Ontology Modeling Method of Complex Product Domain based on OWL and SWRL

Mintao Li, Jun Yu, Hongjun He

Wenzhou Polytechnic, Wenzhou Zhejiang, 325035, China

Abstract

With the increasing complexity of products, product design is no longer a single-domain and single-task design process, but a collaborative design process that combines multiple domains and multiple tasks. However, due to the heterogeneity of knowledge caused by different designers and disciplinary barriers in the process of collaborative design, the efficient promotion of collaborative design is seriously hindered. Therefore, it is urgent to build a good unified model of complex product design knowledge service to support the effective sharing of product knowledge. This paper studies the domain ontology modeling and reuse method of complex product design, analyzes the product design process and functional modules according to the integrated design method of systems engineering, and combines the domain ontology to model the product model collaboratively. The logical rules in the product knowledge service process are extracted through the SWRL ontology reasoning mechanism, and the semantic rules are formed by mutual mapping with the ontology model. On the basis of semantic rules, Pellet and Protégé reasoning engines are used to realize knowledge reasoning of complex products and improve the utilization and reuse rate of knowledge models.

Keywords

Ontology; OWL; SWRL; Complex Product Development.

1. Introduction

Complex products refer to products development with large design scale, complex composition, multi-coupling and multi-interaction. Specifically, it involves the coordination of various design resources, the integration of multi-disciplinary models, and the need to consider multi-stage designing process and full-life cycles. With the continuous improvement of digital manufacturing capabilities, large and complex products are faced with a series of problems such as high design indicators, high degree of integration, and difficulty in actual test simulation. At present, most design units still use single-domain and single-task product modeling and simulation technology. However, at this stage, complex product development needs inevitably require cross-domain and multi-disciplinary collaborative design. The product design field lacks cross-domain and multi-disciplinary designers. The co-design, co-simulation and co-optimization make it difficult to solve the core problems of co-design and modeling. Collaborative design and simulation of complex products is a complex system engineering involving many technical fields such as machinery, hydraulics, and electronic control [1].

Therefore, in order to realize unified normative modeling, unified data and model expression, unified model simulation and optimization of each component module of complex products, and unified model simulation and optimization within a unified design framework, support the collaborative design of complex products, and solve problems caused by single-domain product design. Model and data heterogeneity, simulation execution design and control difficulties, etc, Using SysML and ontology modeling technology to study the unified design knowledge modeling method of complex products [2], aiming to comprehensively obtain the design fields

ISSN: 1813-4890

of various disciplines in the collaborative design process of complex products [3, 4]. The model realizes the unified expression and effective sharing of data during the whole life cycle of the product, and eliminates "process barriers" and "information islands" [5].

2. Multidisciplinary Collaborative Modeling Method for Complex Products

2.1. Complex Product Design Stage Division Process

The design and development of complex products is a complex process that includes the entire design life cycle. Due to the various types of complex products targeted by different industries and fields, the complex functions to be implemented are also very different, resulting in the specific development process of each complex product are also great different. Zhang Feng [6] summarized the complex product virtual prototype design verification process and system engineering comprehensive integration method, and divided the complex product development process into four stages: demand prototype, functional prototype, performance prototype and implemention prototype. The specific division of complex product design and development process (taking vehicle system design as an example) is shown in Fig. 1:

