Title:
Ontology-based Assembly Process Modeling with Element Extraction and Reasoning

Authors:
Qiangwei Bao, baoqiangwei@buaa.edu.cn, Beihang University
Gang Zhao, zhaog@buaa.edu.cn, Beihang University
Yong Yu, yuyong@buaa.edu.cn, [a]Beihang University, [b]Beijing Engineering Technological Research Center of High-Efficient & Green CNC Machining Process
Sheng Dai, daisheng@buaa.edu.cn, Beihang University

Keywords:
Assembly Process, Ontology, Elements Extraction, Elements Reasoning

DOI: 10.14733/cadconfP.2021.1-5

Introduction:
With the research and application of cloud manufacturing [1], [7], a standardized expression method is urgently needed to model manufacturing processes and capabilities. Meanwhile, the requirement is particularly imperative in assembly processes modeling, for that there are serious irregularities in the formulation and record of assembly processes due to the diversity of assembly modes in assembly workshops [2]. The technology of semantic web with mature development provides a suitable modeling method for the assembly processes, considering the extraordinary description and sharing abilities of semantic web languages in modeling. Furthermore, ontology web language (OWL) has been considered as the representative language of semantic web for the rich expression methods of individuals and relationships. The modeling of assembly process with ontology is of great significance to solve the problems of non-standard and poor sharing in traditional assembly process modeling, which is the focus of this research.

In this manuscript, the ontological approach of assembly process modeling is given. Firstly, the assembly processes are divided into worksteps as the modeling units and expressed with the concept of event ontology consisting of six elements. Secondly, the assembly documents are analyzed and decomposed, in which the elements are extracted with natural language processing (NLP). Finally, ontology-based reasoning is used to associate the assembly resources that are not directly indicated in the process documents to further complete the filling of assembly event elements

Modeling of Assembly Process:
Generally, assembly process is formulated and recorded with assembly documents by process designers. The assembly document of certain task expands in order of assembly operations, in which the assembly worksteps are defined corresponding specific operations to guide the assemblers to complete the assembly task in accordance with certain operational orders and requirements.

An assembly workstep can be seen as an event that contains various information related to the operation. In the conceptual framework of event ontology, an event e can be defined as a six-tuple [4], as shown in Eqn. (2.1).

$$e = (A, O, T, V, P, L) \tag{2.1}$$

The elements in the six-tuple are known as the event factors. In the scene of describing an assembly workstep, the elements of an event should cover various kinds of information of the corresponding workstep to guide the assembly operation of this step instead of document and record all resources and data related to the operation. The explanation of the six-tuple of assembly workstep events is shown in Tab. 1.

Factors	*Paraphrases*	*Explanations in assembly workstep*
A	A happening action set	The possible operations performed in the assembly workstep
O	Objects that participate in	The associated assembly resources in this workstep
T	Period of time	The time information recorded in the process of workstep execution
V	Environment of event	The environment information recorded in the process of workstep execution
P	Assertions on the procedure of actions execution	Pre-condition, post-condition and intermediate assertions in the process of workstep execution
L	Language expressions	Description language of assembly workstep

Tab. 1: The explanation of the six-tuple of assembly workstep events.

In the modeling of assembly workstep events, each factor represents a set of information of a certain type for the assembly worksteps. Detailed discussion of factors are as follows:

- *A* represents the actual operation performed in the assembly workstep by assemblers. Assembly operations can be expressed by a variety of actions, such as combining, constructing, cleaning, welding, etc. The operation is the most important part of an assembly workstep event, which is the core of the other elements.
- *O* represents the associated assembly resources in this workstep. Generally, the assembly resources include personnel, materials, tools, equipment and environment in the workshop. The workstep event should contain the use of resources to form an effective carrier for preserving the manufacturing historical data.
- *T* and *V* represent the time and environment information in the assembly workstep events respectively. With the execution of worksteps, the time and environment information will be generated naturally, recorded and transmitted to the event models.
- *P* represents the pre-condition, post-condition and intermediate assertions in the process of workstep execution. The conditions factors, e.g., the sequence of worksteps, the requirements of inspection results, the delivery status of materials, etc. can be modeled in this part to ensure that the assembly worksteps are executed in an orderly and quality manner.
- *L* represents the description language of assembly worksteps. The modeling and corpus extraction methods differ on account of different language expressions, including core words sets, core words expressions and core words collocations.

