Int J Adv Manuf Technol (2018) 94:3413–3425
DOI 10.1007/s00170-016-9338-1

ORIGINAL ARTICLE

Design and development of a CNC machining process knowledge base using cloud technology

Yingxin Ye [1,2] · **Tianliang Hu** [1,2] · **Chengrui Zhang** [1,2] · **Weichao Luo** [1,2]

Received: 28 July 2016 / Accepted: 15 August 2016 / Published online: 25 August 2016

Abstract Nowadays, computer numerical control (CNC) machine tool undertakes more processing tasks than other common machine tools because of its highly automated machining ability and high performance. However, due to the lack of intelligence in machining process planning, machining procedure of products mostly depends on process planners rather than CNC machine tools. To make product quality less dependable on process planner's ability and improve the efficiency of process planning in order to fulfill changeable market, this paper presents an approach to design and develop CNC machining process knowledge base using cloud technology. The general standard STEP-NC is mapped to web ontology language (OWL) to describe machining process-related knowledge in a readable and comprehensible way. This mapping relation also makes knowledge suitable for storage in HBase. Through this ontology model, descriptive and logical knowledge can be collected. Hadoop platform is used in this approach to provide the NoSQL database HBase for large-scale knowledge storage and MapReduce programming model for large-scale knowledge processing. Taking advantage of MapReduce, knowledge query engine and reasoning engine can be developed. Users can submit task and resource descriptive files to the cloud through CNC controller and get machining process solutions from knowledge base. Evaluation mechanism is also adopted to filter low-quality knoweldge.

✉ Tianliang Hu
tlhu@sdu.edu.cn

[1] School of Mechanical Engineering, Shandong University, 17923 Jingshi Road, Jinan 250061, P. R. China

[2] Key Laboratory of High Efficiency and Clean Mechanical Manufacture, Ministry of Education, Shandong University, 17923 Jingshi Road, Jinan 250061, P. R. China

Keywords Knowledge base · Machining process · CNC · Cloud technology

1 Introduction

With the development of computer and information technology, computer numerical control (CNC) machine tools have undertaken more complicated machining tasks. Because of its highly automated machining ability and high performance, CNC machine tools have gradually became the main choice to process mechanical products than other common machine tools.

Among the life cycle of mechanical products, process planning is important as it connects the design stage and machining stage. Process planning gives the instruction in machining a product based on the product design and available machining resources. It could be said that, a good process planning can guarantee products' quality, processing time, and cost efficiency. However, current process planning mostly depends on the experience and expertise of process planners. Due to the lack of intelligence in process planning, even though CNC machine tools can do high-speed and high-precision machining, it cannot decide the machining procedure of a product. Besides, as the product becomes more and more complex, the time used in process planning takes up more proportion of product life cycle, which increases the preparing time for product manufacturing and thus cannot fulfill the demand of flexible and changeable market.

To make product quality less dependable on process planner's ability and improve the efficiency of process planning, some researches on machining knowledge base have been done [1–3]. Yeo et al. employed a frame-based knowledge-representation scheme to build knowledge base, which focused on the selection of turning process [1]. Nagasaka et al.

built a management system of knowledge base by presenting machining knowledge as a table of a predicate-logic representation [2]. Lee et al. developed a prototype knowledge-based system ACES for automatic process selection and sequencing [3]. As a combination of artificial intelligence and database, knowledge base is capable of intelligent behavior, which is based on certain domain rules. It has abilities not only to check and manage knowledge but also to reason new knowledge using the rules.

These studies give methods to build or manage machining knowledge base. However, there are still two aspects that need to be considered. They are the following:

- A common knowledge model. The diversity of knowledge model such as expression approach and classification increases difficulty of knowledge collection. Besides, various knowledge model impedes not only knowledge extension but also knowledge sharing between different knowledge bases.
- Ability to deal with mass knowledge storage and processing. It is obvious that the more knowledge a knowledge base contains, the better performance it would have. With knowledge accumulation, single computer is not enough to offer sufficient storage space and hard to improve efficiency of query and reasoning.

