Received April 27, 2020, accepted May 25, 2020, date of publication June 1, 2020, date of current version June 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2998843

Ontology Construction and Evaluation of UAV FCMS Software Requirement Elicitation Considering Geographic Environment Factors

XUAN HU AND JIE LIU
Information Security Research Center, CEPREI Laboratory, Guangzhou 510610, China
Key Laboratory of Ministry of Industry and Information Technology, Guangzhou 510610, China

Corresponding author: Jie Liu (liujie@ceprei.com)

This work was supported in part by the General Project of Guangzhou Scientific Research Plan under Grant 201804010316, and in part by the National Key Research and Development Program of China under Grant 2019YFC0118800.

ABSTRACT The quality of unmanned aerial vehicle flight control and management system (UAV FCMS) software is crucial to guaranteeing the quality of UAVs. Software requirement elicitation (SRE) is an important part of the UAV FCMS software development process. However, this activity suffers from ambiguity, heterogeneity and incompleteness; furthermore, because the use of UAVs is closely related to their geographic environment, geographic environment factors must be fully considered when conducting UAV FCMS SRE activity. In the knowledge engineering community, an ontology is an explicit specification of a conceptualization. Introducing the ontology method into the SRE process is an effective way to solve the above problems. This paper creates a UAV FCMS SRE ontology (SREO) based on domain knowledge and empirical data, as well as a geo-ontology based on geographic information metadata. Then, the paper integrates the above two ontologies into a new ontology. The goal of ontology integration is to analyze ontology concepts by adopting a hybrid ontology mapping method. The specific process analyzes the semantic similarities between the concepts of two ontologies and then decides whether to use a description logic (DL) strategy based on the analysis results. When the corresponding conditions are satisfied, the DL strategy is used to perform both direct and transitive reasoning for the relationships to achieve the ontology mapping, and the ontology integration is eventually implemented. Finally, this paper uses a criteria-based ontology evaluation approach to evaluate the quality of the newly integrated ontology. The evaluation results show that the UAV FCMS SREO considering geographic environment factors exhibits high quality. Further engineering practices also show that the SRE activities and the generated software requirement specifications (SRSs) exhibit a large increase in quality. Through the above activities, improvements to both the quality and reliability of UAV FCMS software can be achieved.

INDEX TERMS Requirement engineering, ontology, flight control and management system, UAV.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) are extensively used for research, monitoring and assistance in several fields of application ranging from defense, emergency and disaster management to agriculture, delivery of items, filming and so on [1]. And they have become an important developmental direction in international aviation. Moreover, their uses are closely related to the environments in which they operate, especially their geographic environment. This feature is applicable to all types of aircraft. In Feb. 2007, the U.S. Air Force's mighty Raptor was felled by the international date line (IDL). When the group of Raptors crossed over the IDL, multiple computer systems crashed on the planes. Everything from fuel subsystems, to navigation and partial communications were completely taken offline. Numerous attempts were made to "reboot" the systems to no avail. In addition, extensive engineering experience has shown that in UAV software testing, problems caused by incomplete and inconsistent software requirements are frequently detected [2], and such problems are closely related to geographic environment factors, e.g., "binding a route with a latitude of 90 degrees on a target

The associate editor coordinating the review of this manuscript and approving it for publication was Hui Liu.

waypoint may result in a UAV flight path error'', ''when a UAV is in a state of dead reckoning navigation, there may be an unsafe situation when flying at a fixed altitude in plateau'', etc. Clearly, these problems are directly related to UAV flight control and management systems (UAV FCMSs) since the quantities relevant to the geographic environment are often used as inputs to the UAV FCMSs. In addition, the special functions and tasks of UAVs give them remarkable domain characteristics. Moreover, the openness and nondeterminism of UAV operating environments, the complexity of the interactions between system components and environments, and the unpredictability of operating conditions and scenarios all intensify the knowledge-intensive development trend of UAVs and make their effectiveness highly dependent on the quality of domain knowledge.

UAV FCMSs play a decisive role in the flight performance, reliability and safety of UAVs. Moreover, software is an integral part of UAV FCMSs. Therefore, ensuring and improving the quality of UAV FCMS software is crucial to guaranteeing the quality of UAVs. Software requirement elicitation (SRE) is the most critical knowledge-intensive activity in a software development process; however, implementing effective requirement elicitation and obtaining correct, complete, consistent, and unambiguous requirement specifications remains a problem that plagues system analysts and software developers. These issues also exist in UAV FCMS SRE activities and have a significant impact on their quality. An important reason for the above problems is the lack of an effective knowledge sharing bridge between system developers and domain users [3]. In addition, the increasing scale and complexity of software systems add to the difficulties in comprehension and development. Moreover, different teams with multiple-views and multi-paradigm development methods are widely used in the development of such complex software systems, which increases the heterogeneity of software requirement specifications (SRSs) and results in inconsistent and ambiguous SRSs [3]. A knowledge-based requirement elicitation method can be used to solve the above problems; its purpose is to use domain analysis and experience to help software system stakeholders understand the application domain and define requirements. The key is to model the domain knowledge as a shareable knowledge framework. Under this framework, domain users can more easily and conveniently express their needs, while the domain developers can understand the requirements more accurately. Moreover, the heterogeneity brought about by multiple viewpoints and paradigms can be minimized. In the knowledge engineering community, an ontology is a formal and explicit specification of a shared conceptualization [4], [5]. Therefore, introducing the ontology method into the SRE process is an effective way to solve the above problems. By adopting the ontology method, the requirement knowledge can be expressed as an ontological concept and association; therefore, it is clear, complete, and consistent and is conducive to the sharing and reuse of knowledge. Literature [6] designed an ontology in a case study for co-simulation in a model-based system engineering tool-chain. They argued that an ontology refers to content about the types of objects, their properties, and their relationships, which represent domain-specific knowledge. Literature [2] studied a software requirement error pattern (SREP) based on an ontology and illustrated the application process with a certain type of UAV FCMS software, but they did not explicitly consider the influence of geographic environment factors on software function realization and reliability assurance. Because UAVs access knowledge in multiple domains, it is necessary to integrate the ontologies of different domains to achieve a relative completeness of knowledge.

This paper proposes a hybrid ontology mapping method. Based on traditional similarity calculations, this paper adopts the deductive reasoning of a description logic (DL) strategy to perform both direct and transitive reasoning for the relationships to achieve the ontology mapping, and ultimately implement ontology integration. By integrating part of the geographic information of a geo-ontology into a UAV FCMS SRE ontology (UAV FCMS SREO), the semantics of the ontology are enriched; thus, the completeness of ontology knowledge is enhanced. Furthermore, the ontology mapping can also eliminate the ambiguity and heterogeneity to some extent. On this basis, this paper adopts a criteria-based evaluation approach to evaluate the quality of the integrated ontology, including ontology validation and ontology verification. Ontology validation checks if the correct ontology has been built, whereas ontology verification checks if the ontology has been built correctly [7]. The rest of this paper is divided into the following sections: section 2 presents the state of the art in ontology integration and evaluation. Section 3 describes the construction of the UAV FCMS SREO based on domain knowledge [8]–[10], industry standards [11] and experience (software requirement errors); moreover, it describes the elicitation of the related concepts of the UAV FCMS SRE based on the literature [12], [13] and the construction of the geo-ontology. Section 4 describes the integration of the UAV FCMS SREO and the geo-ontology and presents the hybrid ontology mapping method combining the semantic similarity calculation [14] with DL [15], [16]. Section 5 presents a case study involving the implementation of the hybrid ontology mapping method and a quality evaluation of the newly integrated ontology in terms of ontology validation and ontology verification. Ontology validation is achieved by applying two validation methods [7]. The first is the ontology content evaluation, and the second is answering competency questions. Ontology verification is achieved using two methods, also [7]. The first is the ontology taxonomy evaluation, and the second is the improved FOCA methodology [17]. This section also shows the results of engineering applications. Finally, section 6 concludes the study.

II. RELATED WORK

The current state of the art of ontology integration and evaluation is presented in this section.

A. A ONTOLOGY INTEGRATION

Ontology integration refers to the process of establishing and processing mappings between ontology entities to achieve ontology alignment or ontology merging when multiple heterogeneous ontologies are applied to an ontology task [18]. Establishing accurate ontology mapping is a basic task and core component of ontology integration. According to an ontology definition model, studies on ontology mapping can be divided into grammar-based methods, concept instance-based methods, concept definition-based methods, and concept structure-based methods [19]. Grammar-based methods involve calculating the edit distance of the concept name [20], [21] and calculating the basic distance between two nodes [22]. A typical representative of the concept instance-based methods is the GLUE system of Washington University [23]. Concept definition-based methods calculate the similarity between concepts by using concept definition [24]. Concept structure-based methods consider the hierarchical structures between concepts when mapping, e.g., node relationships, semantic neighbor relationships, etc. Because of the large number of latent semantics in the hierarchical relationship of nodes, this approach has been adopted in many mapping methods; typical representatives include [22] and [24]. In addition, there are other ontology mapping methods such as rule-based ontology mapping methods [25], [26], statistics-based ontology mapping methods [23], [26], etc. Although these ontology mapping methods are diverse, their shortcomings are also obvious. Mapping methods based on various similarity measures, such as those using ''recall'' and ''precision'', are mostly limited to measuring the equivalence relationships between the entities, emphasizing grammar implementation and lacking an accurate description of semantics. Rule-based methods extract the ontology connotations through semantics and lack a relationship consistency test. Statistical methods are prone to computational errors. The introduction of a DL strategy can avoid the above deficiencies. Different researchers [27]–[29] have conducted related studies; however, most applications only involved the same or similar domain ontology (DO) concepts and rarely involved role levels; moreover, even if role matching was mentioned, it was limited to direct matching, and no intermediate concept or role delivery mapping was used for indirect matching.

Based on the above review and analysis, this paper proposes a hybrid ontology mapping method that combines semantic similarity calculation with DL.

B. ONTOLOGY EVALUATION

Ontology evaluation can be defined as ''a technical judgment of the content of the ontology with respect to a frame of reference during every phase and between phases of their life cycle'' [30]. To achieve the best results and high-quality ontology, one needs to choose from the available list of aspects of ontology to be evaluated; the right approach to evaluation; the right mix of criteria to be evaluated; and also the right tools to be used [31].

1) ASPECTS

Aspects include the vocabulary, syntax, structure, semantics, representation and context of the ontology, which are defined according to literature [31]–[33].

2) APPROACHES

The different known methods and techniques can be mainly assigned to four different kinds of approaches: technology-based, quality-attribute based, data-driven and application or task-based evaluation [31], [34]. Technology-based evaluation investigates the syntax, consistency and formal semantics and thereby ensures the correctness and usability of the ontology. Its typical representative is OOPS!, a web-based tool which is accompanied by a catalogue of potential and common pitfalls [35]. However, this approach cannot tell anything about the quality of the content and applicability of the ontology [36], [37]. Quality-based approach offers a quantitative evaluation which relies on a set of predefined metrics that measure individual quality attributes of an ontology. Yet, some of those quality metrics tend to be hard to measure and might need human experts to evaluate [33]. Its typical representatives include, OntoClean methodology, OntoMetric [38], OntoQA [39], etc. Data-driven evaluation approach concentrates on the usability of an ontology considering its future application and has also been the current focus of recent research [40]–[42]. This approach attempts to analyze how adequate an ontology covers the domain but is not applicable to determine the correctness or clarity of the ontology [43], [44]. Application or task-based evaluation approach would typically involve evaluating how effective an ontology is in the context of a specific application [45]. This approach exhibits a limitation: the result obtained from one task may not be useful for another task as each task is different [34], i.e., it is not suited for a general evaluation, because every ontology must be evaluated individually depending on the application context [43].

