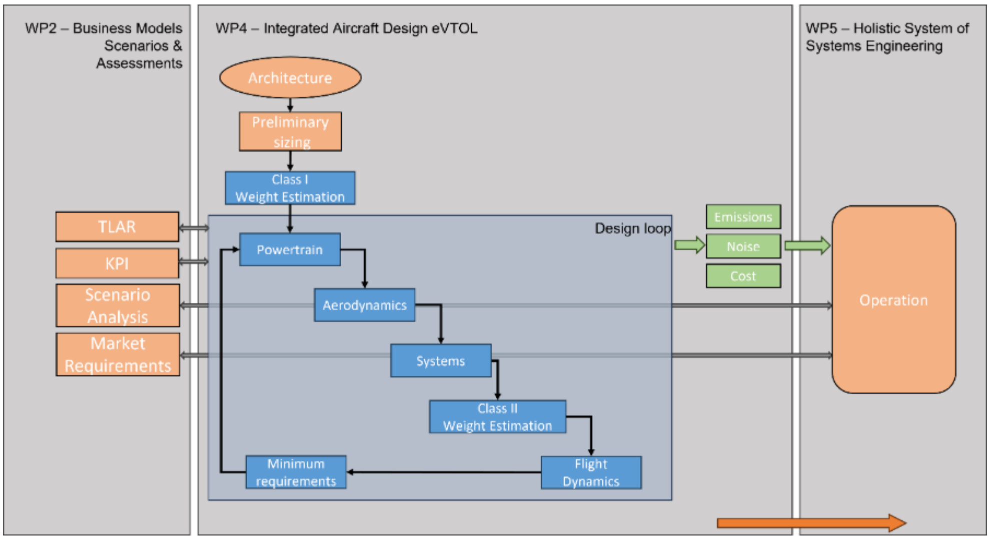

Figure 14 eVTOL concept MDA Formulation for Product-Push Paradigm - ADAM Use Case

Table 2 Baseline requirements for the eVTOL Vehicle

Requirements	Value
Number of Passengers	4
Range	100 km
Cruise Speed	>200 km/h

4.2 Implementation of SoS via ABS

For each of the Use Cases, two scenarios are defined to provide a set of differing yet specific problems which can be solved by the introduction of new assets and their operations in a SoS approach. For each of the two use cases one scenario is presented in this paper. A brief introduction of the Agent-Based Simulation developed for each use case is presented first, followed by several preliminary results.

4.2.1 ADAM – Sustainable Intermodal Mobility

The ADAM use case is centered around complementing the existing transport network with novel aircraft concepts (eVTOL and Seaplane). A European scale transport network is considered to capture the effect of the novel concepts on intracity, regional and European transport scales.

An intermodal mobility simulation is developed by extending the ABS of Advanced Air Mobility [3,21,4,22,23] with the open-source tool SUMO (Simulation of Urban Mobility) [24]. This integrated approach allows for the simulation of both ground and air operations seamlessly and across large scales. Passengers are modelled with origin destination pairs, travel times and priorities. The passengers request transport to an integrated platform (Mobility as a Service Provider) which provides them with a series of intermodal travel plans consisting of ground and air legs for their journeys. The passengers then select a travel plan based on their priorities. The passengers vary in nature from short to long distance travelers. The eVTOL and Seaplanes are embedded into the simulation through their performances, and the vehicle operator is modelled using smart dispatching logic and decides on the allocation of vehicles to passengers. A description of the Simulation and its input and outputs are shown in Figure 15. Whereas the details of each constituting model of the ABS would not be appropriate in this overview paper, since the passenger is the most crucial actor in a transport SoS, the logic prescribed to the passenger agent in the ABS is described in Figure 16, and the passenger objective function in Equation 1.

