Continuous Modeling in Aircraft Mission Analysis and Simulation

Yuchen Zhang(✉) and Jing Zang

Chinese Aeronautical Establishment, Chaoyang District, Beijing, China
zhangyc082@avic.com

Abstract. Mission analysis is the initial process in aircraft development. Its goal is to understand how the aircraft will operate and interact with other systems to accomplish its mission. As a system of systems (SoS) level development process, mission analysis involves various multi-disciplinary and cross-domain models, such as battlefield plotting models, scenario visualization models, SoS architecture models, and multi-agent wargaming models. This results in a heavy modeling workload and presents a significant challenge in maintaining model consistency. To ensure model transferability and consistency, this paper introduces a continuous modeling method for aircraft mission analysis. With the help of a modeling language-independent ontology, it achieves preliminarily model transformation and data transfer between the four kinds of models used in aircraft mission analysis. According to the practice, the continuous method helps improve the continuity and consistency of models and reduce the workload of building and maintaining models.

Keywords: Continuous engineering · Mission engineering · Model transformation · System of systems design

1 Introduction

Mission analysis is the initial process in aircraft development. It initiates the life cycle of aircraft by defining mission problem, developing concept of operations (ConOps), and capturing top-level requirements and validation criteria of aircraft [1]. As a system of systems (SoS) level development process, mission analysis involves various kinds of multi-disciplinary and cross-domain models. This can result in a heavy modeling workload and presents a significant challenge for maintaining model consistency.

Continuous modeling is a fresh idea derived from continuous engineering. It aims to achieve continuous and efficient transfer of models and data between different tools, thus allowing for the reuse or automatic transformation of models to improve the quality and efficiency of modeling [2]. Nowadays, much research has been done on continuous modeling at system-level, with model transformation achieved between SysML, Modelica, AADL, UML models [3–8]. However, there are few theories and methods for model transformation at system-of-systems level.

S. Fu (Ed.): APISAT 2023, LNEE 1050, pp. 491–498, 2024.
https://doi.org/10.1007/978-981-97-3998-1_42

This paper introduces a continuous modeling method for aircraft mission analysis. It first introduces the four types of models used in model-based aircraft mission analysis process. Then, an ontology-based continuous modeling method is introduced. The model transformation mechanism is studied to determine how metamodels of different system-of-systems level models are matched to a language-independent ontology, allowing models to be transformed from or refer to the ontology. In our practice, this method achieves preliminarily model transformation and data transfer between the four kinds of models used in process.

2 Modeling and Simulation in Aircraft Mission Analysis

Mission analysis is located at the upper left corner of the V-model of systems engineering. It initiates the life cycle of an aircraft by defining the mission problem, developing ConOps, and capturing top-level requirements and validation criteria. The goal of mission analysis is to understand how the aircraft will operate and interact with other systems to complete its mission (Fig. 1).

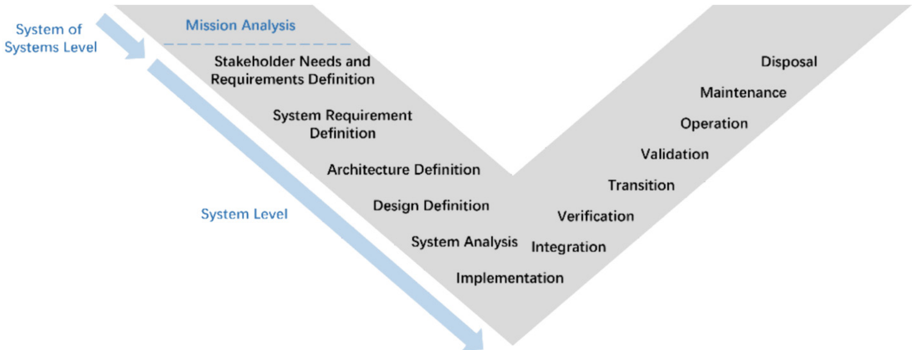

Fig. 1. Mission analysis in V-model of systems engineering

The process of mission analysis can also be represented as a small V-model, including mission definition, ConOps development, SoS architecting, and mission simulation. As a SoS level development process, mission analysis involves various kinds of multidisciplinary and cross-domain models. These include battlefield plotting models, scenario visualization models, SoS architecture models, and multi-agent wargaming models (Fig. 2).