3) CRITERIA

This kind of evaluation approach is done by humans who try to assess how well the ontology meets a set of predefined criteria, standards, requirements, etc. Reference [46] Various criteria have been proposed in literature to evaluate the quality of ontology [31]–[33], [47]: consistency, completeness, accuracy, conciseness, correctness, computational efficiency, adaptability, clarity.

4) TOOLS

Various tools have been developed to support the task of ontology evaluation, each concerned with different aspects of evaluation. There exist tools for checking the consistency, the structure or modeling mistakes of the ontology [37]. Various available tools include: ODEClean, ODEval, AEON, Eyeball, Moki, XD-Analyzer, OQuaRE, OntoCheck, OntoQA, OntoClean, OntoMetric, ACTiveRank, OOPS!, ODEval, oQual [31].

Based on the above review and analysis, this paper adopts a criteria-based evaluation approach to evaluate the quality of the integrated ontology, including ontology validation and ontology verification.

III. CONSTRUCTION OF THE UAV FCMS SREO AND GEO-ONTOLOGY

Currently, the widely accepted ontology construction guideline is the five criteria proposed by Gruber, i.e., clarity, coherence, extendibility, minimal encoding bias, and minimal ontological commitment [4]. Moreover, there are other supplementary rules for specific operations. The most famous rules are Arpirez's three criteria, i.e., the standardization of concept name, the diversification of concept level, and the minimization of semantic distance [48]. Following the above rules, combined with the engineering application background of this research, this paper uses the TOVE method [49] to guide the ontology construction.

A. ONTOLOGY FORMALIZATION

Ontologies provide interrelations between elements, hierarchy among domain concepts, data structure and the integration of heterogeneous information [50]. The different ontology classes, relationships, constraints and axioms define a common vocabulary to share knowledge [51].

Formally, an ontology can be defined as the tuple:

$$O = (C, H, I, R, P, A) \quad (1)$$

where: $C = C_C \cup C_I$ is the set of entities of the ontology. The set C_C consists of classes, i.e., concepts that represent entities that describe a set of objects, while the set C_I is constituted by instances.

$H = \{kind\_of(c_1, c_2)|c_1 \in C_C, c_2 \in C_C\}$ is the set of taxonomic relationships between the concepts, which define a concept hierarchy and are denoted by "kind_of (c_1, c_2)", meaning that c_1 is a subclass of c_2.

$I = \{is\_a(c_1, c_2)|c_1 \in C_I \wedge c_2 \in C_C\} \cup \{\mathrm{prop_K}(\mathrm{c_i}, \mathrm{value})|\mathrm{c_i} \in C_\mathrm{I}\} \cup \{\mathrm{rel_K}(\mathrm{c_1}, \mathrm{c_2}, \ldots, \mathrm{c_n})|\forall \mathrm{i}, \mathrm{c_i} \in C_\mathrm{I}\}$ is the set of relationships between ontology elements and its instances.

$R = \{rel_K(c_1c_2c_n)|\forall i, c_i \in C_C\}$ is the set of ontology relationships that are neither "kind_of" nor "is_a". The relationships between concepts mainly have two types: hierarchical relationships and non-hierarchical relationships [52].

$P = \{prop_K(c_i, datatype)|c_i \in C_C\}$ is the set of properties of ontology entities and its basic datatype.

$A = \{condition_x \Rightarrow conclusion_y(c_1, c_2, \ldots, c_n)|\forall j, c_j \in C_C\}$ is a set of axioms, rules that allow checking the consistency of an ontology and infer new knowledge through some inference mechanism. The term "$\mathrm{condition_x}$" is given by $condition_x = \{(cond_1, cond_2, \ldots, cond_n)|\forall z, cond_z \in H \cup I \cup I \cup R\}$

B. UAV FCMS SREO CONSTRUCTION

1) THE UAV FCMS SREO CONSTRUCTION PROCESS

A UAV FCMS SREO has a variety of contents that involve both the UAV FCMS field and the software engineering field; therefore, the knowledge system can be modeled by a knowledge aided design system (KADS) [5]. The knowledge hierarchy in this model is clearly divided, and each layer of knowledge exhibits good maintainability and reusability. Furthermore, to enable the above knowledge model to play a role in knowledge sharing and reuse, it is necessary to integrate relatively independent knowledge layers through the ontology to form a knowledge system. This paper constructs both generalization layer and domain layer ontologies. The UAV FCMS SREO construction process includes the elicitation of domain knowledge; the elicitation of concepts, concept attributes, concept hierarchies and concept relationships; and the use of a formal language to represent these definitions.

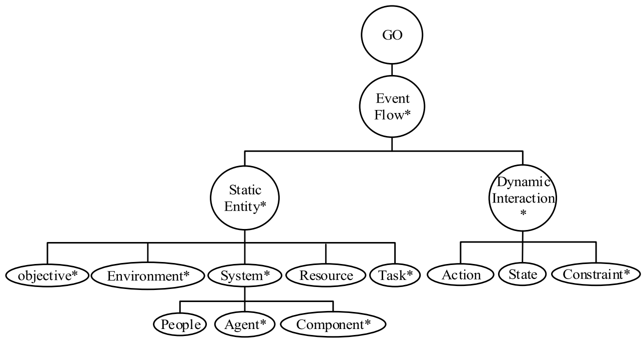


FIGURE 1. Hierarchy of concept classes in GO.

2) UAV FCMS SREO CONCEPTS AND RELATIONSHIPS

- UAV FCMS software-related concepts and relationships

First, a generalized ontology (GO) is constructed according to the KADS. Figure 1 shows the hierarchy of the concept classes in the GO. The concept class with a "$*$" is a non-terminating concept class, and the rest are all terminating concept classes. Furthermore, a portion of the concept dictionary table and a portion of the GO concept space are shown in Table 1-Part A and Table 2 , respectively.

Second, the UAV FCMS SRE DO is built. The UAV FCMS software is the core part of the UAV FCMS. Figure 2 shows the internal structure and main external interfaces of the UAV FCMS.

Due to the variety of concepts involved, in the concept selection stage, this paper uses the term 'weighting technique' along with equation 2 [53].

$$AvgConceptScore = \frac{\sum ConceptScore}{\sum Concepts} \quad (2)$$

A portion of the concept dictionary table is shown in Table 1-Part B.

- The related concepts and relationships of the SREP

Definition 1: The SREP refers to the error produced in the software requirement development stage, which occurs repeatedly in a specific error lifetime scenario, spreads in the subsequent design and implementation, and may cause a system (component) to fail to perform the expected function or affect the maintainability of the system. Such errors

TABLE 1. Portion of concept dictionary table.

	Concepts				
A.GO concepts	EventFlow Resource State	DynamicInteraction Task Agent	StaticEntity Objective People	System Action Component	Environment Constraint
B.UAV FCMS software concepts	FCMS UAV	FCMComputer ADC	FCMSSoftware INS	ServoSystem CNS	Sensor GPS
C.SREP concepts	SREP Scenario Solution		ErrorManifestation DocumentationError EnvironmentalError	PerformanceError SafetyError VersionControlError	InterfaceError MaintenanceError FunctionalError

TABLE 2. Portion of GO concept space.

Concepts/ relationships	P^C	Descriptions
Objective	partial order	Objective ⟶ Sub-objective
Task	partial order	Task ⟶ Sub-task
has	-	Task $\xrightarrow{has}$ Objective
need	$support^{-1}$	Task $\xrightarrow{need}$ Resource
support	$need^{-1}$	Resource $\xrightarrow{support}$ Task
interact	symmetry	System $\xrightarrow{interact}$ Environment
interact	symmetry	Environment $\xrightarrow{interact}$ System
produce	-	Action $\xrightarrow{produce}$ State

Notes: “need” and “$need^{-1}$” are mutually inverse, “support” and “$support^{-1}$” are mutually inverse.

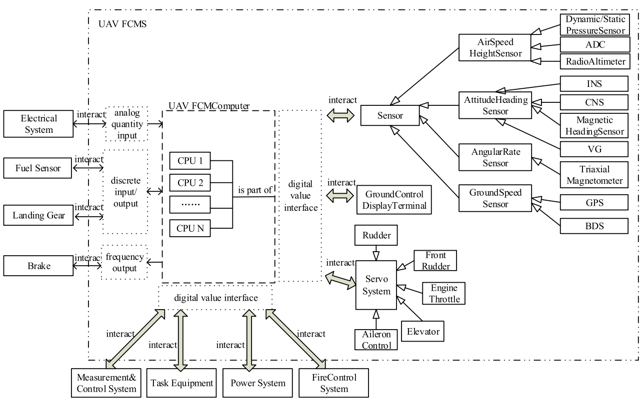


FIGURE 2. Internal structure and main external interfaces of UAV FCMS.

are general and common in a specific scenario and can be corrected by various means.

The definition shows that the core components of an SREP are ‘‘scenario’’, ‘‘error-manifestation’’ and ‘‘solution’’.

Furthermore, for the sake of simplicity, this paper selects these three concepts as part of the collection of ontology concept classes for the UAV FCMS SREO. The concept dictionary table is shown in Table 1-Part C.

- The concept classes and relationships of the UAV FCMS SREO

Because the concepts and relationships associated with the SREP are relatively independent of other concepts and relationships of the UAV FCMS SREO, Figure 3 only shows the unified model language (UML) diagram representations of the concepts and relationships of the UAV FCMS SREO other than the SREP. “⟶” represents the inheritance relationship and “⇢” represents the relationships other than the inheritance relationship.

C. THE SELECTION OF GEOGRAPHIC INFORMATION METADATA AND DOMAIN ONTOLOGY CONSTRUCTION

1) GEOGRAPHIC INFORMATION METADATA

The use of digital geographic data is intended to simulate and describe the real world for computer analysis and the graphical display of information [13]. In the current digital geographic data domain, the authoritative and available domain concept classification standard and domain system structure include the contents given in literature [12] and [13]. The FGDC and ISO TC/211 assert that metadata contain data on data content, quality, conditions, and other characteristics.

However, the contents residing at the geographic information metadata level are insufficient for actual domain use. This is due to significant differences between the metadata and the ontologies, (1) the metadata mainly focus on the external form features of information resources, whereas an ontology mainly focuses on the inherent content characteristics of the information resources; (2) the metadata focus on the description and positioning of the information resources, while an ontology organizes and manages the knowledge content. More critically, metadata lack semantic description capabilities; therefore, they cannot solve the problem of the semantic heterogeneity of data sets or the description of the implicit relationships between data categories. Thus, it is necessary to establish an ontology layer on the top of the metadata and perform semantic description and ontology reasoning.

In this research, domain experts built a hierarchical concept system by selecting the parts associated with the UAV FCMS SRE considering the geographic environment factors in the field of digital geographic data, and each concept was described by a set of attributes. Because our research focuses on the UAV-related geo-ontology construction, the main role of the UAVs in this case is intelligence collection. According to literature [13], the corresponding ontology was defined as an ‘‘intelligent military domain ontology’’ (IMDO). Table 3 outlines a portion of the intelligent military domain-related geographic information metadata.