This research aims at presenting an approach for design and development of CNC machining process knowledge base with a common knowledge model, which has ability to deal with ever-growing knowledge effectively. In Section 2, researches related with the two mentioned aspects earlier are analyzed. In Section 3, method of knowledge collection with an ontology model is introduced. With this method, CNC machining process knowledge base on Hadoop platform is built and the method of usage is given as well in Section 4. Section 5 is the conclusions.

2 Literature review

Knowledge model, knowledge storage, and processing method play important roles in the process of establishing a knowledge base. Knowledge model is closely related to expansibility, readability, intelligibility, and reusability of knowledge in a knowledge base. While storage capacity and processing strategy affect integrity and efficiency of the knowledge base. Related works are listed in the following section.

2.1 CNC machining knowledge model

Currently, G-code (ISO 6983) is still widely used as a CNC machining model. It just gives CNC system route information and digital/analog input command. With this machining model, CNC machine tool is just an executor and lack of intelligence. To overcome disadvantages of currently used G-code, a new standard known as STEP-NC (ISO 14649) is developed by the International Organization for Standardization (ISO). Compared to G-code, STEP-NC offers an object-oriented, well-structured, and information-integrated data model which makes the basis for intelligent CNC manufacturing and the open environment of globalization [4].

Figure 1 shows the structure of STEP-NC. This figure shows that STEP-NC defines clear relation and detailed description about CNC machining process and workpiece. Since it complied with Standard for the Exchange of Product Model Data (STEP), it enables the information flow between upstream CAx (CAD/CAM/CAPP) system and CNC controller without any lost. Therefore, many researches utilize STEP-NC to realize process planning in the phase of CAPP.

Zhang et al. designed a STEP-compliant process planning system [5]. This system can evaluate surface roughness based on each working step and thus provide a reference for process optimization. STEP-NC file is generated as output in this system. Nassehi et al. examined the feasibility of STEP-NC in an adaptive global manufacturing system [4]. Because STEP-NC file is resource independent and interoperable, it can transfer regardless of geographical restrictions. A prototype system was developed to accept this file and generate the final instructions such as G-code or feature-based ShopMill MPF according to local machining resource. Allen et al. provided a methodology of generating STEP-NC code automatically [6]. This system can be regarded as a multi-agent system. With this system, the process planner accepts CAD file and machining specification, then exports STEP-NC file based on a manufacturing resource database. Zhang et al. did a series of research based on STEP-NC and developed a universal process comprehension interface (UPCi) [7–9]. This software can represent different CNC part programs in a unified STEP-NC format and extract process plan through reserved shopfloor knowledge.

Progresses of these researches demonstrate that information contained in STEP-NC is enough to describe machining process and suitable for current globalization trend. However, the modeling language EXPRESS used in STEP-NC lacks sematic and thus leads to difficulty of understanding to both humans and computers. This difficulty in interpreting would impede knowledge sharing and CNC intelligent process planning.

Ontology is a preferred technology for knowledge modeling due to its explicit, sharable, and reusable feature [10]. An ontology model language called web ontology language (OWL) is recommended by the World Wide Web Consortium (W3C). OWL was originally developed for Sematic Web, and gradually adopted by other fields like biological and library management. Recent years, OWL is also