To construct the ontology of assembly worksteps, hierarchical relationships among classes need to be established firstly, as shown in Fig. 1. One major branch represents the inheritance relationship among assembly task information, assembly workstep information and elements describing worksteps. On the other hand, assembly resources of the assembly site should be ontologically modeled according to the types to realize the association between worksteps and resources. It is worth mentioning that the types of assembly resources may vary greatly in different workshops, the modeling method in Fig. 1. provides only one feasible example. For a variety of assembly resources, ontology modeling should be carried out in a similar way according to the actual situation of the specific workshop.

As mentioned above, there are six elements involved in the assembly workstep events. Some elements are recorded by data properties of ontology classes, including action information, time information, environment information, assertion information and language information by predefined ranges of data properties for the corresponding classes. Object information of worksteps reflects in the object properties of specific classes. Assembly workstep events can be semantically expressed through ontology by instantiating worksteps and resources along with the assignments of data and object properties to instances.

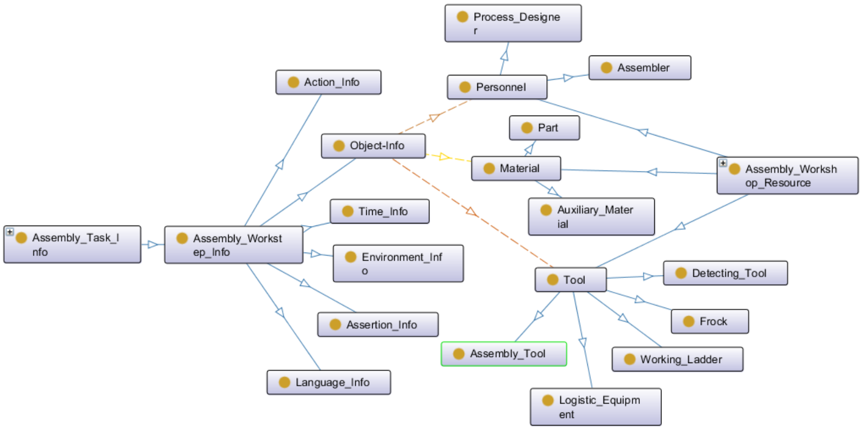

Fig. 1: The ontology structure of assembly workstep events.

Elements Extraction of Workstep Event:

In the case of assembly operation, assembly worksteps are usually in the form of process document with sequence number, workstep name, content, etc. Some process documents also include tools, equipment, rated working hours and other information listed separately, which can be directly extracted to fill in the workstep events. However, for the documents where all the information is gathered in the content, the workstep event elements should be extracted using NLP technology.

NLP is a bridge between machine language and human language to achieve the purpose of man-machine communication. In the case of elements extraction of workstep event, the human language refers to the contents manually specifying the operations by process designer in process documents. On the other hand, the machine language refers to the semantic six-tuple representing the assembly workstep events. Among the functions that NLP technology can achieve, natural language understanding (NLU) plays a key role in the transformation from assembly process documents to workstep events. Considering the characteristics of assembly process and the implementation method of NLU, the steps of corpus preprocessing [5] are shown in Fig. 2.

