MACODA Ontology and Knowledge Management

Vitor Basto-Fernandes*, Diana Salvador, Iryna Yevseyeva, and Michael Emmerich

Instituto Universitário de Lisboa (ISCTE-IUL), University Institute of Lisbon, ISTAR-IUL, Av. das Forças Armadas,1649-026 Lisboa, Portugal
School of Computer Science and Informatics, Faculty of Technology, De Montfort University, LE1 9BH Leicester, United Kingdom
Leiden Institute of Advanced Computer Science, Faculty of Science, Leiden University, 2333-CA Leiden, The Netherlands
{vitor.basto.fernandes,diana_salvador}@iscte-iul.pt,
iryna.yevseyeva@dmu.ac.uk,
emmerich@liacs.nl

Abstract. In this chapter we present a Many Criteria Optimisation and Decision Analysis (MACODA) Ontology and MACODA Knowledge Management Web-Based Platform (named MyCODA, available at http://macoda.club) for the research community. The purpose of this initiative is to allow for the collaborative development of an ontology to represent the MACODA knowledge domain and to make available a set of integrated tools for its use by researchers and practicioners. MyCODA is a knowledge based platform to identify and describe MACODA research constructs, and to explore how these constructs relate to each other. It is designed to model and systematize the knowledge created by the MACODA research community, supporting features such as querying and reasoning, by means of formal logics, and use cases such as training new learners and finding research gaps in the MACODA research domain.

Keywords: Many-Objective Optimisation; MACODA Ontology; Web Ontology Language; Knowledge Management; MACODA Community Platform.

1 Introduction

It is well known and documented that the key driver for countries' economic growth and productivity is their investment in Research and Development (R&D).

* The authors acknowledge the support provided by the Lorentz Center of University of Leiden - The Netherlands, in the Many Criteria Optimisation and Decision Analysis (MACODA) Workshop, 16-21 September 2019.

Over the years, countries have substantially increased their public and private investment in science, technology and innovation [1].

The trend of increasing investment in R&D, results in more researchers being involved in scientific knowledge production and dissemination, for instance by means of scientific publications such as journal and conference papers, MSc and PhD thesis, technical reports and scientific data repositories. Scientific knowledge represents a valuable resource, it gives an ability to solve problems, promotes new ideas, and stimulates new research topics.

Scientific knowledge production's and publications' exponential growth represents a great opportunity for knowledge sharing and development on a global scale, but raises serious difficulties concerning scientific knowledge management. The knowledge is commonly not well-structured, well-defined or harmonized (different taxonomies, same constructs or concepts named differently, different concepts named in the same way, etc.).

Scarbrough, Swan and Preston [2] define knowledge management as a process or practice of creating, acquiring, capturing, sharing and using knowledge, wherever it resides, to enhance learning and performance of organizations and individuals. It enables the creation of value from the expert's domain knowledge.

R&D funding agencies around the world, and specially in Europe, highlighted the difficulty in achieving innovation and industrial productivity from the results of research [1],[3]. The struggle in knowledge discovery and utilization is perceived not only by industries but also by researchers and students, which are overloaded with the amount of knowledge produced in their domains.

Even in narrow fields, such as Multi-Objective Optimisation (MOO) or Many Criteria Optimisation and Decision Analysis (MACODA), the number of studies conducted on these topics is quite extensive. Let us illustrate the situation briefly by looking on the publication trends in *Many-criteria Optimization*. The exponential growth of publications is illustrated in Figure 1. Note, that we consider here a keyword based analysis and often papers in Many-Criteria Optimisation also are methodologies from classical EMO methods.

With the exponential growth of scientific knowledge in the MOO and MACODA fields, the need to develop a new approach to effectively manage, systematize and retrieve the knowledge produced about these fields, has become more obvious.

Domain knowledge can be captured and made available to both machines and humans by means of an ontology. Ontologies are currently the most suited way to formally represent concepts within a domain and the relationships that hold between them [4]. They not only provide a common understanding of the structure of information but also enable knowledge-sharing and reuse. With the help of an ontology, a new researcher or practitioner can easily learn more about an algorithm for a particular application or find a future research topic, considerably decreasing the efforts of searching, finding and selecting the specific knowledge of her/his interest. As an example, let us assume the role of an expert in an engineering domain facing an optimisation problem, that he/she is knowledgeable about, but he/she is not an expert on optimisation. This engi-

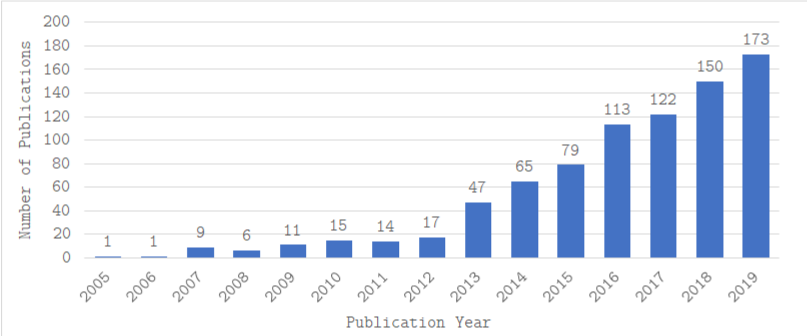

Fig. 1. Number of publications about many-objective optimisation in the Web of Science Core Collection from 2005 to 2019.

neering expert would benefit from querying features of an ontology created and managed by the optimisation research community, that allows him/her to query the ontology for algorithms that have previously proved successful in a problem similar to his/her problem.