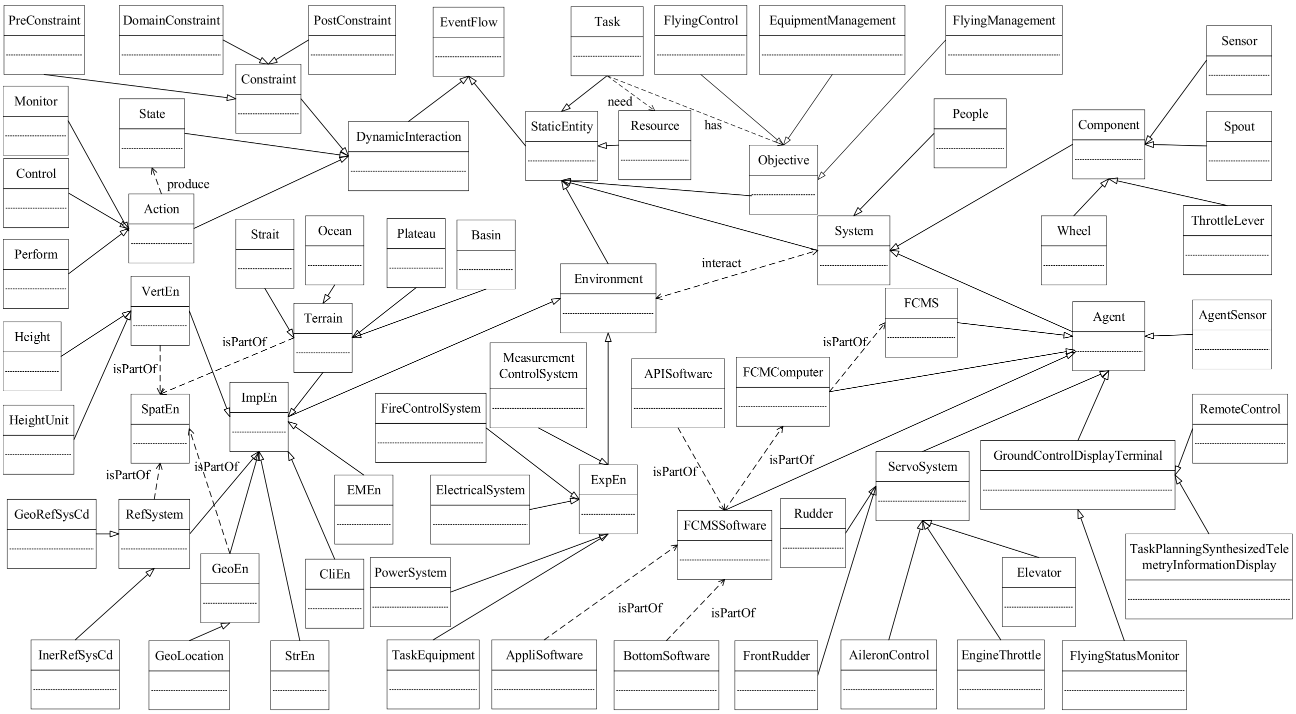


FIGURE 3. UML diagram representations of concepts and relationships of UAV FCMS SREO.

TABLE 3. Portion of intelligent military domain-related geographic information metadata.

Metadata		
DataTypeInfo	Extent	VertExtent
GeoExtent	VertUoM	BL/L
EastBL	WestBL	NorthBL
SouthBL	GeoBndBox	System
Agent	Satellite	AgentSensor
RefSysInfo	RefSystem	GeoRefSysCd
MdCoRefSys	vertMinVal	vertMaxVal

Table 3 shows that these contents contain rich feature information such as longitude, latitude, height, etc. Moreover, they express not only certain geographic information semantics but also other intelligent military domain-related information. Therefore, the relationships between the standard feature sets of different concepts can be found based on the shared feature attribute sets of concepts of different ontologies. Moreover, the integration between different domain concepts can be realized based on the construction of corresponding conceptual systems and architectures. Section 3 addresses this challenge.

2) CONSTRUCTION OF AN IMDO

The greatest difference between a geo-ontology and a general ontology is that the former possesses not only general attribute characteristics but also spatial characteristics; thus, an IMDO also has such characteristics. The main idea in describing geographic element-related concepts in the IMDO is to divide the described objects into two categories: conceptual attributes and spatial attributes. The conceptual attributes describe the non-spatial ontology attributes in terms of five aspects—matter, form, spatial distribution, function, and rank, while the spatial attributes describe the ontology in terms of three aspects—topological relationship, positional relationship, and directional relationship. The following takes the Yangtze River as an example to describe the semantic features. The OWL code for the formal description of the Yangtze River is shown in Figure 4.

Example: the Yangtze River—water (matter) + flow (form) + linear (spatial distribution) + traffic (function) + economy (rank) + separation from the Yellow River (topological relationship)... + east-west direction (directional relationship) . . . + in the south of Beijing (positional relationship) . . .

The components of the IMDO include "Environment", "System", "DataTypeInfo", and "RefSystem". The UML diagram representations of the concepts and relationships are shown in Figure 5.

IV. THE INTEGRATION OF THE UAV FCMS SREO AND THE IMDO

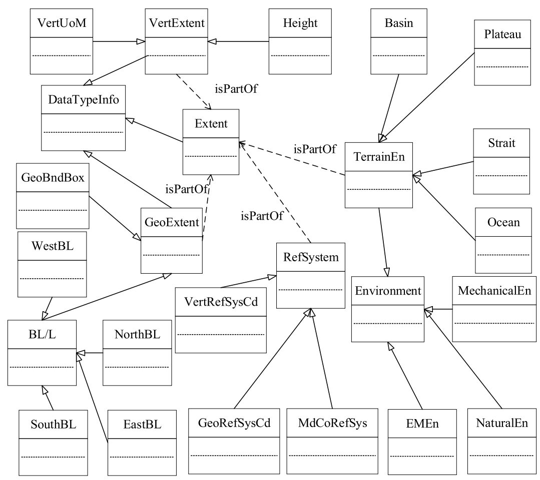
A. THE HYBRID ONTOLOGY MAPPING METHOD

This paper realizes the integration of the UAV FCMS SREO and the IMDO through ontology mapping. It performs a similarity analysis [14] of the concepts in the above two ontologies. The specific processes include, lexical comparison, structural comparison and relational comparison.

However, the semantic similarity analysis method is limited to the measurement of the equivalent relations between the entities and lacks a precise description of the semantics.

```
< owl: Classrdf: ID = "the Yangtze River ">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "river " >
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "water">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "flow">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "linear">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "traffic">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "economy">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "separation from the Yellow River">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "east-west direction">
< rdfs: sub ClassOf >
< owl: Classrdf: ID = "in the south of Beijing">
< rdfs: sub ClassOf >
......
```

FIGURE 4. **OWL code for the Yangtze River formal description.**



Notes: For the sake of simplicity, the "System"-related relationships are not included in this figure.

FIGURE 5. **UML diagram representations of concepts and relationships of IMDO.**

In some cases, the semantic similarity values obtained solely by this method are not accurate. Thus, it is necessary to use DL to detect the matching relationship between the concepts, as well as between the concept and role in different domain ontologies, and realize the matching from one ontology to another. For specific processes, an ontology API is initially used to parse the two ontologies to be integrated, and the concepts and roles are acquired; then, a data dictionary is used to complete the string matching of the concepts and roles; finally, an inference engine performs reasoning to make the concepts and roles in one ontology gradually match the other ontology according to inference rules. The process of the hybrid ontology mapping method is shown in Figure 6.

B. SIMILARITY ANALYSIS

1) LEXICAL COMPARISON

The activity ''lexical comparison'' performs a lexical comparison between the representative terms of the elements obtained from the two ontologies O_1 and O_2. This activity

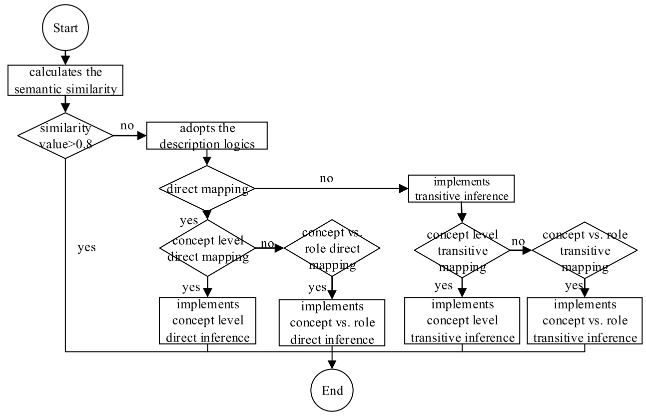


FIGURE 6. **Process of hybrid ontology mapping method.**

takes as input two lists of terms ''list of terms in O_1'' and ''list of terms in O_2'' composed by elements of the sets C_C, P, R and H of the ontology definition in section III. Then, each term is enriched with its synonyms by consulting a lexical dictionary. Then, for each ontology element, the terms and their synonyms are compared with the correspondent terms of the other list. For the result, two types of values can be obtained: 1 for perfect match and 0 for no match. In addition, we also need to define synonyms in combination with domain features.

2) STRUCTURAL COMPARISON

The activity ''structural comparison'' makes a similarity analysis between the terms in the sets C_C of the ontologies O_1 and O_2. A similarity measure which considers the hierarchical structures in which they are inserted was adapted from the work of Mendes and Girardi [54].

$$sim(c_i, c_j) = \frac{2^*|c_{iH} \cap c_{jH}|}{|c_{iH}| + |c_{jH}|} \tag{3}$$

where c_i is a class of ontology O_1; c_j is a class of ontology O_2; C_{iH} is the list of super classes of class c_i in the hierarchy H; C_{jH} is the list of super classes of class c_j in the hierarchy H.

3) RELATIONAL COMPARISON

The activity ''relational comparison'' performs a similarity analysis between the non-taxonomic relationships in the ontologies. Thus, when a lexical similarity is found between two relationships R_1 and R_2 of the ontologies O_1 and O_2, a weight is assigned to the result of the structural comparison of the concepts related by them. This assignment of weights based on the identification of elements of the set R, increases the value of the similarity measure and makes it more adequate once the comparison uses the whole structure of the ontology realizing a semantic comparison between its elements.

C. DESCRIPTION LOGIC

1) THE FEASIBILITY OF DL USAGE

To realize the integration of a UAV FCMS SREO and an IMDO, it is necessary to clarify the semantic relationships between the ontology concepts. Because the concept level can ignore the extension of concepts, i.e., the instance sets, the semantic relationships of concepts are completely determined by their connotation relationships. The calculation of the connotation relationships between the UAV FCMS SREO concepts and the IMDO concepts involves calculating the concept attribute set and its range. It is a set operation that satisfies the typical set operation syntax and can define the concept connotation relationships as four semantic relations: synonymous relationships (semantic equivalence relations), upper and lower semantic relationships (parent/child concept relationships), semantic intersections and semantic non-intersections. Therefore, the ontology integration can be studied by DL.

2) THE MATCHING METHOD BASED ON THE DL

DL is descended from so-called "structured inheritance networks" with the basic components as concepts, roles, and individuals [15] and is widely used as the basis for ontology description languages. The vocabulary consists of concepts, which denote the sets of individuals, and roles, which denote the binary relationships between the individuals. A DL system consists of four parts: a constructor set representing the concepts and relationships, a reasoning mechanism on Tbox/Abox, the Tbox, and the Abox. The Tbox is a set of axioms that describes the structure of a domain, including concept definitions and the inclusion relationships of concepts. It is implemented through a set of statements describing the general attributes of concepts. Connotation axioms are invariant. The Abox is a set of axioms that describes the named individuals. It contains the extended knowledge, including instance assertions and relational assertions. The extended knowledge is often considered to be constantly changing [16]. Usually, a DL system contains at least the following constructors: intersections(∩), unions(∪), negations(-), existential quantifiers(∃), ∃ universal quantifiers(∀∀), bottom concepts (⊥), universal concepts (⊤) [15]. Complex concepts and roles can be constructed through simple concepts and relationships [51].

- The definition of a matching relationship

Definition 2: For a C (a concept or role) of O_i and a D (a concept or role) of O_j, iff, an arbitrary individual, satisfying the following five mappings in turn, means that the relationships between the two are an equivalence relation, a subsumption relation, a supersumption relation, an overlapping relation or a disjoint relation respectively,

$$i:C \xrightarrow{\equiv} j:D \tag{4}$$

$$i:C \xrightarrow{\subseteq} j:D \tag{5}$$

$$i:C \xrightarrow{\supseteq} j:D \tag{6}$$

$$i:C \xrightarrow{\&} j:D \tag{7}$$

$$i:C \xrightarrow{\perp} j:D \tag{8}$$

where:

(4) $|= i:C \xrightarrow{\subseteq} j:D \wedge i:C \xrightarrow{\supseteq} j:D$

(8) $|= i:C \xrightarrow{\subseteq} -j:D$

(5) and (6) are mutually inverse.

