applied sciences

MDPI

Article

A Framework of Modeling and Simulation Based on Swarm Ontology for Autonomous Unmanned Systems

Xinghai Gao [1,*], Gang Xiao [2], Kai Xie [3], Weijia Wang [2], Yuhua Fu [3], Chuangye Chang [1] and Zhuoqi Wang [1]

[1] Institute of Unmanned Systems, Beihang University, Beijing 100191, China; changchy@buaa.edu.cn (C.C.); wangzhuoqi@buaa.edu.cn (Z.W.)
[2] National Key Laboratory for Complex Systems Simulation, Beijing 100101, China; m18611481731@163.com (G.X.); venus_wwj@126.com (W.W.)
[3] Laboratory of Electromagnetic Space Cognition and Intelligent Control Technology, Beijing 100191, China; eeixiekai@163.com (K.X.); xtsfyh@126.com (Y.F.)
* Correspondence: gaoxh@buaa.edu.cn

Abstract: For the emerging autonomous swarm technology, from the perspective of systems science and Systems Engineering (SE), there must be novel methodologies and elements to aggregate multiple systems into a group, which distinguish the general components with specific functions. Here, we expect to provide a presentation of their existence in swarm development processes. The inspiration for our approach originates from the integration of swarm ontology, multiparadigm modeling, multiagent systems, cyber-physical systems, etc. Therefore, we chose the model-driven architecture as a framework to provide a method of model representation across the multiple levels of abstraction and composition. The autonomous strategic mechanism was defined and formed in parallel with Concept of Operations (ConOps) analysis and systems design, so as to effectively solve the cognitive problem of emergence caused by nonlinear causation among individual and whole behaviors. Our approach highlights the use of model-based processes and their artifacts in the swarm mechanism to integrate operational and functional models, which means connecting the macro- and micro-aspects in formalism to synthesize a whole with its expected goals, and then to verify and validate within an L-V-C simulation environment.

Keywords: swarm ontology; autonomous system; model-driven; multiparadigm modeling; model-based systems engineering

Citation: Gao, X.; Xiao, G.; Xie, K.; Wang, W.; Fu, Y.; Chang, C.; Wang, Z. A Framework of Modeling and Simulation Based on Swarm Ontology for Autonomous Unmanned Systems. *Appl. Sci.* **2023**, *13*, 9297. https://doi.org/10.3390/app13169297

Academic Editor: Marco Troncossi

Received: 18 April 2023
Revised: 4 August 2023
Accepted: 9 August 2023
Published: 16 August 2023

1. Introduction

The current trends in the development of unmanned systems are evolving to add the characteristics of autonomy, adaptability, and intelligence; furthermore, completely heterogeneous unmanned systems can be aggregated into a swarm to address various types of missions, making them more complex. The development of swarm is a field of AI that focuses on the architecture of swarm mechanisms to enable a large number of individual autonomous systems to act in a coordinated way, with decentralized control, automatic interactions, and self-organization, even in real time, resulting in the emergence of swarm across multiple levels of causation. When we think of a swarm as a complex system or a System of Systems (SoS), it is difficult to comprehend and analyze, and traditional engineers cannot use closed-form analysis or prediction techniques to fulfill the development of the ConOps of swarm and the design of individual system units. The basic concept of a modeling framework for such complex systems is based on the description of dynamic systems and defined as a formal ontology in a logic-based language [1]. On the other hand, in the emerging field of autonomous swarm technology, the design elements and the overarching architecture used to aggregate multiple unmanned systems into a group are always vague, even lacking or ignored. A proposed swarm Unmanned Aerial System (UAS) mission taxonomy is designed to provide building blocks for an overall top–down design methodology, and further, a decentralized control architecture and layered approach to be integrated as design elements for developing swarm UAS technology [2].

However, today, the application of digital technologies in swarm seems to be very significant for the evolution of digital engineering for mission, specification, design, integration, verification, and validation in complex systems. The mission-based architecture for swarm intends to integrate the mission doctrine, utilize composable elements, demonstrate modularity across missions, and be intuitive to the human operator [3]. Compared with traditional systems with transparent use cases and exact functions, the development of complex systems such as a swarm will intensely depend on the utilization of Modeling and Simulation (M&S), which may be the sole appropriate approach throughout the development of a swarm. In experimental platforms, the different simulation results are organized to group them depending on the missions or behaviors carried out by the swarm. Some behaviors, such as aggregation and collective movement, are quite basic to constitute more complex and high-level swarm tasks [4]. Although swarm technology is based on unmanned systems, artificial intelligence, etc., its application is still in the early stage. The enhanced capabilities of swarm systems can bring about obvious advantages for the achievement of mission tasks, such as distributed delivery and deployment, remote communication and command, persistent surveillance and reconnaissance, multi-sensor data collection and transmission, multitarget search and target, etc. [5].

In the traditional design of swarm systems, most designers are restricted to their respective engineering expertise, mainly focusing on communication networking, vehicle platforms, controls, sensors, individual autonomous agents, etc. Due to the scarceness in the background of system engineering, some significant challenges are faced from the perspective of mission conceptualization and operational contextualization. Therefore, we should rely on the general principles and processes of Model Based Systems Engineering (MBSE) to establish a new swarm-oriented technology application paradigm. To create and evolve a mission-effective swarm, traditional engineers must collaborate with system architects to consider the methodology of operation, design, and test when developing an autonomous unmanned swarm [3].

Here, we focus on a domain-specific application of a swarm and aim to be able to meet the expected mission efficiency and the requirements of the autonomous unmanned system within a single context of the development framework. We explore the ConOps of autonomous swarm and derive the design specifications of the Autonomous System (AS), both in a common architecture of viewpoints and views, as well as build a coherent lifecycle process to capture and track them.

The rest of the paper is organized as follows: Section 2 reviews some related concepts, methodologies, and research works along our research roadmap; Section 3 describes the architecture framework of M&S based on swarm ontology; Section 4 discusses some different methods of M&S and their applications in our approach; and Section 5 concludes the paper.

2. Related Works

The elements of our methodology were mainly from the heuristics of the ontology of knowledge representation and artificial intelligence, such as the following: Cyber-Physical System (CPS) with hybrid-networked computational and engineered physical elements, behavior-based system modeled by the Discrete Events Systems Specification (DEVS), agent-based model for an adaptive system, Multi-Paradigm Modeling (MPM) as the foundation for CPS engineering, digital twins applied in a Live–Virtual–Constructive (L-V-C) platform, the architectural framework models of the lifecycle of a complex swarm system, from conceptualization to contextualization on MBSE, etc. These elements represent a common approach for design and operation in order to meet the capabilities and requirements for any intended swarm missions.

2.1. Swarm Ontology

An ontology is a system of concepts to represent an explicit specification of a conceptualization of a complex system, which is borrowed from philosophy, which is the systematic study of existence (being) in general. Ontology is a top-level abstraction in the field of AI and is applied in a formal declarative modeling method of logic-based statements. In the context of our research, ontology is regarded as a knowledge representation approach reflecting concepts with their properties and relationships in specific domains, and also constraints and rules governing those properties and relationships. For an autonomous swarm, it could combine intelligence and other characteristics expected in humans with technical characteristics such as cognition, behavior, perception, and execution in the form of formal models and structural frameworks. In the context of AI, we can define a set of concepts (such as an object, relationship, interaction, function, or other) in a knowledge domain to describe the ontology of the system, in which human-understandable terms and axioms help to explain its meaning, and also well-formed constraints exist among these terms to support machine reasoning.

Because an autonomous system swarm has the characteristics that individuals follow simple behavior or logic rules and can stimulate collective behavior with a decentralized coordination mechanism, M&S might be the most potential approach to solve problems of such complex systems. However, from the perspective of M&S, the relationship between the emergence behavior embodied by the swarm as a whole and the simple behavior of the individual needs to be effectively represented in the swarm ontology, which is not only the highest level of abstract form, but also the most basic theoretical foundation of this study [1].

With the growing complexity of the collaboration between multiple autonomous systems as well as humans and robots, the IEEE-RAS (Robotics and Automation Society) Robotics and Automation Ontology working group formed a standard ontology and associated methodology for knowledge representation and reasoning in robotics and automation, together with the representation of concepts in an initial set of application domains. In order to cover the domain of robotics and automation, the working group developed a bottom–up and top–down approach with four subgroups: Upper Ontology/Methodology (UpOM), Autonomous Robots (AuRs), Service Robots (SeRs), and Industrial Robots (InRs). The focus of the AuR subgroup is future unmanned systems working in teams with other unmanned vehicles to share situational awareness and coordinate activities, such as Unmanned Aerial Vehicle (UAV), Unmanned Ground Vehicle (UGV), and Autonomous Underwater Vehicle (AUV). For the level of an individual system, the ontology provides help to enable the decision making, control strategies, sensing abilities, mapping, environment perception, motion planning, communication, autonomous behaviors, etc. [6].

2.2. Model-Based Paradigm

Model-based paradigms have become a powerful driver for systems engineering and the launching of system thinking, in which processes and activities of the system life cycle shape the context of systems engineering practice, and M&S constitutes the core mechanism and is beneficial to helping connect the stages of MBSE [7]. The essence of the transformation of MBSE is a continuous shift to a system development process supported by the continuity and traceability of models. When referring to models, we highly advocate executable models for simulation in an Experimental Frame (EF). Therefore, advanced M&S technologies and methods have become the key enabler driving the life cycle of systems engineering.

MBSE relies on the trend of adopting a unified formal model throughout the system's life cycle process activities, and has been struggling to seek an appropriate means to connect the blueprint models that describe the system architecture in an iterative and incremental way, and then be used to validate these stakeholders and verify by its specifications, even early in prototyping development. In order to support the multidisciplinary practice in SE, the communities in systems engineering have been adopting many modeling approaches and tools, which range from mission or business analysis to requirement definition, system structure, and interfaces, and even to system behavior, and so on. We need a formal modeling language to combine visual graphics for human communication and a metamodel with constructs and rules needed to build specific models within a domain of interest, and a specification to be exchanged spanning diverse modeling paradigms and tools. The Unified Architecture Framework (UAF) for SoS, System Modeling Language (SysML) for systems, and its collections of visual diagrams, which are derived from the Unified Modeling Language (UML), have been standardized and continuously updated by the Object Management Group (OMG) and International Council on Systems Engineering (INCOSE) at the initiative of MBSE.

Furthermore, from the perspective of system paradigm evolution, a CPS is defined as an autonomous, adaptive, and intelligent system in which Communication, Computing, and Control (C3) components dominate physical behavior. M&S will involve multiple levels of concept, specification, and operation in CPS Engineering (CPSE) to use formal methods to express basic concepts (such as structures, states, events, concurrency, etc.) and their relationships, to represent the system studies (problem) in the real world as a model, and to verify the implementation of system behaviors and functions (solution) by executing various simulation instructions through a simulation engine [8]. Therefore, a CPS is an advanced hybrid form of system and can be modeled in some very distinctive ways, such as computational elements for discrete modeling; physical elements for continuous modeling; communication networking for probabilistic scenarios; and even game theory for operations research to support planning and decision making [9].