In the optimisation field, large numbers of methods and algorithms have been proposed and published in the last decades. Thus, obtaining a systematical view of the knowledge produced in this field is becoming very complex.

New approaches and techniques are needed in order to systematize the scientific knowledge in the multi- and many-objective optimisation fields and to make it useful. Non-experts in MOO and MACODA fields should be able to explore and easily retrieve the information of interest by means of a platform, that facilitates knowledge search and retrieval. Experts should be able to share their knowledge with the community. Therefore, the development of a platform that serves this purpose is a priority.

The work presented in this chapter proposes the systematization of MACODA knowledge domain by the means of a standardised ontology representation, a Web-based knowledge management platform (named MyCODA, available at http://macoda.club), and a knowledge management process for the MACODA research and practitioners community.

MyCODA platform allows its users to easily access, learn and compare existing optimisation methods, seek an appropriate method for a specific problem, share new scientific knowledge and collaborate with other MACODA researchers.

2 MACODA Ontology

2.1 Ontology Overview

Etymologically, ontology comes from Greek and means essentially "the study or theory of being or that which is". In simple terms, ontology seeks the classification and explanation of entities.

In philosophy, an ontology is defined as "the science of what is, of the kinds and structures of objects, properties, events, processes and relations in every area of reality" [5]. Over the last decades, ontologies became more popular in other areas, namely Knowledge Management, Artificial Intelligence and the Semantic Web, given the need for a shared and common understanding of the domain.

In Computer Science, Gruber [6] and Borst [7] were the pioneers in defining the notion of ontology. Later, Studer *et al.* [8] presented the most accepted definition of an ontology: "An ontology is a formal, explicit specification of a shared conceptualization". 'Conceptualization' refers to an abstract model of a knowledge domain that represents concepts and relationships between them. 'Explicit specification' means that the model should be represented using a coherent, unambiguous and structured language. 'Formal' implies that the ontology should be machine interpretable. 'Shared' means that knowledge represented in an ontology should define a common and consented vocabulary in a given domain, that can be shared across people and application systems.

Ontologies specify the semantics of an area of knowledge by defining concepts (or classes) that represent existing 'things' and the relationships among them, properties that each concept may have, constraints on concepts or properties, and axioms. An instance of a class is known as an individual. Different generality levels of ontologies can be defined [8], namely:

- Domain ontologies, which contain knowledge that is valid for a particular type of domain (*e.g.* medical, mechanic).
- Generic ontologies, which capture general knowledge about the world and, therefore, are valid across several domains.
- Application ontologies, which contain all the necessary knowledge for modelling a particular domain.
- Representational ontologies do not commit to any particular domain. Such ontologies provide representational entities without stating what should be represented.
- Metadata ontologies, which provide a vocabulary for describing the content of on-line information sources.

The process of building an ontology is not straightforward. Various approaches exist to guiding the ontology development. A general proposal to the process of building ontologies is given by Noy [9]:

1. Determine the domain and scope of the ontology.
2. Consider reusing existing ontologies.
3. Enumerate important terms in the ontology.
4. Define the classes and the class hierarchy (taxonomy).
5. Define object properties.
6. Define data properties.
7. Create individuals.

The semantic structure provided by ontologies differs from the formatting of information afforded by relational and XML databases, as they provide an objective specification of domain information, by representing a consensual agreement on the concepts and relations characterizing the way knowledge in that domain is expressed.

By providing a formal and hierarchically structured representation of an area of knowledge with commonly accepted definitions, ontologies minimize misunderstandings and miscommunications and make reasoning possible. By sharing the same underlying vocabulary, ontologies allow computer agents interoperation, as they can understand incoming requests and return the required knowledge. In addition, their semantic structure facilitates the process of precise knowledge indexing and retrieval.

A common understanding of a domain among people and application systems, fosters knowledge sharing and reuse not only between communities of experts, but also new learners [7]. In the present work, an ontology is used as a main mechanism to represent and share a domain knowledge of interest.

2.2 Ontologies in Knowledge Management

The role of an ontology in knowledge management is to facilitate the representation of knowledge, as it provides a common vocabulary about a particular domain of interest. By having explicit knowledge representation, an ontology provides information in machine-understandable form, which allows reasoning from a given set of facts and rules about the domain.

The potential advantages of using an ontology for knowledge management in the MACODA domain are obvious. An ontology is especially suited for representing and processing a large amount of information, providing the required capabilities to systematize the scientific knowledge produced in this field.

A considerable part of the MACODA knowledge domain can be represented by the means of formal logics (predicate logics), supported by OWL ontologies knowledge representation standards. For example, the following excerpt of the MACODA ontology represents a fragment of the MACODA taxonomy (hierarchy of classes/subclasses) by using *isA* type of relation, and *canSolve* type of relation to express which algorithms can succcessfuly be applied to which optimisation problems: JobShop *isA* SchedulingProblem; FlowShop *isA* JobShop; NSGA-II *canSolve* JobShop.

Additionally, we can add our own specific knowledge to the knowledge base, for example: mySpecificSchedulingProblem *isA* FlowShopProblem. When we represent this knowledge by the means of OWL ontologies, we can query/retrieve the explicit knowledge present in the knowledge base (*e.g.* 'what are the algorithms that canSolve JobShop ?' query would result in 'NSGA-II' algorithm), but also benefit from the formal logics-based inference done on the overall knowledge base by the querying engine (*e.g.* the result of 'what are the algorithms that can solve mySchedulingProblem ?' query would include 'NSGA-II' algorithm, because mySpecificScheduling problem is a special case of JobShop and NSGA-II *canSolve* JobShop).