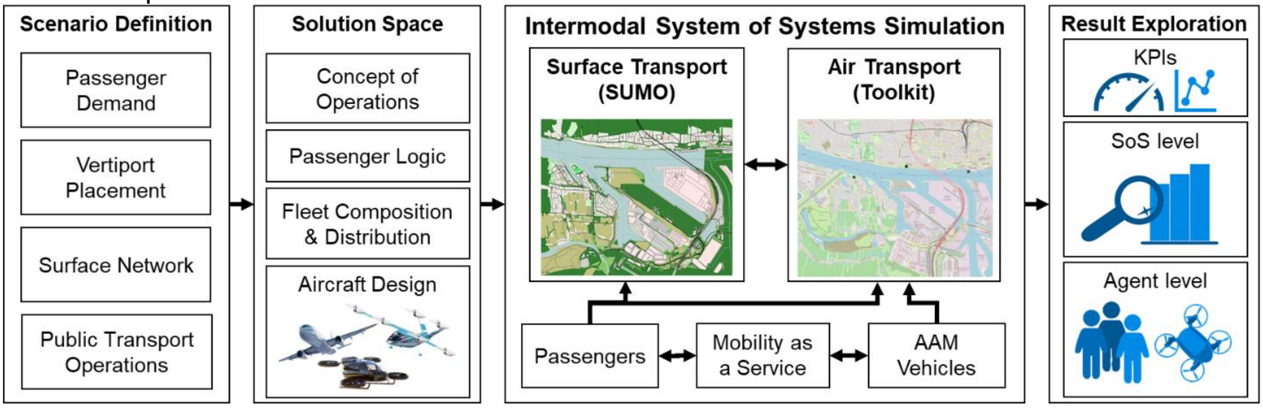

Figure 15 Schematic of the Intermodal SoS simulation, showing the link with SUMO, inputs and outputs to the Intermodal simulation

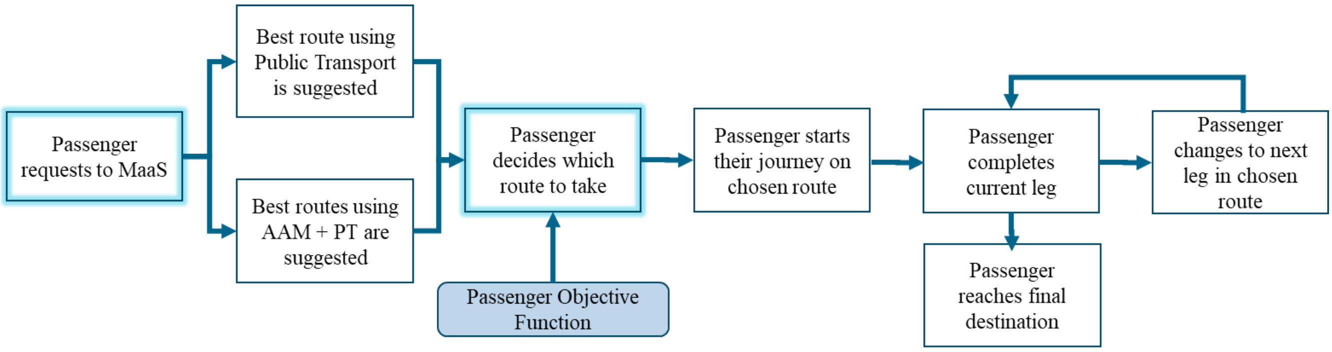

Figure 16 Passenger logic in the ADAM Simulation

Equation 1 Passenger Route Selection Logic

$$\mathrm{Obj} \;= \mathrm{Min}\left(duration * VoT + \sum C_{mode_{km}} * distance_{mode}\right)$$

As discussed in the product push paradigm, see Chapter 2.3, one of goals with the ADAM simulation is to identify how to best operate the novel aircraft products. To demonstrate an example of how insights into the operation can be obtained through the ADAM simulation, Figure 17 shows results from a study around Hamburg using a fleet of eVTOLs. Figure 17 (left) shows the routes in an area that can offer time savings to the set of travellers considered in a study. It can be seen that the longer AAM routes promise more potential time savings to the travellers in this area. If this is compared to those routes actually taken by AAM shown in Figure 17 (right). It can be seen that a disagreement arises between the routes that passengers *should* take and those that they *actually* take. This discrepancy is attributed to the ticket pricing structure assumed for this study where a linear scheme was assumed and though the time saving potentials also increase with distance, the increase in cost does not justify the use. The insights provided by this brief example show that in order to capture the longer distance trips by eVTOLs, non-linear ticket pricing models could be utilized to better take advantage of the cruise-efficiency of the Tiltrotor concepts.