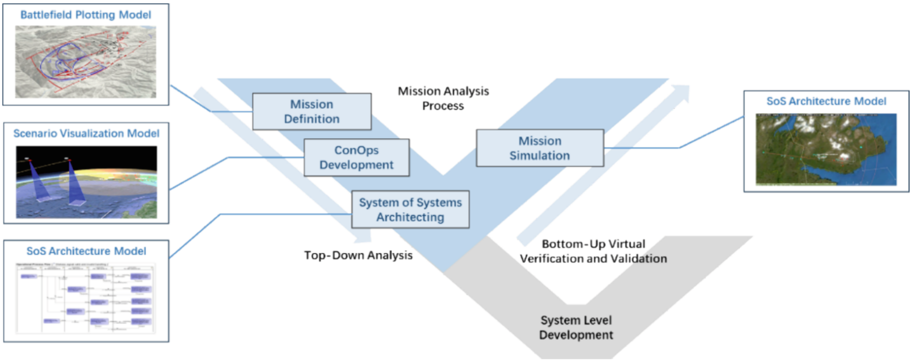

Fig. 2. Mission analysis process

Mission definition provides the overarching context and assumption for a mission. It describes the time frame, operational environment, blue and red forces, and the objectives of mission. Battlefield plotting model, which is a type of geographic information system (GIS) model, is built during this step to help the definition of basic mission information such as operational area, main force, attack direction and more.

ConOps development presents detailed operational scenario, including the ordered set of events, behaviors, and interactions for a specific set of systems. Scenario visualization model is used during this step to show the dynamic operational scenario with 3D simulation and implement time-space analysis.

System of systems architecting designs the mission architecture using a standard framework, such as unified architecture framework (UAF) in our practice. SoS architecture model provides a formal and systematic approach for architecture design, completing the development from strategic, operational and resource viewpoints, as well as from structure, connectivity, activity, sequence, state, and parameter aspects.

While the above three steps implement top-down analysis, mission simulation implements bottom-up virtual verification and validation. Multi-agent wargaming model is built and simulated during this step to determine whether the mission analysis results are correct.

3 Continuous Modeling Method

The four types of models used in aircraft mission analysis are interconnected. They describe the mission of aircraft from various perspectives and share some common data. Ensuring consistency of different models is crucial, and it is beneficial if parts of the model can be transferred and reused.

Continuous modeling aims to achieve seamless and efficient transformation between models, thus improving the consistency of models and efficiency of modeling. In continuous engineering, there are three levels of model transformation based on the level of continuity or automation [9].

The first level is manual modeling, where models are built by human and consistency is also maintained by human. This level can also be considered as discontinuous modeling.

The second level is interface-driven transformation, considered as low-level continuous modeling. Model transformations are achieved through end-to-end interfaces. This level is achieved through the definition of interfaces and directly data exchange between different modeling tools.

The third level is ontology-based transformation, considered as high-level continuous modeling. The smooth transformation is achieved through the definition of meta-model, where the semantic of different meta-models are mapped to the same ontology (Fig. 3).

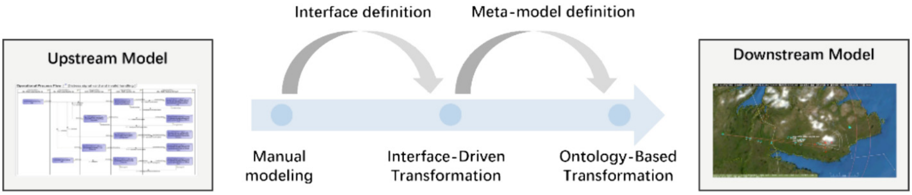

Fig. 3. Levels of continuous modeling

3.1 A Language-Independent Ontology

Ontology, which Originated from philosophy, is the study of the formal description and representation of human knowledge. It describes the basic concepts in a specific domain and captures their essential meaning in a language-independent way. Ontology is usually used for unified and consistent description of different models.