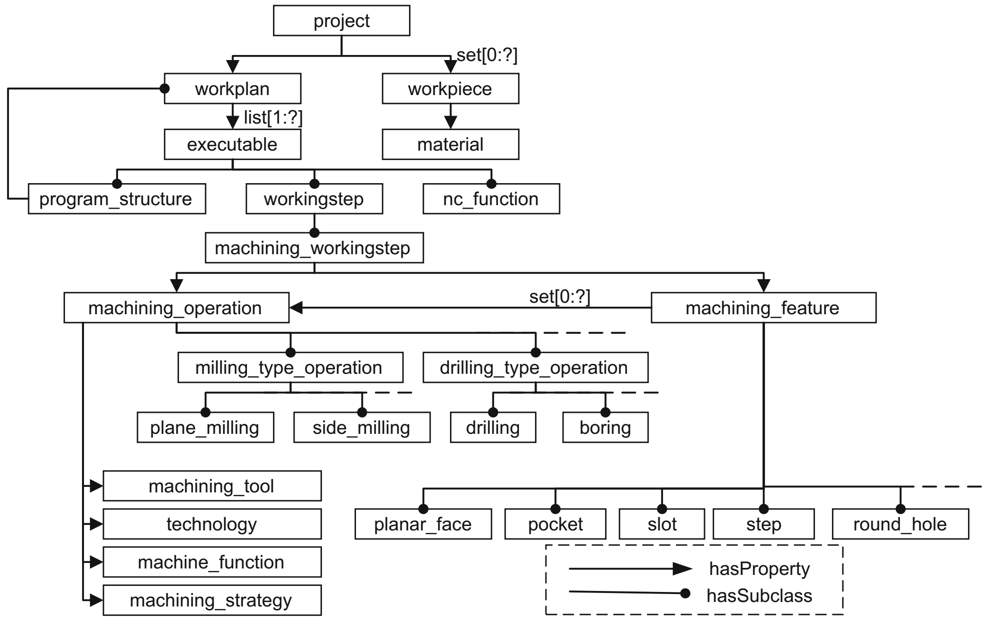


Fig. 1 Structure of STEP-NC model concluded from ISO 14649 Part 10

used in manufacturing field. Manufacturing's Semantics Ontology (MASON) is such an ontology model developed by Lemaignan et al. [11]. This model includes three head concepts (entities, operations, and resources) which aims at describing manufacturing relevant concepts and can be applied in cost estimate and multi-agent system. Chi use OWL building a model of supply chain to help enterprises find the right partner [12]. However, these models are custom defined and lack of widely accepted standards support. Barbau et al. built an ontology model named OntoSTEP [13], which complies with STEP. From this research, OWL shows superiority over EXPRESS in two aspects. One is readable and comprehensible by both humans and machines; the other is its ability to do reasoning. However, OntoSTEP is presented in a text file, which would be too large with knowledge increasing to store in a single computer. Meanwhile, query and reasoning efficiency would be affected as well.

In this paper, focusing on intelligent process planning, machining relevant information is distilled from STEP-NC and transformed to OWL model through a designed mapping rule. Then, this model was used to collect knowledge and build knowledge base, which meets requirements for massive knowledge's storage and processing.

2.2 Large-scale data storage and processing

Distributed and parallel processing would contribute to processing efficiency of large-scale data. Cloud computing is a promising approach which supports distributed and parallel processing. It is firstly put forward by Eric Schmidt in the Search Engine Strategies Conference & Expo (SES), San Jose, 2006. Definition of cloud computing is various, the widely acceptable one is defined by NITS [14]: Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Lots of famous companies such as Google, Amazon, IBM, and Yahoo have successively invested a lot of resources to carry out the cloud project [15–17].

In the face of globalization, cloud computing is also applied in the manufacturing field and generates a new concept of cloud manufacturing. Cloud manufacturing, which is a new computing and service-oriented manufacturing model, has been widely studied in recent years [18].

Tao et al. investigated the relation of cloud computing, Internet of things (IoT), and cloud manufacturing. Based on their previous research works on manufacturing grid (MGrid) service management [19–21], a cloud computing and IoT-based cloud manufacturing service system was proposed [22]. For implementing the concept of cloud manufacturing, a series of theory problems on manufacturing service intelligent management [23] have been investigated by Tao et al. [24–28] too. The concept of manufacturing service SDM simulator (SDMSim) is proposed to provide a uniformed research platform which aims at studying these problems[29].

Ren et al. developed a cloud manufacturing system which gave an application example to provide good solutions for product life cycle based on the globalization of manufacturing resources [30]. Hadoop platform and its programming model MapReduce is introduced in this system to deal with massive data, but data storage is still achieved through traditional relational database management system (RDBMS).

In these researches, it can be concluded that cloud technology has been widely used under the environment of globalization in manufacturing field. Cloud technology shows its advantages of parallel and distributed processing feature, which can adapt to trans-region collaborative manufacturing or services. Compared to traditional machining process database, knowledge base built with cloud technology has advantages of openness, high access efficiency, and powerful storage capacity when faced with large-scale data.