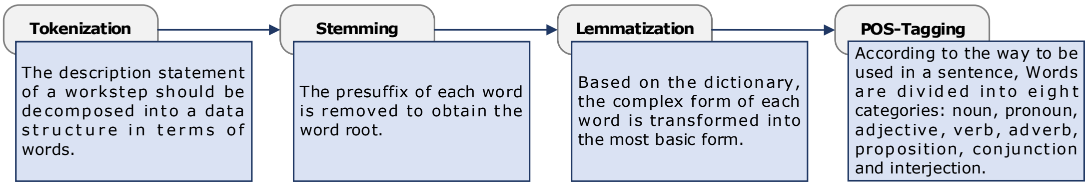

Fig. 2: The steps of corpus preprocessing.

After the corpus preprocessing of process document, each workstep description can be broke up into a data set containing words and phrases. A corpus containing all the assembly resource names in the workshop is constructed as an inherent attribute in the assembly workshop. Meanwhile, the possible assembly operations in the workshop can also be recorded and constitute another action corpus. Through traversal and comparison, the overlap part between the two corpus and the workstep data set can be located. The overlap part may contain several resources and actions, which can be regarded as the associated resources and actions of the certain workstep to fill in the workstep event. It should be noted that corpus of actions and resources should be iterated and updated to ensure that the real-time status of the workshop is better reflected.

For other elements in the workstep events, appropriate data can be obtained from multiple sources. Physical elements such as time and environment elements can be obtained through system time and environment sensor data at specific time points. The condition elements are mainly reflected in the sequence of assembly steps. The description language elements record the language used in the process document.

Elements Reasoning of Workstep Event:
The process documents may not contain the names of all the resources associated with the assembly worksteps, leading to the incompleteness of assembly workstep elements. To deal with this problem, it is necessary to construct an expert system for workstep event elements reasoning, as shown in Fig. 3.

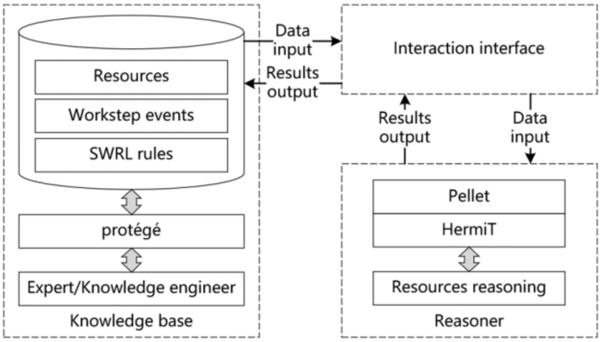

Fig. 3: The expert system for workstep event elements reasoning.

The rules used in reasoning are mainly derived from the knowledge in the assembly workshop, mainly including the following aspects:

- The use of some assembly resources requires the support of other resources. For example, if welding equipment is needed, appropriate welding consumables must be required.
- Certain operations require equipment or tools supports. For example, if heavier materials are to be transported, logistics equipment like carts, AGVs, or cranes are necessary under their respective applicable conditions.
- The locations of the execution of assembly operations are determined by the resources associated with the operations in some cases. For example, if a workstep requires a fixed specialized frock, the location where the workstep is executed will depend on the location of the frock.

Each rule can be expressed using semantic web rule language (SWRL) and applied to ontology reasoning. According to the SWRL syntax, OWL semantics can be abstracted into tuples in the form of formulas with a method of data type mapping [1], as shown in Eqn. (4.1).

$$I = (R, EC, ER, L, S, LV) \tag{4.1}$$

Thereinto, *R* stands for a group of resources. *LV* stands for a group of literal values in *R*. *EC* stands for the mapping between classes and data types along with the mapping between *R* and *LV*. *ER* stands for the mapping of binary relation between properties and *R*. *L* stands for the mapping between typed literal and elements in *LV*. *S* stands for the mapping between individual name and elements in *EC*.

In editing, rules are made up of a series of atoms, which can be data functions $C(x)$, relational functions $P(x, y)$, or other built-in functions $(r, x_1, x_2, \cdots)$. Thereinto, *C* stands for classes in the ontology, while *P* stands for properties, including object properties and data properties. In addition, *x* and *y* are the corresponding variables, representing individuals and data values, respectively. Build-in functions are embedded in SWRL, in which *r* stands for functional relationships and *x* stands for variables with the functions of numeric calculation, string operation, comparison operation, Boolean operation, etc.