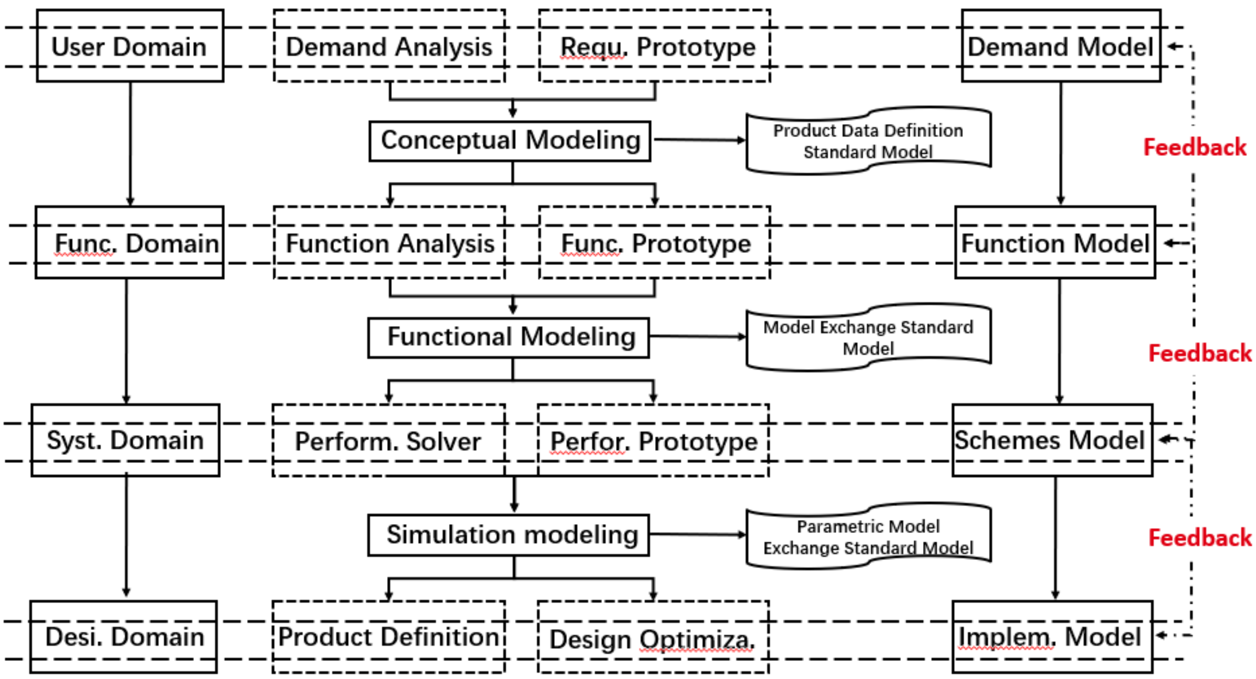

Fig 1. The division process of complex product design stages

(1) Demand prototype.

The demand prototype is designed and developed according to the overall design requirements of the complex product of the vehicle. In vehicle design, the overall design indicators, overall parameter design and overall layout design, body-in-white design, bearing chassis design and electronic control system design of the vehicle need to be determined in the design requirements stage. This process is to build a product concept description model that computers and designers can understand together, and it is the first-level design abstraction of the product system. The general designer and the designers of various disciplines need to convey the domain product knowledge to the product designers with the help of the requirement concept model prototype.

(2) Functional prototype.

The functional prototype is based on the product structure function. In this stage, the design parameters of each functional module, such as the engine system, the chassis system and the electronic control system need to be functionally designed and improved, and then the product functional models of different levels and granularities could be established. The functional prototype will establish various functional modules to achieve various design indicators of the demand prototype, aiming to meet the overall design requirements of complex products.

(3) Performance prototype.

The performance prototype is the detailed design stage of complex products. This stage is the specific design process stage of completing vehicle 3D solid modeling and assembly, product design performance parameter calculation, motion simulation and path planning, cruise accuracy and performance simulation, and reliability checking. This design stage involves the process of product modeling and simulation. The aforementioned demand prototype and functional prototype must eventually be transformed into the simulation model for performance analysis, which is an important intermediate link from the product function to the specific realization of the product.

(4) Implemention prototype.

The design of the implemention prototype marks that the development of complex products has reached the final design stage. In this stage, co-simulation and design optimization will be carried out collaboratively through different discipline models, and then comprehensive decision-making and evaluation of the final molded product will be made, and finally the decision-making evaluation results and design requirements will be fed back. The indicators are compared, and further modifications are made to the design.

The above four stages all have complete design structure models. The core of complex product collaborative design is to build a unified design reuse template for these four stages to improve the efficiency of product design.

2.2. Collaborative Knowledge Modeling Method for Complex Products

The research and development of complex products needs to rely on multidisciplinary designers to work together, which inevitably leads to the heterogeneity of product design knowledge. Because different designers have different disciplinary knowledge backgrounds, their product knowledge modeling methods and operation methods are different, and they also have different understandings and descriptions of the same product knowledge concept, which leads to the inability to effectively support knowledge sharing and interoperability, hindering the process of collaborative design for complex products across regions and with multiple disciplines [7].