In this research, based on the distributed operating environment which cloud technology provided, a CNC machining process knowledge base was built to fulfill the demands of massive trans-region knowledge collection, storage, and processing. The cloud knowledge base designed in this paper was built on the Hadoop platform. This platform is a project maintained by Apache for the open-source software of reliable, scalable, and distributed computing. Hadoop includes lots of modules, not only the MapReduce module for massive data processing but also the HBase module for massive data storage was used in the development of CNC machining process knowledge base.

3 Knowledge collecting method

In the process of developing knowledge base, knowledge collection takes an important role as knowledge is the foundation of querying and reasoning. Three steps are presented to collect knowledge: knowledge scope and acquisition way confirmation, ontology model development, and knowledge collection through ontology model.

3.1 Knowledge scope confirmation

In the process of ontology modeling, clear knowledge scope can guarantee modeling efficiency and model integrity. In this research, CNC intelligent process planning-related knowledge is classified into two parts and shown in Figure 2. One is descriptive knowledge which defines process planning relevant concepts and their according instances. The other is logical knowledge which specifies domain rules to do knowledge reasoning

3.1.1 Descriptive knowledge

Descriptive knowledge consists of three main parts. These three parts are linked by relations to compose a complete machining project. They are listed as below:

- Workpiece: A workpiece can be described by machining features, material, and machining requirements.
- Machining resource: CNC machine tool and its attachments like cutting tools and fixtures belonging to it are all machining resources.
- Machining process: As mentioned in Section 2, according to STEP-NC, a machining process should specify cutting tool, feed rate, spindle speed, machining function, and machining strategy.

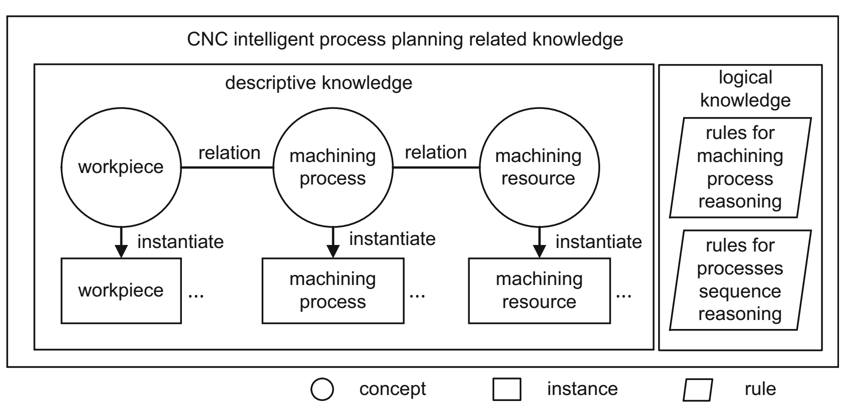
Most above-mentioned information has been already defined by STEP-NC, so concepts of descriptive knowledge can be distilled from different STEP-NC part. Figure 3 shows their acquisition method from STEP-NC.

But some information definition in STEP-NC is not complete enough to realize intelligence machining process planning. In this paper, fixture definition is added and machining requirement definition is enriched.

In a STEP-NC file, fixture is simplified through several points on the surface of workpiece, just indicating champing position to avoid cutting tool collision through machining. So, fixture can be simply described by the number of contact point for workpiece and also their adjustable range.

Machining requirement in STEP-NC file only contains global tolerance as workpiece's property. This shape tolerance is for workpiece and valid where no other tolerances are specified. However, surface roughness of each machining feature is also an important influencing factor in machining processes

Fig. 2 Structure of CNC intelligent process planning related knowledge



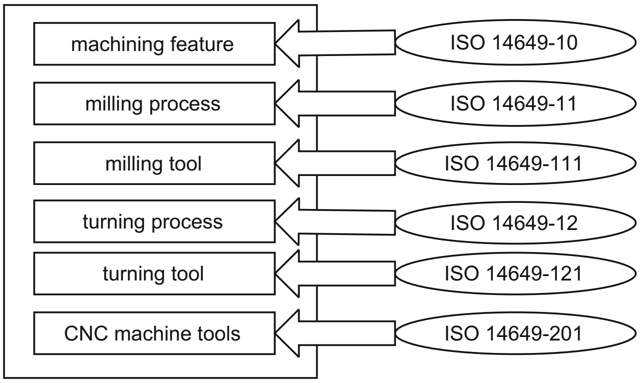


Fig. 3 Concepts of descriptive knowledge acquired from STEP-NC part

planning. So, surface roughness is added as another property of machining requirement in this research.

