IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 18, 2016, accepted October 13, 2016, date of publication November 1, 2016, date of current version November 18, 2016.

Digital Object Identifier 10.1109/ACCESS.2016.2621756

A New Fuzzy Ontology Development Methodology (FODM) Proposal

XIN LI, JOSÉ-FERNÁN MARTÍNEZ, AND GREGORIO RUBIO
Research Center on Software Technologies and Multimedia Systems for Sustainability, Technical University of Madrid, 28031 Madrid, Spain

Corresponding author: X. Li (xin.li@upm.es)

The research leading to the presented results has been undertaken within the SWARMs European project (Smart and Networking Underwater Robots in Cooperation Meshes), under Grant Agreement no. 662107-SWARMs-ECSEL-2014-1, partially supported by the ECSEL JU and the Spanish Ministry of Economy and Competitiveness (Ref: PCIN-2014-022-C02-02).

ABSTRACT There is an upsurge in applying fuzzy ontologies to represent vague information in the knowledge representation field. Current research in the fuzzy ontologies paradigm mainly focuses on developing formalism languages to represent fuzzy ontologies, designing fuzzy ontology editors, and building fuzzy ontology applications in different domains. Less focus falls on establishing a formal methodological approach for building fuzzy ontologies. Existing fuzzy ontology development methodologies, such as the IKARUS-Onto methodology and fuzzy ontomethodology, provide formalized schedules for the conversion from crisp ontologies into fuzzy ones. However, a formal guidance on how to build fuzzy ontologies from scratch still lacks in this paper. Therefore, this paper presents the first methodology, named fuzzy ontology development methodology (FODM), for developing fuzzy ontologies from scratch. The proposed FODM can provide a very good guideline for formally constructing fuzzy ontologies in terms of completeness, comprehensiveness, generality, efficiency, and accuracy. To explain how the FODM works and demonstrate its usefulness, a fuzzy seabed characterization ontology is built based on the FODM and described step by step.

INDEX TERMS Fuzzy ontologies, methodology, generality, vagueness, knowledge representation.

I. INTRODUCTION

Ontology provides a formal and explicit specification of a shared conceptualization [1] and it has become the most promising modelling technique to represent information. Ontology typically consists of concepts (general abstraction for a class of individuals), properties (specification of relationships between concepts or their attributes), instances, and axioms. Different formalism languages, including RDF (Resource Description Framework),[1] RDFS (Resource Description Framework Schema),[2] and OWL (Web Ontology Language),[3] can be used to formalize ontology in a machine-readable format. Due to its major advantages, such as formality, machine-readability and shareability, ontology has attracted growing interests from academia to represent knowledge in real world applications. Despite the undeniable success of ontology, classical ontology, also referred to crisp ontology, lacks the ability to deal with information which has an imprecise or vague meaning [2], [3]. For instance, representation of a piece of information with a quantitative degree, "Jack is tall with at least degree 0.5", cannot be accommodated by crisp ontology.

Due to the importance of dealing with vagueness in the knowledge representation field, a standard way to formally quantify and represent vagueness is required. Since fuzzy set theory and fuzzy logic [4] seem appropriate to manage the vagueness which is inherent to real world information, fuzzy ontology, which introduces those two techniques into crisp ontology, emerged in the early 2000's [5]. By means of encasing fuzzy sets, fuzzy ontology can associate the modelled information which has a vague meaning with a world belief or truth degree.

Essentially, elements which form fuzzy ontologies are similar to those in crisp ontologies from the definition point of view. However, fuzzy ontology elements show more advance than crisp ontology elements in terms of representing vagueness which is inherent to real world information. An exhaustive list of fuzzy ontology elements could be referred to [3]. Elements, which are usually included in fuzzy ontologies, are shown in the following:

- Fuzzy concepts. They refer to concepts which do not have clear-cut boundaries and represent fuzzy sets of

[1] RDF: https://www.w3.org/RDF/
[2] RDFS: https://www.w3.org/2001/sw/wiki/RDFS
[3] OWL: https://www.w3.org/2001/sw/wiki/OWL

individuals. Thus, an individual could be attributed to a fuzzy concept with a certain degree. For instance, Jack aged 45 could be classified as an instance of a fuzzy concept YoungPerson with a degree of 0.4. So instead of being impossible, Jack is regarded as a young person to some extent.

- Fuzzy roles. Fuzzy roles describe fuzzy binary relations between concepts or individuals. They can link different concept instances associated with certain degrees. For instance, a fuzzy relationship "likes" can be used to represent a vague statement "John likes apples to degree 0.8".
- Fuzzy data types. Fuzzy forms of data which contain vague meanings are specified by fuzzy data types. Fuzzy data types are used to fuzzify attributes values, such as the range of data properties.

It is worth noting that fuzzy ontologies have been applied in many applications, including Information Retrieval [6]–[8], Semantic Web [9], [10], Underwater Robotics [11]–[13], and Ambient Assisted Living [14], [15] etc. To explore the applicability of fuzzy ontologies to more domains is becoming more and more active. However, the topic on methodologies for guiding the overall fuzzy ontology development process draws less focus in current research. Alike developing crisp ontologies, the construction of fuzzy ontologies also needs to be completed following a well-defined guideline. The guideline, which essentially refers to a development methodology, should address common questions had by ontology engineers during the development process. Possible questions could be seen as follows:

1. Is the development of fuzzy ontologies the same as the crisp ontologies construction?
2. How to start in order to develop fuzzy ontologies?
3. How to design fuzzy ontologies step by step?
4. What activities should be done in each step?
5. In which way the development of fuzzy ontologies can be completed faster and more efficiently?
6. What issues need to be considered during the development process in order to ensure a good quality of fuzzy ontologies?
7. Once completing the design of fuzzy ontologies, is it the end of the entire development process without further considerations, such as documentation or maintenance?

Existing attempts to present development methodologies for building fuzzy ontologies are the IKARUS-Onto (Imprecise Knowledge Acquisition Representation and Use) [16] methodology and the Fuzzy Ontomethodology [17]. The IKARUS-Onto methodology provides a very comprehensive methodology for developing fuzzy ontologies from existing crisp ones. With the formal guideline provided by the IKARUS-Onto, effectiveness of the development for fuzzy ontologies in domains with the existence of crisp ontologies can be enhanced. Similarly, the Fuzzy Ontomethodology also presents a guideline for the engineering principles of converting crisp ontologies into fuzzy ones. However, a formal guidance on how to build fuzzy ontologies from scratch is still a lack in current literature. Therefore, to fill the gap, a novel fuzzy ontology development methodology (FODM) is presented in this paper with aim to provide the first methodological approach to develop fuzzy ontologies from scratch. The FODM, created by taking existing resources, such as crisp ontology development methodologies and existing fuzzy ontology development methodologies as references, presents a concrete workflow for engineering principles of fuzzy ontology constructions. The entire development process is divided into eleven phases and concrete activities are grouped in each phase. The FODM can also act as a methodology for building crisp ontologies if the target domain or application does not contain any vague or imprecise information. The FODM could provide a schedule of activities or tasks that need to be performed during the fuzzy ontology development process in terms of completeness, comprehensiveness, generality, and ease of use. It is worth noting that the purpose of this paper is not to provide a rigorous scientific evaluation of FODM compared with any other methodology or no methodology. In principle, the proposed FODM is an abstraction of activities for building fuzzy ontologies from scratch. Thus, it is a subjective methodology in nature. As de Hoog [18] says, "it is extremely difficult to judge the value of a methodology in an objective way. Experimentation is of course the proper way to do it, but it is hardly feasible because there are too many conditions that cannot be controlled." In fact, the difficulty and absence of making rigorous evaluation exist in every existing ontology methodology [16]. The value of the proposed FODM is the first guidance on how to develop fuzzy ontologies from scratch in a formal way. It could expect an enhancement in the FODM-based development process compared with an intuitive development. To demonstrate the usefulness and applicability of the proposed FODM, a simple, but realistic fuzzy ontology aiming to represent the characterization of seabed is developed based on the FODM approach and described step-by-step.

The remainder of this paper is organized as follows: related works on methodologies for building ontologies are presented in section II. Specifically, section II reviews existing methodologies for developing crisp ontologies and existing methodology for building fuzzy ontologies. Section III shows the proposed FODM with detailed specifications for each phase. A fuzzy ontology aiming to model the characterization of seabed is constructed based on the FODM and presented in section IV. Afterwards, discussion on the proposed methodology is shown in section V. Finally, in section VI, conclusions are given and future work is also pointed out.

II. RELATED WORK

In this section, the state of the art in ontology development methodologies is presented. Specifically, a summary of the most well-known methodologies for building crisp ontologies is provided. In addition, existing fuzzy ontology development methodologies presented in current research are reviewed.