The major benefits of using ontologies in knowledge-management are given by [10],[11],[12]:

- Ontologies improve knowledge search and retrieval by exploiting ontological background knowledge about the application domain.
- They provide a solid structure for information gathering, integration, and organization.
- Ontologies avoid semantic ambiguities of terms in a domain.
- They support knowledge visualization, valuable for analyzing big amounts of data with complex interconnections and finding useful knowledge.

2.3 Semantic Web

Knowledge representation by the means of OWL ontologies promotes a standardised and open representation of knowledge at the World Wide Web scale. A set of World Wide Web Consortium (W3C) standards, including the OWL (Web Ontology Language, intentionally named as OWL) standard, constitutes what has been identified as the Semantic Web or Web of Knowledge, in contrast to the Web of (HyperText Markup Language - HTML) Content.

Berners-Lee *et al.* [13] describe the Semantic Web as "an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" .

The Semantic Web (or Web 3.0) provides a set of standards and technologies that enable computers to understand and manipulate data in a similar way to humans. It connects pieces of information contained in a document or application, rather than documents or applications itself, *i.e.*, it is concerned with the semantics, not the structure of the data [14].

The collection, structuring and retrieval of data are enabled by a set of standards defined by the World Wide Web Consortium (W3C), that are used to formally represent metadata [15]. These technologies provide a common framework to share information across different applications and systems.

The architecture of the Semantic Web is presented in Figure 2. The lower layer standards of the Semantic Web Protocol Stack allows for resources identification and for basic forms of data representation, such as URI/IRI (Uniform Resource Identifier/Internationalized Resource Identifier) to identify OWL ontologies, classes, properties, etc., and XML (Extensible Markup Language) family of standards to define lexical and syntactical structures and annotations of OWL ontologies. The upper layer standards allow for more abstract concepts and relations representation and modelling of the knowledge domain of interest (representation of knowledge domain relations - meaning/semantics), such as RDF (Resource Description Framework) and OWL (Web Ontology Language) family of standards. Query languages for data and knowledge representation layers are also defined and available in the Semantic Web standards stack (*e.g.* SPARQL query language).

In this work we are focused on the OWL (Web Ontology Language), a knowledge representation language for ontologies. OWL has 3 sub-languages, namely OWL Full, OWL Description Logic (DL) and OWL Lite [17]. OWL DL is the most suitable for our work, due to its well balanced trade-off between language expressiveness and formal logics reasoning features. For the sake of space and

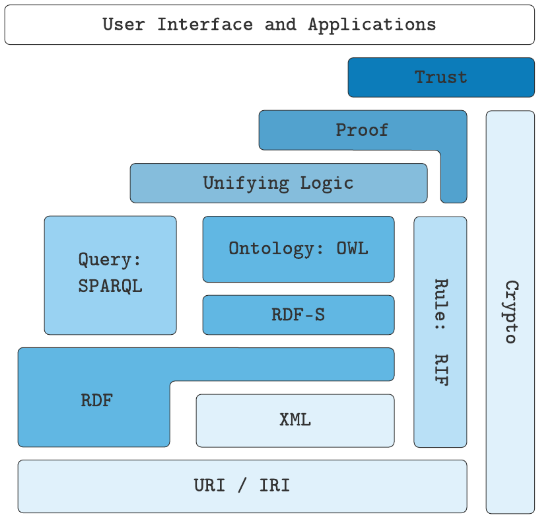

Fig. 2. Semantic Web Protocol Stack. [16]

clarity, we will not get into details on the differences between the OWL sub-languages. Some of the OWL DL features relevant for our work are [18]:

- It allows to set cardinality restrictions to restrict the number of distinct values a property may have (e.g. to express that one algorithm can solve one or more types of optimisation problems, one algorithm has one or more authors, one algorithm has only one creation year, etc..).
- It has the possibility to declare two classes to be disjoint (e.g. to express that optimisation problems are either combinatorial or continuous).
- It also allows to set classes as logical combinations (intersections, complements or unions) of other classes.
- It defines functional, reflexive, symmetric, inverse and transitive properties (e.g. expressing that relation *isExtensionOf* is a transitive relation, and *algorithmX isExtensionOf algorithmY*, and *algorithmY isExtensionOf algorithmZ*, allows the inference and querying engines to deal and process *algorithmZ* as an extension of *algorithmX*).

OWL is an ontology language for the semantic web with formally defined meaning, allowing the use of a reasoner that helps maintaining a consistent and correct classes' hierarchy, as well as formal logics inference and ontology querying, by the means of formal logics. Ontologies are OWL documents that can be published in the Web and may refer to or be referred from other OWL ontologies, enabling a richer integration, sharing and reuse of data.

In 2009, W3C announced a new version of OWL, named OWL 2. OWL 2 has a very similar structure to OWL but introduced new features, such as increased expressive power for properties, extended support for datatypes, simple meta-modeling capabilities, extended annotation capabilities, and keys [19]. Moreover, it introduced three new profiles, OWL 2 EL, OWL 2 QL, and OWL 2 RL [20]. OWL 2 EL is useful in applications employing large-scale ontologies. OWL 2 QL is aimed at applications that use very large volumes of instance data, where query answering is the most important reasoning task. OWL 2 RL is aimed at applications that require scalable reasoning without sacrificing too much expressive power.

An OWL ontology comprises classes, individuals and properties. A class may have subclasses that represent concepts more specific than the superclass. The hierarchy of classes, defines the taxonomy adopted in the ontology. Individuals represent class instances in the domain of interest. Properties are divided into two different kinds: object properties and data properties. An object property is a binary relation to relate classes or individuals, and a data property relates classes or individuals with a designed primitive data-type (*e.g.* integer, string, boolean) [21].