2.3. Multiparadigm Modeling

The essential part of our research is the application of M&S throughout the swarm life cycle to initiate MBSE; thereby, it needs to collaborate on various models at various levels and stages of concepts, specification, and operation, etc. Hereby, we should select Multi-Paradigm Modeling (MPM) as an underlying approach to synthesize many different modeling techniques for a swarm to achieve the integration of modeling paradigms, model transformation techniques, and compositional modeling methods. MPM is a key method that provides a solid foundation for the design process for a CPS.

The most critical issue is the distinction between modeling patterns, which requires us to analyze the respective characteristics of exploratory and constructive modes and the way they are combined. The two modeling modes adopt different properties in order to reach their respective goals, because of the originations from different schools of thought with different goals. For example, exploratory modeling grows from the bottom–up and focuses on describing an open world; constructive modeling is a perspective from the top–down that focuses on proposing a closed-form solution. However, it is the intrinsic differences between the two modeling patterns that form the necessary complementarity. Exploratory modeling aims to explain domain concepts by describing them, usually in the form of classification, and typically uses modeling languages such as Web Ontology Language (OWL) to specify taxonomy and Description Logic (DL) for reasoning. On the contrary, the purpose of constructive modeling is to establish a domain solution by prescribing nominal types for all elements of the domain, and is supported by modeling languages, i.e., SysML and first-order logic via constraint languages such as Object Constraint Language (OCL), understanding instances of all types through instantiation relationships [10].

Given the goal of hierarchical swarm modeling, different models are actually required at different levels of abstraction. However, we can exquisitely apply the same modeling language to simultaneously solve the challenges of cross-abstraction-level semantic association and executable model continuous transformation. For example, exploring the concept of swarm operation is still an open domain of knowledge, which means that a swarm ontology will benefit the advantages of exploratory modeling; for the development of autonomous unmanned systems, we hope to guide the rapid configuration of a system based on a metamodel to specify other models and its instances across multiple domains.

At this point, the metamodel will become the model template for system construction. The most intuitive way to solve the semantic connection between the two modeling methods is to use the same modeling language, such as SysML.

2.4. Behavior-Based System for Autonomy

The concept of a swarm derives from biology and refers to a group of a large number of biological individuals working together to accomplish some collective behaviors, where a single individual or any uncooperative individuals cannot perform such useful tasks without the help of the rest of the swarm, such as the flocking of birds and schooling of fish, colonies of bees, and so on.

A reasonable way to develop and evaluate intelligence is to understand the ability of natural organisms to handle real-world complexity. The main goal of behavior-based systems is to solve the control problems and applications of single or multiple robots (autonomous systems). The concept of basic behaviors has an explicit modular nature, and behavior-based systems can be presented as building blocks with the properties of functional decomposition and sequential interdependencies to enable the autonomous system to reason in a complex challenging environment and grant it adaptive behavior [11]. A swarm ontology creates a common conceptualization that can be shared in model transformation and association by all those involved in an engineering development process [12].

In line with the tendency of the top–down methodology in MBSE, the high-level functional models should be specified before decomposing to lower-level functions. Especially supported by SysML, we make use of the diagrams of use case, activity, sequence, and state machine to model system behaviors in a consistent and coherent way successively throughout the behavior abstraction level of a black box in a larger environment, business flow with several functional divisions, interactions between a group of elements, and a specific state and events in an element unit. Our approach first focuses on behaviors in a high-level swarm and then decomposes them into the distributed system by developing modular, scalable, reusable, and tailored behaviors that execute the intended swarm operation [13].

We refer to the autonomous strategic mechanism of an autonomous system as the process of decision making by taking information about the environment via sensors, which also is a computational architecture on the level of an individual. To effectively apply the State Analysis method in the context of complex control systems, [14] provides an ontological definition of the concepts and relations to map State Analysis onto a practical extension of SysML.

The DEVS in an M&S-driven paradigm should be implemented by integrating a swarm ontology to offer life cycle control according to scenarios, and it becomes particularly important to predict and test the behaviors of both swarm and individuals. DEVS is a popular model-based approach to perform modeling and simulation to connect the activities of MBSE, and combines discrete, continuous, and hybrid models in a formal way. The block-based unit is modeled to build a modular and hierarchical structure, which can be interconnected through input or output ports with self-contained behavior [15]. A model transformation approach is proposed to simulate hybrid SysML models under a DEVS framework and to depict hybrid models, and simulation-related metamodels with discrete and continuous properties are extracted from SysML diagrams, which refer to a Block Definition Diagram (BDD), Internal Block Diagram (IBD), State Machine Diagram (SMD), and Parameter Diagram (PAR). Following the OMG's Meta Object Facility (MOF), DEVS metamodels are constructed based on the definition of DEVS formalism, including discrete, hybrid, and coupled models. Such an approach may facilitate the modelers to use a DEVS-based simulator to confirm complex systems models [16].

2.5. Digital Twins in an L-V-C Platform

An autonomous CPS must have learning capabilities to adapt to external environments, so it is also an agent-based system. The subject of agent-based development and testing platforms makes a clear connection between an intelligent, adaptive, and autonomous CPS and its Digital Twin (DT) in the form of intelligent software agents within a situated virtual context that replicates the nature of the physical environment [8].

Through the digital twin of a system, we have the possibility to analyze and test various operational scenarios on a complex swarm before its physical implementation. A digital twin is a virtual representation which is based on the digitalization of physical systems to allow modeling the state of a physical entity or system, and it is created by digitalizing data collected from physical entities through sensors, so various predictions could be made by understanding the behavior of the physical entity [17]. A digital twin consists of three essential components: a physical product, a virtual representation of that product, and the bidirectional data connections that feed data from the physical to the virtual representation, as well as information and processes from the virtual representation to the physical [18]. Although a DT deeply relies on its current simulation, it has some significant distinctions from simulations, such as DTs dedicated to whole-life-cycle operational processes rather than some detailed design and testing, and behaving in the real world by obtaining real-time data from a physical product, not just for virtual training.

In an M&S environment supported by MPM, we go beyond the limitations of traditional computational models to acquire the full capability needed to investigate the emergent behavior of complex systems, known as an L-V-C simulation, including a live simulation where the model involves humans interacting; a virtual simulation where the model is simulated by a hybrid of humans and computer-generated experiences; and constructive simulations where the model is entirely implemented in a digital computer and even has a high level of abstraction [7].

Throughout the system's lifecycle in MBSE, one of the primary tenets is to reuse the concept of a system. The digital twin in an L-V-C simulation as an analytics framework provides new opportunities to operationalize early investments in system models to perform analysis before the physical asset is fielded. Additionally, by leveraging early efforts to define the DT, analytic processes are used to verify and validate [19].

In order to establish a substantial and permanent linkage between physical products and virtual representations, we adopt a Unified Repository (UR) to include various sensor reports, simulation data, control parameters, etc. Virtual development tools and physical collection tools populate a Unified Repository of data to achieve two-way connectivity between virtual representations and physical products, thereby forming a virtual/real hybrid simulation environment platform that covers the interaction of real and virtual spaces [20].

2.6. The Taxonomy of the Methodologies/Methods

We would like to review the relative research works coming from the references (see Table 1) mainly according to the taxonomy of the methodologies/methods of swarm ontology, modeling and simulation for cyber-physical systems, model-driven engineering, behavioral simulations, and virtual/real hybrid simulation environments, which inspire our approach to build a comprehensive modeling and simulation framework to combine models for swarm ontology, which present complex operational concepts and system specifications, represent the dynamic structure and behavior of autonomous systems, and incorporate decentralized communication, distributed control, and adaptive planning to design elements of cyber-physical systems.

Table 1. The taxonomy of related methodologies/methods.

No.	Methodologies/Methods	Key Ideas and Its Usage	Refs.
1	Swarm ontology	Ontology: A set of concepts of the problem or knowledge and the relationships between them, which is suitable for describing some domain of interest. The aim here was to define a swarm ontology in terms of basic types of the elements in specific domains and their properties connected to each other by a formal language, such as SysML.	[1,5,6,12,14]
2		Generic swarm architecture: An integrated swarm framework with a composable model abstraction of various specific properties depending on individual autonomous units or agents. The aim here was to develop a method to design a swarm architecture from an initial mission and then various models to be iterated and verified to achieve the desired behavior of the swarm.	[2–4]
3	M&S for a CPS	CPS design paradigm: A new design technique to encompass physical and cyber components to embrace the most appropriate M&S at the component level and at the overall abstraction level in which the system can address the general mission or specific problem. The aim here was to use appropriate M&S to facilitate all the phases of the CPS design in MBSE processes, ranging from the conception to the hardware and software design, and then conduct a massive deployment evaluation in a simulation or virtual/real environment.	[7–10,21]
4		Multi-Paradigm Modeling (MPM): A key approach for CPS design processes to support the coordination of different modeling paradigms, model transformation, and compositional modeling approaches to form the system from specification to verification and validation. The aim here was to associate the conventional methods with model-based designs, in which the specification and design solution are executed and implemented as a domain-specific modeling.	[10]
5	Behavioral simulation	Behavior-based system: an AI-based system which forms as a result of the individual behavior of a set of physical components and/or cyber components (e.g., agents) and their interaction in a dynamic context. The aim here was to develop a behavior-based system at a variety of abstraction levels with semantical enrichment, to facilitate the top–down sharing of a set of common concepts and properties from an ontological model and the bottom–up incremental construction of the coupling of sensing and action through behaviors.	[5,11]
6		Discrete Events System Specification: A mathematical formalism to describe hybrid systems including discrete and continuous behaviors, such as a CPS, which combine discrete events, discrete time, and continuous dynamics in a mathematically sound way. The aim here was to provide a model mapping rule to transform discrete and continuous behaviors into general systems models based on the state machine and constraint diagrams of SysML.	[11,15,16]
7		Multi-Agent-based Simulation (MAS): An emerging simulation technique to address very different individual models ranging from simple (reactive agents) to more complex (cognitive agents) within the unified conceptual framework. The aim here was to develop an autonomous CPS computation model to integrate intelligence by communicating/computing from cyber components and a real-time adaptation via the distributed control of physical components.	[22,23]