The reasoning process based on SWRL can be expressed by Eqn. (4.2).

$$(\wedge_{i=1}^{n} Atom_i) \rightarrow (\wedge_{j=1}^{m} Atom_j) \tag{4.2}$$

SWRL consists of reasoning antecedents and reasoning consequents, which are made up of multiple atoms with the relationships of logical AND. When atoms of the reasoning antecedents are all true, reasoning consequents can be deduced. With the ontology reasoning engine Pellet [6] or HermiT [3], SWRL rules can be used for reasoning in the ontology of assembly events to establish the connections between workstep and other participating resources.

By this form of reasoning method, assembly workstep event elements that cannot be obtained in the process of information extraction can be reasoned out to further enrich the assembly workstep events. The standardization and semantic modeling of assembly process can be realized through the establishment of multiple assembly workstep events.

Conclusions:

This research proposes an assembly process modeling method based on ontology technology in order to solve the problems of non-standard description and insufficient semantics. Firstly, assembly process is divided into worksteps, which correspond to the concept of event ontology as basic units. The elements of workstep events are analyzed under the premise of considering the characteristics of assembly process. Secondly, NLP technology is used to process assembly documents, further realize the transformation from human language to machine language, which constitutes a significant step of extraction of workstep event elements. In addition, the workstep event elements is completed with ontology reasoning of non-intuitive assembly resources that are used for the worksteps as well.

Acknowledgements:

The authors would like to express sincere gratitude to the anonymous reviewers for the invaluable comments and suggestions that have improved the quality of the paper. This research is financially supported by the 2017 Special Scientific Research on Civil Aircraft Project.

References:

[1] Boley H.; Paschke A.; Shafiq O.: RuleML 1.0: the overarching specification of web rules, International Workshop on Rules and Rule Markup Language for the Semantic Web, 2010, 162-178. https://doi.org/10.1007/978-3-642-16289-3_15

[2] Cohen, Y.; Naseraldin, H.; Chaudhuri, A.; Pilati, F.: Assembly systems in Industry 4.0 era: a road map to understand Assembly 4.0, International Journal of Advanced Manufacturing Technology, 105(9), 2019, 4037-4054. https://doi.org/10.1007/s00170-019-04203-1

[3] Glimm B.; Horrocks I.; Motik B.; Stoilos G.; Wang Z.: HermiT: An OWL 2 reasoner, Journal of Automated Reasoning, 53, 2014, 245-269. https://doi.org/10.1007/s10817-014-9305-1

[4] Liu, W.; Liu, Z.; Fu, J.; Hu, R.; Zhong, Z.: Extending OWL for modeling event-oriented ontology, 4th International Conference on Complex, Intelligent and Software Intensive Systems, 5447543, 2010, 581-586. https://doi.org/10.1109/CISIS.2010.88

[5] O'Connor J.: The NLP workbook: a practical guide to achieving the results you want, Conari Press, San Francisco, CA, 2013

[6] Sirin E.; Parsia B.; Grau B. C.; Kalyanpur A.; Katz Y.: Pellet: a practical OWL-DL reasoner, Journal of Web Semantics, 5(2), 2007, 51-53. https://doi.org/10.1016/j.websem.2007.03.004

[7] Wu, D.; Greer, M. J.; Rosen, D. W.; Schaefer, D.: Cloud manufacturing: strategic vision and state-of-the-art, Journal of Manufacturing Systems, 32(4), 2013, 564-579. https://doi.org/10.1016/j.jmsy.2013.04.008

[8] Xu, X.: From cloud computing to cloud manufacturing, Robotics and Computer-Integrated Manufacturing, 28(1), 2012, 75-86. https://doi.org/10.1016/j.rcim.2011.07.002