The essence of collaborative design and development of complex products is cross-regional and multi-disciplinary modeling and simulation. There are a large number of design models between disciplines, so it is urgent to solve problems such as integration, sharing and data exchange between different models. The improvement of model integration and data exchange efficiency will significantly improve the efficiency of collaborative design and modeling of products. The collaborative knowledge modeling technology based on ontology metamodel can effectively carry out unified semantic modeling management for complex product system design, and realize product abstraction and analysis of different discipline models at a higher level. The knowledge model of each level for complex product is described by the relation set that constitutes the product model. As shown in Fig. 2, it is a multi-disciplinary and multi-level design model of the vehicle, which is expressed as a set:

$$Q = \{M, R_m\} \quad (1)$$

In the formula,

$M = \{m_{11}, m_{21}, m_{22}, \cdots, m_{ij}, \cdots\}, R_m = \{R_{m,12}, R_{m,23}, \cdots, R_{m,i(i-1)}, \cdots, R_{m,j(j-1)}, \cdots\}$

M (Models) represents the set of all models that make up a complex product, m_{ij} represents the jth model of the ith layer; R_m represents the set of association relationships between the models at each level, $R_{m,j(j-1)}$ represents the set of associations between the (j-1)th layer and the (j)th layer.

A complex product can be divided into multiple subsystems from the module division scheme of the complex product [8], and each subsystem contains multiple components and parts. These systems, components and parts together form a multi-granularity level decomposition model. Ontology-based complex product systems can be expressed as:

$$Onto_{Sys} = \cup_{i=1}^{n} \cup_{j=1}^{P_i} T_{ij} = \cup_{i=1}^{n} \cup_{j=1}^{P_i} \cup_{k=1}^{m} Sub_{ijk} + \cup_{i=1}^{n} \cup_{j=1}^{P_i} M_{ij} + \cup_{i=1}^{n} \cup_{j=1}^{P_i} O_{ij} \quad (2)$$

In the formula,

n——represents that the complex product system model has n levels;

P_j——represents the number of metamodels contained in the jth layer of the hierarchical system model;

T_{ij}——represents the jth model at the ith granularity level;

Sub_{ijk}——represents the jth model of the ith level in the kth domain;

M_{ij}——represents the jth model of the ith layer;

O_{ij} —— represents the jth metamodel at the ith granularity level.