- Direct inference

The implicit knowledge contained in a DL knowledge base can be made explicit through inferences [15]. The following describes the reasoning rules for the two aspects of "concept level" and "concept vs. role" [28], [29].

a: CONCEPT LEVEL

From the perspective of Abox, the mapping rules of the relationships between the concept "X" of O_i and the concept "Y" of O_j are given, and "a" represents an arbitrary individual.

Rule 1: Concept equivalence,

$$\textit{iff}\ \forall a,\quad X(a) \leftrightarrow Y(a)| = (i:X \equiv j:Y)$$

Rule 2: Concept subsumption,

$$\textit{iff}\ (1)\forall a,\quad X(a) \rightarrow Y(a) \wedge$$
$$(2)\exists b,\quad Y(b) \wedge -X(b)| = (i:X \subseteq j:Y)$$

Rule 3: Concept overlapping,

$$\textit{iff}\ \forall a,\quad X(a) \wedge Y(a)| = (i:X\&j:Y)$$

Rule 4: Concept disjoint,

$$\textit{iff}\ \forall a,\quad X(a) \leftrightarrow -Y(a)| = (i:X \leftrightarrow -j:Y)$$

b: CONCEPT VS. ROLE

In practical applications, there is often a semantic matching relationship between a concept of an ontology and a role of another ontology. For example, "WestBL", "EastBL", "NorthBL" and "SouthBL" exist as the classes in an IMDO, yet they exist as object attributes in a UAV FCMS SREO, corresponding to "hasWestBL", "hasEastBL", "hasNorthBL" and "hasSouthBL", respectively. Thus, a concept of one ontology can be described as a role of another ontology. To determine the correspondence between these two ontologies, the analysis of the concept, as well as the range and domain of the role, must be examined. Set $X_i, X_i' \in O_i$, where X_i and X_i' denote the concept, and $R \in O_j$, where R denotes the role. R_D and R_R represent the domain and range of the role R of O_j, respectively.

Rule 5: Tbox Concept vs. role equivalence,

$$\textit{iff}\ (X_i \equiv R_D) \wedge (X_i' \equiv R_R)| = (X_i \equiv R)$$

Rule 6: Tbox Concept vs. role subsumption,

$$\textit{iff}\ (1)(X_i \subseteq R_D) \wedge (X_i' \subseteq R_R) \vee$$
$$(2)(X_i \equiv R_D) \wedge (X_i' \subseteq R_R) \vee$$
$$(3)(X_i \subseteq R_D) \wedge (X_i' \equiv R_R)| = (X_i \subseteq R)$$

Rule 7: Tbox Concept vs. role overlapping,

$$\textit{iff}\ (X_i \wedge R_D) \wedge (X_i' \wedge R_R)| = (X_i\&R)$$

TABLE 4. Transitive mapping relations.

X_1RX_3 \ X_2RX_3	X_1RX_2				
	$\equiv$	$\subseteq$	$\supseteq$	&	$\perp$
$\equiv$	$\equiv$	$\subseteq$	$\supseteq$	&	$\perp$
$\subseteq$	$\subseteq$	$\subseteq$	$\cong$	&	$\cong$
$\supseteq$	$\supseteq$	$\cong$	$\supseteq$	$\cong$	$\cong$
&	&	&	$\cong$	&	&
$\perp$	$\perp$	$\perp$	$\cong$	&	$\perp$

Rule 8: Tbox Concept vs. role disjoint,

$$iff\,(X_i \equiv -R_D) \vee (X_i' \equiv -R_R)| = (X_i \perp R_j)$$

- Transitive inference

A direct inference is limited to the relationships between two concepts or between a concept and a role. In practice, it is often difficult to find a direct mapping relationship between the two; therefore, it is necessary to use a transitive inference through intermediate concepts [16]. As shown in Table 4 , X_1, X_2 and X_3 denote the concept or role. $\cong$ and & denote that the relationship between two concepts is a fuzzy relation ($\subseteq$, $\supseteq$, &) or a disjoint relation.

Rule 9: $X_1 \perp X_2, X_3 \subset X_1, X_4 \subset X_2| = X_3 \perp X_4$

Rule 10: $X_1 \perp X_2, X_1 \perp X_4, X_4 \subset X_2, X_3 \subset X_1| = X_2 \perp X_3$

Rule 11: $X_1 = X_2, X_3 \subset X_1| = X_3 \subset X_2$

Rule 12: $X_1 \subset X_2, X_2 \subset X_3| = X_1 \subset X_3$

Rule 13: $X_1 \cong X_2, X_1 \subset X_3| = X_2 \cong X_3$

In short, guided by the process of the hybrid ontology mapping method, using a semantic similarity analysis and a DL strategy, the integration of a UAV FCMS SREO and an IMDO is finally realized.

V. CASE STUDY

This paper uses the UAV FCMS SREO and the IMDO as experimental objects to perform an ontology integration and evaluation. The UAV FCMS SREO is mainly based on relevant literature, industry standards, and the development and testing experience of multiple continuous versions of a certain type of UAV FCMS software. Because of the need to consider the geographic environment factor on the UAV FCMS SREO, the hybrid ontology mapping method based on semantic similarity analysis and DL is used for ontology integration. Moreover, the improved FOCA method is used to evaluate the quality of the newly integrated ontology. In addition, the results of engineering applications also illustrate the effectiveness of the method. It should be noted that in this study, only a portion of the UAV FCMS SREO is related to the IMDO.

A. IMPLEMENTATION OF THE HYBRID ONTOLOGY MAPPING METHOD

1) SEMANTIC SIMILARITY CALCULATION

- The lexical comparison

a) List of the representative terms in O_1:

TABLE 5. List of equivalent terms.

List of terms in O_1	List of terms in O_2
System	System
Agent	Agent
AgentSensor	AgentSensor
SpatEn	Extent
RefSystem	RefSystem
GeoRefSysCd	GeoRefSysCd
GeoEn	GeoExtent
GeoLocation	GeoBndBox
VertEn	VertExtent
HeightUnit	VertUoM
Height	Height
CliEn	CliEn
StrEn	MechanicalEn
EMEn	EMEn
Terrain	TerrainEn
Basin	Basin
Plateau	Plateau
Ocean	Ocean
Strait	Strait

C_C ={System, Agent, AgentSensor, Environment, SpatEn, GeoEn, VertEn, RefSystem, GeoRefSysCd, GeoLocation, Terrain, EMEn, Strait, Ocean, Basin, Plateau, CliEn, StrEn, Height, HeightUnit};
R={isPartOf (GeoEn, SpatEn), isPartOf (VertEn, SpatEn), isPartOf (RefSystem, SpatEn), isPartOf (Terrain, SpatEn)};
H={see Figure 3}.

b) List of the representative terms in O_2:
C_C ={System, Agent, AgentSensor, RefSystem, GeoRefSysCd, Extent, GeoExtent, GeoBndBox, VertExtent, VertUoM, Height, Environment, TerrainEn, Strait, Ocean, Basin, Plateau, NaturalEn, MechanicalEn, EMEn};
R={isPartOf (GeoExtent, Extent), isPartOf (VertExtent, Extent), isPartOf (RefSystem, Extent), isPartOf (TerrainEn, Extent)};
H={see Figure 5}.

The list of equivalent terms shown in Table 5 is obtained based on the conventional lexical comparison method with the domain features.

- The structural comparison

The structural comparison activity is performed with the aim of analyzing the similarity between the hierarchies of concepts presented in the ontologies; therefore, equation (3) is used. The intersections in this equation are defined from the equivalences described in Table 5 . The result varies from 0 to 1, depending on how similar is the hierarchical structure between the lists. It is noted that for the concepts of different levels (the generalization layer or domain layer), the similarity values should be calculated in combination with the path of the corresponding level. O_1 has been divided into a generalization layer and a domain layer, while for O_2, except for "Thing", "System", "Agent" and "Environment" belonging to the generalization layer, the remaining values should belong to the domain layer.

TABLE 6. Similarity values.

List of terms in O_1	List of terms in O_2	Similarity values
System	System	0.67
Agent	Agent	0.75
AgentSensor	AgentSensor	1.00
SpatEn	Extent	0.50
RefSystem	RefSystem	0.67
GeoRefSysCd	GeoRefSysCd	0.80
GeoEn	GeoExtent	0.50
GeoLocation	GeoBndBox	0.67
VertEn	VertExtent	0.50
HeightUnit	VertUoM	0.67
Height	Height	0.67
CliEn	NaturalEn	0.67
StrEn	MechanicalEn	0.67
EMEn	EMEn	0.67
Terrain	TerrainEn	0.67
Basin	Basin	0.80
Plateau	Plateau	0.80
Ocean	Ocean	0.80
Strait	Strait	0.80

TABLE 7. Similarity values increased by weights.

List of terms in O_1	List of terms in O_2	R	Similarity values
System	System	-	0.67
Agent	Agent	-	0.75
AgentSensor	AgentSensor	-	1.00
SpatEn	Extent	hasPart	**0.60**
RefSystem	RefSystem	isPartOf	**0.77**
GeoRefSysCd	GeoRefSysCd	-	0.80
GeoEn	GeoExtent	isPartOf	**0.60**
GeoLocation	GeoBndBox	-	0.67
VertEn	VertExtent	isPartOf	**0.60**
HeightUnit	VertUoM	-	0.67
Height	Height	-	0.67
CliEn	NaturalEn	-	0.67
StrEn	MechanicalEn	-	0.67
EMEn	EMEn	-	0.67
Terrain	TerrainEn	isPartOf	**0.77**
Basin	Basin	-	0.80
Plateau	Plateau	-	0.80
Ocean	Ocean	-	0.80
Strait	Strait	-	0.80

Notes: "haspart" and "isPartOf" are mutually-inverse.

Taking the term "Agent" as an example,
O_1: Thing→EventFlow→StaticEntity→System→Agent
O_2: Thing→System→Agent
Substituting the values into equation (3),

$$sim(Agent, Agent) = \frac{2^*|c_{iH} \cap c_{jH}|}{|c_{iH}| + |c_{jH}|} = \frac{2^*3}{8} = 0.75.$$

Taking the terms "SpatEn" and "Extent" as an example,
O_1: ImpEn→SpatEn
O_2: DataTypeInfo→Extent
Substituting the values into equation (3),

$$sim(SpatEn, Extent) = \frac{2^*|c_{iH} \cap c_{jH}|}{|c_{iH}| + |c_{jH}|} = \frac{2^*1}{4} = 0.50.$$

Table 6 presents the similarity values between the concepts of the two ontologies.

- The relational comparison

The relational comparison is performed with the aim of analyzing the similarity among non-taxonomic relationships of the ontologies. Thus, for each lexical similarity found between terms belonging of the set R, weights are assigned to the values obtained from the activity of structural comparison. Table 7 presents the similarity values for some concepts of the ontologies O_1 and O_2. The concept "SpatEn" of O_1 and the concept "Extent" of O_2 have a non-taxonomic relationship in common (hasPart). Thus, a weight of 0.1 is added to the value of similarity obtained in the previous activity, being it updated to 0.60. Similar situations exist for "GeoEn" and "GeoExtent", "VertEn" and "VertExtent", "RefSystem" and "RefSystem", and "Terrain" and "TerrainEn".

2) DESCRIPTION LOGIC

It can be seen from Table 7 that the similarity values of some concepts are improved after the weights are added. However, the overall similarity values of the concepts are still low, which is inconsistent with the actual domain situation and does not fully reflect the true semantic information. In addition, there are some concepts with new semantics that require new concepts to be added. Therefore, it is necessary to adopt the DL strategy further. The DL strategy is performed according to the process in Figure 6.

