From Operational Scenarios to Synthetic Data: Simulation-Based Data Generation for AI-Based Airborne Systems

Siddhartha Gupta*
Clausthal University of Technology, Institute of Informatics, 38678 Clausthal-Zellerfeld, Germany

Umut Durak† Oliver Ellis‡ Christoph Torens§
German Aerospace Center (DLR), Institute of Flight Systems, 38108 Braunschweig, Germany

Developing safety-critical AI-based systems is an emerging challenge in aviation. Amongst others, the recent concept paper of the European Union Aviation Safety Agency (EASA), "First usable guidance for Level 1 machine learning applications", provides invaluable insights to tackle this challenge. It particularly highlights the importance of synthetic data for training, validation and testing as a means of complementing real world data for completeness and representatives. The primary source of synthetic data is simulations. The literature recognizes simulation not only as a crucial data source but an effective method for verification. This paper uses EASA guidance as a baseline to propose a simulation-based data generation process for AI-based airborne systems.

I. Introduction

The future of aviation will be smart and connected. European Union Aviation Safety Agency (EASA) Artificial Intelligence Roadmap [1] presents the intention of the agency to approve the first AI-based systems in 2025. To enable the certification of AI-based systems in aviation, there are many ongoing efforts for developing acceptable means of compliance [2]. EASA Concept Paper: First Usable Guidance for Level 1 Machine Learning Applications [3] has been the first one to be published for airborne systems. EASA classifies the AI-based systems to Level 1 when they are human assistance, Level 2 when there is a human/machine collaboration and Level 3 for the autonomous machine.

The EASA guidance presents learning assurance, AI explainability and AI safety risk mitigation as the building blocks of trustworthy AI. Learning assurance aims at extending the current development assurance approaches to cover the learning process. Figure 1 shows the integrated learning assurance process proposed by the EASA concept paper, where there is learning process verification before the implementation. The machine learning model is expected to be verified before going forward with the implementation. Different from the classical non-AI systems, implementation verification includes inference model verification and data verification. As for all other AI-based systems, data is also the key element in the EASA learning assurance process. The model training step uses training and validation data, whereas test data is used for learning process verification and inference model verification.

The data requirements for developing AI-based systems are enormous. Particularly for computer vision applications, while it may be possible to produce real data and label them for typical cases, it is almost impossible to produce data for corner cases if not too dangerous or too expensive. The public extract of the report from the collaboration between EASA and Deadalean AG from the project Concepts of Design Assurance for Neural Networks (CoDANN) [4] recognizes simulation as a crucial data source.

Scenarios are important artefacts in the simulation engineering process[5]. A simulation study begins with a description of a scenario and ends with a successful simulation execution[6].SES has been a vital component in the scenario-based approaches for aviation. It represents elements of a system and their relationships in a hierarchical and axiomatic manner[7]. [8] have used SES metamodel to model components of a scenario in Aviation and have demonstrated its capabilities in Aviation Scenario Definition Language(ASDL)[7] and as well as its use in a research simulator[9] at the German Aerospace Center. However, a scenario based approach using SES needs to evolve for the data demands posed by AI-based systems.

---

*Scientific Staff, Aeronautical Informatics.

†Group Leader, Safety Critical Systems and Systems Engineering, AIAA Associate Fellow.

‡Research Scientist, Flight Dynamics and Simulation.

§Research Scientist, Unmanned Aircraft, AIAA Senior Member.