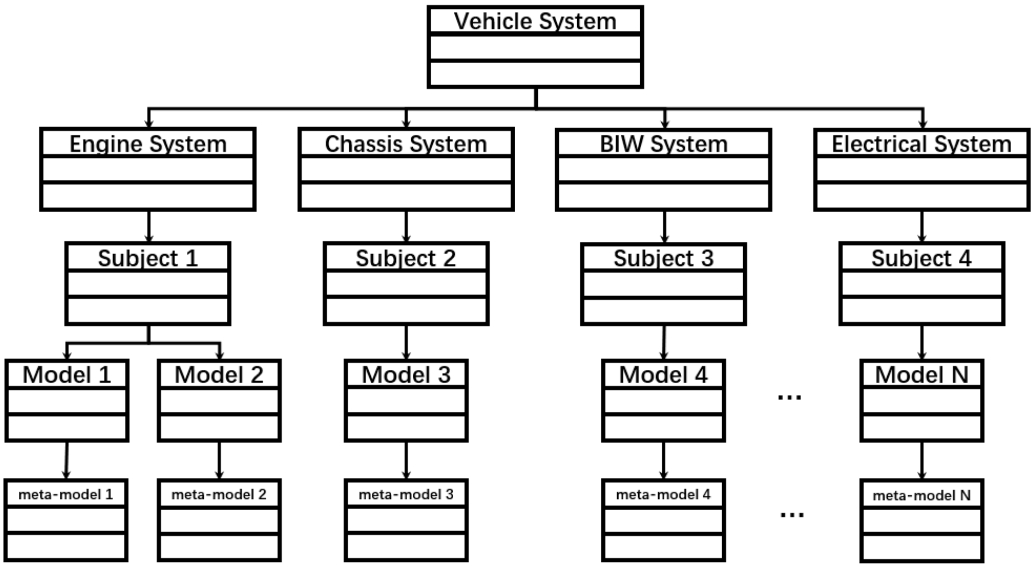

Fig 2. A Metamodel-Based Multidisciplinary System Decomposition Model

2.3. Construction Method of Complex Product Ontology Metamodel

The complex ontology meta-model has complete structural functions, parameter transfer methods, open interfaces, and supports high reuse and interoperability [9]. This paper then uses the internal module diagram in the SysML diagram model to build the standard structure of the metamodel, as shown in Fig.3(a). The SysML-based metamodel construction method first defines the input and output interfaces of the metamodel parameters and the external domain interface, and encapsulates the specific operation methods of the model in the metamodel. These methods include model simulation algorithms and model state transition methods. The coupling of these metamodels can build heterogeneous models of different disciplines. The coupling process of defining the model with the module definition diagram in the SysML diagram model is shown in Fig. 3(b). The meta-model coupling system forms the coupling model through the mapping of the meta-model invocation interface, and also opens the coupling model domain interface and input and output interfaces for model invocation.

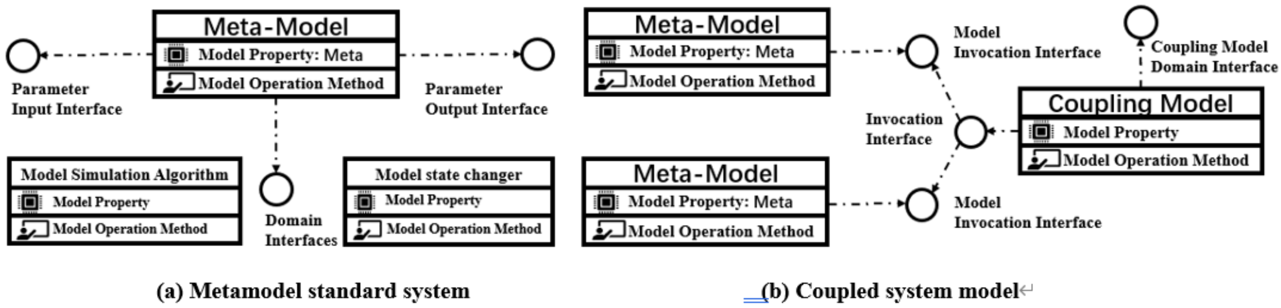

Fig 3. Metamodel standard system and coupled system model based on SysML

The ontology metamodel defined in this study is described by the following 6-tuple:

$$\text{ONTOLOGY\_Meta-Model} = \{\text{Class, Attributes, Constraints, Relation, Methods, Interface}\} \quad (3)$$

In the formula,

(1) Class represents the collection of objects contained in the metamodel;

(2) Attributes represents the set of attributes possessed by the meta-model;

(3) Constraints represents the constraint set of meta-model attributes;

(4) Relation represents the set of mapping relationships between metamodels;

(5) Methods represents a working set of meta-model operation methods;

(6) Interface represents the set of external interfaces of the meta-model for mutual calls between models.

3. Case Analysis

Based on the above analysis of the multidisciplinary collaborative knowledge modeling method for complex products, this section uses the Protégé ontology modeling tool to design and build the vehicle ontology metamodel library.

3.1. Vehicle Domain Ontology Design

3.1.1. Domain Ontology Analysis

The functional design and development of modern vehicles is often done by large teams from many different disciplines in automotive engineering. The content of automobile design includes overall vehicle design, assembly design and parts design. The overall design of the whole vehicle is also called the general layout design of the vehicle. Its task is to make the designed product meet the requirements of the vehicle parameters and performance indicators specified in the design task book, and decompose these vehicle parameters and performance indicators into related total vehicle parameters and performance indicators. into the parameters and functions. As shown in Fig.4, it is the effect diagram of a certain company's model vehicle and the schematic diagram of the mechanism function.