- Concept equivalence

From the concept equivalence rules, the concept "A: GeoRefSysCd" of O_1 is equivalent to the concept "B: GeoRefSysCd" of O_2, i.e., A: GeoRefSysCd $\equiv$ B: GeoRefSysCd (semantic layer). "A: GeoRefSysCd" and "B: GeoRefSysCd" can be merged. Moreover, the concepts of the first column in Table 7 from "A: HeightUnit" to "A: Strait" are each equivalent to the concepts of the second column in Table 7, and the corresponding concepts of the two columns can be merged.

- Concept vs. role equivalence

$$\text{A: hasBL/L (A: SouthBL} \cup \text{NorthBL} \cup \text{WestBL} \cup \text{EastBL, A: GeoLocation)} \quad (9)$$

$$\text{B: GeoBndBox} \equiv \text{A: GeoLocation} \quad (10)$$

$$\text{B: BL/L} \equiv \text{A: SouthBL} \cup \text{NorthBL} \cup \text{WestBL} \cup \text{EastBL} \quad (11)$$

Using equation (9)~(11), the inferred correspondence is,

$$\text{A: hasBL/L (B: BL/L, B: GeoBndBox)} \quad (12)$$

Equation (9) and equation (12) correlate the correspondence according to the class-property similarity condition and the following can therefore be inferred,

$$\text{A: hasBL/L} \equiv \text{B: BL/L}$$

Then, "B: BL/L" of O_2 and "A: hasBL/L" of O_1 can be merged. Similarly, "B: SouthBL" of O_2 and "A: hasSouthBL" of O_1 can be merged; "B: NorthBL" of O_2 and "A: hasNorthBL" of O_1 can be merged; "B: WestBL" of O_2

and "A: hasWestBL" of O_1 can be merged; "B: EastBL" of O_2 and "A: hasEastBL" of O_1 can be merged.

- Transitive inference

a:

$$\text{A: RefSystem} \equiv \text{A:GeoRefSysCd} \cup \text{A: InerRefSysCd} \tag{13}$$

$$\text{B: RefSystem} \equiv \text{B: GeoRefSysCd} \cup \text{B: VertRefSysCd} \cup \text{B: MdCoRefSys} \tag{14}$$

$$\text{A: GeoRefSysCd} \equiv \text{B: GeoRefSysCd} \tag{15}$$

$$\text{A: InerRefSysCd} \leftrightarrow \text{-(B: MdCoRefSys} \cup \text{B: VertRefSysCd)} \tag{16}$$

Using equation (13)~(16), the inferred correspondence is,

A: RefSystem & B: RefSystem

Therefore, the concepts different from O_1 in O_2 can be added to O_1. The integrated ontology contains the following concepts: "A: RefSystem", "A:GeoRefSysCd", "A: InerRefSysCd", "B: MdCoRefSys" and "B: VertRefSysCd".

Similarly,

$$\text{A:Agent} \equiv \text{A:BottomSoftware} \cup \text{A:FCMS} \cup \text{A:APISoftware} \cup \text{A:FCMSSoftware} \cup \text{A:GroundControlDisplayTerminal} \cup \text{A:AgentSensor} \cup \text{A:ServoSystem} \cup \text{A:ApplicSoftware} \cup \text{A:FCMComputer} \tag{17}$$

$$\text{B: Agent} \equiv \text{B: AgentSensor} \cup \text{B: Satellite} \tag{18}$$

$$\text{A: AgentSensor} \equiv \text{B: AgentSensor} \tag{19}$$

$$\text{A: BottomSoftware} \cup \text{A: FCMS} \cup \text{A: APISoftware} \cup \text{A: FCMSSoftware} \cup \text{A: GroundControlDisplayTerminal} \cup \text{A: ServoSystem} \cup \text{A: ApplicSoftware} \cup \text{A: FCMComputer} \leftrightarrow \text{-(B: Satellite)} \tag{20}$$

Using equation (17)~(20), the inferred correspondence is,

A: Agent & B: Agent

Therefore, the concepts different from O_1 in O_2 can be added to O_1. The integrated ontology contains the following concepts: "A: Agent", "A: AgentSensor", "A: BottomSoftware", "A: FCMS", "A: APISoftware", "A: FCMSSoftware", "A: GroundControlDisplayTerminal", "A: ServoSystem", "A: ApplicSoftware", "A: FCMComputer", "B: Satellite".

b:

$$\text{A: HeightUnit} \equiv \text{B: VertUoM} \tag{21}$$

$$\text{A: Height} \equiv \text{B: Height} \tag{22}$$

$$\text{A: VertEn} \equiv \text{A: HeightUnit} \cup \text{A: Height} \tag{23}$$

$$\text{B: VertExtent} \equiv \text{B: VertUoM} \cup \text{B: Height} \tag{24}$$

TABLE 8. List of terms in O_N.

List of terms in O_1	List of terms in O_2	List of terms in O_N
System	System	System
Agent	Agent	Agent
AgentSensor	AgentSensor	AgentSensor
SpatEn	Extent	SpatEn
RefSystem	RefSystem	RefSystem
GeoRefSysCd	GeoRefSysCd	GeoRefSysCd
GeoEn	GeoExtent	GeoEn
GeoLocation	GeoBndBox	GeoLocation
VertEn	VertExtent	VertEn
HeightUnit	VertUoM	HeightUnit
Height	Height	Height
hasBL/L	BL/L	hasBL/L
hasSouthBL	SouthBL	hasSouthBL
hasNorthBL	NorthBL	hasNorthBL
hasWestBL	WestBL	hasWestBL
hasEastBL	EastBL	hasEastBL
CliEn	NaturalEn	CliEn
StrEn	MechanicalEn	StrEn
EMEn	EMEn	EMEn
Terrain	TerrainEn	Terrain
Basin	Basin	Basin
Plateau	Plateau	Plateau
Ocean	Ocean	Ocean
Strait	Strait	Strait

Using equation (21)~(24), the inferred correspondence is,

$$\text{A: VertEn} \equiv \text{B: VertExtent}$$

Therefore, these two concepts can be merged.

Similarly,

The previous inference has merged the concept "B: BL/L" of O_2 with the concept "A: hasBL/L" of O_1, therefore, the inference,

$$\text{B: GeoExtent} \equiv \text{B: GeoBndBox} \tag{25}$$

can be established.

$$\text{A: GeoEn} \equiv \text{A: GeoLocation} \tag{26}$$

$$\text{A: GeoLocation} \equiv \text{B: GeoBndBox} \tag{27}$$

Using equation (25)~(27), the inferred correspondence is,

$$\text{B: GeoExtent} \equiv \text{A: GeoEn}$$

Therefore, these two concepts can be merged.

Similarly,

$$\text{A: SpatEn} \equiv \text{A: GeoEn} \cup \text{A: VertEn} \tag{28}$$

$$\text{B: Extent} \equiv \text{B: GeoExtent} \cup \text{B: VertExtent} \tag{29}$$

$$\text{B: GeoExtent} \equiv \text{A: GeoEn} \tag{30}$$

$$\text{A: VertEn} \equiv \text{B: VertExtent} \tag{31}$$

Using equation (28)~(31), the inferred correspondence is,

$$\text{B: Extent} \equiv \text{A: SpatEn}$$

Therefore, these two concepts can be merged.

The list of terms in the integrated ontology O_N is shown as Table 8. (only for the concepts and attributes that exist in both ontologies before integration.)

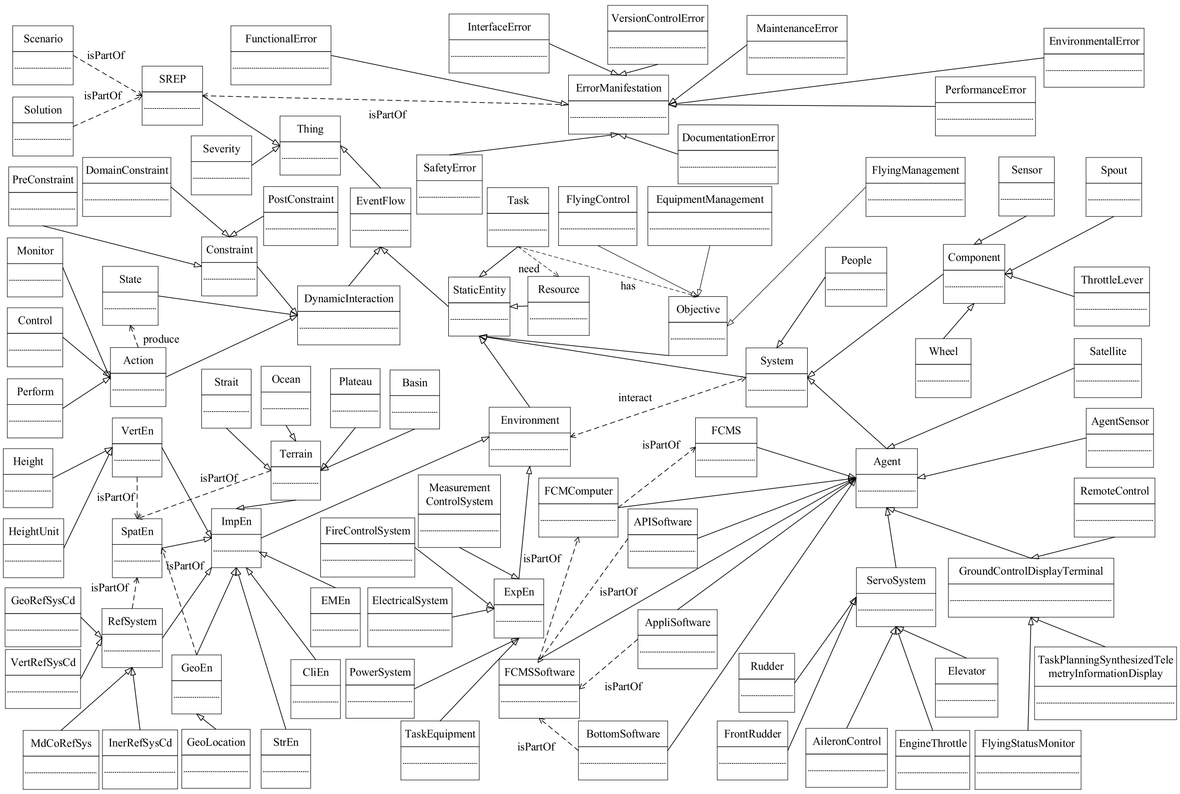


FIGURE 7. **Network diagram of the newly integrated ontology.**

Based on Figure 3, a network diagram of concept classes and relationships of the newly integrated ontology is shown in Figure 7.

The ontology integration process described above is semi-automated. In addition to the automatic reasoning using the inference engine, human participation is also required to delete some concepts or confirm the reservations of certain concepts manually. For example, the concept ''DataType-Info'' in the IMDO does not exist in the newly integrated ontology because from a semantic point of view, this concept is not needed in the new ontology.

B. ONTOLOGY EVALUATION

The development of ontology description languages and tools aids developers in building ontologies according to specific applications. However, due to the complexity of application semantics, ensuring ontology quality remains an important issue. In addition, the widespread use of ontologies has led to an explosive growth in the number of ontologies on the Internet. Ontologies enable reuse, but different ontologies have notable differences in domain coverage, comprehensibility and accuracy. Thus, it is difficult for users to grasp ontology features as a whole and understand their application. Based on the above two points, it is necessary to evaluate ontology quality. According to ontology evaluation results, developers can reconstruct an ontology to optimize its structure, thereby creating high-quality ontologies. Meanwhile, users can also select an optimal ontology between different ontology systems.

This criteria-based evaluation is our approach to measure internal and external semantic structural domains and concept structures in ontologies via our proposed criteria. It consists of ontology validation and ontology verification.