Various environments and tools for building ontologies are available, such as OntoStudio, Protégé, NeOn Toolkit, Swoop and TopBraid Composer. With the growing adoption of the OWL, Protégé has become the most popular and widely-used semantic web ontology editor.

Protégé [22] desktop is a free, open-source, java-based ontology editor and framework for building both simple and complex ontology-based applications. It is supported by a strong community of academic, government, and corporate users, who use Protégé to build knowledge-based solutions in areas as diverse as biomedicine, e-commerce, and organizational modeling. Protégé is fully compliant with the latest OWL specifications and supports collaborative ontology editing as well as annotation of both ontology components and ontology changes.

The Protégé editor screenshot in Figure 3 shows an excerpt of the MACODA ontology. It shows the look and feel of the Protégé editor Graphical User Interface (GUI), and three panels displaying the MACODA taxonomy (left panel) the instances of MACODA classes (middle panel) and relations of MACODA domain concepts (right panel). MACODA ontology visualisation, comprehension and modification can be performed by knowledge engineers, using the Protégé ontology editor, with the help and input provided by domain experts (*e.g.* MACODA researchers/experts), who usually do not have knowledge engineering or semantic web standards expertise.

WebProtégé [23] is a lightweight ontology editor and knowledge acquisition tool for the Web that uses Protégé infrastructure. It can be accessible from any Web browser, has extensive support for ontology collaboration, and a highly customizable and pluggable user interface that can be adapted to any level of user expertise. Protégé and WebProtégé are used in the present work for ontology design and edition.

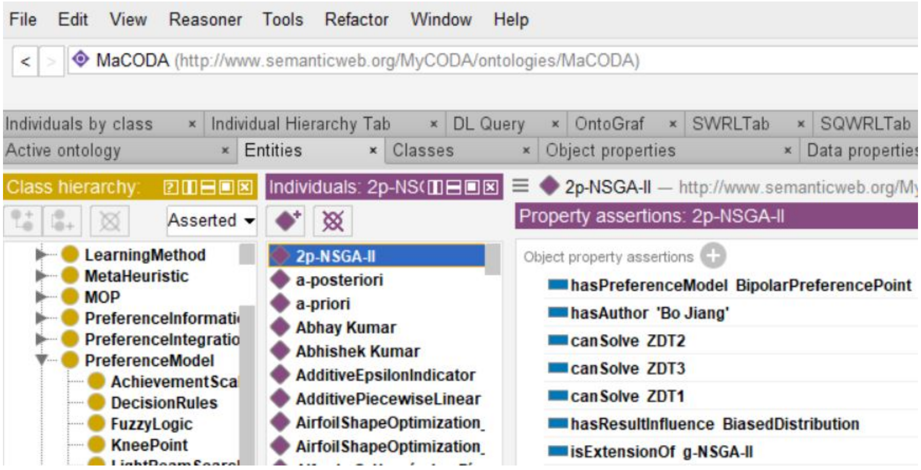

Fig. 3. View of MACODA Ontology with Protégé Ontology Editor GUI. [22]

2.4 Related Work

With the increasing trend in the number of publications about many and multi-objective optimisation, the necessity of systematizing the resulting knowledge in this domain has increased. The advantages of representing this knowledge in the form of ontologies were presented in a few previously published works, which served as basis of our work, namely:

– In [21] "Building and Using an Ontology of Preference-Based Multiobjective Evolutionary Algorithms", Li *et al.* propose an OWL ontology to model and systematize the knowledge of preference based multiobjective evolutionary algorithms (PMOEAs). This ontology aims to help researchers understanding, accessing, and analyzing methods, or identifying future research topics. This work also explains how to build/extend the PMOEA ontology and it presents simple and practical examples for various use cases. The PMOEA ontology was built with the help of Protégé Desktop and made public in WebProtégé.
– In [24] "An Ontology of Preference-Based Multi-objective Metaheuristics", Li *et al.* provide an overview of Preference-based multi-objective metaheuristics (PMOMHs) and propose a novel method to systematize and manage the current knowledge in this field. It also details the process of building the PMOH ontology using Protégé. This work extends and improves the PMOEA ontology, and use cases are provided to demonstrate the benefits of the ontology.
– In [25] "Presenting the ECO: Evolutionary Computation Ontology", Yaman *et al.* present an ontology exclusively focused in evolutionary computation, namely, genetic algorithms, genetic programming, evolutionary programming and evolutionary strategies. The ontology is used for strategies,

operators and parameter selection of evolutionary algorithms, to solve optimization problems, by means of evolutionary computation.
- In [26] "Evolutionary Computation Ontology: E-Learning System", an evolutionary computation ontology is designed and implemented by Kaur and Chaudhary, using Protégé. This ontology identifies the essential features of the subject Evolutionary Computation. It was designed for helping learners to enhance their knowledge level in the subject of evolutionary computation and it facilitates the learner to use the visualization feature and query feature.
- In [27] "A Survey of Diversity Oriented Optimisation: Problems, Indicators, and Algorithms", Basto-Fernandes *et al.*, provide an overview of the various concepts, methods and applications of diversity oriented optimisation. To represent the domain of diversity oriented search in a systematic way, an ontology was developed using Protegé with the intention to help users to classify algorithms correctly and find related work.

Although uncountable OWL ontologies exist in a huge variety of knowledge domains (e.g. Internet of Things, Cyber Security, etc.), very few ontologies were published in the multiobjective optimisation domain and none in the MACODA domain.