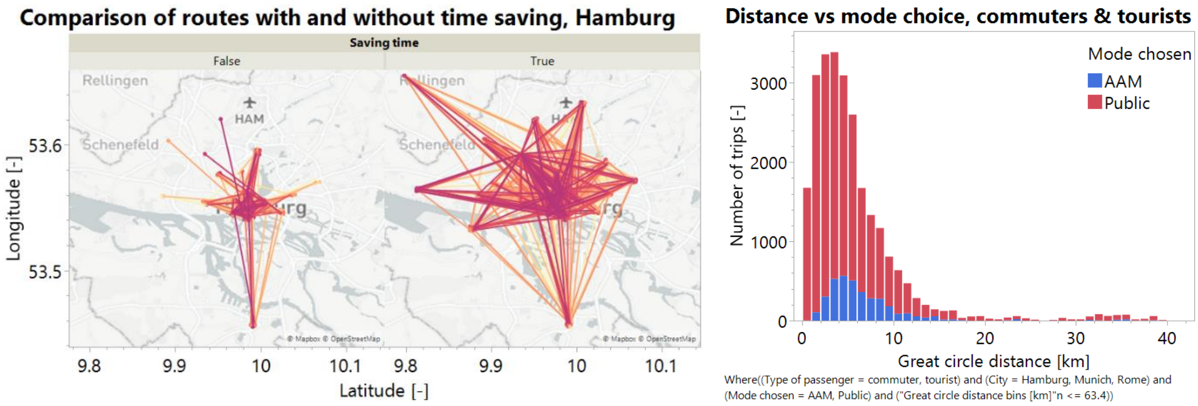

Figure 17 AAM Routes in Hamburg Metropolitan area with and without potential time savings (Left), Number of trips by mode and distance (Right)