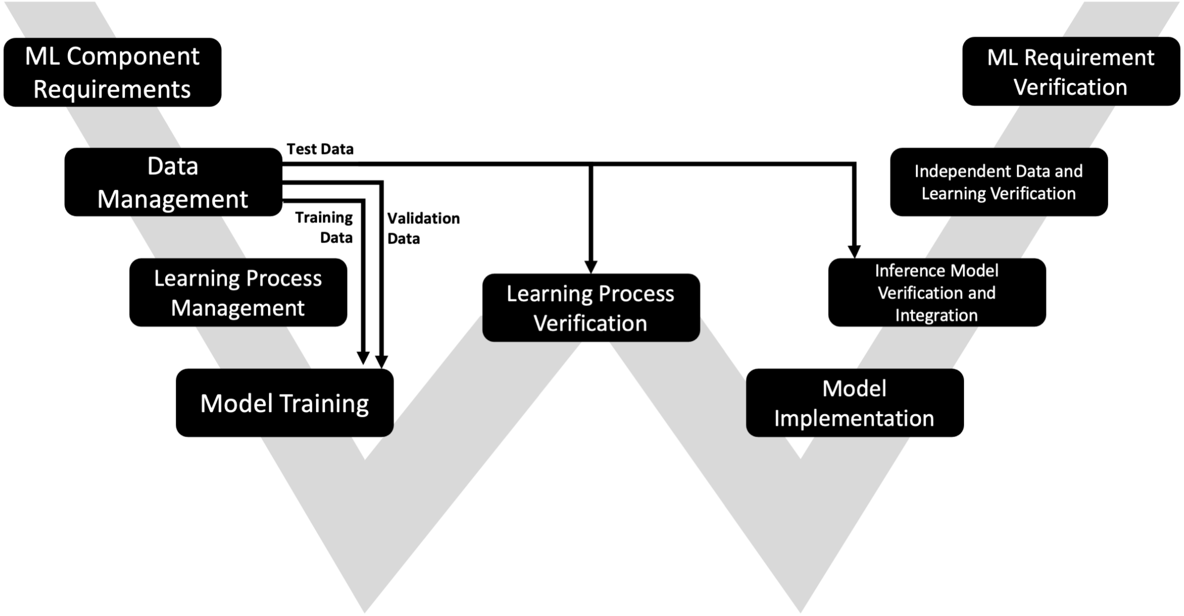

Fig. 1 EASA Integrated Learning Assurance Process [3]

The main concern of the paper is to answer the question: What needs to be simulated for synthetic data generations? We aim to answer this question with a simulation-based data generation process based on the EASA guidance and the scenario based approach using SES. The process will be an extension to SES scenario modelling [10]where we visit some other concepts related to AI based system and exemplify all the steps with a demonstrator.

II. Scenarios. Operational Design Domain and Simulation

The EASA guidance asks the developers of the ConOps for all AI-based systems to first capture the operational scenarios and then the operational limitations and assumptions of the system, namely the Operational Design Domain (ODD).

"A conops describe how a system will be operated during a life cycle phases to meet the stakeholders expectations.It describes the system characteristics from an operational perspective and helps facilitate an understanding of the system goals" [11]

While there is no commonly accepted definition of ODD in aviation, in the automotive domain, ODD is defined as Operating conditions under which a given driving automation system or feature thereof is specifically designed to function, including, but not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain traffic or roadway characteristics [12]. The same definition has been used in Unmanned Aircraft Systems and the concepts of Automation and Autonomy briefing paper[13]. The British Standards Institute released a Publicly available Specification 1883 further provides an ODD taxonomy for automated driving systems [6]. It decomposes ODD into the scenery, environmental conditions and dynamic elements. At the top level, scenery encompasses fixed road structures, drivable areas and junctions; environmental conditions include weather and illumination, and dynamic elements are traffic and subject vehicle. This taxonomy inspired the formal modelling of ODD. While some factors from the Automotive Domain are not directly transferable to Aviation, There exists similarities in the classification and will serve as a good starting point for modelling ODD in Aviation.