Because an (OWL) ontology is not an end by itself, additional artifacts, tools, and knowledge management process must be created, to ensure the use and enjoy the benefits of having a knowledge domain represented in an (OWL) ontology.

An organised and comprehensible knowledge management process (concerned with acquisition, creation, dissemination, sharing and utilisation of knowledge) must be defined for the MACODA domain, to deal with question such as: what is the relevant knowledge in this domain ?; how can it be used ?; in which context may it be used ?; are there restrictions to its use ?; who provided it ?; who created it ?; is it of high quality and reliable ?; how can it be searched, updated and harmonised among the MACODA research and practitioners community members ?; how is change management and quality management supported ?; etc.

Promoting a MACODA research community working environment where knowledge management is done in a research community-wise base, represents a critical success factor to foster MACODA knowledge domain creation and sharing.

MyCODA aims to be a community agreed and specific purpose platform to provide an important support for the MACODA domain knowledge management and curation, able to foster knowledge creation, sharing, use, and innovation in this domain. No other platform is known by the authors, with this purpose, features and knowledge domain.

Since one of the seven steps for developing an ontology suggested by [9] is "Consider reusing existing ontologies.", in this work, much attention was put on adopting the vocabularies and concepts presented in the mentioned ontologies, with adaptations designed to serve the purpose of the MACODA ontology and platform.

3 MyCODA Platform

3.1 Conceptual Model

This section presents an overview of MyCODA platform's features (available at http://macoda.club), use cases, best practices to be adopted in the ontology design and community-built ontology cooperation model. Among the main MyCODA platform features we highlight the following:

- Allow user registration and user authentication by email address and password.
- Allow users profiles management (visitor, learner, optimisation practitioner, expert, moderator).
- Generate a newsletter about MACODA upcoming events and new releases of MACODA ontology.
- Allow users to subscribe to a newsletter by providing their email address, as well as to notify MyCODA platform moderator about upcoming events and news on MACODA area, by the means of a web form.
- Allow users to search for and retrieve knowledge from the ontology by means of predefined, assisted creation or fully customized queries.
- Allow users to visualize and explore visually the MACODA ontology.
- Allow users to participate in discussions (forum) on MACODA ontology design updates/evolution and other MACODA topics.
- Allow users to propose MACODA ontology updates by filling a web form sent to MyCODA platform moderator.
- Allow users to propose topics for discussion, documents (journal and conference articles, tutorials, etc.), software frameworks and other MACODA related materials to be indexed/available in MyCODA platform, by means of a web form available for users to fill, and to be sent to the platform moderator.
- Allow MyCODA moderator to access users proposals, validate and perform changes in MACODA ontology and MyCODA platform contents.
- Allow MyCODA platform visits statistics by content and visitor origin.

We can foresee six different types of users , who have different perspectives of using the MyCODA platform :

- The *Visitor* is an unregistered user who accesses the platform and intends to explore it. This actor has the lowest level of privileges.
- The *Learner* corresponds to a student or a newcomer in the optimisation field, that aims to quickly get familiar with the domain and learn from the platform.
- The *Optimisation Practitioner* doesn't intend to contribute to the knowledge domain, he/she just needs to solve optimisation problems with the help of the platform.
- The *Expert* corresponds to an experienced optimisation researcher that can add knowledge into the platform and propose ontology changes or improvements, which will be then evaluated by domain experts.

- The *Moderator* is a special type of *Expert* that can perform additional actions, such as validating the *Experts* suggestions and updating the ontologies and the platform contents.
- The *Administrator* is responsible for users management, platform design, evolution and maintenance.

This cooperative community-built ontology model allows MACODA researchers and practitioners community to design and evolve a harmonised MACODA ontology in a well-structured, well-defined, systematic, formalized and standard way. MyCODA platform has the role of promoting knowledge management processes and practices in the MACODA domain (creating, acquiring, capturing, sharing, (re)using knowledge), by the means of customised tools, specifically designed for MACODA researchers and practitioners (*e.g.* querying and visualization tools).

In Figure 4 we show the MyCODA platform initial version (available at http://macoda.club), including:

- A (home) welcome area about the MACODA initiative launched in September 2019 at the Lorentz Center of University of Leiden - The Netherlands, and a short introduction to the MyCODA platform.
- An education section pointing to educational and training materials, and courses on MACODA.
- Events, pointing to recent and future events on MACODA.
- Resources, pointing to scientific and technical materials on MACODA (*e.g.* scientific papers and software of reference in this knowledge domain).
- MyCODA tools area, which provides a set of integrated tools to access, browse, visualize and query the MACODA ontology.
- A forum section to support experts and researchers knowledge sharing, suggestions and discussions on ontology corrections, improvements, vocabulary and knowledge harmonization.
- A frequently asked questions section.
- Contact/Join Us form.
- A list of MACODA ontology and MyCODA platform contributors, and their affiliations, in About Us section.

A dynamic and flexible support for the MACODA ontology visualization is provided by the means of the WebVOWL service [28], including a variety of visualization, interaction, filtering and statistical features about the ontology. A visual perspective of the MACODA ontology is shown in Figure **??** of this section.

Text tree-based browsing of the MACODA taxonomy is also available in MyCODA. Figure 5 shows a snapshot of the taxonomy exploration around the MetaHeuristic class branch.

Other types of ontology entities and relations can be explored, searched and related, by providing their names, patterns or filters on their names (see Figure 6).