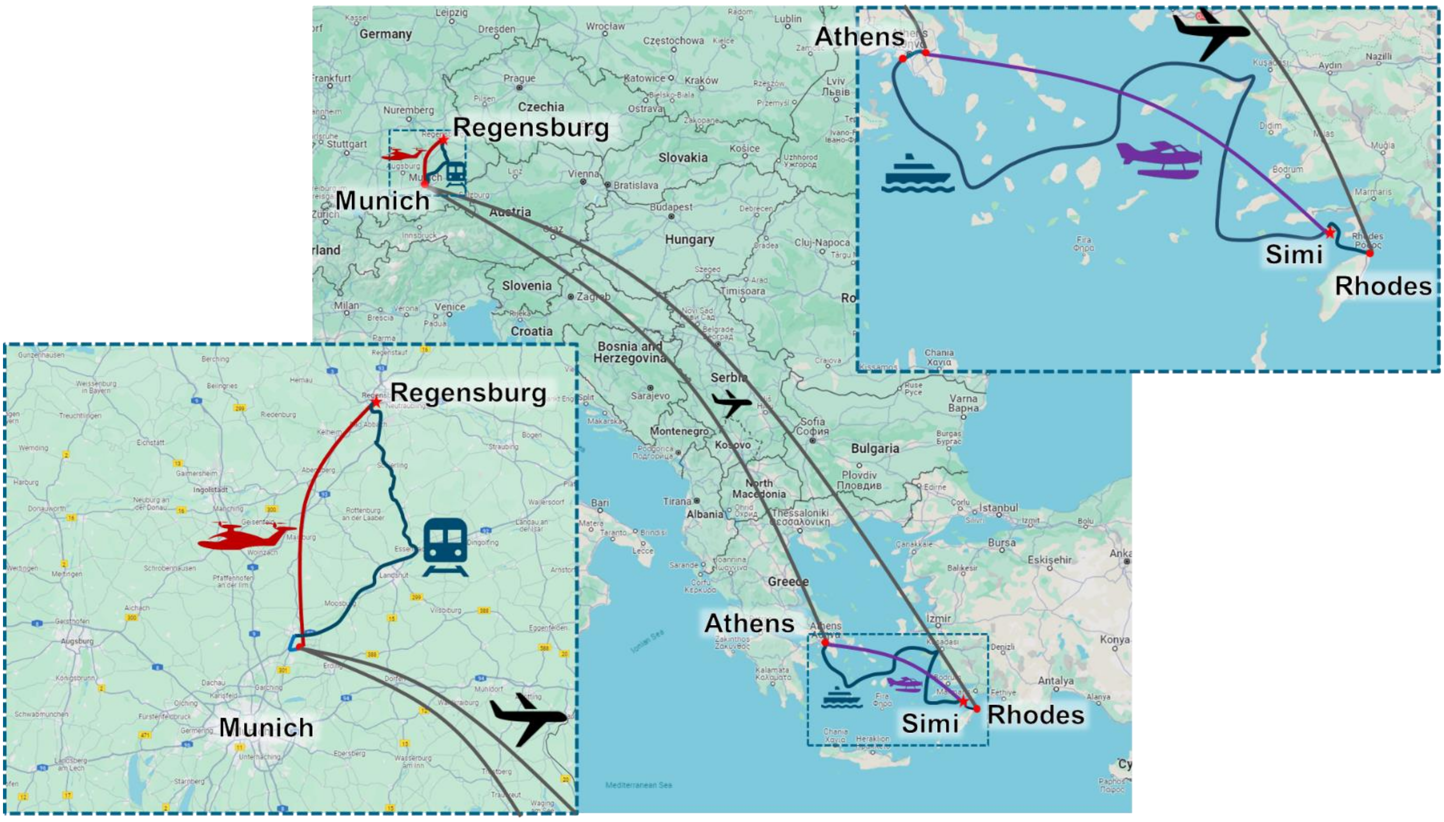

Figure 18 Intermodal Routes for a passenger travelling from Regensburg. Germany to Simi, Greece

Table 3 Intermodal Route description for a passenger travelling from Regensburg, Germany to Simi, Greece

	Legs	Mode	Duration	CO2 Emissions (kg)	Cost (EUR)
	Route 1				
07:50 - 09:30	Regensburg to Munich	Public Transport	1h 40min	4 [25]	29[3]
11:40 - 15:15	Munich to Athens	Flight	2h 35min	106*	154[3]
18:00 – 12:45[+1]	Athens to Simi	Ferry	18h	11 [26]	75[3]
-	Wait Time	-	~5h	-	-
	Total		**~27h**	**122**	**258**
	Route 2				
09:20 – 09:40	Regensburg to Munich	eVTOL	20min	22 [27]	120[1]
11:40 - 15:15	Munich to Athens	Flight	2h 35min	106*	154[3]
16:00 - 17:00	Athens to Simi	Seaplane	1h	163 [29]	180[2]
-	Wait Time		~2h 30min	-	
	Total		**~6h 30min**	**291**	**454**
	Route 3				
23:45[-1] – 03:15	Regensburg to Munich	Public Transport	3h 30min	4 [25]	29[3]
05:50 – 08:55	Munich to Rhodes	Flight	2h 05min	156*	263[3]
10:30 - 11:50	Rhodes to Simi	Ferry	1h 30min	0.98 [26]	17[3]
-	Wait Time	-	~4h	-	-
	Total		**~11h**	**161**	**300**