Referring to the four types of models mentioned above, we propose a reference ontology for aircraft mission analysis. This ontology contains the common concepts of the four models.

Figure 4 presents the ontology in a UML-style way. Concepts are shown as rectangular annotations, and relationships are shown as different lines. A solid line means *Association*, a line with hollow diamond means *Aggregation*, and a line with hollow triangle means *Generalization.*

The ontology is roughly divided into three layers: the mission, operational, and system layers. Concepts in the mission layer are demonstrated in green color. The core concept is *Mission*, which is described by defining its *Objective*, *Location*, and *Environment*. *Capabilities* are required to accomplish a mission. *Capability Gap* is obtained by comparing *Required Capability* and *Current Capability.* To fulfill the capability gap, *Capability Solution* is proposed, which is realized by *System of Systems.*

Concepts in the operational layer are presented in yellow color. They describe the operation of system of systems. Mission is further analyzed by dividing it into *Concept of Operation*, *Operational Scenarios* and *Operational Tasks*. *Operational Performers* are assigned to perform operational tasks. They are the users or stakeholders of the aircraft.

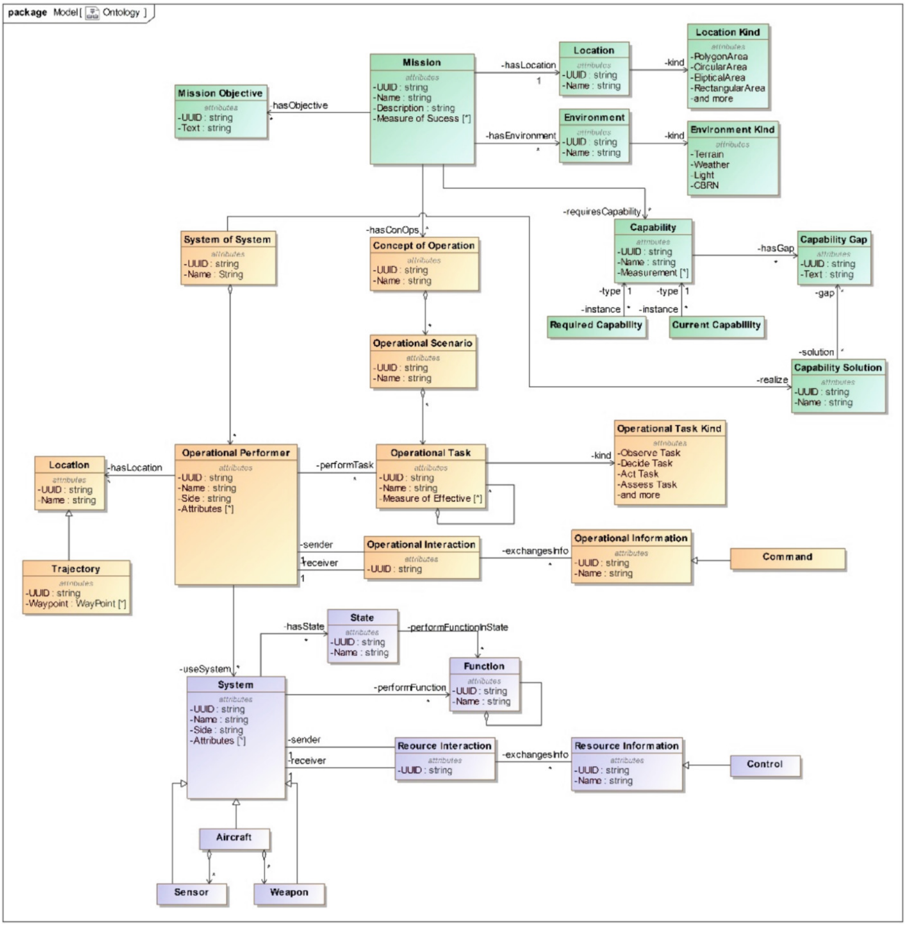

Fig. 4. Ontology for aircraft mission analysis

System layer concepts are shown in blue color. Operational performer uses *Systems* to perform its operational task. An aircraft is a kind of system in this ontology. It is described by designing its behavior (including *State*, *Function* and *Interaction*) and its structural composition.