As the description earlier, concepts give the framework of descriptive knowledge. According to the definition of concepts, instances with specific value will be collected.

3.1.2 Logical knowledge

Logical knowledge provides derivation relations between machining feature and machining process for intelligent process planning. As logical knowledge contains key contents for decision-making, it should be acquired through domain experts and also need critical check to confirm its usability. Since process planning is related to processes of each feature and also their machining sequence, the logical knowledge should be formulated in two aspects as follows.

- Rules can reason one or several processes for one feature.
- Rules can determine processes sequence according to optimizing objective like energy efficient or processing speed.

As it is not feasible to make rules for every machining feature in different sizes, machining features concepts have to be generalized to cooperate with rules and thus make it suitable for machining process reasoning. This procedure will be specified in Section 3.2. Besides generalized feature, concrete feature is still maintained to describe a workpiece in project.

3.2 Ontology model development

As described in Section 2.1, OWL was chosen to build the ontology model in this paper. Primary concepts about OWL are listed below:

- Class: Define the concept of an object with some certain properties.
- Property: Explain the features of an object owns with some restriction, also known as a relationship expression between the domain and range it links. Two kinds of properties exist in OWL. The object property links class to class, while the data property links class to data types such as int, double, boolean, string, and so on.
- Individual: Declare an instance of a class with initialized property value.
- Restriction: Restrict the relation between domain and range to make ontology expression more rigorous. There are five restriction types: some, only, min, exact, and max.

This ontology model corresponds with the concept part of descriptive knowledge aforementioned in Section 3.1 to describe machining process-related knowledge in a clear and rigorous way.

An open-source ontology editor named protégé becomes a good selection for ontology modeling because it has friendly HMI for users to edit ontology and can generate an XML file in OWL schema according to the user editing in HMI. This software is developed by Stanford University with several useful reasoners to check logic conflicts of ontology in modeling.

All concepts described in STEP-NC can be distilled and transformed to OWL format through protégé. Table 1 shows a specific mapping relation between EXPRESS and OWL.

Aggregation type in STEP-NC means a property defines a one-to-many relationship. Considering the subsequent knowledge storage in database, this mapping is simplified by using restriction "only" to restrict the type of domain.

Besides direct mapping from STEP-NC to this ontology model, concepts about "machining_feature" should be generalized for process reasoning. To distinguish generalized class and concrete class, we add "_g" to the end of generalized class's name.

Take entity "planar_face", subtype of "machining feature", for example, Figure 4 shows its definition in STEP-NC. Two classes in OWL are related to this entity. They are "planar_face" and "planar_face_g". Class "planar_face" can be directly transformed from entity using the mapping relation. While class "planar_face_g" will be elaborated because some modification of its properties are required during the transformation.

All attributes of entity "planar_face" are listed in Fig. 4. Some of them are inherited from its parent class, which means that modification to these properties will be applied on their parent class and the parent class is also a generalized class.

In "planar_face_g", three kinds of modifications are proposed, they are listed below:

Table 1 Mapping relation between EXPRESS and OWL

EXPRESS			OWL
Entity			Class
Subclass of			SubclassOf
Attribute with an entity type			ObjectType
Attribute with a simple data type			DataType
Attribute without optional	Domain is single type		use "only" and "exact 1" to restrict type and cardinality
	Domain is aggregation type (LIST, SET, etc.)	SET[0:?], $n \neq 0$	use "only" to restrict type
		SET[n:?], $n \neq 0$	use "only" and "min n" to restrict type and cardinality
		SET[?:n], $n \neq 0$	use "only" and "max n" to restrict type and cardinality
		SET[n:m], $n \neq 0$, $m>n$	use "only", "max n" and "max m" to restrict type and cardinality
Attribute with optional	–		use "only" to restrict type

- Elimination: Properties indicate relationships between machining feature and whole workpiece as well as contribute less to machining process reasoning, like property "its_workpiece".
- Addition: Properties are required in machining process reasoning, like "its_machining_requirement" and "its_material".
- Generalization: Properties with exact value which are diversity and not suitable for reasoning, like "course_of_travel" and "removal_boundary".