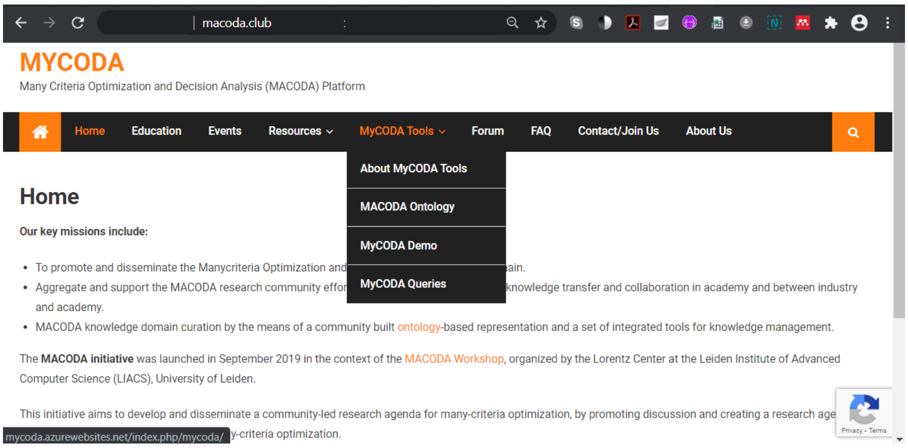

Fig. 4. MyCODA Web-Based Platform.

The MyCODA platform user will be able to run predefined queries, or build his/her own custom made queries, with MyCODA platform support. The user is not required any acquaintance with OWL ontology design or OWL query languages syntax. Figure 7 shows a predefined query example "What are the metaheuristics published after 2015 ?" to be run on the MACODA ontology.

Other predefined (or user custom made) queries could be: "What are the Python libraries implementing NSGA-III ?"; "What are the metaheuristics that were tested in the Knapsack problem ?" "What are the metaheuristics that were tested in the Knapsack problem having Java libraries implementations ?"; "Which order relations have been proposed to many-objective optimisation ?"; "Who are the researchers working both in decomposition-based and indicator-based metaheuristics ?".

More advanced uses of the ontology can be achieved by adding knowledge to the ontology in the form of necessary and sufficient conditions, and description logic rules. It means that experts knowledge inserted in the ontology, can result in great benefit for learners and optimisation practitioners, in cases where the queries involve complex levels of inference and a considerably high number of concepts and relations.

For instance, the query "What are the metaheuristics that can be used to solve the knapsack problem ?", might benefit from knowledge inserted into the ontology by a MACODA domain expert, stating that all heuristics that have been applied successfully to solve a combinatorial problem, can be used to solve any other combinatorial problem.

As Knapsack is a combinatorial problem, all metaheuristcs used in any combinatorial problem, are candidates to solve the knapsack problem.

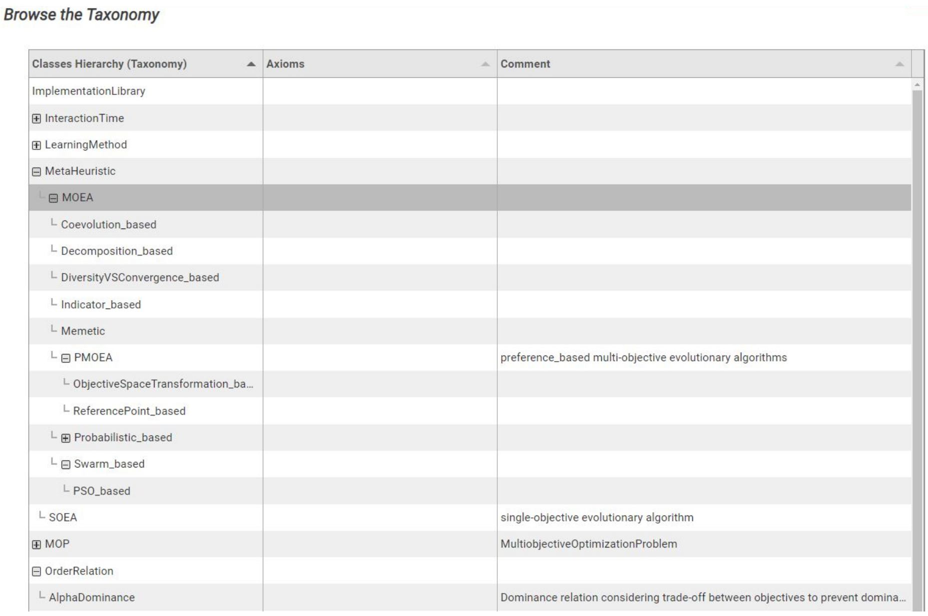

Fig. 5. View of MACODA Taxonomy in the MyCODA platform.

The reasoning ability provided by the OWL inference engine, is also expected to support MACODA domain experts in their search for research gaps, emerging research topics and research communities.

3.2 Ontology Design Best Practices

As a researchers and practitioners community scale effort, to represent, manage and access the MACODA knowledge domain in a systematised and standard way, a set of conventions and best practices (e.g. naming, design, commenting and annotation, etc.) must be adopted, to create a comprehensible, consistent, easy to maintain, easy to update and easy to query ontology, and also to avoid some common modeling mistakes [9],[29].

The conventions and best practices described in the following intend to provide the basic guidelines to be adopted in the construction and evolution of the MACODA ontology. We grouped them in three categories, general best practices, naming conventions and versioning conventions.