1) ONTOLOGY VALIDATION

- Ontology content evaluation

This method checks the content of the ontology based on the following main criteria [31]–[33], [47]: consistency, completeness, accuracy, conciseness, expandability, and sensitiveness. The criteria and their compatibility to UAV FCMS SREO considering geographic environment factors are shown in Table 9.

- Competency questions evaluation

The competency questions for determining the scope and designing purposes of UAV FCMS SREO considering geographic environment factors are used here for the evaluation. Answers and justifications are shown in Table 10. Competency questions ensure that the ontology implementation fulfills the scope of UAV FCMS SREO considering geographic environment factors.

2) ONTOLOGY VERIFICATION

- Ontology taxonomy evaluation

TABLE 9. Ontology content evaluation.

Criteria	Explanation	Satisfaction
Consistency	Enables reasoner to infer knowledge and interpret the ontology, whether it is according to specification with no contradictions or conflicting information.	Yes, since no contradictory knowledge can be inferred by reasoner. Also, reasoner shows no errors.
Completeness	Assures that all important information is included in the ontology.	Yes, it is complete based on specifications determined in the design phase of the ontology.
Accuracy	Determines if the expert knowledge about the domain approves of the contained knowledge in the ontology, i.e., distance between real world and conceptualization.	Yes, the activity of interviewing experts has been conducted. And experts participate in the ontology construction process.
Conciseness	To make sure no irrelevant information is included which makes the ontology unnecessarily large and increases computational resources.	Yes, because the ontology does not contain any unnecessary concepts, i.e., no redundancies.
Expandability	This refers to the effort required to add new definitions to an ontology and more knowledge to its definitions without altering the set of well-defined properties already guaranteed.	Yes, it is easily expanded since there is no need to make big changes in a set of well-defined definitions when adding new definitions.
Sensitiveness	This relates to how small changes in a definition alter the set of well-defined properties already guaranteed.	The ontology is not sensitive since small changes in definition will not alter a set of well-defined contents.

The taxonomy evaluation method is used for checking the taxonomy of the ontology based on main criteria mentioned in [55]. These criteria and their compatibility to UAV FCMS SREO considering geographic environment factors are shown in Table 11.

- The improved FOCA evaluation

a: THE FOCA METHOD AND ITS SHORTCOMINGS

FOCA is a method that can be used for evaluating the quality of an ontology. FOCA includes determining the type of ontology, a questionnaire to evaluate the components, a framework to follow, and a statistical model that calculates the quality of the ontology. FOCA goes through three verification steps, as shown in Figure 8 [17]. Ontology type verification defines two types of ontology: a domain or task ontology and an application ontology. Questions verification possesses questions to serve the goals. Quality verification calculates the ontology quality.

The FOCA evaluation criteria do not include a quantitative evaluation of ontology cohesion reflecting the close relationship between the ontology concepts. The structure of an ontology is consistent with object-oriented structure and should also meet the principle of ''high cohesion, low coupling''. The higher the ontology cohesion is, the closer relationship between the concepts. Therefore, the ontology cohesion can reflect the degree of ontology modularization to a certain extent. More importantly, because of the ontology integration technology used in this paper, the cohesion of the related concepts in the integrated ontology is also an important indicator reflecting the ontology quality. This paper calculates the cohesion of the parts related to both original ontologies of the newly integrated ontology and adds this indicator to the FOCA to evaluate the ontology quality.

b: ONTOLOGY MODULE AND DIRECTED ACYCLIC GRAPH

An ontology module is a collection of the closely related concepts, relationships, and axioms reflecting a common theme. The ontology module is divided or extracted from an original ontology and is part of the original ontology [56].

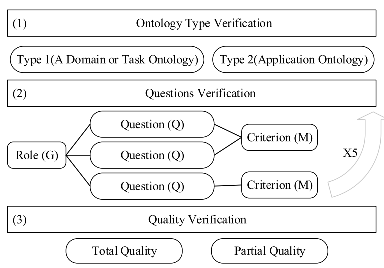


FIGURE 8. FOCA method.

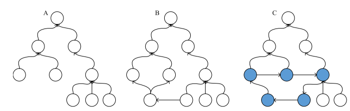


FIGURE 9. (a) Simple hierarchical tree, (b) DAG, (c) graph that contains a cycle, indicated in blue.

The modularization of an ontology helps reduce complexity and enhances comprehensibility, testability, maintainability, and reliability. The module has its own cohesion and can be used independently [57].

Ontology classes are arranged in a hierarchy from the general (high in the hierarchy) to the specific (low in the hierarchy). Despite the hierarchical organisation, most ontologies are not simple trees. Rather, they are structured as directed acyclic graphs (DAGs). This is because it is possible for classes to have multiple parents in the classification hierarchy, and furthermore ontologies include additional types of relationships between entities other than hierarchical classification (which itself is represented by is_a relations). All relations are directed and care must be taken by the ontology editors to ensure that the overall structure of the ontology does not contain cycles, as illustrated in Figure 9 [58].

c: THE EVALUATION METRICS OF COHESION

This paper uses the following evaluation metrics of cohesion [59]: the ontology module cohesion ''Coh'', the original ontology cohesion ''AOC'', the leaf node average

TABLE 10. Competency question answers and justifications.

CQ	Justification
1 Why build a UAV FCMS SREO considering geographic environment factors?	Engineering experience has shown that many UAV FCMS software errors are closely related to geographic environment factors. The reason is that the SRE process does not consider geographic environment factors and scenarios, and the generated SRSs may have problems in correctness, completeness, consistency, and unambiguity. An ontology is a formal and explicit specification of a shared conceptualization. By adopting this ontology method, the requirement knowledge is expressed as the ontology concept and association; therefore, it is clear, complete, and consistent. In addition, it greatly promotes the quality of UAV FCMS SRE and SRSs.
2 Who needs this ontology?	The stakeholders, including system analysts/software developers, customers/users and domain experts, all need this ontology. The purpose of an ontology is to use domain analysis and experience to model domain knowledge as a shareable knowledge framework to help stakeholders understand the application domain and define requirements. Under this framework, system analysts and software developers can understand the requirements accurately, and the heterogeneity brought about by multiple viewpoints and paradigms can be minimized. Moreover, the implementation of a higher quality UAV FCMS SRE activity based on this ontology benefits the development of UAV FCMS software and the entire system, helps to obtain products with fewer defects and higher quality, and enables developers to shorten the development cycle and reduce costs in a highly competitive market. Meanwhile, because the domain knowledge possessed by customers, users, and domain experts plays an important role in high-quality SRE, this ontology can help them accurately express requirements that developers can understand in accordance with software development guidelines and improve system development quality and efficiency.
3 Who is responsible for managing and maintaining this ontology?	This ontology exists as a part of UAV FCMS software and system development. Software and system developers are responsible for the management and maintenance of ontologies. In addition, there is a dedicated team in the development team responsible for the initial ontology development and subsequent ontology management and maintenance.
4 What are the main contents in this ontology?	Contents include the GO and DO level ontology-related concepts, attributes, hierarchies, and relationships. These are specifically based on 1) the UAV FCMS SREO concept classes and relationships: the UAV FCMS software-related concepts and relationships, the SREP-related concepts and relationships; 2) the concepts and relationships of the IMDO; 3) the concepts, attributes, hierarchies, and relationships of the integrated ontology.
5 When is this ontology needed?	This ontology is needed when developing the same or similar UAV FCMS software system, which can realize the sharing and reuse of the ontology.
6 How is this ontology managed and maintained?	A dedicated team is responsible for the management and maintenance of the ontology. According to actual usage and user feedback, the ontology is continuously updated, including adding new necessary content and deleting outdated content. In addition, logs are used to record the management and maintenance process.

TABLE 11. Ontology taxonomy evaluation.

Criteria	Types	Explanation	Satisfaction
Inconsistency	Circularity errors	A class is stated as a specialization of itself	No error, reasoner shows no errors.
	Partition errors	Wrongly defines concept classifications as disjoint/complete	No error, reasoner shows no errors.
	Semantic errors	Incorrect semantic classification	No error, reasoner shows no errors.
Incompleteness	Incomplete concept classification	Concepts are overlooked by classification	No error, all concepts of the knowledge specified in the design phase are included.
	Partition errors	A partition between a set of classes is omitted	No error, because all the instances of the base classes belong to the sub classes.
Redundancy	Grammatical redundancy	More than one explicit definition of any hierarchical definition	No error, each class has only one definition.
	Identical formal definition	Classes or instances with same formal definition	No error, there is no two classes with the same definition. No error, in GO or DO there are no instances.

inheritance depth “ADIT-LN”, and the comprehensive cohesion “TCOO”.

- ✓ Coh(M) indicates the connection tightness of the concepts in an ontology module. The calculation equation is,

$$Coh(M) = \begin{cases} 0 & n = 0 \\ \dfrac{\sum_{i=1}^{n} \sum_{j>i}^{n} R(c_i, c_j)}{n(n-1)/2} & n > 1 \\ 1 & n = 1 \end{cases} \tag{32}$$

where M denotes an ontology module; n denotes the number of nodes in the DAG of M; n(n-1)/2 denotes the number of edges of the full connected graph in the DAG of M; and $R(c_i, c_j)$ denotes the relationship between the concepts c_i and c_j. If there is a direct or indirect inheritance relationship between c_i and c_j, then $R(c_i, c_j) = 1$, otherwise $R(c_i, c_j) = 0$. If there is no concept in M, then Coh(M) = 0. If there is only one concept in M, Coh(M) = 1 because this concept does not depend on any other concept, and it is the closest structure. From equation (32), the range of Coh(M) is [0, 1] because the largest relation number in a DAG is the number of edges of the full connected graph.

- ✓ The calculation equation of AOC is,

$$AOC = \frac{\sum_{i=1}^{n} Coh(M_i)}{n} \tag{33}$$

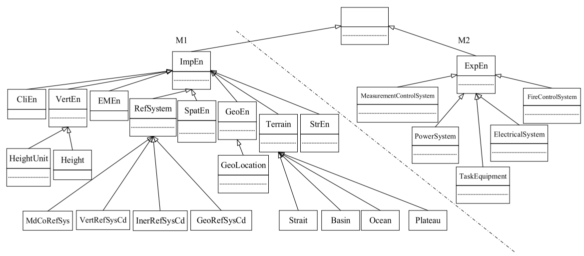


FIGURE 10. **Concept class hierarchy of "Environment" in the integrated ontology.**

where n is the number of modules partitioned by the original ontology, and Coh(M_i) is the cohesion of the ontology module M_i.

✓ ADIT-LN represents the depth of conceptual hierarchy in the ontology and depicts the degree of richness and refinement of the concepts. The calculation equation is,

$$ADIT-LN = \sum_{i=1}^{n} \frac{DOI_i}{TNOP} \tag{34}$$

where DOI_i represents the inheritance depth of a path i from its root node to a leaf node in a DAG, and the inheritance depth refers to the total number of edges of the path i from its root node to a leaf node in a DAG. n = TNOP, i.e., the total number of different paths from the root node to the leaf nodes in a DAG.

✓ TCOO is calculated as follows,

$$TCOO(O) = \alpha^{*}AOC(O) + \beta^{*}ADIT - LN(O) \tag{35}$$

where $\alpha + \beta = 1$, O represents the original ontology. The values of AOC and ADIT-LN are relatively high, indicating that the relationship between the original ontology concepts is closely connected; moreover, the concept level of the original ontology is relatively deep, and the concepts are rich.

d: THE CALCULATION OF ONTOLOGY COHESION

In this paper, the ontology cohesion is calculated for a portion of the integrated ontology related to both original ontologies (also an ontology module). It should be noted that because "Agent" and the related concepts account for a relatively small amount, the influence on cohesion is not significant. Therefore, this paper only calculates the cohesion of the "Environment" module. "Environment" is the apex of this ontology module, as shown in Figure 10.