A. METHODOLOGIES FOR DEVELOPING CRISP ONTOLOGIES

It is widely accepted that there is no single "correct" way or methodology for developing ontologies [19]. Aiming to provide good guidelines for crisp ontology constructions, various ontology development methodologies have been presented. An ontology development methodology provides a formalization for scheduling activities or tasks that should be followed and performed during the design process. Workflows proposed by different methodologies might fare better or worse regarding efficiency, ease of use, comprehensiveness and rationality. A well-organized schedule of activities proposed by ontology development methodologies can provide methodological supports for ontology engineers. The most well-known ontology methodologies proposed in current literature are METHONTOLOGY [20], NeOn [21], DILIGENT [22], On-To-Knowledge [23], HCOME [24], and DOGMA [25]. In addition, Noy and McGuiness [19] presented a very descriptive yet simple guide to create crisp ontologies. A set of survey papers, such as [26]–[28], are also available providing good references to existing ontology development methodologies and their features. To conclude, a considerable amount of methodologies can come in handy for developing crisp ontologies. However, these methodologies dedicated to crisp ontologies cannot be directly applied to construct fuzzy ontologies due to major differences between fuzzy ontologies and crisp ones. In order to develop fuzzy ontologies, additional procedures, such as including fuzzy logic to approximate vagueness and conceptualizing the fuzzified vagueness, should be considered in the development process.

B. METHODOLOGIES FOR DEVELOPING FUZZY ONTOLOGIES

Current research on fuzzy ontologies mainly focuses on dealing with conceptual formalisms. In other words, how to represent fuzzy ontologies in a formalized language is the most active work. How to develop fuzzy ontologies in a standard and effective way is under-researched. The IKARUS-Onto methodology [16] is a methodology for fuzzy ontology development. It focuses on the provision of a methodological guideline for the conversion from crisp ontologies into fuzzy ones. It consists of five formal steps, including acquiring crisp ontology, establishing need for fuzziness, defining fuzzy ontology elements, formalizing fuzzy elements, and validating fuzzy ontology. The IKARUS-Onto methodology represents a comprehensive guidance for fuzzifying crisp ontologies. Thus, it is suitable to be used to develop fuzzy ontologies in domains with the existence of crisp ontologies. Similarly, the Fuzzy Ontomethodology [17] also emphasizes on formalizing the activities for developing fuzzy extensions based on available crisp ontologies. The Fuzzy Ontomethodology consists of three steps, including conceptualization, ontologisation, and operationalization. Processes grouped in each step are too ambiguous to be understood and used in practice. In addition, the Fuzzy Ontomethodology is devoted to providing guidelines for building ontologies for semantic web search. Reusing fuzzy elements (e.g., fuzzy concepts, fuzzy sets, fuzzy relationships, or fuzzy data types) that have been defined in existing fuzzy ontologies can enhance the interoperability and shareability in the ontology community as well as guaranteeing less workload. Nevertheless, neither of existing fuzzy ontology methodologies does consider the inclusion of an important step, which is reusing existing fuzzy ontology elements, in the development process. While attempting to model knowledge in domains where no existing crisp ontologies are available, the development of fuzzy ontologies should be guided in a formal way. Since existing fuzzy ontology methodologies rely on the existence of crisp ontologies, it is apparent that a methodological approach for developing fuzzy ontologies from scratch is still a lack in current literature.

Ontologies should be built following a methodological guideline in order to better model imprecise and vague information. To this end, this paper presents a fuzzy ontology development methodology which could provide well-defined engineering principles to improve the development and building of fuzzy ontologies from scratch. This proposed method could enable good treatments and utilizations of vague or imprecise knowledge in terms of generality, accuracy, reusability, efficiency, and shareability.

III. THE PROPOSED FUZZY ONTOLOGY DEVELOPMENT METHODOLOGY (FODM)

In this section, a formal fuzzy ontology development paradigm is presented based on existing ontology development methods. Its emphasis lies on introducing new changes brought by fuzzy ontologies into the development process. The proposed FODM assumes prior knowledge of principles of crisp and fuzzy ontology from potential readers. It does not aim to completely reform current crisp ontology development methods. Instead, it is built on the basis of existing crisp ontology development methods with additional fuzzy related considerations.

A. INPUTS FOR THE CREATION OF THE FODM

As shown in Fig. 1, the proposed FODM is grounded on the basis of three major resources, including existing methodologies for building ontologies, practical experiences on constructing fuzzy ontologies and lessons learned from fuzzy

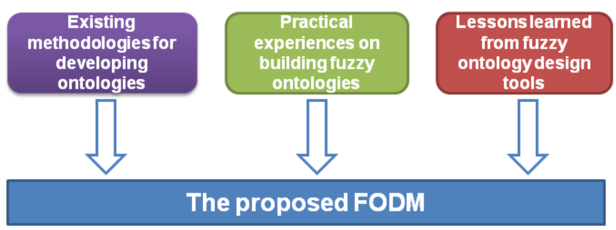


FIGURE 1. Inputs inspiring to conceive the FODM.

ontology design tools. All these knowledge resources are inspiring to create the new FODM.

- **Existing methodologies for developing ontologies**. In nature, the development of fuzzy ontologies would not completely reform the crisp ontology development process. Instead, the general flow to construct fuzzy ontologies should be compliant with conventional crisp ontology development methodologies. Nevertheless, new changes will be introduced into conventional methodologies with additional fuzzy considerations. Thus, conventional crisp ontology development methodologies are selected as the starting point to create the new FODM. It is worth noting that as stressed in section II each crisp ontology development methodology fares better or worse in terms of some specific evaluation considerations, such as consideration for reusing existing ontologies. Hence, several methodologies, including Methontology and NeON etc., are comprehensively studied so that strengths of each method can be correctly collected and applied in the new FODM. In addition, the IKARUS-Onto methodology and the Fuzzy Ontomethodology are also taken as valuable references to the proposed FODM.
- **Practical experiences on building fuzzy ontologies**. Experiences of ontology engineers in the *Grupo de Redes y Servicios de Próxima Generación (GRyS)*[4] obtained from designing a lot of ontologies, including crisp and fuzzy ontologies, are beneficial to the creation of the new FODM. Though different ontology engineers have different preferences to design fuzzy ontologies, an initial group of informal steps could be abstracted from their practical experiences. These informal steps could provide a preliminary foundation which could afterwards be formalized as formal methodological activities or processes.
- **Lessons learned from fuzzy ontology design tools**. Various fuzzy ontology software tools, here particularly referring to fuzzy ontology editors, have been created and been off-the-shelf. The Fuzzy Ontology Generation Framework (FOGA) [9] provides support in automatically generating fuzzy ontologies. The Fuzzy OWL 2 plug-in [3] enables ontology engineers to define fuzzy related knowledge by means of OWL 2 annotations in a very visualized and easy way. By practicing with fuzzy ontology design tools, especially referred to Fuzzy OWL 2, lessons can be learned, such as the way a conceptual model is implemented by editors. The practice with fuzzy ontology tools can imply an informal workflow, which is the default process specified in those tools, to develop fuzzy ontologies.

The FODM obtains inspirations from three aforementioned resources. After a thorough study on the state of the art in those research fields, valuable knowledge are extracted and applied into the creation of the new FODM with additional fuzzy introduced modifications. The proposed FODM will be elaborated in the following section.

B. SPECIFICATIONS FOR THE PROPOSED FODM

The aim of the proposed FODM is to provide a formal abstraction of activities that need to be done throughout the development process. The proposed methodology is dedicated to presenting the first methodological approach to build fuzzy ontologies from scratch, rather than converting existing crisp ontologies into fuzzy ones. The whole workflow of the proposed FODM can be viewed in Fig. 2. In general, all the activities or tasks are grouped into eleven phases to form the entire lifecycle of building a fuzzy ontology. Each phase and its associated purposes and activities are elaborated in the following subsections.

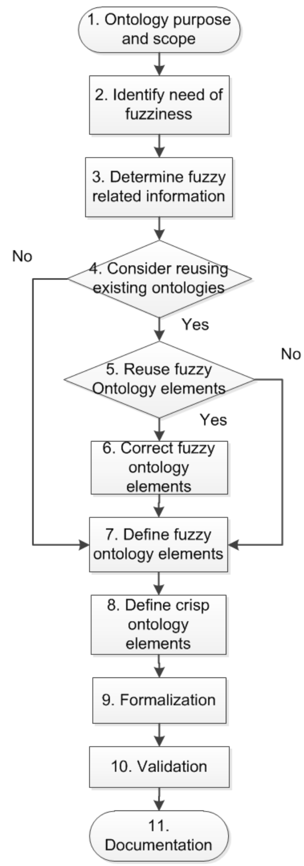


FIGURE 2. Workflow of the proposed FODM.