The following are some general best practices adopted for the MACODA ontology design:

- The ontology must be documented in sufficient quality and detail.
- Structure and vocabularies of existing ontologies should be reused as much as possible , to promote the web semantic view of a harmonised and integrated

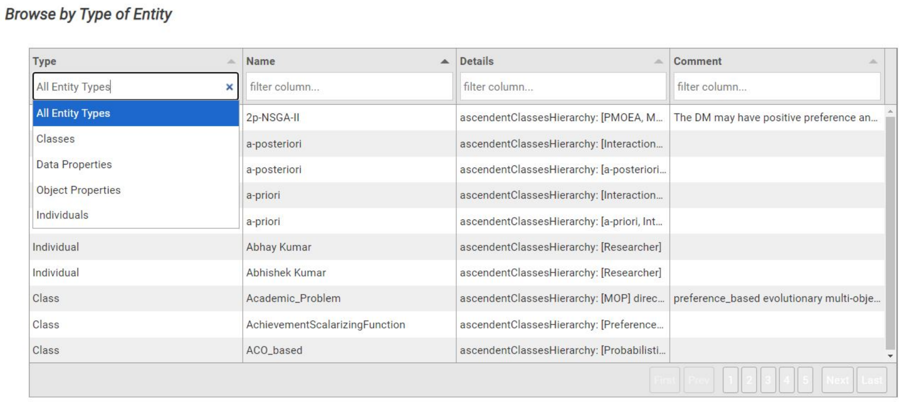

Fig. 6. View of MACODA Ontology Browsing in the MyCODA platform.

web scale knowledge base, improving the overall reasoning and querying potential.

- External ontologies should be mapped to new created ontologies, to increase the likelihood of sharing and interoperability , i.e., new ontologies should reuse and link their entities (classes, properties, etc.) to the corresponding entities of existing ontologies, stating the eventual synonyms and equivalence relations.
- Each class and property should have an URI to address identifier space.
- All classes and properties should have a definition.
- Predicates must be clear and precisely defined.
- The relationships in the ontology should be coherent.
- Disjoint classes should be used to separate classes from each other, where the logic makes sense and dictates.
- Property restrictions should be assigned sparingly and judiciously.
- Annotation properties should be used to promote the usefulness and human readability of the ontology.
- Information on how to contact the authors and how to contribute to the ontology should be available.

The following are some naming conventions adopted for the MACODA ontology design:

- Class names should start with a capital letter and should not contain spaces.
- When a class name contains more than one word, the words should be together and the first letter of each word should be capitalized, *e.g.* 'PreferenceModel'.
- Classes should be named as single nouns.
- Reserved words such as “class”, “property” and so on should not be added to class names.

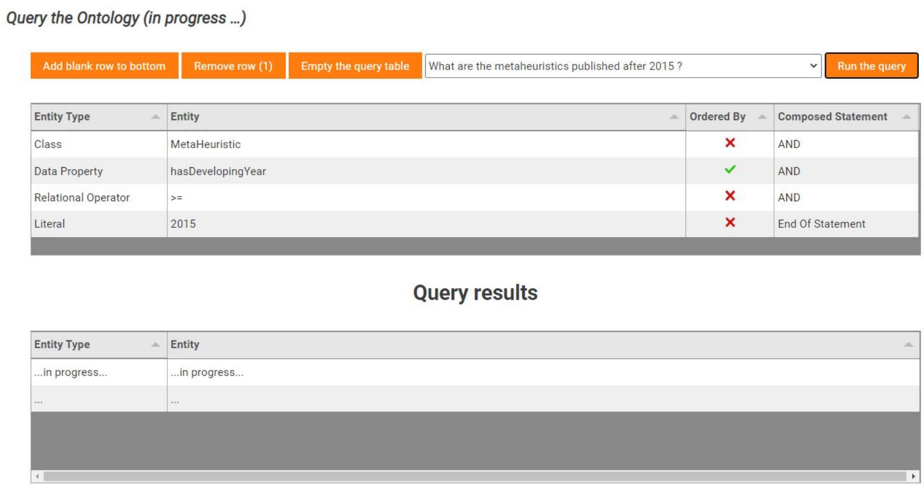

Fig. 7. View of MACODA Ontology Querying in the MyCODA platform.

- Abbreviations in classes names should be avoided. Exceptions are algorithms that are very commonly referred by their abbreviation, such as for instance NSGA-II or SMS-EMOA.
- Property names should start with lower case, have no spaces and have the remaining words capitalised, *e.g.* 'hasAuthor'.
- A 'has' or 'is' prefix should be added to property names.
- Properties should be named as verbs.
- The verb sense should be adjusted for inverse properties, for example, *Book hasAuthor John* would be expressed inversely as *John authored Book*.

In MyCODA ontology versioning will be adopted in order to control the following ontology evolution stages:

- Ontology initial version.
- Ontology version resulting from experts updates proposals.
- Ontology version resulting from MyCODA platform moderator validation.
- Ontology version resulting from MACODA researchers and practitioners community acceptance.

MyCODA relies on ontology storage at github.com platform, which offers distributed version control and access control. MACODA ontology updates proposed by the research and practitioner community, using the MyCODA platform, will follow a validation and verification lifecycle, according to the progressive acceptance of the proposal.

4 Conclusions and Future Work

This chapter gave a brief introduction of domain knowledge management using ontologies.

It also introduced an initial version of the MACODA ontology that summarizes current knowledge in the field of Many-Criteria Optimisation and Decision Analysis.

MyCODA platform content and features were presented, and highlighted the role it may have for researchers, practitioners and learners in MACODA scientific knowledge management.

Ontology design best practices and tools were suggested, and a collaborative ontology development model was proposed.

MACODA ontology and MyCODA platform will be maintained and further enriched by the authors. All researchers, experts, practitioners and learners are encouraged to contribute and keep MyCODA as a knowledge repository of the progress made in this emerging scientific field.