According to the guidelines for prioritizing the protection of the hierarchical relationship, M1 and M2 are obtained by the modularization using the module partitioning tool SWOOP [60], i.e., the hierarchical relationship between concepts is not destroyed in modularization process.

Using equation (32) for M1 and M2, Coh(M1) = 0.16, Coh(M2) = 0.33. Substituting these two values into equation (33), AOC = 0.245. Using equation (34), ADIT-LN = 1.55.

Let $\alpha\alpha = 0.60$ and $\beta\beta = 0.40$; using equation (35), TCOO(O) = 0.767. This is the comprehensive cohesion of the ontology module with "Environment" as its apex in the integrated ontology. Referring to the results of the case study section in [59], the comprehensive cohesion of the main integrated part of the newly integrated ontology in this paper is slightly low. It should be noted that the AOC value is low, indicating that the relationship between the concepts in the ontology is not very close. Furthermore, the ADIT-LN value is not high, indicating that the concept hierarchy is not sufficiently deep. The main reason for the above results is that the hierarchical structure of the conceptual classes in the ontology is not sufficiently complete; meanwhile, there are limited connections between the concepts other than the hierarchical relationships.

e: THE IMPROVED FOCA EVALUATION METHOD

This paper uses the improved FOCA to evaluate the quality of the integrated ontology.

✓ Ontology type verification

FOCA defines two types of ontology, a domain or task ontology and an application ontology. The UAV FCMS SREO considering geographic environment factors is a DO (type 1); therefore, a type 1 ontology should answer Q5 instead of Q4 for Goal 2 shown in Table 12.

✓ Questions verification

When a cohesion metric is added, it needs to answer the 13 questions in Table 12 (should answer Q5 instead of Q4). These answers should then be scored by the evaluator. The set of questions corresponding to Goal 2 is expanded by adding a question of ontology cohesion metric, i.e., "Was the ontology cohesion metric value acquired?". The scores refer to the experimental data in [59]. The cohesion metric values and corresponding scores are shown in Table 13. These 13 questions serve five goals. The goal/question/metric (GQM) approach for the improved FOCA is shown in Table 12.

✓ Quality verification

Ontology quality can be calculated in two ways: total quality and partial quality. This paper uses the total quality verification because most goals are considered in the evaluation. Total quality verification is calculated using beta regression models, proposed by Ferrari [61], and shown in (36), as shown at the bottom of next page.

- Cov_S is the mean of the grades from Goal 1.
- Cov_C is the mean of the grades from Goal 2.
- Cov_R is the mean of the grades from Goal 3.
- Cov_{Cp} is the mean of the grades from Goal 4.
- LExp is the variable for evaluator experience, with 1 being very experienced and 0 being not experienced at all.
- Nl is 1 only if some Goal is impossible for the evaluator to answer all the questions.
- Sb = 1, Co = 1, Re = 1, Cp = 1, because the total quality considers all the roles.

TABLE 12. GQM for the improved FOCA.

Goal	Question		Metric
1.Check if the ontology complies with Substitute.	Q1. Were the competency questions defined?	Does the document define the ontology objective? Does the document define the ontology stakeholders? Does the document define the use of scenarios?	Completeness
	Q2. Were the competency questions answered?		Completeness
	Q3. Did the ontology reuse other ontologies?		Adaptability
2.Check if the ontology complies Ontological Commitments.	Q4. Did the ontology impose a minimal ontological commitment? (an application ontology)		Conciseness
	Q5. Did the ontology impose a maximum ontological commitment? (a domain or task ontology)		Conciseness
	Q6. Are the ontology properties coherent with the domain?		Consistency
	Q7. Was the ontology cohesion metric value acquired?		**Cohesion**
3.Check if the ontology complies with Intelligent Reasoning.	Q8. Are there contradictory axioms?		Consistency
	Q9. Are there redundant axioms?		Conciseness
4.Check if the ontology complies Efficient Computation.	Q10. Did the reasoner bring modeling errors?		Computational efficiency
	Q11. Did the reasoner perform quickly?		Computational efficiency
5.Check if the ontology complies with Human Expression.	Q12. Is the documentation consistent with modeling?	Are the written terms in the documentation the same as the modeling? Does the documentation explain what each term is and does it justify each detail of modeling?	Clarity
	Q13. Were the concepts well written?		Clarity
	Q14. Are there annotations in the ontology that show the definitions of the concepts?		Clarity

TABLE 13. Cohesion metric values and corresponding scores.

M	0≤M<0.5	0.5≤M<1	1≤M<1.5	1.5≤M
Scores	25	50	75	100

By Substituting these values into equation (36), the following result is obtained,

$$\hat{\mu}_i = \frac{\exp\{-0.44+0.03(100\times 1)+0.02(83.3\times 1)+ 0.01(100\times 1)+0.02(75\times 1)-0.06\times 1-25(0.1\times 1)\}}{1+\exp\{-0.44+0.03(100\times 1)+0.02(83.3\times 1)+ 0.01(100\times 1)+0.02(75\times 1)-0.06\times 1-25(0.1\times 1)\}} = 0.9725$$

The total quality of the ontology is 0.9725, which is near to 1. This shows the high quality of the UAV FCMS SREO considering geographic environment factors.

C. ENGINEERING APPLICATIONS

The ontology proposed in this paper has already been applied in engineering, i.e., the requirement elicitation of a certain type of UAV FCMS software has been carried out based on the ontology this paper proposes. To illustrate the effectiveness of this method, further verification is necessary using a comparative experiment (a software requirement inspection). The research selects two continuous versions of the UAV FCMS software and adopts a conventional method for the requirement elicitation of a version 3.3.x; after a defined period, the requirements of a version 3.3.(x + 1) are elicited based on the ontology this paper proposes. Table 14 records the detected SREP error-manifestations and number distributions of these two SRSs by a requirement inspection. The same group of inspectors is used to conduct the comparative experiment. SRS I is developed based on the conventional method, and SRS II is developed based on the ontology proposed in this paper.

TABLE 14. SREP Error-manifestations and number distributions.

Error-manifestations	SRS I				SRS II			
	C	I	A	T_t	C	I	A	T_t
Functional errors	4	1	1	6	0	1	1	2
Performance errors	0	0	1	1	0	0	1	1
Interface errors	3	6	0	9	1	2	0	3
Safety errors	4	1	0	5	1	0	0	1
Environmental errors	2	2	0	4	0	0	0	0
Maintenance errors	0	1	1	2	0	0	2	2
Documentation errors	0	3	4	7	0	2	4	6
T_s	13	14	7	-	2	5	8	-
T	34				15			

Notes: C—"critical", I—"important", A—"average". T_t—"total number of errors in each type", T_s—"total number of errors in each severity", T—"total number of errors".

The results that the total number of errors in SRS I is much higher than in SRS II. In addition, the severity of the errors detected in SRS I is higher, and they occur in more significant error types such as functional errors, interface errors, safety

$$\hat{\mu}_i = \frac{\exp\{-0.44 + 0.03(Cov_S \times Sb)_i + 0.02(Cov_C \times Co)_i + 0.01(Cov_R \times Re)_i + 0.02(Cov_{Cp} \times Cp)_i - 0.06LExp_i - 25(0.1 \times Nl)_i\}}{1 + \exp\{-0.44 + 0.03(Cov_S \times Sb)_i + 0.02(Cov_C \times Co)_i + 0.01(Cov_R \times Re)_i + 0.02(Cov_{Cp} \times Cp)_i - 0.06LExp_i - 25(0.1 \times Nl)_i\}} \quad (36)$$

TABLE 15. Quality metrics results of SRS I and SRS II.

Metrics	SRS I	SRS II
Correctness	31/40=77.5%	38/41=92.7%

errors, and environmental errors. The direct cause of the above results can be initially identified as SRS II using the ontology-based method; SRS I uses the conventional method.

Intuitively, because the ontology is a complete set of domain knowledge, the quality information of SRS can be obtained indirectly by considering the correspondence between the SRS and ontology element. This paper evaluated the quality of SRS based on the metrics in [62]. The quality metrics results of the SRS I and SRS II are shown in Table 15.

From the results, the difference between SRS I and SRS II in "Correctness" is more obvious. An ontology is a semantic basis for building a specific problem domain. Ideally, all requirements items should be able to find the corresponding elements in the ontology. **(the number of items that can be mapped to the ontology / the total number of requirements items)** can reflect the proportion of the mapped elements. The higher the ratio is, the higher the SRS quality. This ratio of SRS I to SRS II is significantly lower. This shows that some of the requirements items of SRS I are not included in the ontology library, implying nonconformity with the actual application. This fact also explains the results of the requirement inspection in Table 14. Therefore, the requirement knowledge ontology has a major impact on the entire requirement development process. In general, the quality of the SRS obtained based on the proposed ontology is higher than the quality of the SRS obtained based on the conventional method. Therefore, the ontology elements, i.e., the knowledge elements, should be fully integrated in the early stage of the requirement development process.

VI. CONCLUSION

This paper focused on the problems of ambiguity, heterogeneity, and incompleteness in a UAV FCMS SRE, especially geographic environment-related factors, to use an ontology to solve the above problems. By constructing a UAV FCMS SREO and an IMDO and integrating these two ontologies, a UAV FCMS SREO considering geographic environment factors was obtained. A hybrid ontology mapping method was adopted to analyze the ontology concepts. Based on a traditional similarity calculation, a DL strategy was used to detect the matching relationships between different domain ontologies through deductive reasoning, realize the mapping between two ontologies, and finally complete the ontology integration. This method avoided the shortcomings of the similarity calculation method, which was limited to measuring the equivalence relation between the entities, merely emphasizing the grammar implementation, and lacking an accurate description of the semantics. The ontology evaluation results showed the higher quality of the integrated ontology. Moreover, the engineering applications showed that the SRE activities and the generated SRS based on the proposed ontology enabled a notable increase in quality.

However, the results in this paper are still insufficient, and the new cohesion index in the improved FOCA is not very satisfactory. This suggests that the relationship between the concepts in the UAV FCMS SREO considering geographic environment factors is not very close; furthermore, the concept hierarchy is not sufficiently deep. The main reason for the above results is that the concept hierarchy in the ontology is not sufficiently complete, and the relationships between the concepts are limited except for the hierarchical relationships. Therefore, it is necessary to further improve the ontology, enrich and refine the ontology concepts, and fully explore the implicit relationships between the concepts.

REFERENCES

[1] D. Cavaliere, V. Loia, and S. Senatore, "Towards an ontology design pattern for UAV video content analysis," *IEEE Access*, vol. 7, pp. 105342–105353, Aug. 2019.

[2] X. Hu, B. Liu, Y. Wang, and C. Jiang, "Application of software requirement error pattern based on ontology," *J. Beijing Univ. Aeronaut. Astronaut.*, vol. 35, no. 6, pp. 723–727, Jun. 2009.

[3] X. Hu, "Study on ontology-based avionics system requirements knowledge model construction and software requirements elicitation method," Ph.D. dissertation, Dept. Rel. Syst. Eng., Beihang Univ., Beijing, China, 2009.

[4] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?" *Int. J. Hum.-Comput. Stud.*, vol. 43, nos. 5–6, pp. 907–928, Nov. 1995.

[5] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquisition*, vol. 5, no. 2, pp. 199–220, Jun. 1993.

[6] J. Lu, G. Wang, and M. Törngren, "Design ontology in a case study for cosimulation in a model-based systems engineering tool-chain," *IEEE Syst. J.*, vol. 14, no. 1, pp. 1297–1308, Jun. 2019.

[7] A. A. Alsanad, A. Chikh, and A. Mirza, "A domain ontology for software requirements change management in global software development environment," *IEEE Access*, vol. 7, pp. 49352–49361, Apr. 2019.

[8] W. Sentang and F. Yuhua, *Flight Control System*. Beijing, China: Beihang Univ. Press, 2005.