1) PHASE 1: ONTOLOGY PURPOSE AND SCOPE

As defined in the majority of crisp ontology development methodologies, such as Methontology, the primary task is to clarify the motivation of building a fuzzy ontology. In other words, the purpose and scope of modelling information using fuzzy ontology should be clearly defined. Basic questions should be raised and explicitly answered in order to make the purpose and scope of ontology clear. For example,

[4]http://www.upm.es/observatorio/vi/index.jsp?pageac=grupo.jsp&idGrupo=400

a set of questions could be 1) What is the domain or scope of information that needs to be modelled? 2) Is ontology the best modelling technique over other solutions, such as text, key value, and Unified Modelling Languages (UML) etc.? 3) What is the type (including domain-specific, generic or core, application specific, and representational ontologies) of ontology depending on the determination of domain or scope? 4) Who will be involved in the development of ontology and what roles they are going to play? 5) How to ensure a tight collaboration between different participants so as to guarantee a successful development of ontology? Once questions are accurately addressed, the purpose and scope of ontology could be established. Though answers to those questions might slightly change during the development process, the general purpose and scope could retain at given moments. Until now, it is clear that an ontology is going to develop in order to model information within a specific domain or scope.

2) PHASE 2: IDENTIFY THE NEED OF FUZZINESS

With using fuzzy ontologies to manage vagueness and impreciseness born in mind, the second phase aims to identify whether fuzziness should be introduced into the ontology design. The ultimate goal of this step is to determine what type of ontology is going to build: either crisp ontology or fuzzy ontology. In this step, both ontology engineers and domain experts should participate and cooperate with each other to establish the need of fuzziness. The reason behind the involvement of domain experts is because domain experts could provide specialized knowledge to analyze if fuzziness is needed. To obtain a proper answer, a set of activities should be conducted. Firstly, a deeper identification on the domain or scope of ontology should be done. A first check on the information that is going to be modelled can enrich the understanding on the necessity of fuzziness. After the check, information that is vague present in the domain or scenario could be found out. Secondly, domain experts will justify whether fuzziness will be taken into account in the ontology design. Before the emergence of the fuzzy ontology technique, crisp ontology is widely used in a diversity of domains where actually vague information exists. However, all information in those domains is assumed to be accurate and uncertainty inherent to information is neglected. Now with the fuzzy ontology technique, it is feasible to deal with vagueness that crisp ontologies could not. Nevertheless, the need of fuzziness should be decided by domain experts because of the balance between degree of vagueness and complexity of building fuzzy ontologies. In other words, to what extent the planned ontology is going to represent the information should be justified. Thirdly, fuzziness might exist in different ontology elements according to the definition of fuzzy ontologies. Different types of fuzziness should also be identified, such as indetermination of individuals in instantiating concepts (namely, fuzzy concepts), blurry relations in pairs of individuals (namely, fuzzy relations) etc. The identification of specific fuzzy elements which are likely to be included need not be exhaustive but need be sufficient to get a rough grasp. After all these actions, the need of fuzziness can be determined and also a general cognition of specific types of fuzziness underlying in the planned ontology can be obtained.

3) PHASE 3: DETERMINE FUZZY RELATED INFORMATION

Since research on methodologies for building crisp ontologies is quite mature and also it falls out of the focus of this paper, the default setting for the result of step 2 is true which denotes that fuzziness is required in the ontology design. Hence, the main focus of the step 3 is put on determining fuzzy related information. Following the step 2, a better understanding for vague information present in the domain could be achieved. In this step, information that really has vague meanings could be identified to a greater extent. A distinction between precise and vague information can be established which could provide valuable inputs for further definitions. Based on the results obtained in this step, the knowledge base in the intended domain could be split into two parts: precise and fuzzy related information. With a clear awareness of the differentiation, ontology engineers could provide different treatments tailored for precise information or fuzzy related information in a well-defined manner.

4) PHASE 4: CONSIDER REUSING EXISTING ONTOLOGIES

Checking existing ontologies relevant to the domain or scope of interest and determining their reusability are the main tasks defined in this phase. Reusing existing resources can give a lot of credits for the ontology design. Mainly, benefits brought by reusing existing ontologies are two-fold: 1) reducing workload of designing ontologies and saving the design time, and 2) enabling interoperability and compatibility with other applications which commit to the same ontologies. It is worth noting that here existing ontologies refer to not only crisp ontologies but also fuzzy ontologies. Existing fuzzy ontologies are firstly considered and included into the list to check for reusability. It is worth noting that compared with crisp ontologies, existing fuzzy ontologies are fewer and more difficult to navigate. To the best of our knowledge, there is not such a database or hub dedicated for publishing fuzzy ontologies. However, traditional ontology resources, such as W3C wiki,[5] Swoogle[6] webpages, domain relevant documents, project documentations, and academic publications, could be visited for existing fuzzy ontologies. For instance, to find existing fuzzy ontologies for recognition of human behaviour, a web search using key words ‘‘fuzzy ontology for human behaviour recognition’’ could bring some useful information, such as the source link to an existing fuzzy human behaviour ontology (http://users.abo.fi/ndiaz/public/FuzzyHumanBehaviourOntology/) and many research papers on fuzzy human behaviour ontologies. With the existing fuzzy ontology

[5] W3C wiki: https://www.w3.org/wiki/MainP age

[6] Swoogle: http://swoogle.umbc.edu/

```
Inputs:
ExistingCrispOntologies; ExistingFuzzyOntologies; DomainOrScopeOfOntology;
Outputs:
ExistingOntologiesToBeReused; FuzzyElementsToBeReused; CrispElementsToBeReused;
DefinedFuzzyElements;
Functions:
ReusingExistingOntologies (ExistingCrispOntologies, ExistingFuzzyOntologies,
DomainOrScopeOfOntology): checking if existing crisp and fuzzy ontologies could be reused,
returning ExistingOntologiesToBeReuse d;
FuzzyElements (ExistingOntologiesToBeReused): checking whether selected ontology elements
to be reused are fuzzy or not;
CorrectFuzzyElements (FuzzyElementToBeReused): refining fuzzy elements which are inherited
from existing fuzzy ontologies to fit in the domain or scope of the planned ontology;
DefineFuzzyOntologyElements: defining fuzzy ontology elements from scratch to represent
corresponding fuzzy related information.
Main:
If ReusingExistingOntologies (ExistingCrispOntologies, ExistingFuzzyOntologies,
DomainOrScopeOfOntology) == True then
        if FuzzyElements (ExistingOntologiesToBeReused)==True
        then CorrectFuzzyElements (FuzzyElementToBeReused)
        else DefineFuzzyOntologyElements
else DefineFuzzyOntologyElements
```

FIGURE 3. Flow of phase 4, 5, 6, and 7.

resources, ontology engineers and domain experts should further examine their relevance to the target domain. Fuzzy ontology elements, which provide approximation and modelling for similar vagueness, could be inherited. In addition, crisp ontology elements defined in existing fuzzy ontologies could also be useful if they are considered as relevant to the target modelling information. This extension of introducing fuzzy ontologies into the existing ontology base can increase the possibility to reuse ontological elements in the ontology design. In this way, reusability of existing ontological resources could be maximized. Apart from existing ontologies, non-ontological resources, such as literal classifications and domain specifications, can also be used to extract useful terminologies and hierarchies. Depending on the fuzziness of existing ontologies which are selected as candidates to be reused, different actions are defined to process crisp or fuzzy elements in order to integrate existing ontology elements into the intended ontology. Fig. 3 illustrates the specific treatment to ontology elements that could be reused in terms of fuzziness.

5) PHASE 5: REUSE FUZZY ONTOLOGY ELEMENTS

The answer to that whether existing ontologies could be reused could become clear after step 4. If an or several existing ontologies are analyzed to be useful in the ontology design, a fine-grained check should be made on those potential ontologies. The check-up is focused on inspecting whether selected ontology elements from existing ontologies are fuzzy. Three different kinds of check results may be got: 1) only crisp ontology elements, 2) only fuzzy ontology elements, and 3) both crisp and fuzzy ontology elements could be reused in the planned ontology. If only crisp ontology elements from existing ontologies are identified as useful, then it leads to step 7 which will be specified in subsection Phase 7. Taking into account vague information in the domain of interest, existing fuzzy ontologies might have already provided similar specifications and corresponding modelling to those impreciseness and vagueness. Thus, some fuzzy ontology elements could be picked out from existing ontologies and be potential elements to be reused in the planned ontology. If the check result falls into this case, then further inspection and correction on those fuzzy ontology elements should be made which are explicitly defined as step 6. If the check result is the last case, then both phase 6 and 7 should be activated.