A scenario is defined with the major entities, their initial conditions, a timeline of significant events, and the environment [14]. Scenario development can be prescribed as a process beginning with the stakeholders' descriptions of the scenario and finishing with the generation of the corresponding executable specifications to be simulated. Based on the classification of Siegfried and his colleagues [15, 16], there are three types of scenarios that are produced in successive stages of the scenario development process: operational scenarios, conceptual scenarios, and executable scenarios: Operational scenarios are described in plain language describing the intended use of the system, conceptual scenarios are modelled formally in detail based on the simulation environment and executable scenarios contains the specifications to be executed in the simulator[17]. The critical elements in this operational scenario are the entities, their initial states and the events. They are usually required to be refined and augmented with additional information pertaining to simulation. They correspond to the operational scenarios that are presented as a part of ConOps at EASA guidance. The refinement of operational scenarios needs to be done by the simulation experts to achieve completeness and consistency. The result is conceptual scenarios that specify the piece of the world to be represented in the simulation environment in detail. The executable scenario is finally the specification of the conceptual scenario in a particular format to be processed by the simulation applications.

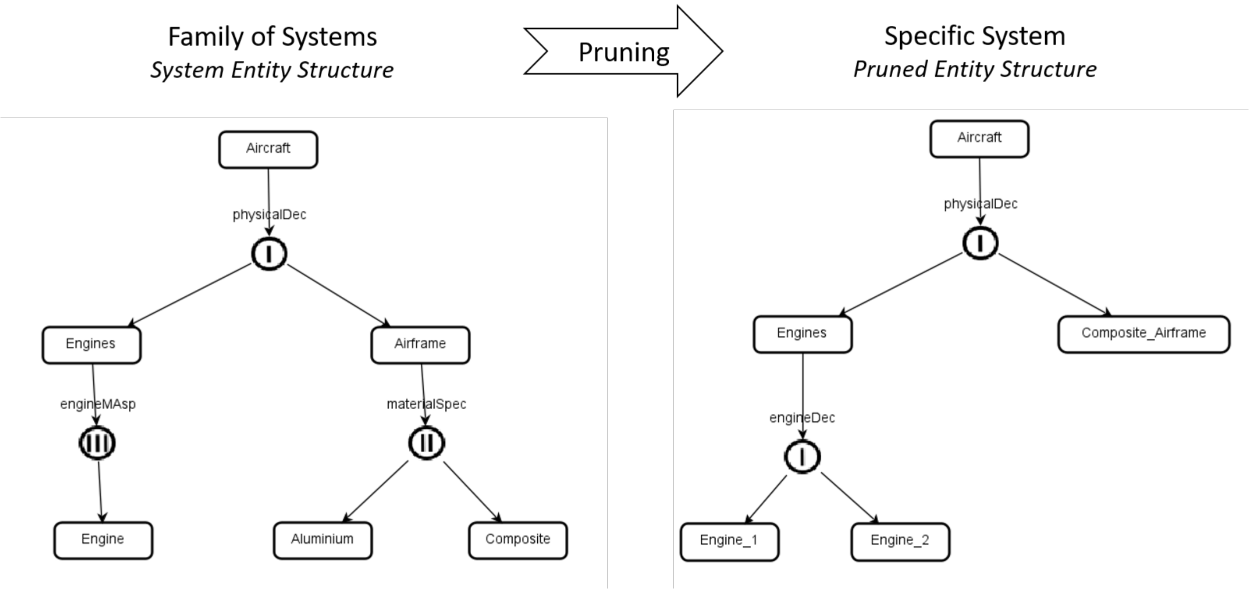

Fig. 2 An Example SES and PES

III. From Domain Model to Synthetic Data

On the one hand, the EASA guidance proposes an anticipated means of compliance for data completeness which says the data sets shall be evaluated against the completeness concerning the ODD; on the other hand, ODDs shall be developed using the operational scenarios from ConOps. This paper proposes to use the very same scenarios for data generation, and thereby we aim to achieve data completeness, in other words, ODD coverage automatically. The following sections explain how the operational scenarios defined in natural language within a ConOps can be used to design a domain model, which can then be used for generating scenarios and the ODD.