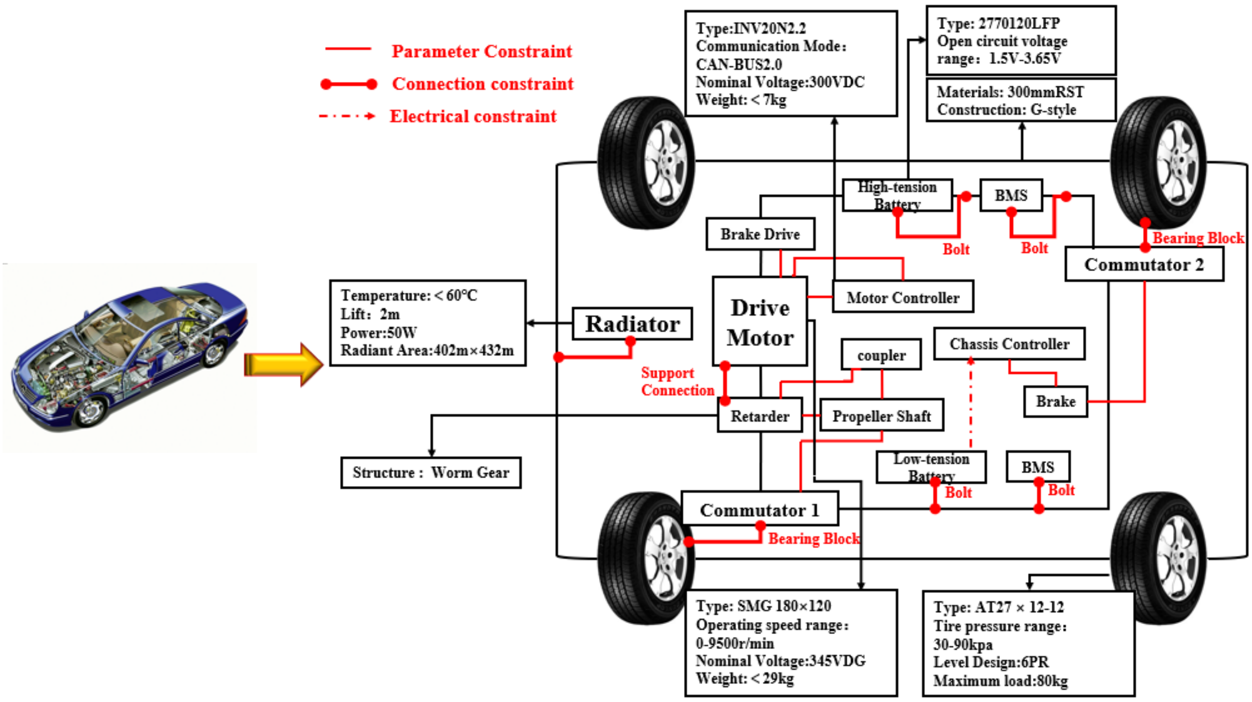

Fig 4. A company's model vehicle renderings and mechanism diagrams

Based on the design model of this model of vehicle, this paper establishes a complex product ontology model of the vehicle, and determines the core semantic scope and boundary of the research.

3.1.2. Domain Ontology Scope Determination

According to the above analysis, the vehicle system is an extremely complex, multi-disciplinary coupled complex system. The system design includes body-in-white, power drive, electronic control and energy supply, etc. It is composed of many subsystems, and these systems are also formed by the coupling of multiple disciplines. According to the whole life design cycle of vehicle design, the vehicle design domain can be decomposed into body-in-white design, power drive design, electronic control design and energy supply design. From the analysis of the above modeling requirements in the design and development of complex products, it can be seen that the establishment of multidisciplinary domain ontology for complex products requires model designers and domain experts to convert domain model knowledge into OWL language models described by Web ontology. The design of OWL language model can not only present the semantic information of complex product design to designers, but also make the Web content created readably and understandably in the machine, whose expressive ability is stronger than XML, RDF and RDF Schema, and can be accessed on the network side with a unique identifier. At the same time, OWL also has the ability to collect distributed design knowledge to achieve collaborative sharing, which can effectively support collaborative design sharing of complex products.

3.2. Design of Complex Product Domain Ontology

3.2.1. Domain Ontology Terminology

A vehicle is a complex design product that contains a large set of terms and involves different disciplines, requiring collaborative modeling across disciplines. As shown in Table 1 below, it is a semantic classification table for vehicle design, which lists the semantic knowledge of design module objects and module parameters.