6) PHASE 6: CORRECT FUZZY ONTOLOGY ELEMENTS

In this phase, the involvement of domain experts is required to correct fuzzy ontology elements which are inherited from existing fuzzy ontologies. Specifications and modelling for vagueness provided by existing fuzzy ontology elements may not guarantee a perfect fit to capture the information that is identified as vague in the domain of interest. Therefore, fuzzification for ontology elements should be refined to accommodate the target ontology requirements. For instance, a fuzzy data type YoungAge defined in an existing fuzzy ontology O_1 is considered to be reused in the planned ontology O_2. However, the fuzzy definition for the data type YoungAge with range restricted by a leftshoulder membership function [29] ls(0,90,10,30) is identified by domain experts as a mismatch to the vague information *'people aged from 10 to 40 could be regarded as young people*' in O_2. Based on information provided by domain experts, the fuzzy data type YoungAge could be reused in O_2 with a corrected fuzzy set, such as ls(0,90,10,40). It is worth noting that to model the same piece of vague information, different solutions which include different fuzzy ontology elements can be available. To choose the most suitable one from existing modelling is also considered in this phase. Taking the same piece of vague information *'people aged from 10 to 40 could be regarded as young people*' as an example, the vagueness in the definition of young age can be captured using different solutions. One is described previously using a fuzzy data type YoungAge to express the vagueness in the definition of young age. Another possibility is to define a fuzzy modifier [29] which could be a function very=ls(0,90,10,40) and use this fuzzy modifier to restrict the property (*isClassifiedAs*) between concept People and YoungPeople. Therefore, the vague information can be expressed as People (and very (*isClassifiedAs*) YoungPeople) or People (and *hasAge* YoungAge). With activities undertaken in this phase, existing fuzzy ontology elements can be corrected to ensure an accurate approximation to information which has a vague meaning present in the intended domain or application.

7) PHASE 7: DEFINE FUZZY ONTOLOGY ELEMENTS

The output of phase 3, which is a comprehensive understanding of distinction between fuzzy related information and crisp information, could be regarded as a valuable input in this phase. The goal of this phase is to define different fuzzy ontology elements to provide correct approximations to the nature of vague and imprecise information in the domain. Tight collaborations between domain experts and ontology engineers are needed in this phase. Domain experts are required to provide a clear and specific definition/quantification for vague information based on their expertise or historical statistics. Fuzzification, such as membership functions and certain degree etc., set by domain experts can reflect imprecise and vague information. Ontology engineers should model vague information by means of fuzzy ontology elements, such as fuzzy concepts, fuzzy relations, and fuzzy data types etc., in a well-organized manner. The procedure to define fuzzy ontology elements is essentially in line with activities defined in crisp ontology development methodologies, such as enumerating (fuzzy) concepts, building the hierarchy, establishing (fuzzy) relations, and defining specific (fuzzy) data types. However, the significant difference between building fuzzy ontology elements and crisp ontology elements is to accurately capture the vagueness in the specifications and represent it using fuzzy sets. The vagueness and its interpretation of fuzzy degrees need to be precisely modelled based on context, namely, particular knowledge domain or scope. Therefore, domain experts play an important role in this stage. Though there might be just a very small amount of vague information present in the whole domain of interest, to model them associated with fuzzy logic is a key task in the whole development process. Up to this point, all precise and vague information could be correctly addressed and modelled by means of corresponding fuzzy elements within the fuzzy ontology.

8) PHASE 8: DEFINE CRISP ONTOLOGY ELEMENTS

This phase focuses on dealing with certain knowledge in the domain. Apart from fuzzy related information, the rest of knowledge base in the domain is defined as crisp ontology elements depending on their specific attributes. Activities defined in conventional ontology development methodologies could be applied in this phase to model crisp information. For instance, taking the method proposed in [19] as an example, to enumerate important terms and organize them in a hierarchical manner could be the first step in this phase. To develop the class hierarchy, three approaches can be followed: 1) top-down (starting with the most general concepts and detailing them to a fine-grained manner), 2) bottom-up (defining the most specific concepts and generalizing them to a higher level), and 3) combination (a mix of the top-down and bottom-up approaches). Relationships could be defined to link different concepts. Other crisp ontology elements, such as data properties, axioms, instances etc., are also developed in this phase. Up to this point, all elements that form the fuzzy ontology have been defined. The conceptual model for the fuzzy ontology has been completed.

9) PHASE 9: FORMALIZATION

A certain language should be selected to formalize the designed ontology into a machine-readable format. Classical ontology languages might not be suitable to express vagueness and imprecision defined in fuzzy ontologies [29]. Hence, different formalism languages have been developed to support the representation of fuzzy ontologies. Syntax and semantics of RDF are extended to support real number on the interval [0,1] to express the certain degree of subject, object and predicate [30]. A set of fuzzy extensions of DLs [2], could also be adopted to enable the transformation from fuzzy ontology elements to a standard formalization. Besides, Bobillo and Straccia [3] presented a concrete methodology to formalize fuzzy ontologies using OWL 2 annotation properties. Fudholi et al. [31] put forward to represent fuzzy ontology elements by means of rules formulated in SWRL. The SWRL-based approach is easy to be used despite it considerably increases the amount of rules and limits the scalability of fuzzy ontologies.

It is worth noting that different fuzzy ontology formalism languages vary from each other in terms of characteristics and capabilities they hold. There is not a standard mechanism to evaluate different formalism languages because they have different strengths and weaknesses with regard to represent specific ontology elements. For example, fuzzy data types are not supported by the fuzzy description logic f-SHIN [32] and the SWRL-based approach while they can be easily expressed by fuzzy OWL 2 annotations. Therefore, a specific formalism language should be chosen according to specific fuzzy ontologies' requirements to enable fuzzy expressions.

10) PHASE 10: VALIDATION

The success of creating a fuzzy ontology is subject to the validation result. The designed ontology should go through a thorough check to ensure it has represented the intended model of the world. In this phase, the designed ontology needs to be validated in terms of several features as follows:

- **Correctness**. The developed ontology should be able to accurately reflect information that is included in the target domain. A clear borderline between crisp information and fuzzy related information is established in the ontology. Accordingly, crisp and fuzzy information are correctly modelled. Particularly, with a focus on fuzzy elements, it is necessary to ensure that real vague meanings in the domain have been correctly captured, understood, approximated, and treated in the ontology.
- **Consistency**. Local inconsistency in the ontology network should be checked. This feature could be automatically checked by some fuzzy ontology reasoners, such as fuzzyDL reasoner [33], and DeLorean [34]. The consistency issue exists in mainly two aspects: the structure level and the content level. In terms of the structure-based consistency, inclusions of constructors, such as owl:disjointWith, and rdfs:subClassOf etc., should be ensured to avoid any conflicts in the ontology hierarchy. Basic observations should be made on the ontology statements to check if any of them contains controversial definitions for the same specification. In this way, the content-based consistency could be guaranteed.
- **Completeness**. The completeness feature ensures that the designed ontology has been able to cover all the aspects of information that belongs to the target domain. It could provide a complete representation of the real world knowledge. With a focus on fuzzy related infor-

mation that is identified by domain experts as significant in the domain, it is a must to ensure that vagueness has been fully captured and included in the fuzzy ontology.

- **Rationality**. The inclusion and quantification for fuzziness, such as fuzzy set and certain degree, make sense to get a good approximation to real information that has vague meanings. A common agreement on the designed treatment for vague information between domain experts and ontology engineers should be achieved.
- **Understandability**. The nomenclature for ontology elements should be easily understandable to all stakeholders, including domain experts, ontology engineers and ontology users. The naming mechanism used in the ontology should be easy, self-explanatory and intuitive. Understandability could strengthen the ease of use of the designed ontology and promote its usability.
- **Conciseness**. Conciseness is also a significant criterion to be considered to evaluate the quality of ontology. Ontology terms are expected to express the most by using the least number of words. To model the same domain of interest, a lightweight and concise ontology is usually preferable than a heavy one under the condition that they cover the same knowledge base. Redundancies in the ontology will increase the volume of the ontology and applicable complexities as well.

In general, the aforementioned properties, except consistency, are subjectively examined by humans who have been involved in the development process, including domain experts, ontology users and ontology engineers. To minimize the side effect of subjectivity in the validation process, it is better to involve as many people as possible, such as another group of domain experts and ontology developers, in verifying the developed ontology. The consistency feature of the developed ontology is usually evaluated by an existing fuzzy ontology reasoner.