The intermodal ADAM simulation not only allows insights into the vehicle operator perspective to be taken as demonstrated by the previous example. The attractive routes for the operation of the novel aircraft can also be identified. An exemplary route analysis for a passenger travelling for their holidays from Regensburg, Germany to Simi, Greece is presented in Figure 18 and Table 3. The route analysis was performed using real data augmented with the new eVTOL and Seaplane modes. All of the route options involve a flight through Munich International Airport, however the intermodal combinations with these flights make a significant difference with respect to time, cost and emissions. Three unique routes are presented, offering varying options attractive to different types of passengers. Route 1 consists of taking public transport, particularly a regional train and a bus, from Regensburg to Munich International Airport. A flight from Munich to Athens follows, ending with an 18h long ferry ride from Athens to Simi. Route 2 has an eVTOL flight from Regensburg directly to Munich International Airport, following a flight from Munich to Athens, and terminating with a direct Seaplane flight to Simi. Route 3 has the distinction of the flight leg taking place between Munich International Airport, to Rhodes Airport, from where a short ferry ride takes the passenger to Simi. However, while Route 3 saves significant time from the ferry journey by flying to an airport closer to Simi, the early departure time comes at the cost of a significantly longer train journey to the airport as well as a longer wait time at the airport. In addition, it comes with an intangible cost of loss of sleep, as the departure from origin takes place just before midnight. A comparison of the routes, show that Route 1 is the cheapest and the most environmentally friendly, however it comes with a significant time penalty. Route 1 may be attractive still to a budget or an environmentally conscious traveler. Route 2 comes at 1.5x the cost of Route 3 and with more emissions, yet offers significant time savings of 4h and 30 min over the next fastest route and has the added advantage of a morning start to the journey. Route 2 may prove the most attractive to those who don't have the luxury of time, or do not want the inconvenience of travelling throughout the night. Route 3, may be favorable yet due to its attractive price performance despite the inconvenience associated with it. The choice between the routes, are entirely dependent on the passenger and what their values are. The ADAM simulation models the different passenger types

through their priorities, and allows them to select among a set of intermodal routes involving regional

[1] Estimated based on cost per pax-km value of 1.25 EUR per pax-[28] with an added margin of 25%
[2] Estimated based on operating cost of Sea Planes [30.
[3] Data retrieved directly from Ferry or Flight Operator's portal

aircraft, trains, busses, ferries and the novel eVTOLs and Seaplanes, similarly as done in this exemplary result. While analysis of a single passenger may not provide enough data to make conclusions on the attractive routes or aircraft requirements, thousands of passengers with varying priorities and origin-destination pairs may. By modelling the existing transport network, and representative travel patterns, insights into how novel modes can best be introduced and operated may be drawn.

4.2.2 EVE – Aerial Wildfire Fighting

The Aerial Wildfire Simulation [31–33] recreates the defined scenario, in terms of its environment and assets. The environment is recreated by modelling the atmospheric conditions either through mathematical models or historical/forecasted weather data, and the terrain is modelled by recreating the elevation profile and the features of the terrain including water sources and vegetation types. The aerial and ground assets are modelled such that they collaborate together to fight and contain the wildfire. Different Concept of Operation of the asset are also modelled and can be varied to find the ideal SoS Constellation. These include direct attack tactics which constitutes of attacking the fire front and indirect attack tactics which constitutes of building a fire block ahead of the fire to prevent it from advancing further. A Cellular Automata wildfire model [34] is developed to model the spread of the wildfire with respect to the atmospheric conditions and terrain definition. The assets are modelled with their characteristics, and the logic governing their behavior. The aerial assets are modelled in the Simulation using their performance and energy characteristics such as energy consumption per flight phase and mass, total energy capacity, payload, etc. Different turnaround procedures can also be modelled based on the powertrain architecture of the assets, such as re-charging or battery swap for assets with an electric powertrain. Furthermore, the bases which the assets can operate from can be flexibly defined including the position and the type of infrastructure available at the bases. This allows for certain bases to be made accessible only to Seaplanes by way of having water runways, and for eVTOLs by way of having vertipads. The composition of the assets, both air and ground, can be controlled by input including the number of each type of asset and their positions at the start. A schematic demonstrating an active simulation is given in Figure 19.

Figure 19 Schematic demonstrating active Simulation of a wildfire in Salamina, Greece using eVTOLs and Seaplanes employing an Indirect Attack strategy.