ISSN: 1813-4890

Table 1. Vehicle Glossary

Classification	Terminology
Product Type	Vehicle, Ship or Drone
Top System	BIW System, Power Drive System, Electronic Control System and Energy Supply System
Performance Parameter	Dynamics, Economy, Handling Stability, Braking, Ride Smoothness and Passability
Parts	Chassis, Transmission System, Engine, Radar, Laser Rangefinder, Power Supply Unit, Communication System
Test Parameters	Maximum Speed, Obstacle Clearance Height, Control Accuracy, Tightness, Mileage

3.2.2. Define the Term Hierarchy

Vehicle design is composed of a variety of heterogeneous models from different disciplines, different professional fields, and different design departments. Table 1 shows that the module models of each level are obtained through hierarchical division, and then the vehicle model is designed and modeled through the semantic expression of these models and module relationships. This paper takes the design of a certain type of vehicle as an example, starts from the top-level system and divides it into a meta-model, and uses SysML to express the system model structure, as shown in Fig.5. The design model interface in vehicle design includes two types of interfaces: one is the discipline input/output interface used to transfer parameters and calls between discipline models, and the other is the design task interface used to describe model activities.

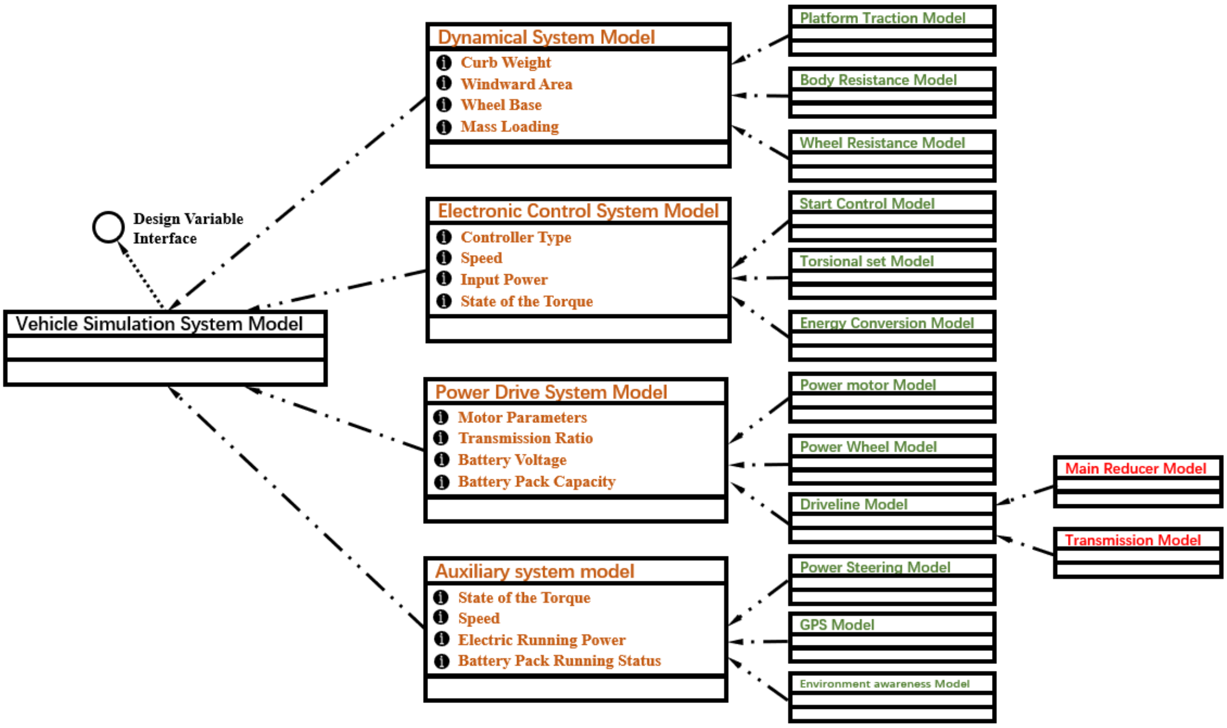

Fig 5. Vehicle Hierarchical Module Structure Model

3.2.3. Ontology Attribute Expression

In domain ontology, properties are the bridge to describe the relationship between two classes. The definition of properties in domain ontology is to add various properties to the classes of vehicle design, such as the term set shown in Table 1. These terms are used to describe a set of properties for a certain type of vehicle. For example, a certain type of vehicle has properties such as top-level systems, components, and basic information, and the Object Properties object properties are just used to characterize these properties. As shown in the following code, the ontology extends the hierarchy through the transfer of attributes, generally through <? ,Attributes,? > the form of triples, the inheritance relationship between layers can be achieved through the side relationship of attributes. By combining the above-mentioned attribute definitions of triples, the ontology representation of vehicle design is given. A small part of the code is as follows:

/code 1 triple definition of vehicle's ontology object attributes/

<"Vehicle", "hasPart", "Engine System">

<"Vehicle", "hasPart", "Chassis System">

<"Vehicle", "hasPart", "BIW System">

...