11) PHASE 11: DOCUMENTATION

In this stage, documentation to introduce the engineering principles of the designed ontology, including descriptions for different ontology elements, design details, method of usage, and maintenance etc., should be written up. As communicable materials to the public, the documentation should be concise, illustrative, understandable, and comprehensive so that non-experts (e.g., ontology users) can easily identify the potential usage of this ontology in their own applications by looking up the document. Besides, enabling the developed ontology as open source to the ontology community is another step forward. Open access to the ontology can expand its dissemination and increase the possibility of reusability in other projects or applications. In addition, valuable feedback from the ontology community can also be collected and used to make a better revision or maintenance on the ontology development.

IV. A USE CASE BASED ON FODM: A FUZZY SEABED CHARACTERIZATION ONTOLOGY

To show the applicability and usefulness of the proposed FODM, a simple use case from the Smart and Networking Underwater Robots in Cooperation Meshes (SWARMs) project,[7] which aims to model the characterization of seabed by means of fuzzy ontology, is presented in this section.

Description of the target domain: AUVs (Autonomous Underwater Vehicles) can get information about seabed from visual, acoustic, and position sensors. Afterwards, context recognition and 3D mapping can be conducted by AUVs to generate a 3D map. The aim of generating 3D map is to characterize different types of seabed so that operators can get a better understanding of the underwater environment and make better decisions to plan tasks. The seabed which is going to be inspected should be clearly characterized as a specific type. And also size of the inspected seabed region is of interest. So a formal vocabulary for representing the seabed types and sizes is needed for operators and AUVs to achieve a common understanding.

Following the steps formalized in the proposed FODM, a fuzzy seabed characterization ontology is going to be constructed in the following sections.

A. PHASE 1: ONTOLOGY PURPOSE AND SCOPE

With an aim at the target domain description, the motivation is to model different characteristics of seabed in order to provide a semantic annotated 3D map for operators and also enable AUVs to carry out context-aware navigation and mission execution. A list of questions is sketched as follows and answers to them can be useful to determine the ontology purpose and scope in a fine-grained manner.

- What kind of information needs to be modelled? Answer: seabed that is going to be inspected and its different features, including type and size of area, are the modelling of interest. The modelling domain is limited to seabed classifications instead of the whole underwater environment (including seabed, water, and surface etc.).
- Is ontology chosen as the modelling technique over other solutions, such as key-value modelling, UML, graphical modelling or multidisciplinary modelling? Answer: as interoperability between different AUVs and operators is expected to be achieved by using a formalized vocabulary for expressing the characterization of seabed, ontology is the most promising modelling technique to provide this specification of conceptualization.
- What will be the type of the intended ontology? Answer: according to the description, it is going to model information limited to a specific domain which is seabed. The seabed ontology aims to model general information related to the seabed which could be reused or inherited by a diversity of underwater robotics related applications. So the planned seabed ontology will be a domain-specific ontology.

[7] The SWARMs project: http://swarms.eu/

- Who will be involved in the development of ontology and what roles they are going to play? Answer: ontology engineers could be the main participant while marine experts and operators could provide insightful knowledge to the characterization.
- How could different people involved in the development facilitate the tight collaboration so as to ensure a successful development of ontology? Answer: ontology engineers will collect valuable knowledge from marine experts and operators. By conceptualizing obtained knowledge, a general framework of the ontology could be built. In addition, all of them will be involved in the majority of development activities, such as refinement and correction. Maintenance work will be mainly done by ontology engineers.

With answers shown above, it is clear that an ontology is going to be developed in order to model the seabed domain-specific information.

B. PHASE 2: IDENTIFY THE NEED OF FUZZINESS

According to the description of the target domain, seabed and its two important features, including type and size, are the intended modelling information. It is assumed that with advanced techniques, such as 3D scanning technique, 3D mapping and 3D SLAM, AUVs are able to explicitly recognize the type of a specific seabed region and calculate/quantify its numeric size. All the information is certain and could be conceptualized by crisp ontology elements. However, marine experts and operators put forward a special requirement to the ontology modelling. Apart from concrete numeric quantifications for the size of region, they would like to know linguistic specifications for the size of seabed. Thus, how to map a seabed region with explicit numeric area known to a linguistic specification implies vagueness. For instance, the size of seabed regions could be classified into three classes, namely, *small*, *medium*, and *large*. The borderline between each type is blurry and overlap between each type could exist. Based on this analysis, fuzziness is needed to manage vagueness inherent to the region size. The conclusion drawn in this phase is that a fuzzy ontology, instead of a crisp ontology, is determined to model the seabed domain.

C. PHASE 3: DETERMINE FUZZY RELATED INFORMATION

Since a fuzzy ontology is determined to model the seabed domain, a tight cooperation between marine experts and ontology engineers is demanded in this stage. A clear distinction between fuzzy related information and certain information should be established. After collecting knowledge and suggestions from marine experts, ontology engineers come up with an accurate diagnosis for the border between fuzzy related and certain information. The results are shown as follows:

Precise information: different seabed regions can be explicitly characterized as corresponding types, such as ground, vegetation, rocks, human-made walls, mud, cliff and spring in seabed. Numeric area values of different seabed regions can be calculated to explicitly represent the size of corresponding seabed regions.

Fuzzy related information: linguistic specifications for the size of seabed region, e.g., large, medium and small, contain vague meanings because a seabed region could be described as large to some extent while it could also be labelled as medium with a probability. The definitions for linguistic classifications for the size of seabed region should be fuzzified to meet the domain needs.

The knowledge base of the seabed domain is accurately divided into two parts: *precise information* and *fuzzy related information*. Afterwards, they can be modelled with different treatments, respectively.

D. PHASE 4: CONSIDER REUSING EXISTING ONTOLOGIES

Having known the domain and scope of the intended fuzzy ontology, existing ontology resources, such as W3C wiki, Swoogle, project webpages, and publications etc., should be checked to find candidates to be reused. Not only crisp ontologies, but also fuzzy ontologies present in existing ontology databases, are reviewed and analyzed for potential reengineering. After querying those ontology resources using keywords, such as *Seabed*, *Seabed types*, *Seabed characterization*, and *Size*, a set of ontologies is found as potential candidates to be reused. After in-depth analyses and comparisons on their content and granularity, the CO3-AUV (Cooperative Cognitive Control for Autonomous Underwater Vehicles) ontology [35], developed in the CO3-AUV project,[8] is selected due to its high relevance to the seabed domain requirements. The CO3-AUV ontology includes a classification of 3D sonar scan points of seabed texture into different structural classes. Different structural classes for the seabed imply potential usage of being imported as a portion of the seabed ontology.

E. PHASE 5: REUSE FUZZY ONTOLOGY ELEMENTS

Since the CO3-AUV ontology, which is selected as the ontology candidate to be reused from phase 4, is a crisp ontology, then a conclusion, that only crisp ontology elements could be reused, can be drawn in this phase. Specifically, seabed types defined in the CO3-AUV ontology which are selected to be reused are as follows:

- Ground. It refers to a patch of seabed region which is smooth and faces upwards.
- Wall. It is regarded as a class of seabed regions which is usually man-made for particular usages, such as supporting AUVs during operations.
- Rock. This concept contains a collection of seabed which is made of rock.
- Vegetation. It is a class generally describing seabed regions which are covered by different kinds of vegetation, such as sea weed and kelp etc. AUVs should avoid this kind of seabed so that they would not get stuck in it.

[8]http://robotics.jacobs-university.de/projects/Co3-AUVs/

- Unknown. This is a catch-all concept which represents a class of seabed that is difficult to be recognized as a specific type. Or the specific type of the seabed region is out of the operators' interest and therefore there is no need to classify it.

F. PHASE 6: CORRECT FUZZY ONTOLOGY ELEMENTS

Since no existing fuzzy ontology elements are considered to be reused in the seabed fuzzy ontology, this phase could be skipped.

G. PHASE 7: DEFINE FUZZY ONTOLOGY ELEMENTS

In phase 3, the seabed knowledge base has been partitioned into two categories: precise information and fuzzy related information. Aiming at representing vague and imprecise information using fuzzy ontology, different fuzzy ontology elements are defined in this phase. To provide linguistic classifications for the size of seabed regions, three fuzzy data types and four fuzzy concepts are defined by ontology engineers in collaboration with marine experts. Definitions of fuzzy data types which follow the fuzzyDL reasoner syntax and vague information they intend to model are shown in Tab. 1. Specifications for fuzzy concepts defined in the fuzzy seabed characterization ontology are also presented in Tab. 2. The expressions for fuzzy concepts follow the syntax of fuzzy Description Logics [29]. In principle, the definition of fuzzy data type aims to provide corresponding specification for the data format of fuzzy concept, such as SmallSize restricts the numeric size of SmallRegion seabed and also generates a specific probability for a seabed area to be classified as small. Thus, a crisp data property, *hasNumericValueSize*, should be defined in order to specify the relationship between fuzzy concepts (Small, Medium, and Large) and fuzzy data types (SmallSize, MediumSize, and LargeSize).