As discussed in Chapter 2.3 the EVE use case employs the product pull paradigm. Simply meaning, the desired goal is to identify the requirements for the aircraft based on the SoS evaluations. To demonstrate how this may be done with the EVE simulation, Figure 20 shows an exemplary result from a study where the response time and the fleet size were investigated with respect to the total burnt area. The failed missions (i.e. where the fire could not be suppressed by the fleet) are indicated in red and the successful missions in blue. It can be seen that as the response times increase the required fleet size scales with it. This shows the importance of the SoS simulation to identify the required SoS Constellations while ensuring robustness. Similary TLAR sensitivity studies, ConOps sensitivity studies and more can be performed to identify the optimal SoS constellations.

Figure 20 Effect of increasing response time on burnt area with varying fleet size

5. Conclusions and Future Work

Overall, the COLOSSUS project focuses on setting up and solving an SoS problem. The project implements the developed approaches and tools on two contrasting use cases namely Aerial Wildfire and Sustainable Intermodal Mobility, for which European scenarios are considered. The problem setup phase has largely been completed with the identification of representative scenarios, involved stakeholders, and their key value indicators. The stakeholder interests are formulated into a Value Function which quantifies the effectiveness of an SoS from a holistic perspective. A coupling of the vehicles designs and SoS evaluations are achieved using state of the art Agent Based Simulations of the two use cases. A cellular automata wildfire model is integrated into the Agent-Based Simulation of the Aerial Wildfire Fighting use case to model the fire behavior thereby creating a platform on which to test the novel aircraft designs and tactics from the project. Furthermore, to explore the operational design space of the Aerial Wildfire Fighting, Reinforcement Learning methodologies will be leveraged in the Agent-Based Simulation to derive the optimal tactics. An intermodal mobility simulation modelling the ground transport modes, regional air transport, together with a representative travel demand scenario create the platform on which the eVTOL and Seaplane designs can be injected into, to evaluate and optimize their business models. Multi-level optimization of the multi-level SoS spanning subsystem, constituent system, system of systems and business models are underway where initial explorations have already been made. The optimization, and overall results analysis will be performed using a holistic approach considering the effectiveness of the SoS as well as it's life cycle analysis and cost. The results from the performed studies such as sensitivity analyses and optimization will be visualized in an interactive dashboard where multi-criteria decision making according to stakeholder interests can be performed. Finally, a transformative digital collaborative framework (TDCF) to efficiently solve an SoS problem, bringing together all the aforementioned capabilities is under development. The TDCF will also enable the tracking and fulfilment of requirements of the SoS and the proposed solutions, and enable the architecting and design space exploration of the SoS.

6. Acknowledgements

The research presented in this paper has been performed in the framework of the COLOSSUS project (Collaborative SoS Exploration of Aviation Products, Services and Business Models) and has received funding from the European Union Horizon Europe program under grant agreement No. 101097120. The Swiss participation in the Colossus project is supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under contract number 22.00609.

7. Contact Author Email Address

The contact author email address: prajwal.prakasha@dlr.de

8. Copyright Statement

References

[1] Martin Spieck, Ludvig K. Franzén, Kristian Amadori, Prajwal S. Prakasha, and Nabih Naeem, "A Capability-Focused Approach To Model Complex, Multi-Layered Systems-of-Systems," *AIAA Aviation Forum and Ascend 2024*.

[2] Kilkis, S., Shiva Prakasha, P., Naeem, N., and Nagel, B., "A Python Modelling and Simulation Toolkit for Rapid Development of System of Systems Inverse Design (SoSID) Case Studies," *AIAA Aviation 2021 Forum,* Virtual Event, 2021.
doi: 10.2514/6.2021-3000

[3] Naeem, N., Ratei, P., and Shiva Prakasha, P., "Modelling and Simulation of Urban Air Mobility: An Extendable Approach," *12th EASN International Conference,* Barcelona, 2022.