Table 1. *Cont.*

No.	Methodologies/Methods	Key Ideas and Its Usage	Refs.
8	Model-driven engineering	Model Based Systems Engineering: The formalized application of modeling to support system requirements, design, analysis, verification, and validation activities throughout system life cycle phases. The aim here was to introduce a top–down, hierarchical approach within an overarching ConOps to decompose into requirement, structure, behavior, and even agent algorithms.	[3,13,24–26]
9		Formal modeling with SysML: With formal logic and within a formal method, SysML can be used to maintain consistency as a design evolvement to provide formal semantics and enable engineers to reason in the model-based development process. The aim here was to apply a single formal modeling language throughout the whole process of MBSE from concept, requirements, architectures, high-level design, V&V, etc., and SysML was extended upward to ontology definition and downward to the definition of opaque behavior and equation of agents; also, the aim was to build a single source of truth to avoid the ambiguity of model semantics.	[12,16,27]
10	Virtual/real hybrid simulation environment	Digital Twin: A virtual representation to model the states of a physical system by collecting information/data from the physical system or components through sensors, so one can experiment or predict the behavior of the physical system in a real environment. The aim here was to construct a swarm digital twin model to support swarm experiments ranging from the concept of operations to integrating verification in an enhanced M&S environment.	[17–20]
11		Live–Virtual–Constructive simulation: A broadly used taxonomy for classifying M&S. A Live simulation involves humans interacting (play acting, etc.) with real systems; a Virtual simulation is a fusion of human and computer-generated experiences; and a Constructive simulation is entirely implemented in a digital computer and may have high levels of abstraction. The aim here was to incorporate Live–Virtual–Constructive simulations into a single M&S environment to leverage the best features of each domain-specific modeling technique to effectively present swarm emergencies and evolve operational capabilities.	[8]

3. The Architecture Framework for Swarm M&S Based on Swarm Ontology

The point of beginning this research was to apply state-of-the-art formal system modeling language to a swarm ontology and effectively solve the problem of executable representation in swarm systems. From the viewpoints of various stakeholders, the purpose of the ConOps is to facilitate a common understanding of a future complex system to help develop operational capabilities to address some emerging problems, which contains the top-level functional thread (i.e., the Functional Flow Block Diagram or Activity Diagram) in the proposed situation and also the operational architecture to bridge the system capabilities and the specific technical requirements needing to be achieved. Therefore, we introduced the ConOps of swarm in a specific domain, and then embedded some semantics into the abstract models and transferred specific design features to the development process of unmanned systems in MBSE. Further, we were able to build a codesign and co-simulation architecture framework which integrated the macrosystem model of the swarm and the micro-individual agent-based run-time model, and supported virtual and real hybrid patterns, thus developing a technical path for the conceptualization development and contextualization evaluation of the autonomous system swarm.

The overall research blueprint of this study is shown in Figure 1; it is very useful to bridge the gap between the operational level of planning and the design solution level of autonomous system units. The content and work of this article mainly lie in the following four aspects:

- Firstly, at the beginning of framing the problems in the complex context, the innovative ConOps of swarm and its novel capability requirements should be derived and deducted, and currently, we give full play to the integrated application advantages of Model-Driven Engineering (MDE) and Multi-Paradigm Modeling (MPM) to break through the traditional feature-based modeling function of Web Ontology Language (OWL) technology and Protégé 5.6.1 software to define a swarm ontology in a conceptual model based on System Modeling Language (SysML), which is more formal and executable. The swarm ontology with a descriptive form should support hierarchical model refinement and translation. From the top–down levels of abstraction, the macro-behavior presents the model of a swarm population, the meso-behavior presents the model of a group of individuals, and the micro-behaviors present the model of an individual. The meso-model is in between the individual and population levels [21]. To achieve explicit knowledge representation and logical reasoning throughout the three levels of macro–meso–micro, it will support the linkage of transition from the swarm overall characteristics to the system design features in the way of decomposition and breakdown, and then convey and map the component specifications in the development of unmanned systems.
- With the application of the process and method of MBSE and the flexible extension mechanism in a system model based on SysML, we are particularly interested in the dominant features of intelligence, adaptability, and autonomy within heterogeneous unmanned systems in multiple domains (such as space, air, ground, sea, etc.) and dedicated to establishing a metamodel framework and its corresponding metamodeling process for those systems. Therefore, focusing on the functional and logical model (mainly by SysML) and the mathematical–physical model (mainly by Modelica), our approach further enhances the pattern of the domain-specific modeling language (DSML) and its integration framework (via the SysML Extension for Physical Interaction and Signal Flow Simulation, SysPhS for short, the specification from OMG) of general unmanned systems to define, develop, integrate, and verify the implementation under the use cases of vehicle maneuvers, autonomous control, information interconnection, mission coordination, and so on.
- For the application of the "Real" and "Virtual" nodes in a hybrid pattern to simulate a typical complex swarm scenario, we define the format of a Unified Repository (UR) for both the digital model (digital twin) and the physical entity in a common representation model of an unmanned system. In the current mature spatiotemporal information system, it embeds agent-based mathematical models and collects data about the movement, navigation, command and control, communication, etc., of the physical entity. We built a codesign and co-simulation environment which supports virtual/real mixing operations to visualize the overall and global swarm application and to verify and validate the conceptualization of autonomous unmanned swarm.
- And finally, considering the swarm ontology technology of autonomous unmanned systems as the main thread in our research, and across the conceptual ontological model–functional and logical model–mathematical physical model, we develop the technology of the integration environment of a multilevel and multiparadigm collaborative model and simulation, which will become a technical evolution platform of the experimental frame to support the development and evaluation of complex behaviors [24], such as swarm environment awareness and cognition, collaborative task planning and decision making, information interaction and autonomous control, and others. We take a hierarchical, composable approach to swarm development and the experimental framework is mainly composed of ConOps, capabilities, architecture, and parameters.

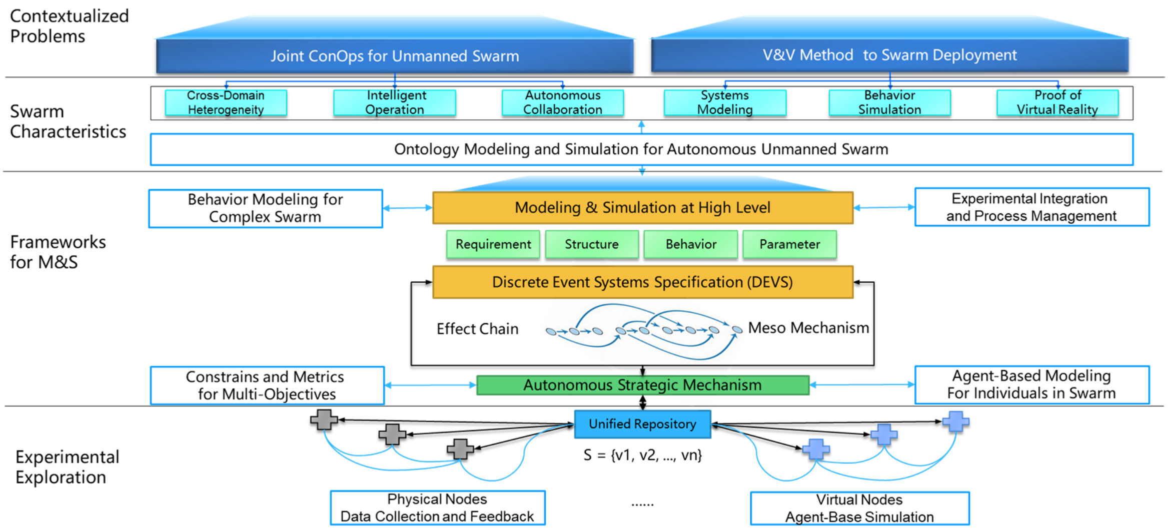

Figure 1. The overall research blueprint of this study.

4. Modeling and Simulation Methods and Their Applications

The M&S and its applications in the autonomous swarm framework involve the following four levels: the definition of swarm ontology, the system development with the metamodel and metamodeling, multiagent M&S for an autonomous CPS, and V&V in a virtual/real hybrid integration environment.

4.1. Descriptive Modeling for Swarm Ontology

This section serves as the basic theoretical and top-level guidance for our methodology and emphasizes a complete description of system conceptualization and contextualization from the loop of Conceive–Design–Implementation–Operation (CDIO) to make the ontology a fundamental approach to addressing swarm complexity, thereby also reflecting the basic key drivers and processes of MBSE and the innovation of specific engineering applications. Therefore, in the context of complex operations, the architecture framework is placed at the intersection of the above four CDIO domains of a complex system. It is highlighted that the functionality and characteristics of the systems and elements within the overall framework should be considered during the mission concept, which requires establishing a top-level framework for analysis and synthesis, concentrating on the architecture model of the swarm ontology. The goal of the work is to entirely model a problem in business terms without refining the solution or its implementation, which relates to the Computation-Independent Model (CIM) of Model-Driven Architecture (MDA) [25].

For our study and other AI systems, what "exists" is that which can be represented in models. For the research of complex systems such as swarm, the formal representation of autonomous systems based on ontology is currently one of the research hotspots, while modeling and simulation are regarded as the most effective solution. Our aim in the research is to provide a comprehensive modeling and simulation framework for future applications of swarm ontology that enables us to leverage the advances in graphical modeling languages (such as SysML) and the process of MBSE, while further enabling us to perform formal analyses of consistency and correctness with respect to the ontology of the domain of swarm.

The ontology is traditionally defined in OWL2 with open-source ontology tools, such as Protégé [10]. In order to reason about the properties of the concept model and particularly facilitate the simplification of the model-to-model transformation from one domain into another in MBSE, SysML has the same abilities as OWL2 to map the domain concepts into SysML entities or relations without affecting the concepts. In SysML, the Block Definition Diagram (BDD) has enough expressiveness to represent a detailed design. When we suitably restrict SysML's BDD, it can be transformed into OWL2 to achieve the equivalent effect. A SysML BDD is a kind of first-order equational logic and provides an abstract syntax for the kind of terms in which logical axioms are expressed using equality, instances, and subclass relations between terms. The knowledge presentation of a system of concepts is suitable for representing designs which will have distinct "has a part of" properties, with domains and range classes that represent the graph structure of the BDD, and with a cardinality restriction on these properties to depict the number of instances of the class during implementation [27].

The general principle is to map the swarm ontology into SysML to define the ontological concepts and relationships as SysML constructs that can be applied to appropriate modeling entities: concepts to blocks, and relationships to a semantically compatible SysML relationship. See Figure 2 for an example of an autonomous swarm ontology model for a specific mission. The macro-behavior corresponds to the swarm tasks; that is, the autonomous swarm needs to possess top-level capabilities to achieve its mission. The meso-behavior to corresponds to a teaming strategy, which refers to the collective behavior of an autonomous system to be negotiated in interaction with the outside world. The micro behavior corresponds to atomic actions, which are various operations that autonomous-system individuals should possess.