Table 2 shows property difference between "planar_face" and "planar_face_g".

Through protégé, ontology model can be manually built according to the mapping relation between EXPRESS and OWL. This ontology model does not contain any individuals which corresponding to the instance part of descriptive knowledge. Logical knowledge is not contained in this model too. Both parts are considered as the knowledge which are collected through this ontology model.

3.3 Knowledge collection through ontology model

According to the ontology model, knowledge collection is divided into two parts as below:

- Descriptive knowledge collection can be considered as individual accumulation in knowledge base. All machining process planning-related personnel like experts, process planners, and machine tool operators can contribute to this part.

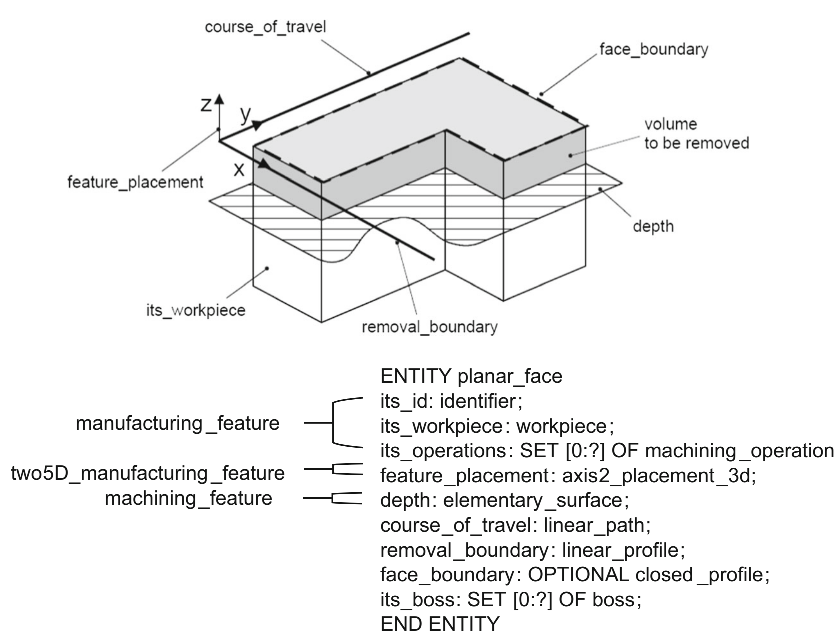


Fig. 4 Definition of "planar_face" in ISO 14649-10

Table 2 Property differences between "planar_face" and "planar_face_g" in OWL

Property of "planar_face"	Property of "planar_face_g"
its_workpiece	no exist
course_of_travel removal_boundary	ratio_of_length_width
its_boss	no exist
no exist	its_machining_requrement
no exist	its_material

- Logical knowledge collection can be considered as filling the set of rules. Different from descriptive knowledge collection, logical knowledge should be acquired only from experts to guarantee its correctness and authority.

Figure 5 shows the collection method.

3.3.1 Logical knowledge collection

Logical knowledge also known as rules are described using Semantic Web Rule Language (SWRL). Though OWL has its own rules to do simple reasoning [31], SWRL provides more detailed user-defined rules to express implicit domain knowledge which OWL may not represent. It enhances the integrity and preciseness of ontology-based domain knowledge.

SWRL inherits rule format from RuleML and represents with OWL, it can be said that, rules are made up by descriptive knowledge through some logic. A complete rule consists of "head" and "body" which means reasoning result and premise separately.

Logical knowledge should be collected before descriptive knowledge as it enables knowledge reasoning and the reasoning result may contribute to descriptive knowledge collection. Experts can use concepts defined in ontology model and combined with SWRL schema to make rules. Two frequently used schema are "variable" and "atom". Variable specifies an unknown individual and in front of its name is always a "?". Atom determines type of "variable" or the relationship between "variable". This type and relation must be defined in the ontology model. "head" and "body" in SWRL is constituted of "atom" with connector "^".