[4] Shiva Prakasha, P., Naeem, N., Ratei, P., and Nagel, B., "Aircraft architecture and fleet assessment framework for urban air mobility using a system of systems approach," *Aerospace Science and Technology*, Vol. 125, 2022.
doi: 10.1016/j.ast.2021.107072

[5] Martins, J. R. R. A. and Lambe, A. B., "Multidisciplinary Design Optimization: A Survey of Architectures," *AIAA Journal*, Vol. 51, No. 9, 2013, pp. 2049–2075.
doi: 10.2514/1.J051895

[6] Bartoli, N., Lefebvre, T., Dubreuil, S., Olivanti, R., Priem, R., et al., "Adaptive modeling strategy for constrained global optimization with application to aerodynamic wing design," *Aerospace Science and Technology*, Vol. 90, 2019, pp. 85–102.
doi: 10.1016/j.ast.2019.03.041

[7] Jones, D. R., Schonlau, M., and Welch, W. J., "Efficient Global Optimization of Expensive Black-Box Functions," *Journal of Global Optimization*, Vol. 13, No. 4, 1998, pp. 455–492.
doi: 10.1023/A:1008306431147

[8] Michael James Sasena, "Flexibility and efficiency enhancements for constrained global design optimization with kriging approximations," PhD, *Mechanical Engineering,* United States, 2002.

[9] Priem, R., Gagnon, H., Chittick, I., Dufresne, S., Diouane, Y., et al., "An efficient application of Bayesian optimization to an industrial MDO framework for aircraft design," *AIAA AVIATION 2020 FORUM,* AIAA AVIATION 2020 FORUM, VIRTUAL EVENT, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2020.
doi: 10.2514/6.2020-3152

[10] Saves, P., Bartoli, N., Diouane, Y., Lefebvre, T., Morlier, J., et al., "Multidisciplinary design optimization with mixed categorical variables for aircraft design," *AIAA SCITECH 2022 Forum,* AIAA SCITECH 2022 Forum, San Diego, CA & Virtual, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2022.
doi: 10.2514/6.2022-0082

[11] Grapin, R., Diouane, Y., Morlier, J., Bartoli, N., Lefebvre, T., et al., "Constrained Multi-Objective Bayesian Optimization with Application to Aircraft Design," *AIAA AVIATION 2022 Forum,* AIAA AVIATION 2022 Forum, Chicago, IL & Virtual, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2022.
doi: 10.2514/6.2022-4053

[12] Bussemaker, J., Boggero, L., and Ciampa, P. D., "From System Architecting to System Design and Optimization: A Link Between MBSE and MDAO," *INCOSE International Symposium*, Vol. 32, No. 1, 2022, pp. 343–359.
doi: 10.1002/iis2.12935

[13] Page Risueño, A., Bussemaker, J., Ciampa, P. D., and Nagel, B., "MDAx: Agile Generation of Collaborative MDAO Workflows for Complex Systems," *AIAA AVIATION 2020 FORUM,* AIAA AVIATION 2020 FORUM, VIRTUAL EVENT, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2020.
doi: 10.2514/6.2020-3133

[14] Brigitte Boden, Jan Flink, Robert Mischke, Kathrin Schaffert, Alexander Weinert, et al., RCE, 2019.

[15] Donelli, G., Mello, J. M., Odaguil, F. I., Van der Laan, T., Boggero, L., et al., “Value-driven Tradespace Exploration for Aircraft Design, Manufacturing and Supply Chain,” *ICAS 2024*.

[16] Alder, M., Moerland, E., Jepsen, J., and Nagel, B. (eds.), *Recent Advances in Establishing a Common Language for Aircraft Design with CPACS*, 2020.

[17] DLR, “PRELIMINARY AIRCAFT DESIGN WITHIN A MULTIDISCIPLINARY AND MULTIFIDELITY DESIGN ENVIRONMENT - OpenAD,”.

[18] Polaczyk, N., Trombino, E., Wei, P., and Mitici, “A review of current technology and research in urban on-demand air mobility applications,”.