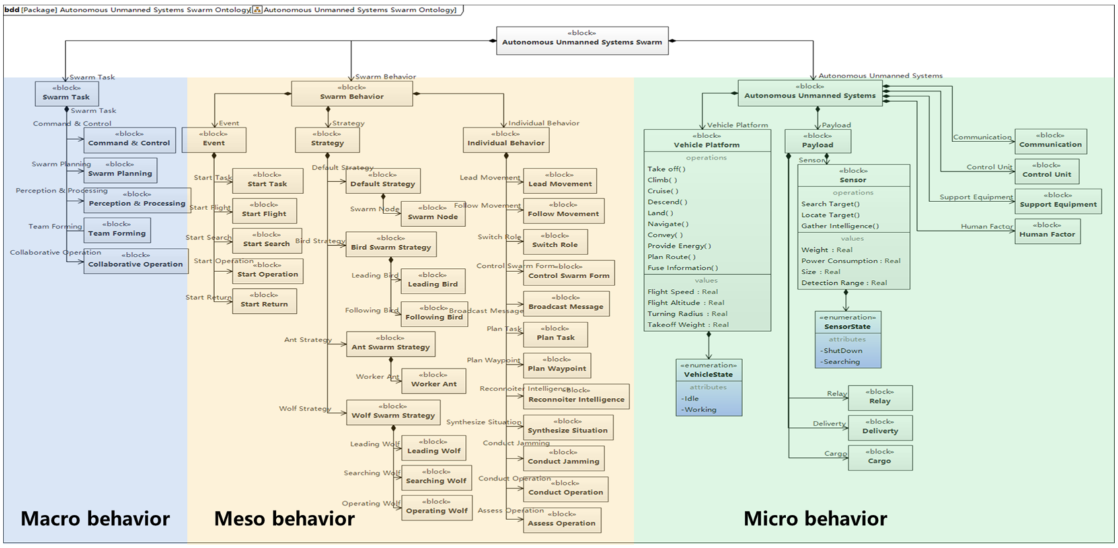

Figure 2. Example of swarm ontology model for a specific mission.

4.2. Metamodel and Metamodeling Supporting an Autonomous System

Modeling languages such as SysML provide particularly useful graphical syntax for human understanding. However, because they only contain abstract semantics, it is necessary to add concrete semantics related to domain knowledge for domain-specific applications. The essence of a metamodel is to create an intermediate layer between a common abstract modeling language and specific implementation instances to represent domain knowledge. Its key role lies in achieving business knowledge extraction and model reuse. The process of creating a metamodel, which we call metamodeling methods, has also become a core activity in implementing specific practices of the MBSE methodology. In order to overcome the learning curve of modeling languages, methods, and tools, metamodels and metamodeling will become key enablers for promoting the transformation of traditional systems engineering processes into methods of MBSE.

Model-driven architecture is a process that focuses on models and is driven by model mapping. The system development approach in the MDA environment aims to accurately describe different problem spaces by creating various models during development activities and using model transformations to drive the entire development process, including analysis, design, and implementation. The Platform-Independent Model (PIM) is then built by specializing the run-time properties in SysML diagrams, such as the unmanned adaptive control units and their dynamic evolution, which reflect the structures and behaviors of the design components.

This section mainly introduces metamodels with unified semantic interpretation for a behavior model of artificial intelligence systems, which can be used as a reference to guide the specification and implementation of various autonomous unmanned systems across many domains in the MBSE process, such as Unmanned Aerial Vehicle (UAV), Unmanned Ground Vehicle (UGV), Unmanned Underwater Vehicle (UUV), etc.

From the viewpoint of Command, Control, and Communication (C3), the interoperability allows information interactions between an unmanned system and others and information sharing and task allocation between different command levels and different units, e.g., STANAG 4586—Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability, which defines data formats, interface requirements, communication protocols, etc. [28]. Meanwhile, from another viewpoint of Open System Architecture (OSA), we should consider a common/open architecture, modular component, test verification, and data integration for unmanned systems. As another example, the SAE standard Joint Architecture for Unmanned Systems emphasizes capturing and categorizing common interfaces and services to enable the continued growth of standard sets and robotic technology [29].

The primary paradigm of Artificial Intelligence (AI) is mainly knowledge-based systems, in which the knowledge related to application domains, the external environment, and the decision making process are defined and presented via symbolic models. On the other hand, MBSE advocates for the specification, analysis, design, verification, and validation of systems using formal models. Therefore, from the perspective of the development of modern AI systems, our first choice is the representation language of symbolic knowledge and the automatic reasoning mechanism based on logical language.

With the V-model, it depicts the development activities that go through the ConOps to the integration and the V&V that helps identify errors early in the life cycle. However, the V-model has the drawback of being rigid and less flexible. However, the walking skeleton model is a lean approach for incremental development, popularly used in software design, and especially suitable for a systems approach to AI implementation to rapidly adjust the scope of a system. It focuses on creating a skeleton framework and looks like a metamodel, which will become the heart of the model-based approach, and the architecture can be configured and optimized to ensure that the system is enhanced [26].

Different from the traditional application of a V-model, working toward complex development and simulation processes such as swarm, we should have a digital system prototype at the beginning to connect the development of the top-level SoS and the design of the underlying multidisciplinary components of unmanned systems. We should incrementally create and deploy a coherent and consistent Digital System Model (DSM) integrating specific models as a source of digital twins of system specification, design, analysis, verification, and validation. The system architecture represents the structure, behavior, and constraints of complex systems to deliver an effective solution satisfying the needs of stakeholders. Therefore, the metamodel is the initial prototype for the system architecture, which will serve as the starting point for system development and support the system's evolution in a M&S environment.

The process to develop a metamodel is also a micro-cyclic iterative development process within the whole framework. Following the idea of the Model-Based System Architecture Process (MBSAP) [13] to connect or transform an SoS mission architecture to a system architecture, the first mapping converts capabilities to Operational Viewpoint (OV); the next mapping transforms the OV into Logical/Functional Viewpoint (LV), where the refinement of system elements, services, functions, interactions, and behaviors is carried out. Then, the development of physical specifications is accomplished by mapping the LV to Physical Viewpoint (PV). Synchronously, digital system models of autonomous systems support the M&S of complex dynamic systems, particularly swarm, and allow engineers to continuously express new solutions and conduct L-V-C online testing before implementation. See Figures 3–8 for examples of SysML models and Modelica models as the metamodels for Unmanned Aerial System (UAS) with the typical composition and synthesis.

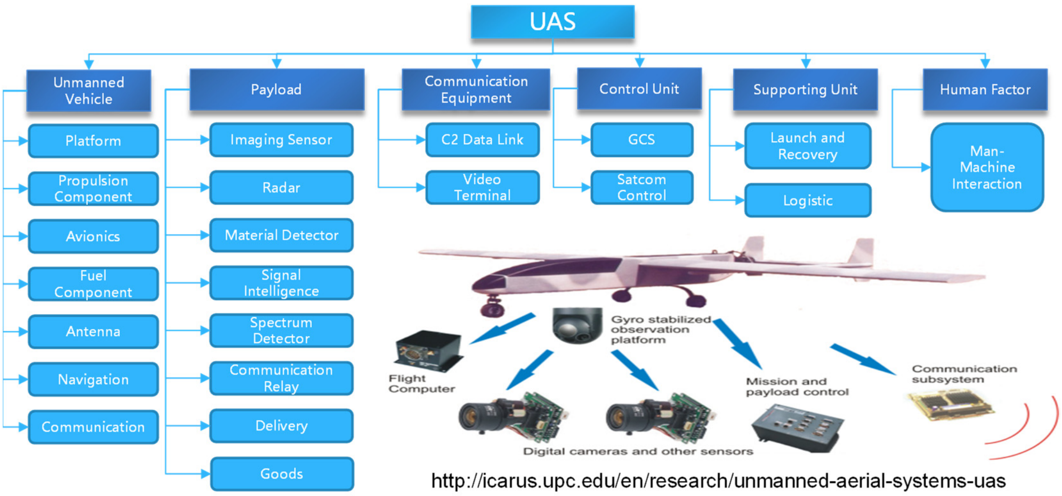

Figure 3. Example of a metamodel of UAS.

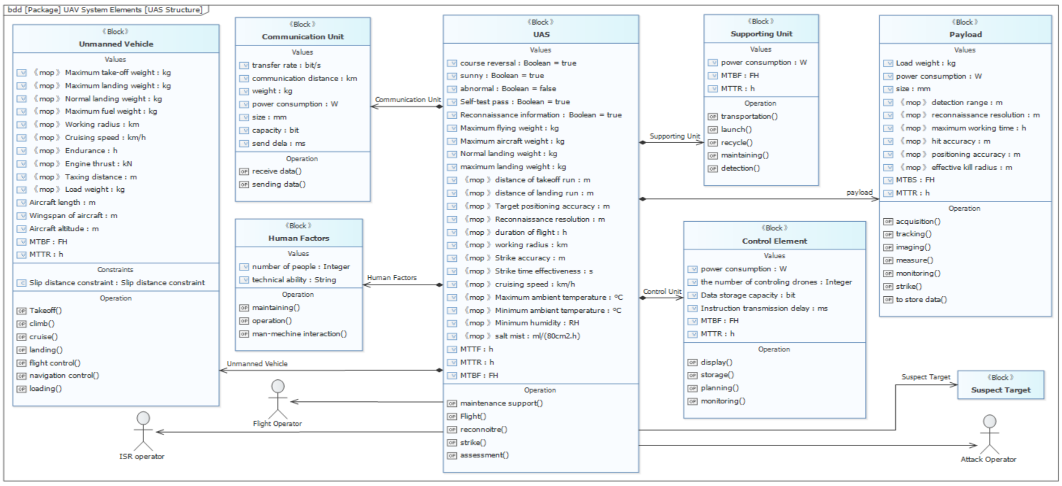

Figure 4. Example of SysML block models for a metamodel of UAS.

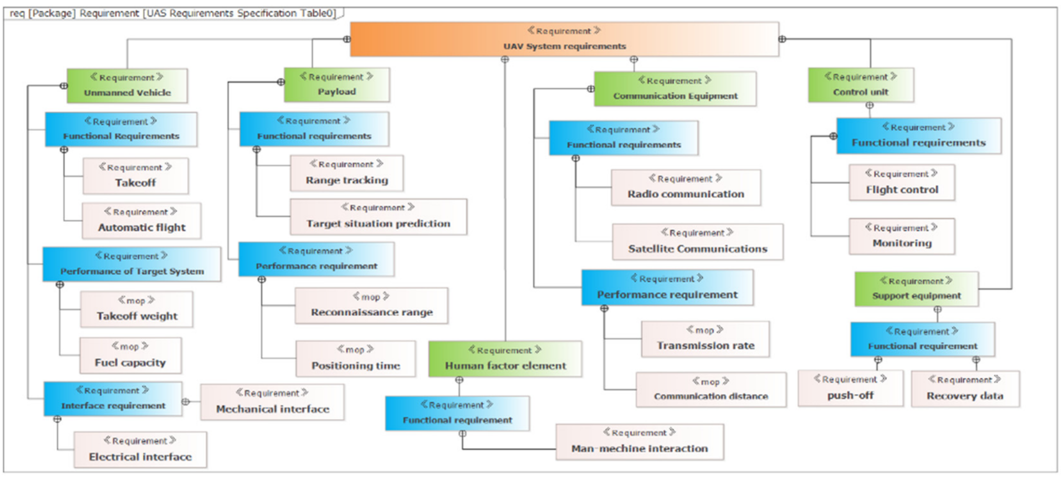

Figure 5. Example of SysML requirement models for a metamodel of UAS.