<"Electronic Control System Model", "hasPart", "Start Control Model">

<“Electronic Control System Model”, “hasPart”, “Torsional set Model”>

<“Electronic Control System Model”, “hasPart”, “Energy Conversion Model”>

...

<"Chassis System", "hasPart", "Transmission System">

<"Chassis System", "hasPart", "Dynamic System">

<"Chassis System", "hasPart", "Steering System">

3.3. Ontology Design Interface

The construction of vehicle domain ontology relies on the ontology building tool protégé, which is a good tool for building ontology with its friendly visual interface and simple operation performance. As shown in Fig.6 is the main body of the vehicle.

In protégé, this paper divides the structure through the vehicle module hierarchy [8], which can express and classify the knowledge of design activities in the relevant subject areas in the design structure matrix, and sort out the correlation between the knowledge, which is foundation for the sharing and reasoning of the knowledge ontology in the subsequent fields [9, 10]. Select the Classes panel in Protégé to display the vehicle's class hierarchy as shown in the left half of Fig.6.

Using the relationship visualization provided by the OntoGraf function module of the Protégé tool, the vehicle ontology relationship network shown in the right half of Figure 6 is generated. The tool also provides a knowledge field query function, as shown in Fig.7, enter "Power_System" in the Search space to retrieve the superior and subordinate terms of the field, and designers can query according to their own needs.