[19] Brown, A. and Harris, W. L., “Vehicle Design and Optimization Model for Urban Air Mobility,” *Journal of Aircraft*, Vol. 57, No. 6, 2020, pp. 1003–1013.
doi: 10.2514/1.C035756

[20] Adu-Gyamfi, B. A. and Good, C., “Electric aviation: A review of concepts and enabling technologies,” *Transportation Engineering*, Vol. 9, 2022, p. 100134.
doi: 10.1016/j.treng.2022.100134

[21] Naeem, N., Ratei, P., Shiva Prakasha, P., Zill, T., and Nagel, B., “System of Systems Driven Fleet Design for Urban Air Mobility: Heterogeneity over Homogeneity,” 2022.

[22] Shiva Prakasha, P., Papantoni, V., Nagel, B., Brand, U., Vogt, T., et al., “Urban Air Mobility Vehicle and Fleet-level Life-Cycle Assessment Using a System-of-Systems Approach,” *AIAA Aviation 2021 Forum,* Virtual Event, 2021.
doi: 10.2514/6.2021-2457

[23] Shiva Prakasha, P., Ratei, P., Naeem, N., Kilkis, S., and Nagel, B., “Towards System of Systems Driven Urban Air Mobility Aircraft Design,” *DICUAM 2021*.

[24] Lopez, P. A., Wiessner, E., Behrisch, M., Bieker-Walz, L., Erdmann, J., et al., “Microscopic Traffic Simulation using SUMO,” *2018 21st International Conference on Intelligent Transportation Systems (ITSC),* 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, 04-Nov-18 - 07-Nov-18, IEEE, 2018, pp. 2575–2582.
doi: 10.1109/ITSC.2018.8569938

[25] Alice Favre, *Rail statistics on energy consumption and emissions*, 24 Apr 2024.

[26] Sims R., R. Schaeffer, F. Creutzig, X. Cruz-Núñez, M. D’Agosto, D. Dimitriu, M.J. Figueroa Meza, L. Fulton, S. Kobayashi, O., “Mitigation of Climate Change. Contribution of Working Group III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change: Transport,” *Climate Change 2014: Mitigation of Climate Change*.

[27] Melo, S. P., Cerdas, F., Barke, A., Thies, C., Spengler, T. S., et al., “Life Cycle Engineering Modelling Framework for batteries powering electric aircrafts – the contribution of eVTOLs towards a more sustainable urban mobility,” *Procedia CIRP*, Vol. 105, 2022, pp. 368–373.
doi: 10.1016/j.procir.2022.02.061

[28] Xiaolong Wu, “Performance evaluation of UAM configurations cost estimation for the eVTOLs,” Doctoral, 2021.

[29] Ostrom Climate, “Harbour Air Seaplanes: 2021 GHG Emissions Summary Report,”.

[30] Thomas Denz, Stephanie Smith, and Rajeev Shrestha, “Seaplane Economics: A quantitative cost comparison of seaplanes and land planes for Sea Base operations,” 2007.

[31] Cigal, N., Naeem, N., Ratei, P., Shiva Prakasha, P., and Nagel, B., “Sensitivity Analysis for Aerial Wildfire Fighting Tactics with Heterogeneous Fleet Via System of Systems Simulation Framework,” 2022.

[32] Shiva Prakasha, P., Nagel, B., Kilkis, S., Naeem, N., and Ratei, P., “System of Systems Simulation Driven Wildfire Fighting Aircraft Design,” *AIAA Aviation 2021 Forum,* Virtual Event, 2021.
doi: 10.2514/6.2021-2455

[33] Shiva Prakasha, P., Naeem, N., Ratei, P., Cigal, N., and Nagel, B., “Exploration of Aerial Firefighting Fleet Effectiveness and Cost by System of Systems Simulations,” *32nd Congress of the International Council of Aeronautical Sciences*, 2021.

[34] Rui, X., Hui, S., yu, X., Zhang, G., and Wu, B., “Forest fire spread simulation algorithm based on cellular automata,” *Natural Hazards*, Vol. 91, 2018, pp. 1–11.
doi: 10.1007/s11069-017-3127-5