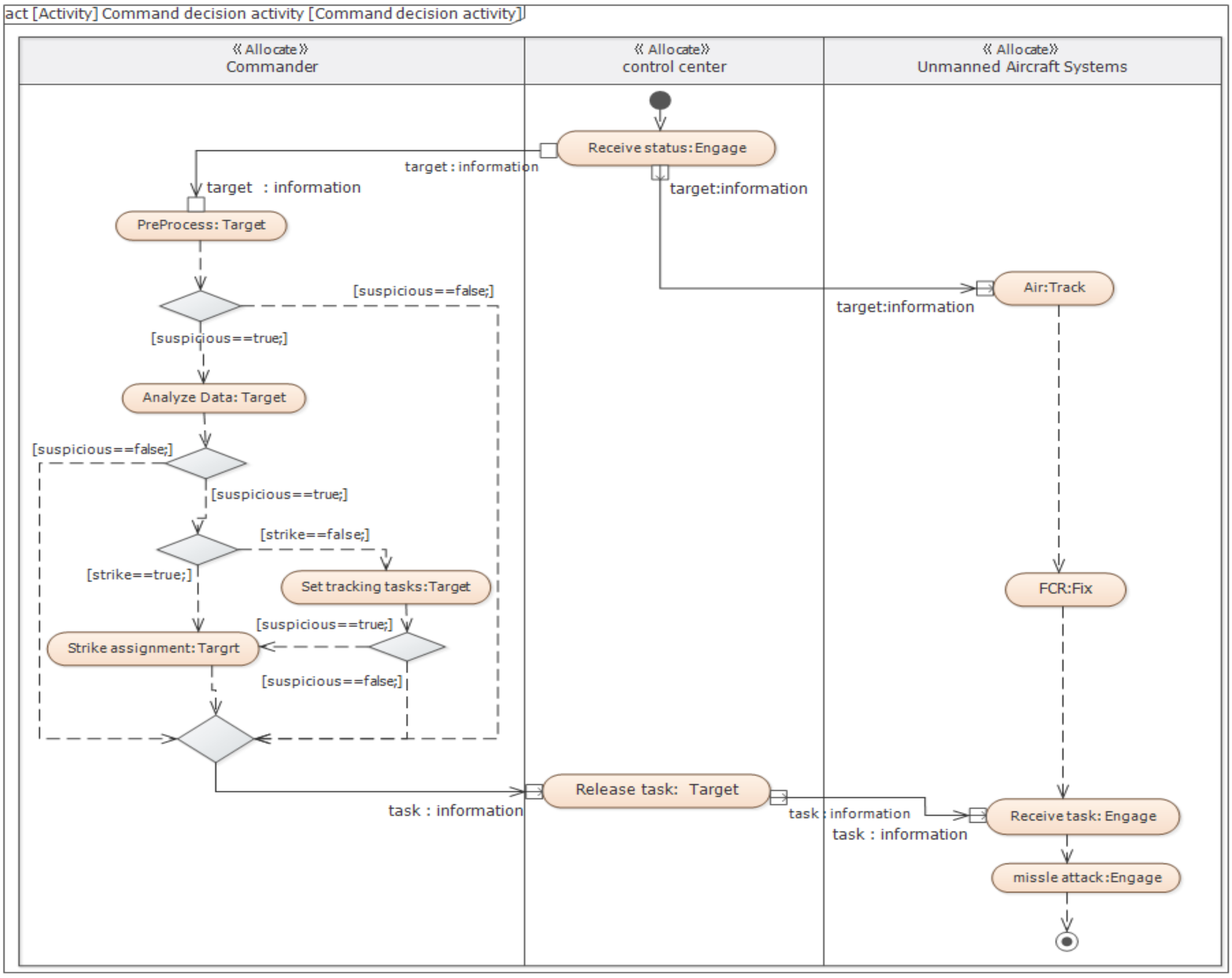

Figure 6. Example of SysML activity models for a metamodel of UAS.

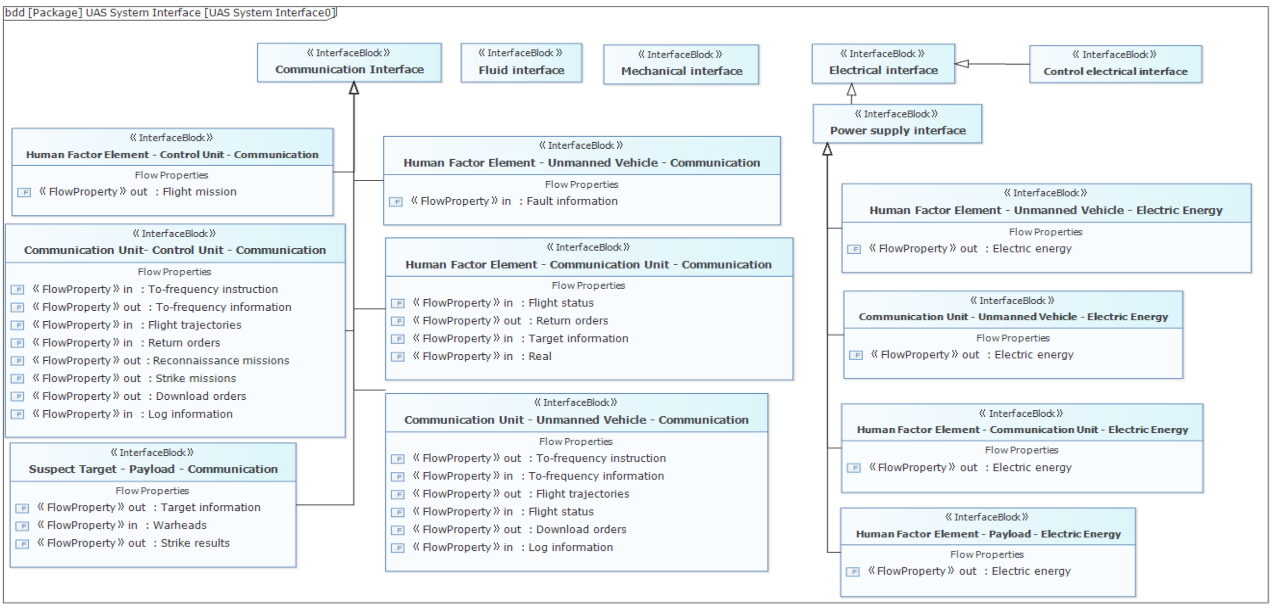

Figure 7. Example of SysML interface models for a metamodel of UAS.

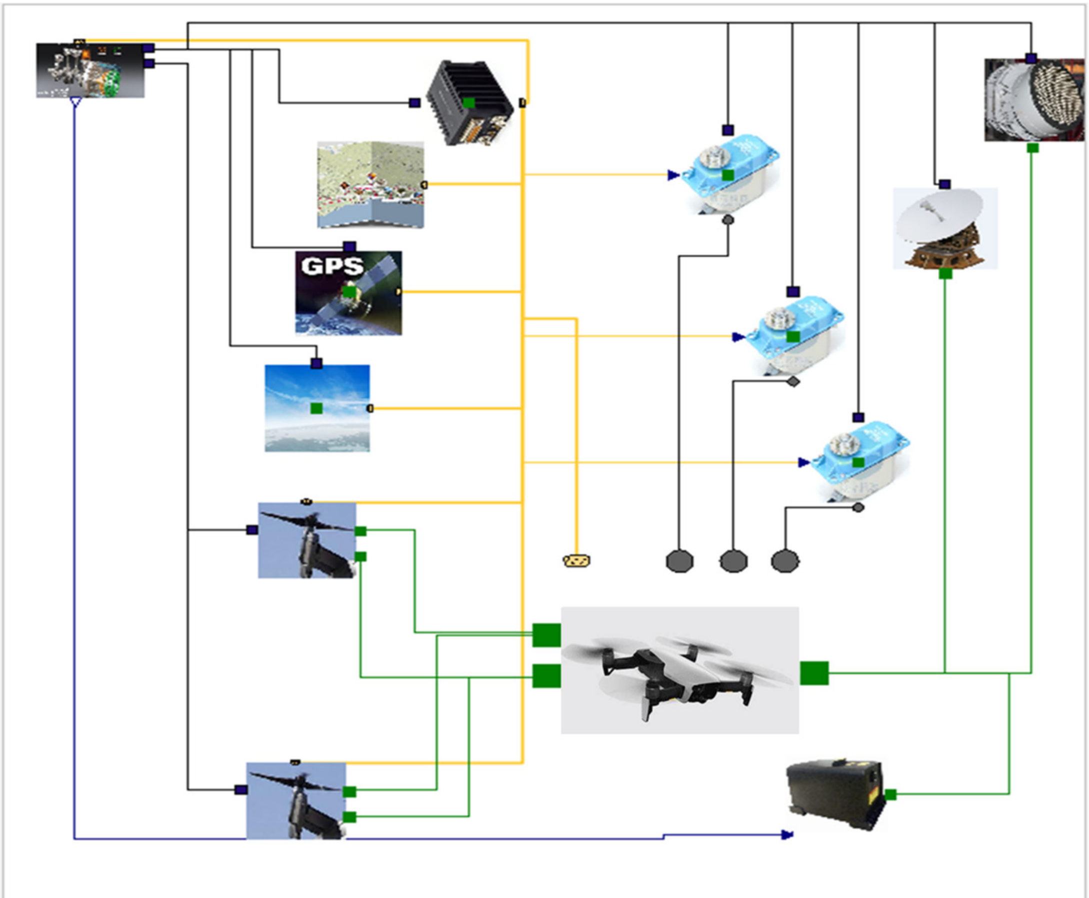

Figure 8. Examples of Modelica models for a metamodel of UAS.

In our methodology, we firstly clarify and explain the aforementioned conceptual models in swarm ontology for the stakeholders and/or team members, which include the whole iterative context of the system definition and the engineering process of autonomous vehicles, and then apply the approach of the metamodel to capture and specify the technical information needing to be developed for the DSM in communication and computation, command and control, motion planning, perception, other knowledge representations about the problem description, and the solution specification to support decision making within MBSE.

4.3. Multi-Agent-Based M&S for CPS

The intelligence of an artificial system is due to the emergent properties in a complex context, such as a swarm, which can be described as results of the interactions between their components and the environment. There is a reasonable expectation that the intelligence of a system should not only be formed from an abstract symbolic system in advance of its operations, such as automatic reasoning based on logic. However, the intelligent behavior of a system should emerge as a composition of simpler agents structured in a certain way and exerting their interactions with others and with the environment [7].