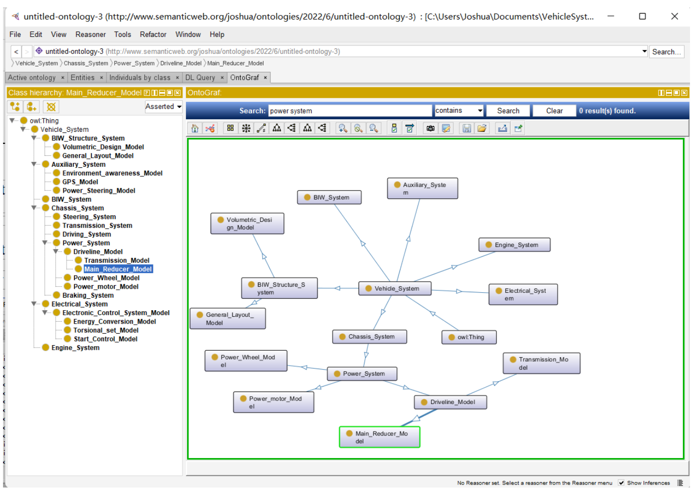

Fig 6. Vehicle Domain Ontology Relation Network

ISSN: 1813-4890

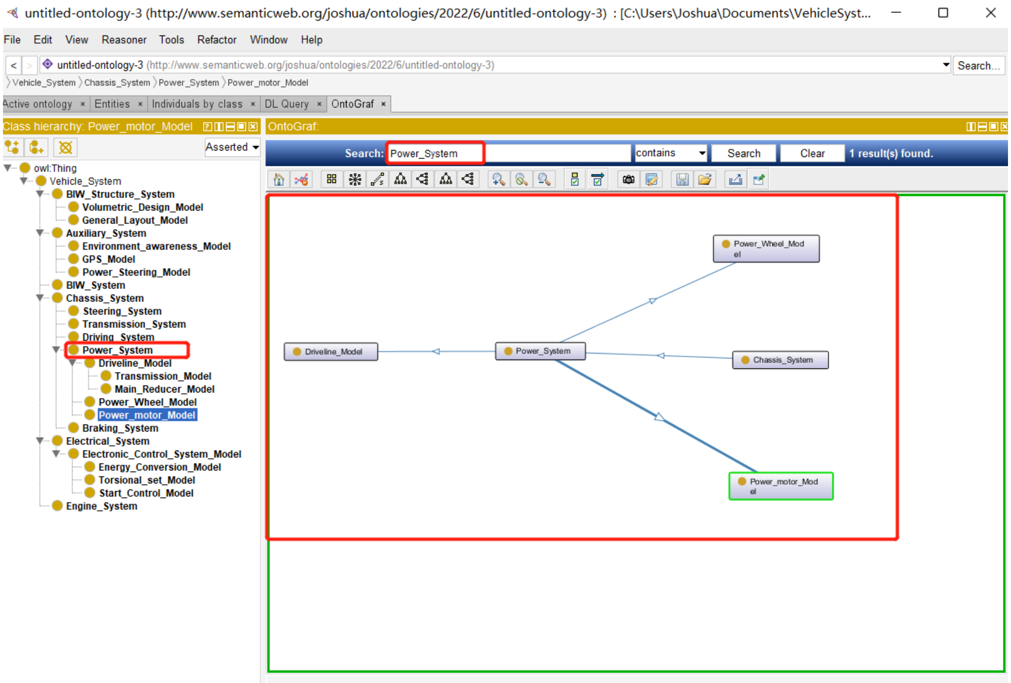

Fig 7. OntoGraf's search function interface

4. Conclusion

Aiming at the deficiency of the lack of collaborative shared semantic model in the current vehicle system design, this paper focuses on the collaborative modeling and knowledge reasoning methods of product systems. On the basis of analyzing the functional structure design process and hierarchical module division model of complex products, an ontology-based collaborative concept modeling method is established, which provides a theoretical basis for cross-disciplinary collaborative modeling of complex vehicle products. And apply SysML to describe the architecture of the vehicle, analyze the construction process of the ontology model, and use the protégé tool to build the vehicle domain ontology model library. At the same time, the model relational database and rule base are expressed, which can effectively support the explicitization of product design knowledge. Finally, in the future, the transformation of domain ontology to relational database can lay a solid foundation for the realization of ontology in the system.

Acknowledgments

The authors are grateful to the anonymous reviewers for their comments, which have helped to improve this paper. The authors acknowledge supporting funds from horizontal projects of cooperation between Wenzhou Polytechnic and Wenzhou Xinmo culture and creativity Co., Ltd (Project No. H2022007), the third batch of scientific and technological innovation team projects of Wenzhou Polytechnic (Project No. WZYKJTD2022002), research project on teaching innovation team of national vocational education teachers (Project No. SJ2020010102).

References

[1] Al-Fedaghi S. Conceptual Modeling of Time for Computational Ontologies[J]. 2020.

[2] Wu L, Yan Y, Gao F, et al. Research on Modeling and Verification Methods for Embedded Software Systems Based on SysML: 2019 IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), 2019[C].

[3] Yu J, Ming Z, Wang G, et al. Ontology-Based Unified Representation of Dynamic Simulation Models in Engineering Design: ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2018[C].

[4] Yu J, Ming Z, Wang G, et al. Integrated Simulation Modeling Method for Complex Products Collaborative Design Using Engineering-APP[M]. Beijing Institute of Technology No. 5 S. Zhongguancun Street Haidian District Beijing 100081 China.

[5] Kim W. Classifying Schematic and Data Heterogeneity in Multidatabase Systems[J]. IEEE Computer Society Press, 1991.

[6] Zhang Feng, Xue Huifeng, Xu Yuan. Ontology-based collaborative modeling method for aerospace product performance prototype[J]. Computer Integrated Manufacturing System, 2016,22(8):13.

[7] Liu H, Chen G. Research on Information Integration for Multidisciplinary Collaborative Design of Complex Customized Product[J]. Procedia Engineering, 2011,23(none):296-302.

[8] Jun Y, Mingzhe Z, Jiayi C, et al. Clustering Collaborative Method for Complex Product Hierarchical Modules Based on HDSM and SysML[J]. International Journal of Science, 2022,9(3):163-173.

[9] Ernadote D. Ontology reconciliation for system engineering: 2016 IEEE International Symposium on Systems Engineering (ISSE), 2016[C].

[10] Denham B, Pears R, Naeem M A. HDSM: A distributed data mining approach to classifying vertically distributed data streams - ScienceDirect[J]. Knowledge-Based Systems,189.