As shown in Tab. 1 and Tab. 2, marine experts provide fuzzification for the blurry borderlines between small, medium and large size using three fuzzy sets, namely membership functions. More specifically, fuzzy sets, which are encased in the fuzzy seabed characterization ontology to describe fuzzy data types, can be seen in Fig. 4.

TABLE 1. Fuzzy data types defined in the fuzzy seabed characterization ontology.

Fuzzy data type	Definition	Vague information modelled
SmallSize=leftshoulder (0,1000,50,100)	Denoting that the numeric size of a small seabed region should comply with a leftshoulder membership function leftshoulder (0,1000,50,100).	Seabed with its size ranging from 0-100 m^2 could be regarded as small to some degree. The degree distribution complies with a leftshoulder membership function.
MediumSize=trapezoidal (50,100,150,200)	Denoting that the numeric size of a medium seabed region should comply with a trapezoidal membership function trapezoidal (50,100,150,200).	Seabed with its size ranging from 50-200 m^2 could be regarded as medium to some extent. The degree distribution complies with a trapezoidal membership function.
LargeSize=rightshoulder(0,1000,150,200)	Denoting that the numeric size of a large seabed region should comply with a rightshoulder membership function rightshoulder(0,1000,150,200).	Seabed with its size ranging from 150-1000 m^2 could be regarded as large with a possibility. The degree distribution complies with a rightshoulder membership function.

TABLE 2. Fuzzy concepts defined in the fuzzy seabed characterization ontology.

Fuzzy concept	Definition	Vague information modelled
SeabedRegion	Representing the superclass of a set of subconcepts, including SmallRegion, MediumRefion, and LargeRegion.	Seabed region could be described by linguistic variables, such as small region, medium region, and large region.
SmallRegion	Containing a collection of seabed whose size is assigned with the SmallSize fuzzy data type. $SmallRegion \equiv \exists hasNumericValueSize.SmallSize$	Seabed area size ranging from 0-100 m^2 is classified as small by complying with a leftshoulder membership function.
MediumRegion	Containing a collection of seabed whose size is assigned with the MediumSize fuzzy data type. $\mathrm{MediumRegion} \equiv \exists \mathrm{hasNumericValueSize.MediumSize}$	Seabed area size ranging from 50-200 m^2 is classified as medium by complying with a trapezoidal membership function.function.
LargeRegion	Containing a collection of seabed whose size is assigned with the LargeSize fuzzy data type. $\mathrm{LargeRegion} \equiv \exists \mathrm{hasNumericValueSize.LargeSize}$	Seabed area size ranging from 150-1000 m^2 is classified as large by complying with a rightshoulder membership function.

H. PHASE 8: DEFINE CRISP ONTOLOGY ELEMENTS

In this phase, the rest part of ontology, namely, crisp ontology elements, should be defined to model precise information in the target domain. Besides, fuzzy ontology elements already

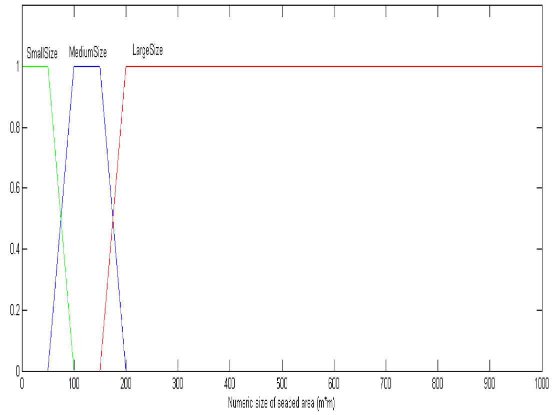


FIGURE 4. Fuzzy data types for seabed region size.

defined in the previous stage and crisp ontology elements which are inherited from existing ontology should be considered for the creation of new crisp ontology elements to avoid any inconsistency. With valuable inputs from marine experts, the following crisp ontology concepts, object and data properties are defined and shown in Tab. 3, Tab. 4 and Tab. 5, respectively.

TABLE 3. Crisp concepts defined in the fuzzy seabed characterization ontology.

Crisp concept	Definition	Certain information modelled
SeabedTypes	Representing a super class of different seabed types.	A specific region of seabed could be classified as a specific type. The type of a specific region of seabed is a significant feature for AUVs to be considered during missions.
Cliff	Representing seabed which geologically is vertical, or near vertical. It is disjoint with other seabed types.	Marine experts think that the recognition of seabed region as cliff is very important for AUVs' operations.
Mud	Representing seabed which is a sticky mixture of earth and water. It is disjoint with other seabed types.	Whether the inspected seabed is mud or not is identified by marine experts to be an important criterion for navigation.
SpringInSeabed	Referring to seabed which contains spring. It is disjoint with other seabed types.	Marine experts consider the fact that whether the inspected seabed contains spring is significant.

As shown in Tab. 3, apart from five seabed types inherited from the CO3-AUV ontology, three more seabed types are included in the seabed fuzzy ontology. The reason behind the addition of new seabed types is because that the existing classification for seabed types by the CO3-AUV ontology is unable to cover all the requirements in the seabed domain. Apart from Ground, Wall, Rock, Vegetation, and Unknown, marine experts have identified that three more types (Cliff, Mud, and SpringInSeabed) are important information for operators and AUVs. All the seabed types are disjoint with each other.

I. PHASE 9: FORMALIZATION

In this use case, OWL 2 is selected as the formalism language to represent the designed ontology model. To easily carry out the transformation from the co nceptual model into the OWL 2-formatted expressions, the ontology editor protégé and its Fuzzy OWL extension are employed in this step. Protégé allows visualized and easy implementations of the designed fuzzy seabed characterization ontology (seen as Fig. 5). Automatic generation of the ontology code in different languages, such as OWL, RDF etc., is also enabled by protégé. The OWL file of the developed fuzzy seabed characterization ontology can be accessible in the web.[9]

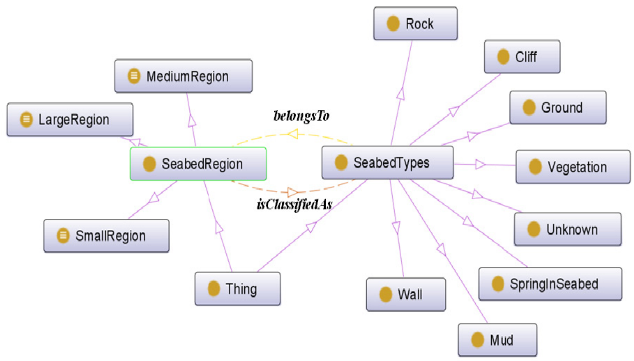


FIGURE 5. The overall visualized structure of the fuzzy seabed characterization ontology.

J. PHASE 10: VALIDATION

The validation results serve as a proof of the usefulness of the developed ontology. The consistency feature of the developed ontology is evaluated by the fuzzyDL reasoner. Other features are subjectively examined by the ontology engineers, domain experts, and ontology users who have been involved in the development process. Specifically, the validation results are presented in the following.

- **Correctness**. No information from the seabed domain is wrongly interpreted and modelled in the developed seabed ontology. Marine experts have provided a clear and correct borderline between certain information and vague information. Vagueness existing in the seabed domain is also correctly captured and represented by corresponding fuzzy ontology elements associated with correct fuzzy sets. In addition, crisp ontology elements and fuzzy ontology elements have been accurately linked via relationships defined by ontology engineers and marine experts.
- **Consistency**. The fuzzy seabed characterization ontology is identified as consistent by invoking the fuzzyDL reasoner. Observations on the structure and content by

[9] https://archive.org/download/FuzzySeabedCharacterizationOntology

ontology engineers and marine experts show that there are no elements containing controversial definitions in the developed ontology.

- **Completeness**. The seabed fuzzy ontology has met all the requirements raised in the first stage and covered the overall knowledge base. Particularly, all information that has vague meanings has been captured and represented in the seabed ontology.
- **Rationality**. The borderline defined by marine experts for distinguishing certain information and uncertain information is rational. Fuzziness introduced into data types and classes for approximating the vagueness of different region sizes makes sense to other marine experts and ontology engineers.
- **Understandability**. The developed ontology can be easily understood by domain experts, ontology engineers, and ontology users. The ontology terms defined in the fuzzy ontology are identified to be self-explanatory.
- **Conciseness**. In the view of domain experts and ontology users' inspection, ontology terms are concise enough to express the intended meanings with the least number of words. There is no redundancy in the ontology naming or structure.

K. PHASE 11: DOCUMENTATION

In this example, documentation to introduce the fuzzy seabed characterization ontology is omitted as it falls out of the main focus of this paper.