A multi-agent-based simulation is an advantageous solution due to its excellent ability to cope with diverse models, ranging from simple entities—usually called reactive agents—to more complex cognitive agents. Within the unified conceptual framework, the modelers can easily handle different levels of representation, for instance, individuals and swarm [22]. Within the framework of artificial intelligence, Multi-Agent System (MAS) are characterized by offering a potential solution to the development of complex problems with distributed properties [23]. Due to the nature of hybrid and real-time control in CPSs with a controller and physical components to sense, control, and operate in a complex physical environment, multi-agent-based development will be of great importance in the domains of CPSs such as unmanned vehicles.

As for the definition of a swarm ontology, it is deemed to be a macro-model to reflect the overall operational mission tasks. Now, it is necessary to consider the autonomous teaming strategic mechanism in a swarm as a meso-model, and individual actions as a micromodel. Now, we choose a multi-agent-based approach to convert the latter two behaviors into computational models. Among them, the determination of the autonomous team strategy lies in bionic research on teaming collaboration rules in a creature swarm, such as the flocking of birds, schooling of fish, or colonies of bees, aiming to achieve the simulation and verification of social behaviors such as the grouping, following, negotiation, divisions, and cooperation of an autonomous system. Alternatively, individual behaviors mainly present activities such as maneuvering, avoidance, detection, communication, control, and others. Both involve discrete, continuous models or their combination. See Figures 9–17 for an autonomous teaming strategic model of a swarm and a state machine model of an individual autonomous system, respectively.

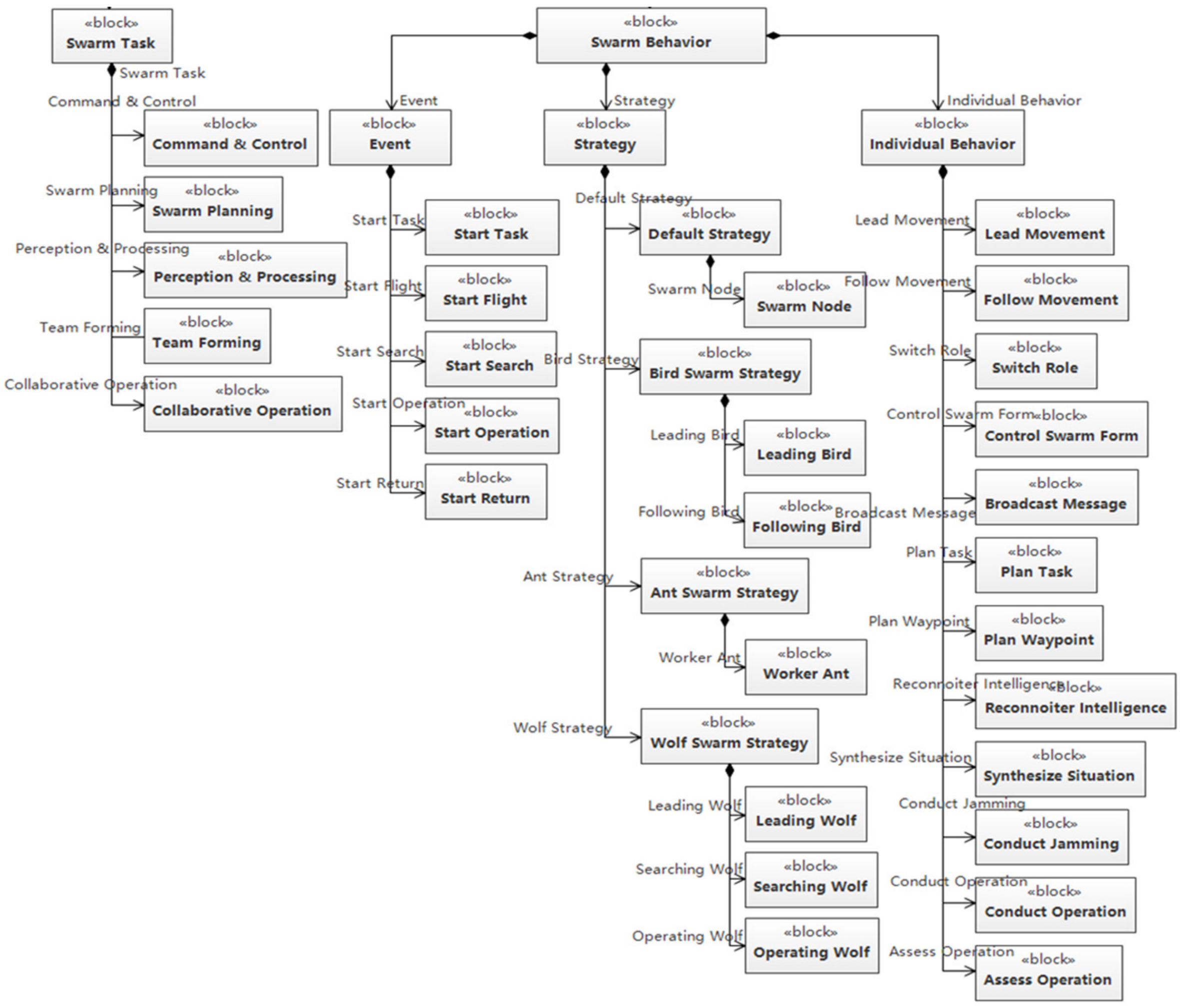

Figure 9. The task and behavior of swarm.

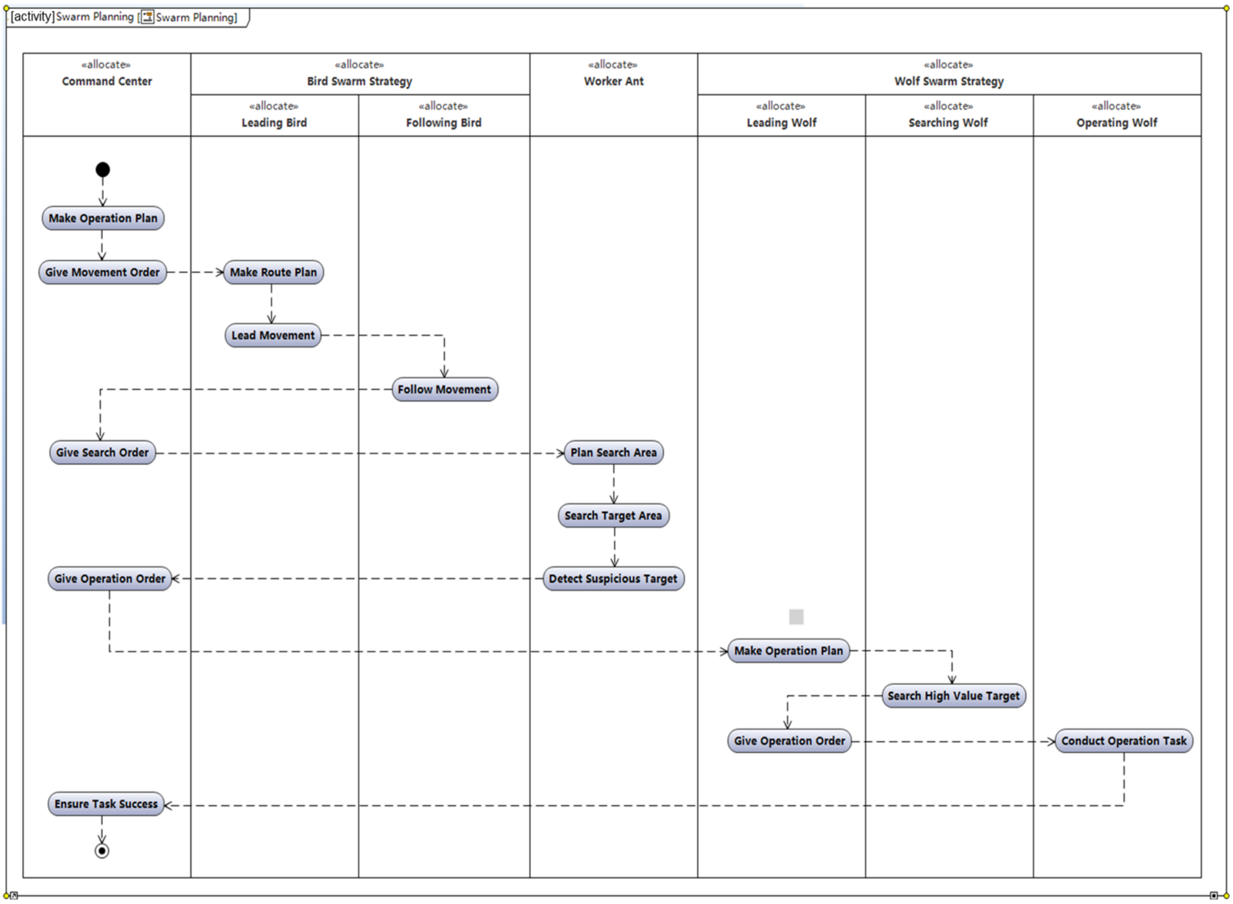

Figure 10. The autonomous teaming strategic model (activity diagram) of swarm.

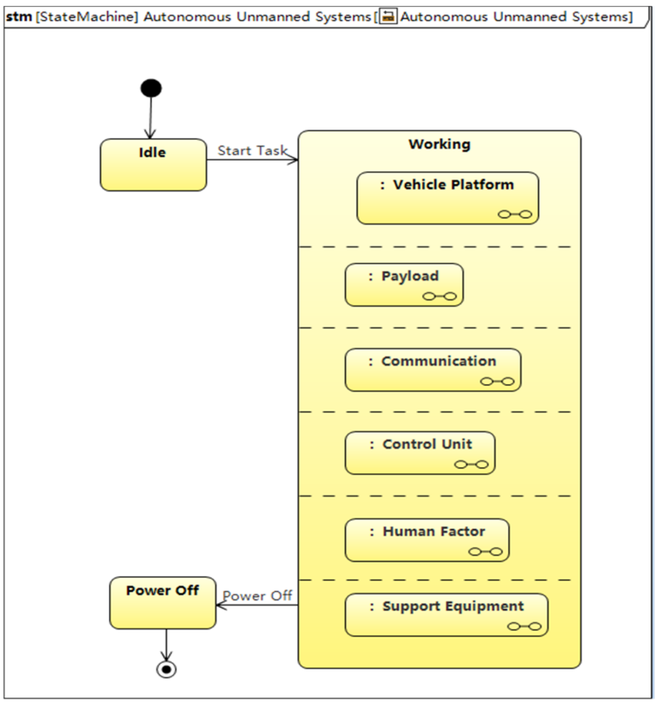

Figure 11. The behavior model (state machine diagrams) of individual autonomous system.