A. Background

Durak and his colleagues have been working using System Entity Structures (SES) for scenario definition since 2018. They present their model-based approach for scenario modelling using SES at [9].In [10] they propose a computational representation, a serialization approach for the SES. [18] introduces the tooling, the SES Editor [*].

SES is a high-level ontology that was developed for modelling variable-structure systems [19, 20]. It is a declarative knowledge representation scheme that specifies the structure of a family of systems using decompositions, component taxonomies, and coupling specifications and constraints. SES is a directed and labelled tree composed of Entity, Aspect, Specialization and Multi-Aspect nodes. An entity is an object of interest to which Variables can be attached; an Aspect specifies a decomposition relationship of an entity, while Specialization denotes its taxonomy. Vertical lines and Specializations represent aspects with a double line. A special kind of Aspect is Multiple-Aspect which denotes that the parent entity is a composition of multiple entities of the same type. Three vertical lines represent it.

Uniformity, alternating mode, strict hierarchy, attached variables, valid brothers, and inheritance are the six axioms

---

[*] https://github.com/aeronautical-informatics/SESTools

that regulate an SES. Uniformity enforces that any two nodes with the same labels have isomorphic subtrees. Alternating mode states that if a node is an Entity, the successor is either Aspect or Specialization, and vice versa. Strict hierarchy constraints a label from appearing more than once down any path of the tree. The attached variables endorse that variable types attached to the same item shall have distinct names. Valid brothers prevent having two brothers with the same label. Finally, inheritance endorses that Specialization inherits all variables and Aspects.

Pruning is the operation of assigning values to the variables and resolving the choices in Aspect, Multi-Aspect and Specialization relations, resulting in the Pruned Entity Structure (PES), a selection-free tree. During pruning, amongst several Aspect nodes for different decompositions of a system, a particular subset can be chosen based on the purpose. Specializations are resolved by choosing one of the variants. The cardinalities in Multi-Aspect relations are also needed to be specified in pruning. An example of SES and PES are given in Figure 2, where on the left-hand side is the SES for a family of aircraft, on the right-hand side is the PES of a particular aircraft with two engines and composite airframe. An automated pruning approach for simulations has been presented in [21].

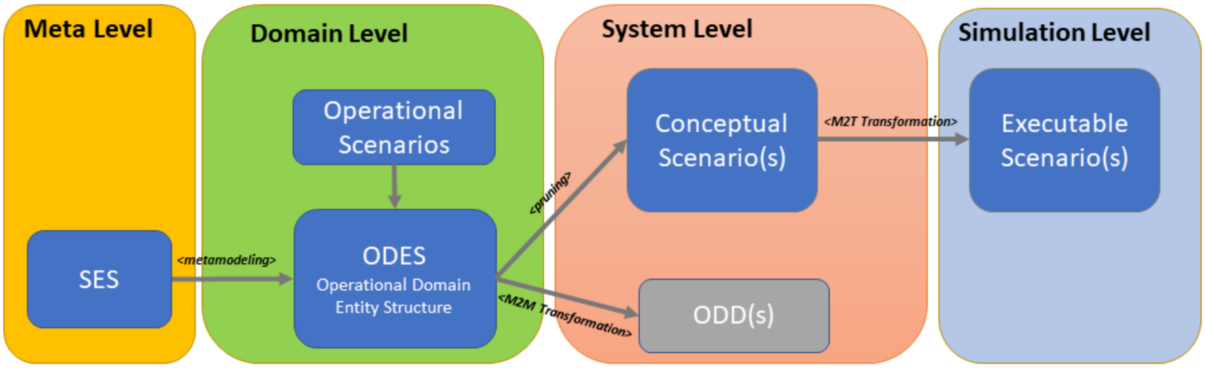

Fig. 3 Simulation-Based Data Generation

B. Approach

As presented in Figure 3, the proposed simulation-based data generation process is built upon the aforementioned previous work. SES is proposed at the meta-level as the high-level ontology. At the domain level, the operational scenarios given in ConOps are formalized with domain modelling into the Operational Domain Entity Structure (ODES). ODES' idea extends Operation-Specific Scenario Description Language (OSDL)[22], which helped define safe scenarios for a particular Unmanned Air Systems (UAS). OSDL utilizes the ranges of the types for the Variables to specify and enforce the safety limits given in CONOPS. ODES extends it with a specific focus on domain modelling to capture the ODD and create a formal basis for a scenario specification.