V. DISCUSSIONS

As shown in section IV, the fuzzy seabed characterization ontology has been successfully developed following the instructions provided by the proposed FODM. During the development process, each phase has been set with clear purposes and the to-do list. It could expect that by using the formal FODM, efficiency and accuracy can be enhanced in the construction process. In principle, the FODM is an abstract description of activities that should be done in order to build a fuzzy ontology in a logic order. The ultimate aim of the proposed methodology is to provide a methodological guideline for the fuzzy ontology construction, so it is of nature to ensure an outperformance than intuitive work. Nevertheless, as emphasized in the introduction section, being a theoretical methodology, it faces a difficulty in making quantitative and rigorous analyses and comparisons with other existing ontology methodologies or no methodology. It is a fact that the lack of quantitative evaluation exists in all existing ontology methodologies [16], including those dedicated to building crisp ontologies, fuzzy ontologies or probabilistic ontologies. For instance, the well-known METHONTOLOGY does not include any evaluation though it does provide a principled methodology for building crisp ontologies from scratch. Likewise, the NeON methodology proves its applicability in different experimental scenarios without providing any rigorous evaluation. Diligent, as a methodology for developing crisp ontologies, offers some use cases without any sort of evaluation. Similarly, the newest probabilistic ontology development methodology [36], published in August 2016, also excludes the evaluation part. Thus, current research just accepts the way an ontology development methodology is proposed as because of the subjective nature of this field. Since ontology development methodologies cannot be rigorously evaluated, it becomes clear that ontology developers choose their methods from existing ones simply by their subjective judgements based on theoretical analyses or experimental experiences.

Due to the aforementioned reasons, the FODM, like other existing ontology development methodologies, is left as unevaluated with other relevant development methods, including non-methodological fuzzy ontology development and existing ontology development methodologies. However, the FODM could be expected to bring enhancement in the fuzzy ontology development process due to the following features:

- Compared with non-methodological ontology development, namely building fuzzy ontologies based on ontology engineers' preferences or intuitions, the only overload introduced by the FODM is the extra time required to learn and practice with the methodology. But in theory, building fuzzy ontologies in a formal and well-ordered manner would speed up the construction process to some extent and probably ensure a better quality of an ontology design.
- The proposed FODM provides the first methodological guideline for building fuzzy ontologies from scratch, starting from determining motivation and ending up with documenting to introduce the designed ontology. This methodology is more complete compared with existing work. Non-methodological development could easily omit steps that are actually significant for the ontology development. The same problem also exists in the existing fuzzy ontology development methodologies. For instance, reusing fuzzy ontology elements from existing fuzzy ontologies is out of consideration in existing fuzzy ontology development methodologies.
- The proposed FODM could be more generally applicable than existing fuzzy ontology development methodologies. Existing methodologies provide the first approach towards the fuzzification of existing crisp ontologies. The dependence on existing crisp ontologies in those methodologies imposes additional constraint for their usage. Namely, their applicability is limited to be used in domains or applications where crisp ontologies have been previously developed. Differently, the FODM aims to provide a generic solution to develop fuzzy ontologies from scratch. It offers different treatments and utilizations for the target domain which either contains existing ontologies or not.
- The FODM divides the target knowledge base into two parts: precise information and fuzzy related information. In this way, ontology engineers can clearly know the borderline between those two parts and provide different

methodological strategies to model them. For precise information, existing conventional methodologies can be employed. And ontology engineers can focus on dealing with information which contains vague meanings by means of defining corresponding fuzzy ontology elements.
- Though the FODM aims to provide a methodological approach to build fuzzy ontologies, it could also be practically used for crisp ontology constructions. If the intended world of the model is identified as crisp during the development process, then the proposed methodology goes through with a set of steps which are essentially in line with conventional methodologies. Thus, the proposed methodology can also act as a standard guide for building crisp ontologies. To conclude, the FODM can be applicable to guide the construction of both crisp and fuzzy ontologies due to its generality and comprehensiveness.

VI. CONCLUSIONS AND FUTURE WORK

A novel fuzzy ontology development methodology, abbreviated as FODM, has been presented in this paper. The FODM provides the first methodological guideline for building fuzzy ontologies from scratch. Based on lessons learned from existing ontology development methodologies, the FODM has been conceived focusing on standardization of the development activities to deal with the vagueness which is inherent to knowledge representation. The FODM abstracts the entire development process into eleven engineering phases, and concrete activities which are necessary to be done in each phase have been enumerated and described. The FODM has the following outstanding features:

- Different from conventional crisp ontology development methodologies, changes introduced by additional considerations for fuzzifying vague information have been included in the proposed methodology. The FODM has provided a standard methodological approach to represent vague information by taking advantage of fuzzy logic.
- Essentially, the FODM does not completely transform the development workflow defined by conventional ontology development methodologies. If the intended world of the model is identified as certain, the FODM could accommodate (such as skipping steps tailored for dealing with vagueness) to develop crisp ontologies. Therefore, the FODM can also be used as a methodology to build crisp ontologies.
- A clear differentiation between precise information and fuzzy related information has been included in the FODM. In this way, different treatments and utilizations can be provided to represent certain and fuzzy related information. Domain experts can focus on analyzing the fuzzy related information and providing accurate specifications to approximate the vagueness based on their expertise or historical statistics.
- The FODM is conceived to be general and it can be applied to develop fuzzy ontologies with or without the existence of crisp or fuzzy ontologies in the same domain or application. The applicability of FODM goes beyond existing fuzzy ontology methodologies to some extent.
- The FODM could be regarded as comprehensive and complete due to its attempt to formalize all necessary activities in the development process. It includes significant phases which are dismissed in some of the existing work. For instance, reusing existing fuzzy ontologies is not considered in the existing fuzzy ontology development methodologies.

Apart from detailed specifications for the FODM, a fuzzy seabed characterization ontology has been developed following the proposed FODM. Design details have been shown step by step. The success of building the fuzzy seabed characterization ontology has demonstrated that the proposed FODM can be applicable to build fuzzy ontologies from scratch and also guarantee the quality of the designed fuzzy ontologies.

Future work can be emphasized in the following aspects:

- As evaluation on the performance of a proposed methodology is a common lack in all existing ontology development methodologies, quantitative analyses and comparisons should be figured out to rigorously prove the outstanding performance of the proposed methodology.
- The proposed FODM should be tested with more experiments, such as building fuzzy ontologies from scratch, constructing crisp ontologies from scratch, building fuzzy ontologies by reusing existing fuzzy ontologies, building fuzzy ontologies by reusing existing crisp ontologies, or developing crisp ontologies by reusing existing crisp ontologies. Afterwards, valuable feedback can be obtained in order to refine or correct the proposed methodology.

REFERENCES

[1] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," *Data Knowl. Eng.*, vol. 25, nos. 1–2, pp. 161–197, 1998.
[2] T. Lukasiewicz and U. Straccia, "Managing uncertainty and vagueness in description logics for the Semantic Web," *Web Semant. Sci. Services Agents World Wide Web*, vol. 6, no. 4, pp. 291–308, Nov. 2008.
[3] F. Bobillo and U. Straccia, "Fuzzy ontology representation using OWL 2," *Int. J. Approx. Reasoning*, vol. 52, no. 7, pp. 1073–1094, Oct. 2011.
[4] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965.
[5] V. V. Cross, "Fuzzy ontologies: The state of the art," in *Proc. IEEE Conf. Norbert Wiener 21st Century (21CW)*, Jun. 2014, pp. 1–8.
[6] J. Zhai, Y. Yu, Y. Liang, and J. Jiang, "Traffic information retrieval based on fuzzy ontology and RDF on the semantic web," in *Proc. 2nd Int. Symp. Intell. Inf. Technol. Appl. (IITA)*, 2008, pp. 779–784.
[7] M. Baziz, M. Boughanem, Y. Loiseau, and H. Prade, "Fuzzy Logic and Ontology-based Information Retrieval," in *Fuzzy Logic*, vol. 215, P. P. Wang, D. Ruan, and E. E. Kerre, Eds. Berlin, Germany: Springer, 2007, pp. 193–218.
[8] J. Zhai, L. Shen, Y. Liang, and J. Jiang, "Application of fuzzy ontology to information retrieval for electronic commerce," in *Proc. Int. Symp. Electron. Commerce Secur.*, 2008, pp. 221–225.

[9] Q. T. Tho, S. C. Hui, A. C. M. Fong, and T. H. Cao, "Automatic fuzzy ontology generation for semantic Web," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 6, pp. 842–856, Jun. 2006.

[10] E. Sanchez and T. Yamanoi, "Fuzzy Ontologies for the Semantic Web," in *Flexible Query Answering System*, vol. 4027, H. L. Larsen, G. Pasi, D. Ortiz-Arroyo, T. Andreasen, and H. Christiansen, Eds. Berlin, Germany: Springer, 2006, pp. 691–699.