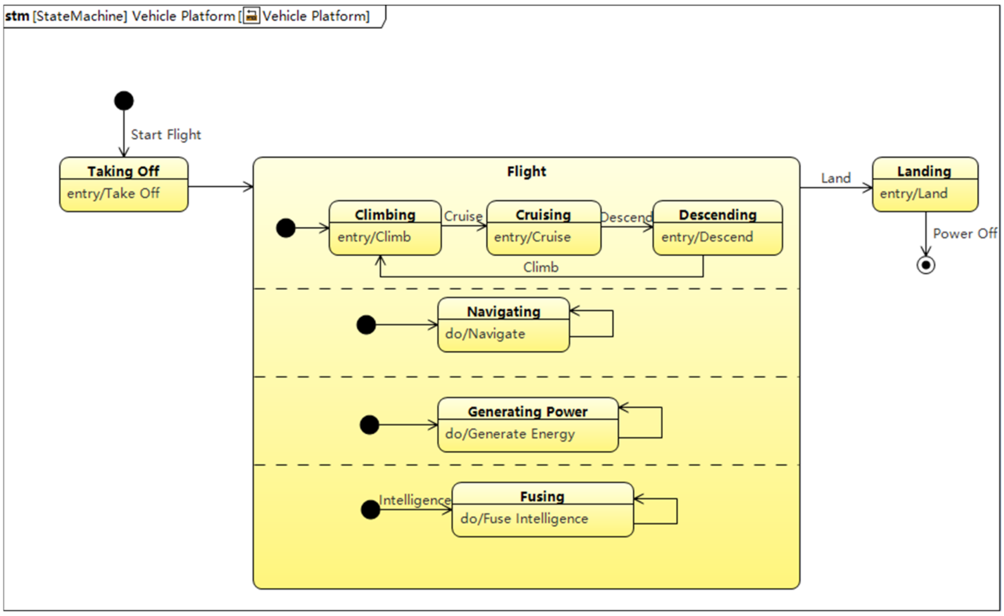

Figure 12. The behavior model (state machine diagrams) of vehicle platform.

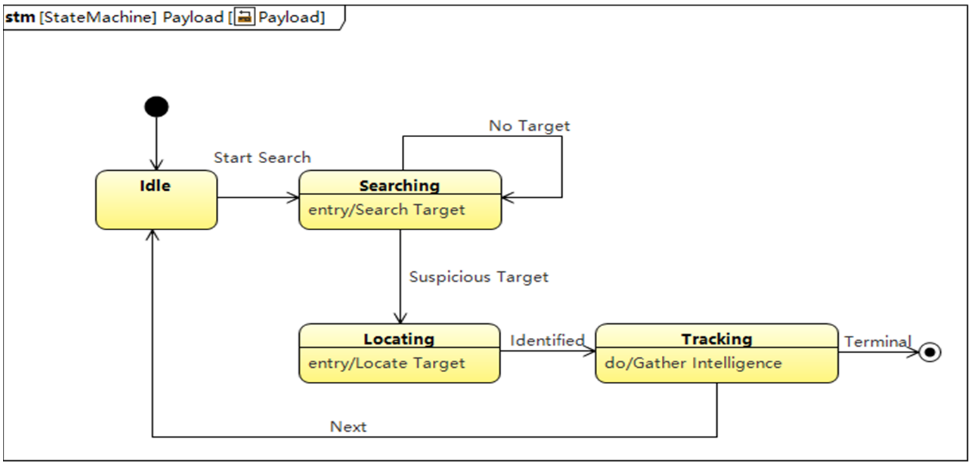

Figure 13. The behavior model (state machine diagrams) of payload.

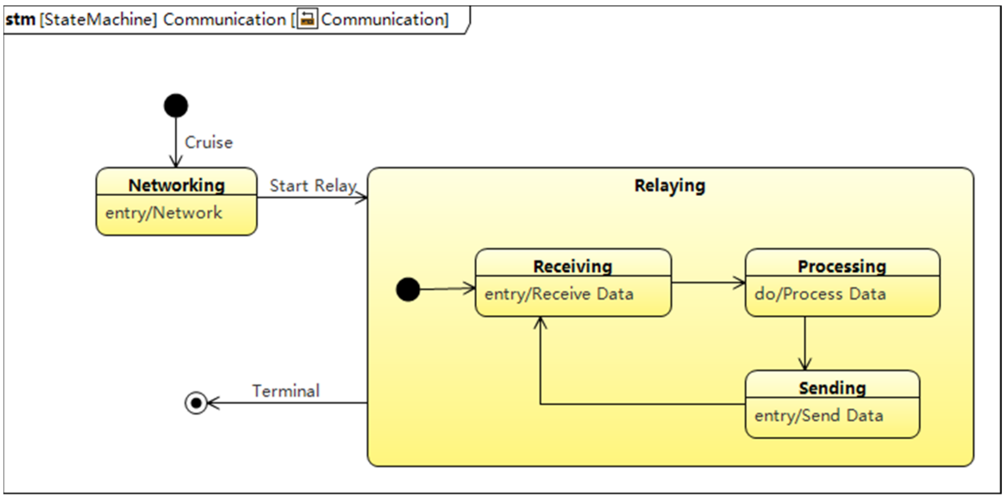

Figure 14. The behavior model (state machine diagrams) of communication.

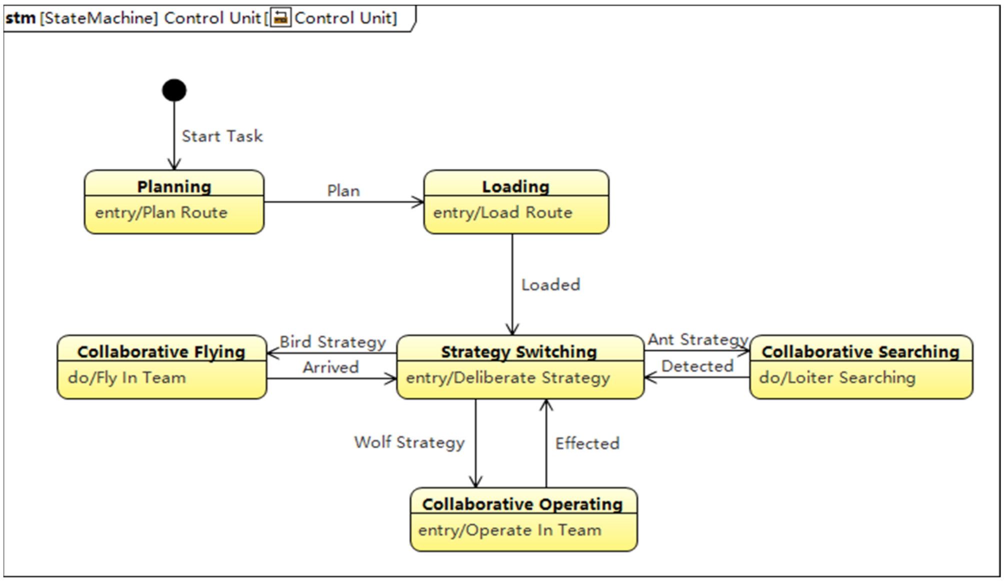

Figure 15. The behavior model (state machine diagrams) of control unit.

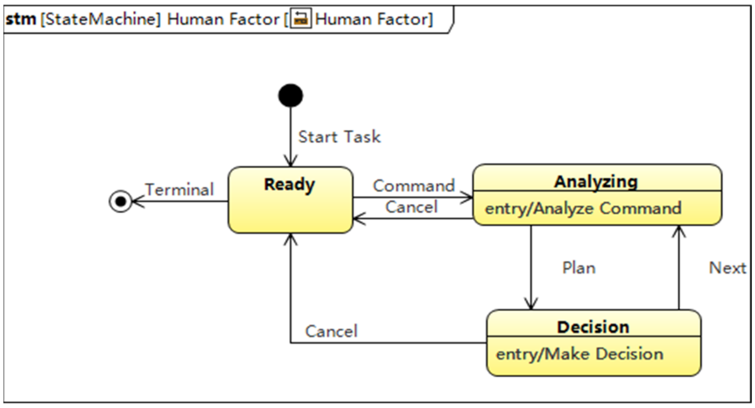

Figure 16. The behavior model (state machine diagrams) of human factor.

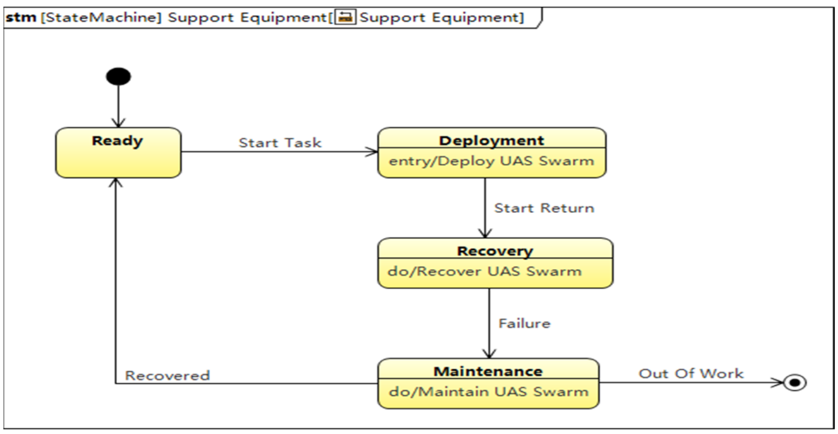

Figure 17. The behavior model (state machine diagrams) of support equipment.

4.4. V&V in a Hybrid Virtual/Real Integration Environment

A swarm is a dynamic System of Systems (SoS) in which components are autonomous systems or other related enabling systems, such as legacy regional communication networks, supporting infrastructure, etc., which adapt to the current context and mission. Although involving so many different modeling paradigms, it is necessary to establish an overall simulation environment to support the analysis, verification, and validation of operational concepts.

The application of M&S throughout MBSE needs to integrate structural modeling and dynamic behavior modeling in the architecture of the DSM, which provides a complete picture of the swarm. In order to meet the needs of various ConOps, design elements are effectively integrated (static structure) and use cases at all levels are employed to justify the requirements and interactions (dynamic behaviors). The characteristics of MBSE ensure that the architecture can cover all use cases, systems, and components, and drive end-to-end M&S to verify the attributes and behavior of the models of systems and components [24].

Hybrid virtual/real integration is a digital Experimental Frame (EF) to support the scenario-based verification and validation of a swarm, where the discrete events of the swarm are the engine of constructive models to drive the behavior of multiple distributed autonomous individuals (simulation). According to the logical structure and dynamic behavior of the autonomous system in a swarm, a run-time computational model of virtual individuals is implemented as agent-based models. The format of a Unified Repository (UR) is defined to receive, send, and store data with the real ones in the M&S environment. This is a comprehensive demonstration of a platform for integrating a swarm ontology in a concept model, DEVS model, system logical structure and behavior model, and multiagent model, and collecting data to contribute to the employment and evolution of swarm. See Figure 18 for an integrated L-V-C modeling and simulation framework for a swarm V&V.