Take a simple rule for example, a rule specifies "when roughness of a plane is smaller than Ra3.2, the federate is smaller than 0.04 m/s" would be represented in SWRL as below:

machining_operation(?o)^its_technology(?o,?t)^feedrate(?t,?f)^planar_face_g(?x)^its_machining_requrement(?x,?rq)^its_roughness(?rq,?rn)^swrlb:lessThanOrEqual(?rn, 3.2)→swrlb:lessThanOrEqual(?f,0.04)

As ontology model derived from STEP-NC and offers an understandable and unified definition, experts can make rules in their familiar domain without additional attention to expression format. Each SWRL item is represented in its own textual format and relatively independent, so they can be collected in different time and regions which facilitates knowledge collection speed and richness.

3.3.2 Descriptive knowledge collection

Approach of collecting descriptive knowledge is more various than logical knowledge collection. Actually, a STEP-NC file can be treated as an individual of class "project". Therefore, STEP-NC files generated from other STEP-compliant systems can be directly submitted to knowledge base as a piece of descriptive knowledge through a transformation from EXPRESS to OWL. Besides, after actual application and

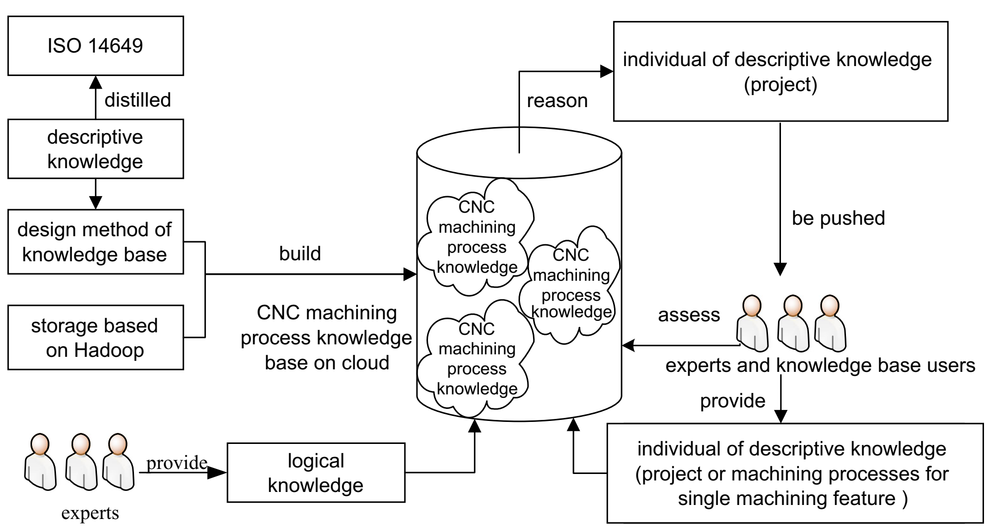


Fig. 5 Knowledge collection method

assessment, the new reasoning result generated from knowledge base can feed back to it as one of descriptive knowledge acquisition approach. Knowledge base users can also edit machining process for each feature according to ontology model and submit them as part of "project" individual.

4 Implementing knowledge base in the cloud

Knowledge base is responsible for knowledge storage and processing. In addition to the realization of basic functions, a reasonable application mechanism can also affect the usability of knowledge base. Design and application methods of knowledge base are introduced in this chapter.

4.1 Design of knowledge base

Knowledge base consists of three parts. Database is used to store classes and individuals of descriptive knowledge. Meanwhile, it cooperated with reasoning engine, which utilizes logical knowledge to do reasoning and query engine to retrieve existed knowledge.

4.1.1 Design of database

Traditional relational database on single computer like SQL Server and MySQL is incapable to store large-scale knowledge and access them efficiently. Hadoop Database (HBase) is one of not only SQL (NoSQL) database which provided by Hadoop platform. Different from relational database, HBase uses distributed key value for data storage. It stores each data as an object, which not restricted to two-dimensional tables. Figure 6 shows the object stored in a key value store table. Each object can be considered as one class or individual in OWL. Its property value corresponds to the property domain in OWL.