While the ASAM OpenODD Concept Paper [23] highlights the difference between ODD and a scenario, in this paper, we concentrate on the similarities and try to answer if we can use a single domain model for both of them. As stated in ASAM Concept Paper, ODDs and scenarios are related but surely not the same. ODD specifies the system's operating environment, whereas a scenario specifies the behaviour of the entities in an ODD. They both use the same ontology and the same entity pool. To our understanding, ODD constraints what can exist in a scenario, where the scenario describes what happens within the ODD. The behavioural specifications in scenarios are usually described with events. While a scenario provides the discrete events, it is expected that the continuous behaviour of the entities starting from the scenario initial conditions are simulated. The approach promises natural traceability between the ODD and the scenarios due to their standard domain model.

Figure 3 depicts that the conceptual scenarios will be generated from ODES with pruning and ODD with the model to text (M2T) transformations. It is possible then to conduct pruning automatically and generate scenarios for data generation that confirms the completeness objective required from the EASA guideline. The pruning can be carried out with different parameters, particularly for the values of the parameters, to generate a distinct set of scenarios for training, validation and verification. The next step is to generate executable scenarios from conceptual scenarios. Generating a scenario file for a particular simulator is usually a model to text (M2T) transformation. The execution of the scenarios with the simulator leads to a set of synthetic data that can be used for training, validation or verification purposes.