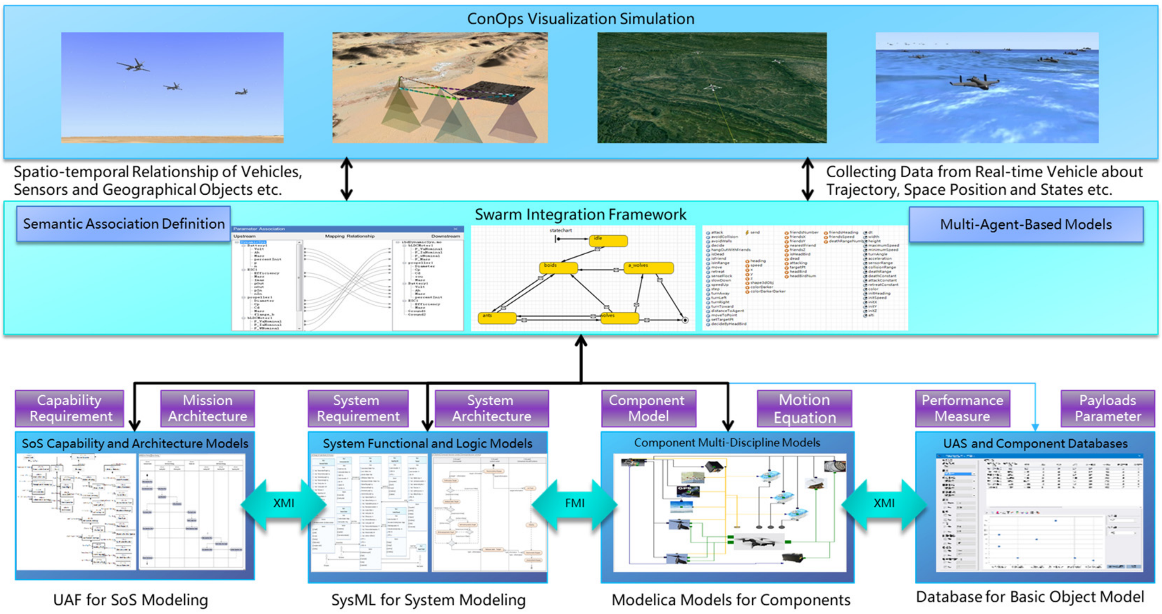

Figure 18. An integrated L-V-C modeling and simulation framework for swarm V&V.

5. Conclusions

The prominent feature of our proposed approach to swarm ontology is mainly the suitable levels of abstraction and their coherent integration framework in a single formal modeling language, which is a swarm mechanism to capture domain-specific knowledge from problems to solutions within the context of the swarm ConOps, and study the model specification, domain-specific semantics, and their transformation throughout conceptual, logical models and agent-based computation. This research also presents the methods of a metamodel and metamodeling in MBSE development, which aim to develop and verify a more reasonable architecture of autonomous systems in an evolving environment. Moreover, with a virtual/real hybrid simulation platform, we intend to establish a comprehensive modeling and simulation environment to continuously enhance human–machine cognitive collaboration and human–autonomy teaming to support MBSE and AI.

Author Contributions: Conceptualization, G.X.; methodology, X.G.; software, C.C. and Z.W.; validation, W.W.; formal analysis, Y.F.; investigation, K.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Burkhart, R. A Swarm Ontology for Complex Systems Modeling. In Proceedings of the Symposium on Complex Systems Engineering, Santa Monica, CA, USA, 11–12 January 2007.
2. Giles, K. A Framework for Integrating the Development of Swarm Unmanned Aerial System Doctrine and Design. Available online: https://docslib.org/doc/7281488/a-framework-for-integrating-the-development-of-swarm-unmanned-aerial-system-doctrine-and-design (accessed on 5 September 2021).
3. Giles, K.; Giammarco, K. A mission-based architecture for swarm unmanned systems. *Syst. Eng.* **2019**, *22*, 271–281. [CrossRef]
4. Navarro, I.; Matía, F. An Introduction to Swarm Robotics. *ISRN Robot.* **2013**, *2013*, 608164. [CrossRef]
5. Gorodetsky, I.; Samoylov, V.V.; Trotskii, D.V. The Reference Ontology of Collective Behavior of Autonomous Agents and Its Extensions. *Comput. Syst. Sci. Int.* **2015**, *54*, 765–782. [CrossRef]
6. Schlenoff, C.; Prestes, E.; Madhavan, R.; Goncalves, P.; Li, H.; Balakirsky, S.; Kramer, T.; Miguelanez, E. An IEEE standard Ontology for Robotics and Automation. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 1337–1342.
7. Castro, R.D.; Marcosig, E.P.; Giribet, J.I. Simulation Model Continuity for Efficient Development of Embedded Controllers in Cyber-Physical Systems. In *Complexity Challenges in Cyber Physical Systems: Using Modeling and Simulation (M&S) to Support Intelligence, Adaptation and Autonomy*, 1st ed.; Saurabh, M., Andreas, T., Eds.; Willy: Hoboken, NJ, USA, 2020; Volume 1, pp. 193–220.
8. Saurabh, M.; Andreas, T. The Complexity in Application of Modeling and Simulation for Cyber Physical Systems Engineering. In *Complexity Challenges in Cyber Physical Systems: Using Modeling and Simulation (M&S) to Support Intelligence, Adaptation and Autonomy*, 1st ed.; Saurabh, M., Andreas, T., Eds.; Willy: Hoboken, NJ, USA, 2020; Volume 1, pp. 3–16.
9. Walid, M.T.; Abd-Elhamid, M.T.; Johan, T. *Cyber-Physical Systems: A Model-Based Approach*, 1st ed.; Springer: Cham, Switzerland, 2021; pp. 5–12.
10. Dominique, B.; Rima, A.; Mauro, I.; Bedir, T.; Holger, G. An ontological foundation for multi-paradigm modelling for cyber-physical systems. In *Multi-Paradigm Modeling Approach for Cyber-Physical Systems*, 1st ed.; Bedir, T., Dominique, B., Hans, V., Miguel, G., Paulo, C., Vasco, A., Eds.; Elsevier: London, UK, 2021; pp. 9–43.
11. Matarić, M.J.; Michaud, F. Behavior-Based Systems. In *Handbook of Robotics*; Siciliano, B., Khatib, O., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 899–909.
12. Bermejo-Alonso, J.; Sanz, R.; Rodrguez, M.; Hernndez, C. An Ontological Framework for Autonomous Systems Modelling. *Int. J. Adv. Intell. Syst.* **2010**, *3*, 211–225.
13. Borky, J.M.; Bradley, T.H. *Effective Model-Based Systems Engineering*; Springer: Cham, Switzerland, 2019; pp. 23–96.
14. Wagner, D.A.; Bennett, M.B.; Karban, R.; Rouquette, N.; Jenkins, S.; Ingham, M. An ontology for State Analysis: Formalizing the mapping to SysML. In Proceedings of the IEEE Aerospace Conference, Big Sky, MT, USA, 3–10 March 2012; pp. 1–16.
15. Zeigler, B.P.; Kim, D. Discrete Event Systems Specification (DEVS), MBSE, and Digital Engineering. Available online: https://www.researchgate.net/publication/355108235_Discrete_Event_Systems_Specification_DEVS_MBSE_and_Digital_Engineering (accessed on 15 October 2022).
16. Wu, X.Q.; Yan, X.F.; Li, X.C.; Wang, Y.Z. Simulating hybrid SysML models: A model transformation approach under the DEVS framework. *J. Supercomput.* **2023**, *79*, 2010–2030.
17. Mendi, A.F.; Erol, T.; Doğan, D. Digital Twin in the Military Field. *IEEE Internet Comput.* **2021**, *6*, 33–40. [CrossRef]
18. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [CrossRef]
19. Bickford, J.; Van Bossuyt, D.L.; Beery, P.; Pollman, A. Operationalizing digital twins through model-based systems engineering methods. *Syst. Eng.* **2020**, *23*, 724–750. [CrossRef]
20. Grieves, M. Digital Twin: Manufacturing Excellence through Virtual Factory Replication. *White Pap.* **2014**, *1*, 1–7.
21. Saurabh, M.; Andreas, T. Multi-Perspective Modeling and Holistic Simulation–A System-Thinking Approach to Very Complex Systems Analysis. In *Complexity Challenges in Cyber Physical Systems: Using Modeling and Simulation (M&S) to Support Intelligence, Adaptation and Autonomy*, 1st ed.; Saurabh, M., Andreas, T., Eds.; Willy: Hoboken, NJ, USA, 2020; Volume 1, pp. 83–110.

22. Drogoul, A.; Vanbergue, D.; Meurisse, T. Multi-Agent Based Simulation: Where are the Agents. In *Multi-Agent-Based Simulation II, MABS 2002, Proceedings of the Third International Workshop on Multi-Agent Systems and Agent-Based Simulation, Bologna, Itlay, 15–16 July 2002*; Simão Sichman, J., Bousquet, F., Davidsson, P., Eds.; Springer: Berlin, Heidelberg, Germany, 2003; Volume 2581, pp. 1–15.
23. Julián, V.; Botti, V. Multi-Agent Systems. *Appl. Sci.* **2019**, *9*, 1402. [CrossRef]
24. Gao, X.H.; Zhang, Y.Q.; Wang, Z.Q. The Architecture of Digital System Model for Systems-of-Systems Engineering. In Proceedings of the Chinese Automation Congress (CAC), IEEE, Hangzhou, China, 22–24 November 2019; pp. 2744–2747.
25. Van Hien, N.; Diem, P.G. A Practical Specialization of MDA/MBSE Approach to Develop AUV Controllers. *J. Mar. Sci. Appl.* **2020**, *20*, 102–116. [CrossRef]
26. Badiru, A.B. *Artificial Intelligence and Digital Systems Engineering*, 1st ed.; CRC: Miami, FL, USA, 2022; pp. 43–63.
27. Graves, H. Integrating SysML and OWL. In Proceedings of the 6th International Conference on OWL: Experiences and Directions, Chantilly, VA, USA, 23–24 October 2009; Volume 529, pp. 117–124.
28. STANAG 4586–Standard Interfaces of UAV Control System (UCS) for NATO UAV Interoperability, docslib.org. Available online: https://docslib.org/doc/13475156/stanag-4586-standard-interfaces-of-ucs-for-nato-uav-interoperability (accessed on 5 January 2021).
29. English, R. *The Joint Architecture for Unmanned Systems, A Set of SAE Interoperability Standards*; SAE Technical Paper 2009; SAE International: Warrendale, PA, USA, 2009; Volume 2009-01-3250. [CrossRef]