[11] F. Ali, E. K. Kim, and Y.-G. Kim, "Type-2 fuzzy ontology-based semantic knowledge for collision avoidance of autonomous underwater vehicles," *Inf. Sci.*, vol. 295, pp. 441–464, Feb. 2015.

[12] Z.-S. Mi, A. C. Bukhari, and Y.-G. Kim, "An obstacle recognizing mechanism for autonomous underwater vehicles powered by fuzzy domain ontology and support vector machine," *Math. Problems Eng.*, vol. 2014, pp. 1–10, Aug. 2014.

[13] A. C. Bukhari and Y.-G. Kim, "A research on an intelligent multipurpose fuzzy semantic enhanced 3D virtual reality simulator for complex maritime missions," *Appl. Intell.*, vol. 38, no. 2, pp. 193–209, Mar. 2013.

[14] N. Díaz-Rodríguez, O. Cadahía, M. Cuéllar, J. Lilius, and M. D. Calvo-Flores, "Handling real-world context awareness, uncertainty and vagueness in real-time human activity tracking and recognition with a fuzzy ontology-based hybrid method," *Sensors*, vol. 14, no. 10, pp. 18131–18171, Sep. 2014.

[15] N. Díaz Rodríguez, M. P. Cuéllar, J. Lilius, and M. D. Calvo-Flores, "A fuzzy ontology for semantic modelling and recognition of human behaviour," *Knowl.-Based Syst.*, vol. 66, pp. 46–60, Aug. 2014.

[16] P. Alexopoulos, M. Wallace, K. Kafentzis, and D. Askounis, "IKARUS-Onto: A methodology to develop fuzzy ontologies from crisp ones," *Knowl. Inf. Syst.*, vol. 32, no. 3, pp. 667–695, Sep. 2012.

[17] H. Ghorbel, A. Bahri, and R. Bouaziz, "Fuzzy ontologies building method: Fuzzy ontomethodology," in *Proc. Annu. Meeting North Amer. Fuzzy Inf. Process. Soc. (NAFIPS)*, 2010, pp. 1–8.

[18] R. De Hong, "Methodologies for building knowledge based systems: Achievements and prospects," in *The Handbook of Applied Expert Systems*. Boca Raton, FL, USA: CRC Press, 1998.

[19] N. F. Noy and D. L. McGuiness, "Ontology development 101: A guide to creating your first ontology," Stanford Med. Informat., Stanford, CA, USA, Tech. Rep. SMI-2001-0880, 2001.

[20] M. Fernández, A. Gomez-Perez, and N. Juristo, "METHONTOLOGY: From ontological art towards ontological engineering," Stanford Univ., Stanford, CA, USA, Tech. Rep. SS-97-06, 1997.

[21] M. C. Suarez-Figueroa, "NeOn methodology for building ontology networks: Specification, scheduling and reuse," M.S. thesis, Universidad Politécnica de madrid, Tech. Univ. of Madrid, Spain, 2010.

[22] D. Vrandečić, S. Pinto, C. Tempich, and Y. Sure, "The DILIGENT knowledge processes," *J. Knowl. Manage.*, vol. 9, no. 5, pp. 85–96, Oct. 2005.

[23] Y. Sure, S. Staab, and R. Studer, "On-to-knowledge methodology (OTKM)," in *Handbook Ontologies*, S. Staab and R. Studer, Eds. Berlin, Germany: Springer, 2004, pp. 117–132.

[24] K. Kotis and G. A. Vouros, "Human-centered ontology engineering: The HCOME methodology," *Knowl. Inf. Syst.*, vol. 10, no. 1, pp. 109–131, Jul. 2006.

[25] M. Jarrar and R. Meersman, "Ontology engineering—The DOGMA approach," in *Adv. Web Semantics I*, vol. 4891, T. S. Dillon, E. Chang, R. Meersman, and K. Sycara, Eds. Berlin, Germany: Springer, 2009, pp. 7–34.

[26] M. Fernández-López and A. Gámez-Pérez, "Overview and analysis of methodologies for building ontologies," *Knowl. Eng. Rev.*, vol. 17, no. 2, Jun. 2002.

[27] M. Cristani and R. Cuel, "A survey on ontology creation methodologies," *Int. J. Semantic Web Inf. Syst.*, vol. 1, no. 2, pp. 49–69, 32 2005.

[28] A. Zouaq and R. Nkambou, "A survey of domain ontology engineering: Methods and tools," in *Advances in Intelligent Tutoring Systems*, vol. 308, R. Nkambou, J. Bourdeau, and R. Mizoguchi, Eds. Berlin, Germany: Springer, 2010, pp. 103–119.

[29] U. Straccia, *Foundations of Fuzzy Logic and Semantic Web Languages*. London, U.K.: Chapman & Hall, 2013.

[30] U. Straccia, "A minimal deductive system for general fuzzy RDF," in *Web Reasoning Rule Syst.*, vol. 5837, A. Polleres and T. Swift, Eds. Berlin, Germany: Springer, 2009, pp. 166–181.

[31] D. H. Fudholi, N. Maneerat, R. Varakulsiripunth, and Y. Kato, "Application of Protégé, SWRL and SQWRL in fuzzy ontology-based menu recommendation," in *Proc. Int. Symp. Intell. Process. Commun. Syst. (ISPACS)*, 2009, pp. 631–634.

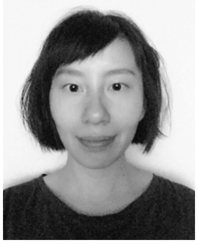
[32] G. Stoilos, U. Straccia, G. Stamou, and J. Z. Pan, "General concept inclusions in fuzzy description logics," in *Proc. 17th Eur. Conf. Artif. Intell. (ECAI)*, Riva del Garda, Italy, 2006, p. 456.

[33] F. Bobillo and U. Straccia, "The fuzzy ontology reasoner *fuzzyDL*," *Knowl.-Based Syst.*, vol. 95, pp. 12–34, Mar. 2016.

[34] F. Bobillo, M. Delgado, and J. Gómez-Romero, "DeLorean: A reasoner for fuzzy OWL 2," *Expert Syst. Appl.*, vol. 39, no. 1, pp. 258–272, Jan. 2012.

[35] M. Pfingsthorn, A. Birk, and N. Vaskevicius, "Semantic annotation of ground and vegetation types in 3D maps for autonomous underwater vehicle operation," *OCEANS*, Waikoloa, HI, USA, 2011, pp. 1–8.

[36] R. N. Carvalho, K. B. Laskey, and P. C. G. D. Costa, "Uncertainty modeling process for semantic technology," *PeerJ Comput. Sci.*, vol. 2, p. e77, Aug. 2016.



XIN LI was born in Sichuan, China, in 1990. She received the B.Eng. degree in communication engineering from Northwestern Polytechnical University, Xi'an, China, in 2012, and the M.Sc. degree (Hons.) in systems engineering and services for the information society from the Technical University of Madrid, Spain, in 2014. She is currently pursuing the Ph.D. degree with the Technical University of Madrid. Since 2014, she has been involved in two European projects, ACCUS and SWARMs. Her research interests include knowledge representation, context reasoning, semantic middleware, and underwater robotics.

JOSÉ-FERNÁN MARTÍNEZ received the degree in electronic and telecommunications engineering in 1993, where he started Research and Development tasks, and the Ph.D. degree in telematic engineering from the Technical University of Madrid, Spain, in 2001. From 1993 to 1996, he was a technical responsible in research projects, National Telecommunications Company, and he was a Technical Manager in his own Company S&H S.L. He is a pioneer and one of the most respected international scholars in the field of Communication. He has authored several national and international publications included in the Science Citation Index in his interest areas. He has been a Guest Professor in the area of Robotics and Avionics with the School of Innovation, Design, and Engineering, Mälardalen University, Sweden, since 2013, after being invited as a an International Expert in Sweden. His main interest areas and expertise are ubiquitous computing and Internet of Things, underwater cooperating robots, new advanced services for wireless sensor and actuators networks, next-generation telematic network and services for internet new generation, smart cities, service management, advanced telecom services, advanced software architectures, component-based distributed applications and intermediation platforms (middleware), and high performance and fault tolerant for resilient systems. He is a Technical Reviser and the Chair of technical national and international events on Telematics. He is a member of different international and scientific committees.

GREGORIO RUBIO received the degree in telecommunication engineering from the Technical University of Madrid (UPM) and in documentation from the Open University of Catalonia. He has been a Professor with the Department of Telematics and Electronic, UPM, since 1989. His main interests include the semantic and ontologies related to services for wireless sensors networks, service management, advanced telecommunication services, and the Internet of Things.