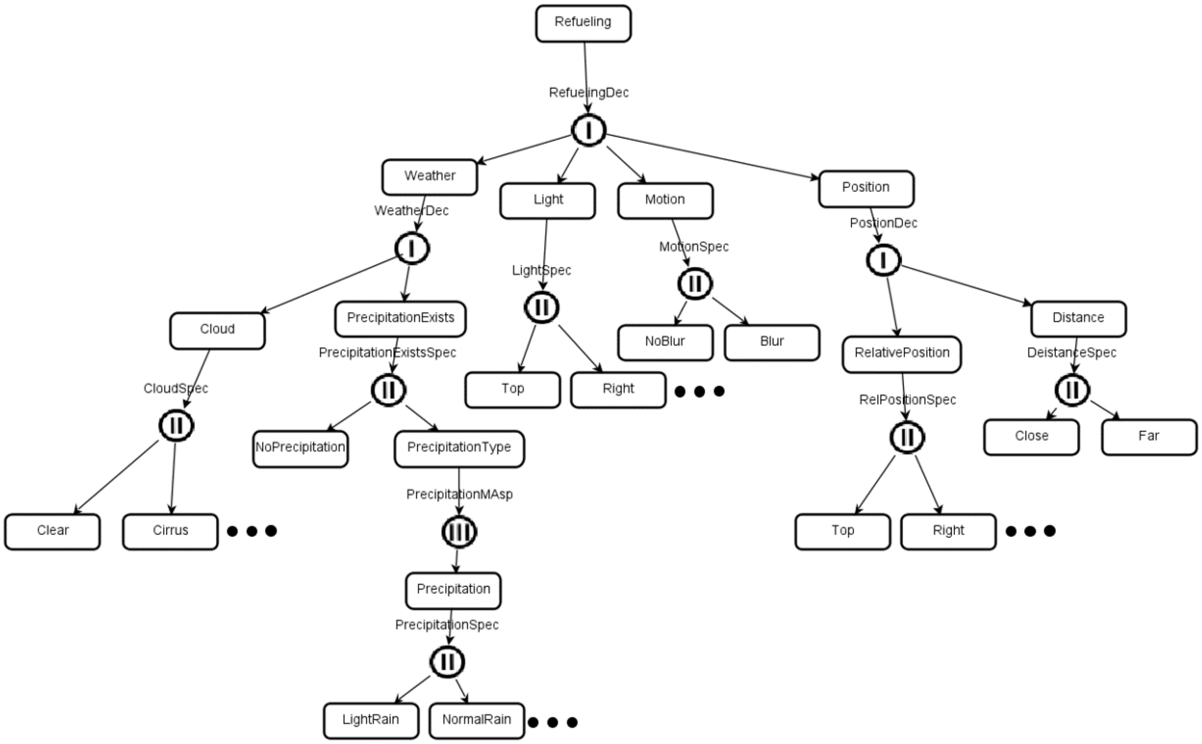

Fig. 4 An Excerpt from the ODES for Drogue Detection

IV. An Example: Drogue Detection for Autonomous Aerial Refueling

Aerial refuelling extends the airborne time by transferring fuel from a tanker aircraft to a follower aircraft during the flight. In probe and aerial drogue refuelling, the follower approaches the tanker from behind and inserts a probe through the drogue basket into the hose, where the follower receives the fuel. For automation of the docking phase, object detection is used to identify the drogue basket. This section is based on [24] where simulation-based verification is used for a detection algorithm that utilizes Convolutional Neural Networks (CNNs). The scenarios are executed in the simulation environment that is built using Unity 3D †.

An excerpt from the ODES for the drogue detection is depicted in Figure 4. The ODES for this application incorporated various aspects of refuelling, such as weather, light or relative position. Then each of them is elaborated and further modelled to represent all possible options. As an example, precipitation can be light rain, light snow, heavy rain and heavy snow. The quantitative characteristics of these enumerations are captured in their variables. The corresponding ODD excerpt is presented in a tabular format in Table 1. It describes the operating environment of the drogue detection algorithm in aerial refuelling. For example, as defined in the domain model, ODD presents the different types of clouds that we expect the drogue detection algorithm to work. An excerpt from a sample conceptual scenario specified in PES is given in Figure 5. All the elements of a scenario to simulate refuelling in this context is captured with this tree. There will be front light on a blue sky without precipitation where there are cirrus clouds. Motion blur also exists. The drogue will move towards the centre and far from the camera. This conceptual scenario converts to a format that executes in the simulation environment in Unity 3D.

The synthetic data being produced incorporates an image and the bounding box for the drogue on the ground truth. Figure 6 presents a sample result of a test that is being done using the synthetic data. The CNN-based drogue detection algorithm detected the drogue with 97% accuracy calculated using one of the standard metrics, Intersection over Union (IoU).

---

† https://unity.com

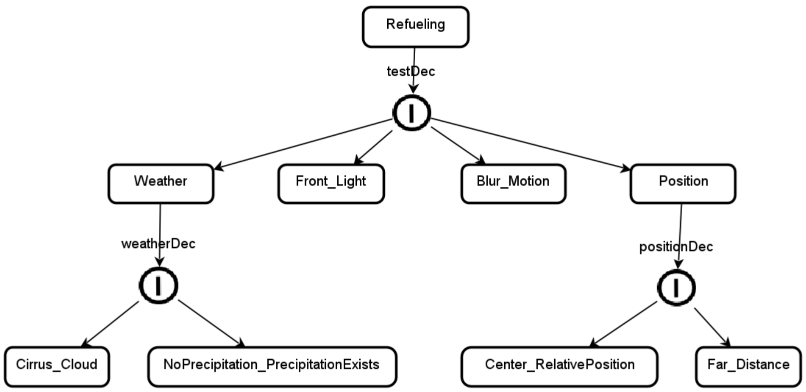

Fig. 5 An Excerpt from a Sample Conceptual Scenario

Environmental Conditions	
Precipitation	None, Light Rain, Light Snow, Heavy Rain, Heavy Snow
Light	Front, Top, Right, Left, Bottom
Motion Blur	Yes, No
Dynamic Elements	Drogue
Relative Position	Top, Bottom, Center, Right, Left
Distance	Close, Far
Scenery	
Clouds	Clear, Cirrus, Cirrostratus, Altostratus, Stratocumulus, Stratus, Cumulus, Cirrocumulus, Cumulonimbus , Altocumulus