The ontology demonstrates the analysis thread of aircraft mission analysis process, captures and identifies common and essential concepts from different models.

3.2 Ontology Based Model Transformation

The ontology proposed above provides a formal description framework for aircraft mission analysis. It can serve as a middleware for model transformation and data storage. Based on the ontology, this paper presents an ontology-based continuous modeling method for aircraft mission analysis.

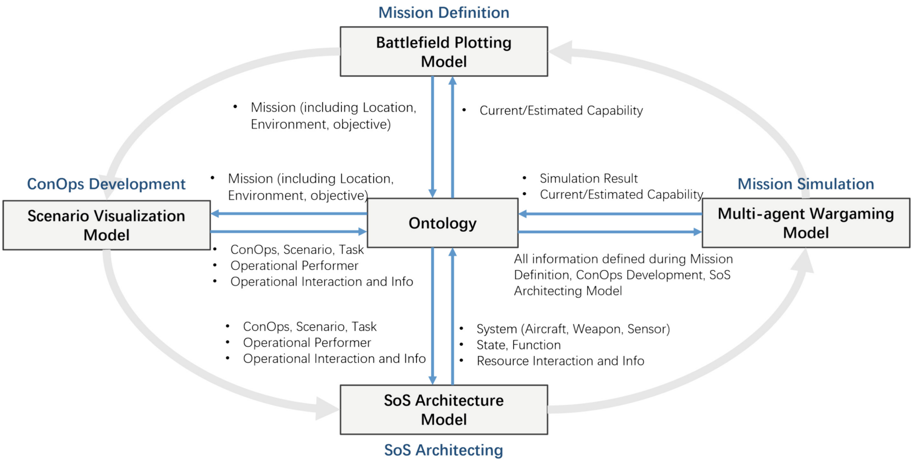

Fig. 5. Ontology as middleware of model transformation and data storage

Figure 5 presents the input and output of the four types of models used in aircraft mission analysis. They are matched to the reference ontology so that models can be transformed to and from the ontology.

Models can first transform to the ontology representation, then other kinds of models can be generated from the ontology. In this way, information defined in the upstream models can be transferred to or reused in the downstream models. This makes it possible to improve the model continuity and consistency. The model transformation process is automated or semi-automated when mappings between metamodels and ontology are defined, thus it avoids repeated and redundant modeling and avoids the inconsistency occurring in the modeling process.

3.3 Model Transformation Example

With the ontology-based continuous modeling method, consistency of different models can be improved. Blow an example of the model transformation is introduced. The transformation mechanism involves mapping the metamodel semantics of different modeling languages to the reference ontology. With the ontology as a medium, different models can be translated to each other.

Figure 6 shows an example of mission-level behavior model transformation. The two models are the SoS architecture model and the multi-agent wargaming model. In our aircraft mission analysis process, the operational process of a fix wing UAS is first described in SoS architecture model using UAF framework. Then, the operational process is simulated and evaluated with a multiagent wargaming model in Maxsim software. In the past, these two types of model were built independently, and the consistency was ensured by engineers. With the continuous modeling method, the wargaming model now can be automatically transformed from the SoS architecture model, and the consistency is ensured by transformation mechanism.