References

1. P. Busquin and Others, "Third european report on science & technology indicators: Towards a knowledge-based economy," tech. rep., European Commission, Brussels, Belgium, 2003.
2. H. Scarbrough, J. Swan, J. Preston, I. of Personnel, and Development, *Knowledge Management: A Literature Review*. Issues in People Management, London, United Kingdom: Institute of Personnel and Development, 1999.
3. A. Rodríguez-Pose, "Leveraging research, science and innovation to strengthen social and regional cohesion," tech. rep., European Commission, Brussels, Belgium, 2015.
4. M. R. Genesereth and N. J. Nilsson, *Logical Foundations of Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1987.
5. B. Smith, *The Blackwell Guide to the Philosophy of Computing and Information*, ch. 11, pp. 153–166. Hoboken, NJ, USA: John Wiley & Sons, Ltd, 2008.
6. T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquis.*, vol. 5, p. 199–220, June 1993.
7. W. Borst, *Construction of Engineering Ontologies for Knowledge Sharing and Reuse*. PhD thesis, University of Twente, Netherlands, 9 1997.
8. R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: Principles and methods," *Data Knowl. Eng.*, vol. 25, p. 161–197, Mar. 1998.
9. N. F. Noy, D. L. McGuinness, *et al.*, "Ontology development 101: A guide to creating your first ontology," Mar. 2001.
10. A. Abecker and L. van Elst, "Ontologies for knowledge management," in *Handbook on Ontologies* (S. Staab and R. Studer, eds.), pp. 435–454, Berlin / Heidelberg, Germany: Springer, 2004.
11. M. Park, K.-w. Lee, H.-s. Lee, P. Jiayi, and J. Yu, "Ontology-based construction knowledge retrieval system," *KSCE Journal of Civil Engineering*, vol. 17, pp. 1654–1663, Jan. 2013.

12. I. Jurisica, J. Mylopoulos, and E. Yu, "Using ontologies for knowledge management: An information systems perspective," in *Proceedings of the Annual Meeting-American Society For Information Science*, vol. 36, pp. 482–496, Information Today, 1999.
13. T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, 2001.
14. C. Semantics, "Introduction to the semantic web." "`https://www.cambridgesemantics.com/blog/semantic-university/intro-semantic-web/`", accessed 2020-05-06.
15. I. R. M. Association, *Information Retrieval and Management: Concepts, Methodologies, Tools, and Applications*. Hershey, PA, USA: IGI Global, 1st ed., 2018.
16. A. Walker, "A wiki for business rules in open vocabulary, executable english," *CoRR*, vol. abs/1103.0697, 2011.
17. W3C, "Owl web ontology language overview." "`https://www.w3.org/TR/2004/REC-owl-features-20040210/#s2`", accessed 2020-05-11.
18. A. Tatnall, *Web Technologies: Concepts, Methodologies, Tools, and Applications*. Contemporary Research in Information Science and Technology, Hershey, PA, USA: IGI Global, 2009.
19. W3C, "Owl 2 web ontology language document overview (second edition)." "`https://www.w3.org/TR/owl2-overview/`", accessed 2020-05-11.
20. W3C, "Owl 2 web ontology language profiles (second edition)." "`https://www.w3.org/TR/owl2-profiles/`", accessed 2020-05-13.
21. L. Li, I. Yevseyeva, V. Basto-Fernandes, H. Trautmann, N. Jing, and M. Emmerich, "Building and using an ontology of preference-based multiobjective evolutionary algorithms," in *Evolutionary Multi-Criterion Optimization* (H. Trautmann, G. Rudolph, K. Klamroth, O. Schütze, M. Wiecek, Y. Jin, and C. Grimme, eds.), (Cham), pp. 406–421, Springer International Publishing, 2017.
22. "Protégé." "`https://protege.stanford.edu/`", accessed 2020-05-13.
23. T. Tudorache, C. Nyulas, N. F. Noy, and M. A. Musen, "Webprotégé: A collaborative ontology editor and knowledge acquisition tool for the web," *Semant. Web*, vol. 4, p. 89–99, Jan. 2013.
24. L. Li, I. Yevseyeva, V. Basto-Fernandes, H. Trautmann, N. Jing, and M. Emmerich, "An ontology of preference-based multiobjective metaheuristics," 2017.
25. A. Yaman, A. Hallawa, M. Coler, and G. Iacca, "Presenting the eco: evolutionary computation ontology," in *Applications of Evolutionary Computation* (G. Squillero and K. Sim, eds.), Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), (Germany), pp. 603–619, Springer, 2017. 20th European Conference on the Applications of Evolutionary Computation (EvoApplications 2017), April 19-21, 2017, Amsterdam, The Netherlands, EvoApplications 2017 ; Conference date: 19-04-2017 Through 21-04-2017.
26. G. Kaur and D. Chaudhary, "Evolutionary computation ontology: E-learning system," in *2015 4th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions)*, pp. 1–6, 2015.
27. V. Basto-Fernandes, I. Yevseyeva, A. Deutz, and M. Emmerich, *A Survey of Diversity Oriented Optimization: Problems, Indicators, and Algorithms*, pp. 3–23. Cham: Springer International Publishing, 2017.
28. S. Lohmann, S. Negru, F. Haag, and T. Ertl, "Visualizing ontologies with VOWL," *Semantic Web*, vol. 7, no. 4, pp. 399–419, 2016.

29. O. S. Framework, “Ontology best practices.” ”https://wiki.opensemanticframework.org/index.php/Ontology_Best_Practices#cite_note-odp3-3”, accessed 2020-06-03.