Table 1 An Excerpt from Drogue Detection ODD

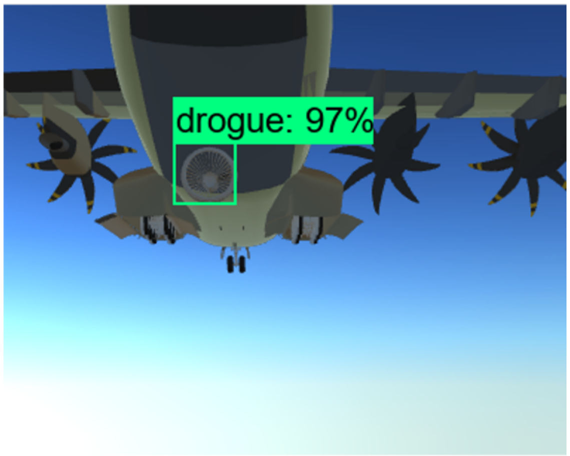

Fig. 6 An Example Test Result with a Synthetic Data.

V. Outlook

Using synthetic data comes with various challenges, such as data verification. However, it can also provide means to generate evidence regarding its traceability, completeness and consistency automatically. This paper presents a model-based approach that promotes domain modelling for specifying operational scenarios and the Operational Design Domain (ODD). When the standard domain model is taken as the single source of truth, it may naturally provide the required traceability between operational scenarios and the ODD. The automated scenario generation may then be used to guarantee consistency and completeness.

The data management process is yet to be exercised and matured in various industrial AI-based systems development activities. This paper, nonetheless, includes an early example to illustrate the realization of the proposed approach. Both the approach and the tooling require further development. The lessons learned from the initial use case are now driving our future work that extends SES at the meta-level further towards supporting intelligent system entity structures. Furthermore, the tooling is under further development to support scalability and modelling ergonomy.

Acknowledgements

This work is partially funded by the H2020 ECSEL 2018 IA under Grant Agreement No 826276 - CPS4EU: Cyber-Physical Systems for Europe.

References

[1] EASA, "Artificial Intelligence Roadmap, A Human-Centric Approach to AI in Aviation, Version 1.0," , 2020.

[2] Torens, C., Durak, U., and Dauer, J. C., "Guidance and Regulatory Frameworks for Machine Learning in Aviation," *AIAA Scitech 2022 Forum*, 2022.

[3] EASA, "Concept Paper First Usable Guidance for Level 1 Machine Learning Applications," , 2021.

[4] EASA, "Concepts of Design Assurance for Neural Networks (CoDANN)," `https://www.easa.europa.eu/document-library/general-publications/concepts-design-assurance-neural-networks-codann`, 2020.

[5] 1730™-2010, I. S., "IEEE Recommended Practice for Distributed Simulation Engineering and Execution Process (DSEEP)," 2011.

[6] Jafer, S., and Durak, U., "Tackling the complexity of simulation scenario development in aviation," *Proceedings of the Symposium on Modeling and Simulation of Complexity in Intelligent, Adaptive and Autonomous Systems*, 2017, pp. 1–10.

[7] Jafer, S., Chhaya, B., Zeigler, B. P., and Durak, U., "SES and Ecore for Ontology-based Scenario Modeling in Aviation Scenario Definition Language (ASDL)," *International Journal of Aviation, Aeronautics, and Aerospace*, Vol. 5, No. 5, 2018, p. 4.

[8] Jafer, S., Chhaya, B., and Durak, U., "Graphical specification of flight scenarios with aviation scenario defintion language (ASDL)," *AIAA modeling and simulation technologies conference*, 2017, p. 1311.