[9] W. Honglun and W. Yingxun, "Flight control and management system of unmanned aerial vehicles," *Acta Aeronautica Et Astronautica Sinica*, vol. 29, pp. s1–s7, 2008.

[10] J. Yinqing, "Flight control system software design for small-scale UAVs based on C/OS-II," M.S. thesis, Dept. Automat. Eng., Nanjing, Univ. Aeronaut. Astronaut., Nanjing, China, 2009.

[11] *Terms and Definitions of Aircraft Flight Control System*, Standard HB6486-2008, Aviation Industry Standard of P.R.China, 2008.

[12] *Geographic Information-Metadata*, Standard ISO 19115-1, 2014.

[13] *Geographic Information-Metadata*, Standard (ISO 19115: 2003, MOD), GB/T 19710-2005, National Standards of P.R. China, 2005.

[14] F. Silva and R. Girardi, "An approach to join ontologies and their reuse in the construction of application ontologies," in *Proc. IEEE/WIC/ACM Int. Joint Conferences Web Intell. (WI) Intell. Agent Technol. (IAT)*, Warsaw, Poland, Aug. 2014, pp. 424–431.

[15] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and F. P. Patel-Schneider, Eds., *The Description Logic Handbook*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[16] L. Junli, H. Zongyi, K. Dongliang, and Z. Qiaoli, "An approach for insight into geo-ontology merging based on description logics," *Geomatics Inf. Sci. Wuhan Univ.*, vol. 39, no. 3, pp. 317–321, 2014.

[17] J. Bandeira, I. I. Bittencourt, P. Espinheira, and S. Isotani, "FOCA: A methodology for ontology evaluation," 2016, *arXiv:1612.03353*. [Online]. Available: http://arxiv.org/abs/1612.03353

[18] J. F. Sowa, *Building, Sharing, and Merging Ontologies*. [Online]. Available: http://www.jfsowa.com/ontology/ontoshar.htm

[19] H. Yanbo, Z. Hongyu, L. Jianhua, T. Liqiu, and L. Zhi, "Survey of ontology mapping approaches," *Comput. Eng. Appl.*, vol. 18, pp. 27–33, Sep. 2005.

[20] *Diogene's Ontology Mapping Prototype*. [Online]. Available: http://diogene.cis.strath.ac.uk/prototype.html

[21] A. Maedche and S. Staab, "Measuring similarity between ontologies," in *Proc. EKAW*, Sigüenza, Spain, 2002, pp. 251–263.
[22] S. Sekine, K. Sudo, and T. Ogino, "Statistical matching of two ontologies," in *Proc. Standardizing Lexical Resour. (SIGLEX)*, MD, USA, 1999, pp. 69–73.
[23] A. Doan, J. Madhavan, R. Dhamankar, P. Domingos, and A. Halevy, "Learning to match ontologies on the semantic Web," *VLDB J.*, vol. 12, no. 4, pp. 303–319, Nov. 2003.
[24] M. A. Rodríguez and M. J. Egenhofer, "Determining semantic similarity among entity classes from different ontologies," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 2, pp. 442–456, Mar. 2003.
[25] M. Ehrig and Y. Sure, "Ontology mapping-an integrated approach," in *Proc. ESWS*, Heraklion, Crete, Greece, 2004, pp. 76–91.
[26] S. Prasad, Y. Peng, and T. Finin, "A tool for mapping between two ontologies using explicit information," in *Proc. AAMAS Workshop Ontologies Agent Syst.*, Bologna, Italy, 2002.
[27] G. Shen, Z. Huang, X. Zhu, L. Wang, and G. Xiang, "Using description logics reasoner for ontology matching," in *Proc. IITA*, Zhang Jiajie, China, Dec. 2007, pp. 30–33.
[28] M. Kolli and Z. Boufaida, "A description logics formalization for the ontology matching," *Procedia Comput. Sci.*, vol. 3, pp. 29–35, 2011.
[29] S. K. Kumar and J. A. Harding, "Ontology mapping using description logic and bridging axioms," *Comput. Ind.*, vol. 64, no. 1, pp. 19–28, Jan. 2013.
[30] A. Gómez-Pérez, M. Fernandez-Lopez, and O. Corcho, *Ontological Engineering: With Examples from the Areas of Knowledge Management, E-Commerce and the Semantic Web*. London, U.K.: Springer, 2004.
[31] S. Jain and V. Meyer, "Evaluation and refinement of emergency situation ontology," *Int. J. Inf. Edu. Technol.*, vol. 8, no. 10, pp. 713–719, 2018.
[32] D. Vrandecic, *Ontology Evaluation, Handbook on Ontologies*. Berlin, Germany: Springer, 2009, pp. 293–313.
[33] J. Pak and L. Zhou, "A framework for ontology evaluation," in *Proc. Workshop E-Bus.* Berlin, Germany: Springer, 2009, pp. 10–18.
[34] R. Lourdusamy and A. John, "A review on metrics for ontology evaluation," in *Proc. ICISC*, Coimbatore, India, Jan. 2018, pp. 1415–1421.
[35] M. Poveda-Villalón, M. C. Suárez-Figueroa, and A. Gómez-Pérez, "Validating ontologies with OOPS!" in *Proc. 18th Int. Conf. (EKAW)*, Galway City, Ireland, 2012, pp. 267–281.
[36] N. Casellas, "Ontology evaluation through usability measures," in *Proc. OTM*, Vilamoura, Portugal, 2009, pp. 594–603.
[37] A. Gangemi, C. Catenacci, M. Ciaramita, and J. Lehmann, "Modelling ontology evaluation and validation," in *Proc. ESWC*, Budva, Montenegro, 2006, pp. 140–154.
[38] J. Hartmann, P. Spyns, A. Giboin, D. Maynard, R. Cuel, M. C. Suárez-Figueroa, and Y. Sure, *D1.2.3 Methods for Ontology Evaluation*, document EU-IST Network of Excellence IST-2004-507482 KWEB Deliverable D, 2005, vol. 1.
[39] S. Tartir, I. B. Arpinar, and A. P. Sheth, "Ontological evaluation and validation," in *Theory and Applications of Ontology: Computer Applications*. Dordrecht, The Netherlands: Springer, 2010, pp. 115–130.
[40] C. Brewster, H. Alani, S. Dasmahapatra, and Y. Wilks, "Data-driven ontology evaluation," in *Proc. 4th Int. Conf. Lang. Resour. Eval.*, Lisbon, Portugal, 2004.
[41] H. Hlomani and A. D. Stacey, "Contributing evidence to data-driven ontology evaluation: Workflow ontologies perspective," in *Proc. 5th Int. Conf. Knowl. Eng. Ontology Develop.*, Vilamoura, Portugal, 2013.
[42] L. Ouyang, B. Zou, M. Qu, and C. Zhang, "A method of ontology evaluation based on coverage, cohesion and coupling," in *Proc. 8th Int. Conf. (FSKD)*, Shanghai, China, Jul. 2011, pp. 2451–2455.
[43] G. Bilgin, I. Dikmen, and M. T. Birgonul, "Ontology evaluation: An example of delay analysis," *Procedia Eng.*, vol. 85, pp. 61–68, Jan. 2014.
[44] A. Zouaq and R. Nkambou, "Evaluating the generation of domain ontologies in the knowledge puzzle project," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 11, pp. 1559–1572, Nov. 2009.
[45] H. Hlomani and D. Stacey. *Approaches, Methods, Metrics, Measures, and Subjectivity in Ontology Evaluation: A Survey*. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.682.1950&rep=rep1&type=pdf
[46] M. Amirhosseini and J. Salim, "OntoAbsolute as a ontology evaluation methodology in analysis of the structural domains in upper, middle and lower level ontologies," in *Proc. Int. Conf. Semantic Technol. Inf. Retr.*, Putrajaya, Malaysia, Jun. 2011, pp. 26–33.
[47] A. Gómez-Pérez, "Ontology evaluation," in *Handbook on Ontologies (International Handbooks on Information Systems)*, S. Staab and R. Studer, Eds. Berlin, Germany: Springer, 2004, pp. 251–273.
[48] J. Arpirez, A. G. L. A. Perez, and S. Pinto, "An ontology-based WWW broker to select ontologies," in *Proc. Workshop Appl. Ontologies Problem-Solving Methods*, U.K., 1998, pp. 16–24.
[49] M. Gruninger and M. S. Fox, "Methodology for the design and evaluation of ontologies," in *Proc. Workshop Basic Ontol. Issues Knowl. Sharing (IJCAI)*, Montreal, QC, Canada, 1995.
[50] M. V. Kureychik and B. I. Safronenkova, "Ontology-based approach to design problem formalization," in *Proc. Int. Seminar Electron Devices Design Prod. (SED)*, Prague, Czech Republic, 2019, pp. 1–5.
[51] L. Lin and W. Hong, "Classification of fundamental geographic information based on formal ontology," *Geomatics Inf. Sci. Wuhan Univ.*, vol. 31, no. 6, pp. 523–526, 2006.
[52] A. Hamdulla, H. Yilahun, K. Abdurahman, and S. Imam, "A hierarchical clustering based relation extraction method for domain ontology," in *Proc. 9th Int. Symp. Parallel Archit., Algorithms Program. (PAAP)*, Taipei, Taiwan, Dec. 2018, pp. 36–40.
[53] Y. Liu, H. T. Loh, and A. Sun, "Imbalanced text classification: A term weighting approach," *Expert Syst. Appl.*, vol. 36, no. 1, pp. 690–701, Jan. 2009.
[54] W. C. Mendes and R. Girardi, "An ontology-based architecture of an RBC agent," in *Proc. 8th Iberian Conf. Inf. Syst. Technol.*, Braga, Portugal, 2013, pp. 776–781.
[55] S. Lovrenčić and M. Čubrilo, "Ontology evaluation—Comprising verification and validation," in *Proc. 19th Central Eur. Conf. Inf. Intell. Syst. (CECIIS)*, Varaždin, Hrvatska, 2008, pp. 657–663.
[56] S. Oh, H. Y. Yeom, and J. Ahn, "Cohesion and coupling metrics for ontology modules," *Inf. Technol. Manage.*, vol. 12, no. 2, pp. 81–96, Jun. 2011.
[57] M. Page-Jones, *Practical Guide to Structured Systems Design*. New York, NY, USA: Yourdon Press, 1980.
[58] C. Dessimoz and N. Škunca, *The Gene Ontology Handbook*. Totowa, NJ, USA: Humana Press, 2017.
[59] L. Lili, S. Guohua, H. Zhiqiu, and K. Shuanglong, "An approach for ontology cohesion metrics based on directed acyclic graph," *Comput. Eng. Sci.*, vol. 37, no. 7, pp. 1297–1303, 2015.
[60] A. Kalyanpur, B. Parsia, E. Sirin, B. C. Grau, and J. Hendler, "Swoop: A Web ontology editing browser," *J. Web Semantics*, vol. 4, no. 2, pp. 144–153, Jun. 2006.
[61] S. Ferrari and F. Cribari-Neto, "Beta regression for modelling rates and proportions," *J. Appl. Statist.*, vol. 31, no. 7, pp. 799–815, Aug. 2004.
[62] H. Kaiya and M. Saeki, "Using domain ontology as domain knowledge for requirements elicitation," in *Proc. 14th IEEE Int. Requirements Eng. Conf. (RE)*, St. Paul, MN, USA, Sep. 2006, pp. 186–195.

XUAN HU received the Ph.D. degree in aerospace science and technology from Beihang University, China, in 2010. She is currently a Senior Engineer with the CEPREI Laboratory. Her main research interests include ontology modeling, software requirement engineering, and software reliability.

JIE LIU received the B.S. degree in computer science from Xidian University, China. He is currently the Director of the Information Security Research Center, CEPREI Laboratory. His main research interests include information security, software reliability and security, and software evaluation technology.