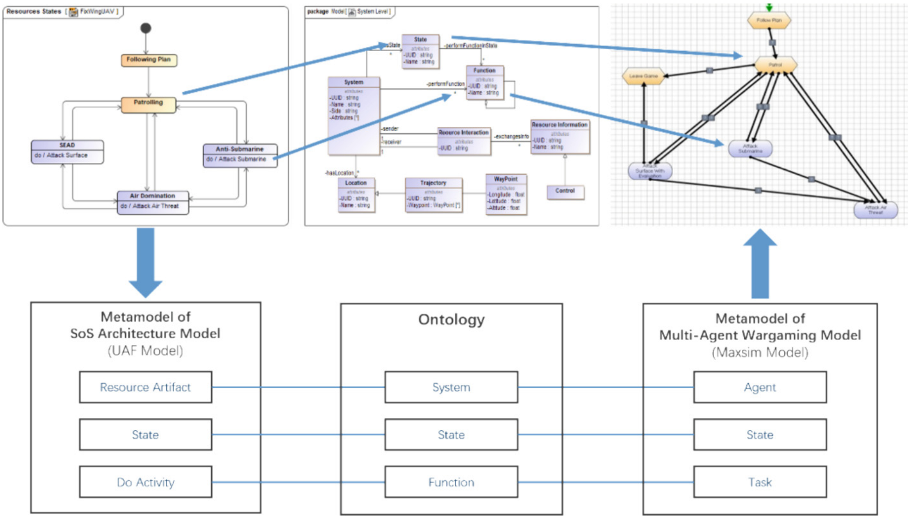

Fig. 6. Model transformation example

The metamodel semantics of two modeling languages are mapped to the ontology. As shown in Fig. 6, a fix wing UAS is described as a *Resource Artifact* in UAF and as an *Agent* in Maxsim. However, they can both be matched to the *System* concept in ontology to have a unified description. Similarly, the behavior of a fix wing UAS is described as *Do Activity* in the state machine in UAF, which has the same meaning as *Task* in Maxsim. Thus, they are matched to *Function* in the ontology. With the ontology as medium, semantics in two modeling languages are mapped together, allowing the wargaming model to be transformed from SoS architecture model. There is no longer a need to build model twice in different software, and fewer inconsistencies will occur.

4 Conclusion

In this paper, a continuous modeling method for aircraft mission analysis is introduced. First, a language-independent ontology is proposed. It captures essential concepts for aircraft mission analysis and provides generic description framework. Second, the ontology-based model transformation mechanism is studied, demonstrating how different models are transformed by matching them to the reference ontology. With the continuous modeling method, four kinds of models used in mission analysis are successfully transformed.

According to our practice, this continuous method initially confirms its feasibility and usability. It can help improve the continuity and consistency of models while reducing the workload of building them.

5 Future Work

At the end of the paper, we would like to discuss some deficiencies of this work and outline our expectation for future work.

First, our primary motivation for this work is to ensure the consistency of the four types of models used in our aircraft mission analysis practice. The ontology built in this paper also serves for this purpose. However, mission analysis involves not only these four models, so the ontology may be not complete and may require further extension. Our ontology can serve as a foundation for future work.

Second, our current work still lacks a synchronization mechanism for models. When one model changes, it is necessary to determine how other models will know about the change and synchronize accordingly. Currently, model transformation is triggered by humans not automatically triggered by changes. An automated change synchronization mechanism is needed.

References

1. Walden, D.D.: Systems engineering handbook: a guide for system life cycle processes and activities (2015)
2. Fitzgerald, B., Stol, K.-J.: Continuous software engineering and beyond: trends and challenges. In: Proceedings of the 1st International Workshop on Rapid Continuous Software Engineering (2014)
3. Nikolaidou, M., et al.: Simulating SysML models: overview and challenges. In: 2015 10th System of Systems Engineering Conference (SoSE). IEEE (2015)
4. Paredis, C.J.J., et al.: 5.5. 1 An overview of the SysML-modelica transformation specification. INCOSE Int. Symp. **20**(1) (2010)
5. Kordon, F., (eds.) et al.: Embedded Systems: Analysis and Modeling with SysML, UML and AADL. Wiley (2013)
6. Lasalle, J., et al.: SysML to UML model transformation for test generation purpose. ACM SIGSOFT Softw. Eng. Notes **36.1**: 1–8 (2011)
7. Sindico, A., Di Natale, M., Panci, G.: Integrating SysML with Simulink using open-source model transformations. SIMULTECH (2011)
8. Morozov, A., Mutzke, T., Ding, K.: Automated transformation of uml/sysml behavioral diagrams for stochastic error propagation analysis of autonomous systems. ASCE-ASME J. Risk Uncertainty Eng. Syst. Part B Mech. Eng. **8**(3), 031102 (2022)
9. Jing, Z., Qian, W., JiaQiang, Z.: Continuous modeling in aircraft top-level demonstration (2018)