[9] Durak, U., Pruter, I., Gerlach, T., Jafer, S., Pawletta, T., and Hartmann, S., "Using System Entity Structures to model the elements of a scenario in a research flight simulator," *AIAA Modeling and Simulation Technologies Conference*, 2017, p. 1076.

[10] Durak, U., Jafer, S., Wittman, R., Mittal, S., Hartmann, S., and Zeigler, B. P., "Computational representation for a simulation scenario definition language," *2018 AIAA Modeling and Simulation Technologies Conference*, 2018, p. 1398.

[11] Kapurch, S. J., *NASA systems engineering handbook*, Diane Publishing, 2010.

[12] SAE, "J3016_202104 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," `https://www.sae.org/standards/content/j3016_202104/`, 2021.

[13] "Unmanned Aircraft Systems and the concepts of Automation and Autonomy," , Apr 2020. URL `https://www.eurocockpit.be/positions-publications/unmanned-aircraft-systems-and-concepts-automation-and-autonomy`.

[14] Topçu, O., Durak, U., Oğuztüzün, H., and Yilmaz, L., *Distributed simulation: A model driven engineering approach*, Springer, 2016.

[15] Siegfried, R., Laux, A., Rother, M., Steinkamp, D., Herrmann, G., Lüthi, J., and Hahn, M., "Scenarios in military (distributed) simulation environments," *Spring Simulation Interoperability Workshop (S-SIW)*, 2012.

[16] Siegfried, R., Oguztüzün, H., Durak, U., Hatip, A., Herrmann, G., Gustavson, P., and Hahn, M., "Specification and documentation of conceptual scenarios using Base Object Models (BOMs)," *Spring Simulation Interoperability Workshop*, Simulation Interoperability Standards Organization (SISO) San Diego, CA, 2013.

[17] Durak, U., Jafer, S., Beard, S. D., Reardon, S., Murphy, J. R., Crider, D. A., Gerretsen, A., Lenz, H., Macchiarella, N. D., Rigby, K. T., et al., "Towards a Standardization for Simulation Scenario Development in Aviation-Panel Discussion," *2018 AIAA Modeling and Simulation Technologies Conference*, 2018, p. 1395.

[18] Chandra Karmokar, B., Durak, U., Jafer, S., Chhaya, B. N., and Hartmann, S., "Tools for Scenario Development Using System Entity Structures," *AIAA Scitech 2019 Forum*, 2019, p. 1712.

[19] Kim, T. G., Lee, C., Christensen, E. R., and Zeigler, B. P., "System entity structuring and model base management," *IEEE transactions on systems, man, and cybernetics*, Vol. 20, No. 5, 1990, pp. 1013–1024.

[20] Zeigler, B. P., *Multifacetted modelling and discrete event simulation*, Academic Press Professional, Inc., 1984.

[21] Folkerts, H., Pawletta, T., Deatcu, C., and Zeigler, B. P., "Automated, reactive pruning of system entity structures for simulation engineering," *2020 Spring Simulation Conference (SpringSim)*, IEEE, 2020, pp. 1–12.

[22] Torens, C., Durak, U., Nikodem, F., Dauer, J. C., Adolf, F., and Dittrich, J. S., "Adapting Scenario Definition Language for Formalizing UAS Concept of Operations," *2018 AIAA Modeling and Simulation Technologies Conference*, 2018, p. 0127.

[23] ASAM, “ASAM OpenODD: Concept Paper,” `https://www.asam.net/index.php?eID=dumpFile&t=f&f=4544&token=1260ce1c4f0afdbe18261f7137c689b1d9c27576#top-02-04-understanding_odd`, 2021.

[24] Ellis, O. S., and Durak, U., “Simulation Based Verification of Drogue Detection Algorithms for Autonomous Aerial Refueling,” *AIAA Scitech 2020 Forum*, 2020.