Table 3 shows conceptual view of HBase table. Columns in HBase can be added dynamically while column family should be designed in advance. Each column will be named with a prefix of its column family. Each cell in HBase can store several values distinguished by its timestamp, then an exact value can be get through the combination of row key, column and timestamp.

HBase is chosen to store knowledge represented in OWL, not only because its ability of distributed storage is suitable for dealing with massive data but also because its features listed below:

Table 3 Conceptual view of HBase table

RowKey	Timestamp	Column Family: cf1		Column Family: cf2	
		Column	Value	Column	Value
r1	t1	cf1:c1	v1-11a		
	t2	cf1:c2	v1-12a		
	t3			cf2:c1	v1-21a
	t4	cf1:c1	v1-11b		
r2	t5	cf1:c1	v2-11a		
	t6			cf2:c2	v2-21a

- Several values can be stored in one cell. Some properties in OWL describe a one-to-many relation, which corresponds to several domain values. Restricted by the two-dimensional table, this type of relationship is difficult to store and query in relational database.
- All data stores in HBase in the form of string. As OWL file is textual, there is no conversion between expression of knowledge and storage of knowledge.
- Value is allowed to be null and will not take up any space. Though concepts store in one table are similar, some differences still exist in their properties. It is null when the class does not own this property or this property is optional. This will save more space than using relational database.

Query for HBase is always aiming at one table that means joint query does not get good support in HBase. So, the lesser the number of tables in HBase design, the better to some extent. However, considering efficiency of query and reasoning, a clear classification of tables is important too. In this design, six tables are built as follows. One of them focuses on class storage, while others concentrate on individual storage.

- Table for class storage. This table is the basis for other four tables design. Two column families exist in this table named "basic relation" and "custom relation". Columns belonging to "basic relation" are relationships OWL supports, such as "SubclassOf", "DisjointWith", "EqualTo" etc. While columns belonging to "custom relation" are

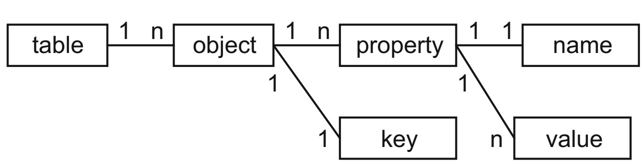


Fig. 6 Object stored in a key value store table

object properties and data properties. Each property takes up two columns to store because of the additional restriction type. Relations and restrictions recorded in this table can be used for ontology integrity and consistency check.
- Tables for machining feature/machining process/machining resource/project individual storage. These tables are designed based on their corresponding classes. Only one column family "properties" exists, all properties owned by the basic class and its subclasses are added to this column family. Considering search efficiency, individual's type is designed as part of row key.
- Table for common used individual. This table is designed for individuals whose class is usually used as a simple structure. Entities such as "cartesian_point", "direction", and "axis2_placement_3d" are the type that almost every class has relation to them. Like other individual storage table, only one column family "properties" exists, but columns belonging to it would be more diversity because classes of these individuals have a few inheritance relation.

Rule storage does not need database for storage. Because they exist in their textual format and need no query and reasoning. Therefore, rules are just stored in a file in the HDFS system as rule pool for reasoning engine.

4.1.2 Design of reasoning engine and query engine

Aiming at ontology reasoning, some reasoners like RACER, Hermit, Jena etc. are developed by researchers. But these reasoners cannot be directly used as reasoning engine in this research because their limited performance when dealing with massive data. Different from above reasoners only supports OWL rules, Jess supports custom rules made by SWRL to do reasoning in various domain. As a rule engine for the java platform. Jess provide API for users to develop reasoning engine according to the actual application condition.

In this research, query is an inner process during the work of knowledge base and not open for users. Ontology query language such as SPARQL was not used because knowledge is retrieved from database. This process can be directly realized in the program through API without a transformation from SPARQL to database query language. HBase API provides two methods to query data, which are "Get" and "Scan". "Get" method returns one record according to the row key and "Scan" method returns a set of records according to some specified parameters such as column name or timestamp range.

As knowledge scale is large, its query and reasoning process would take a lot of time on one computer. Taking advantage of Hadoop platform, a programming model named MapReduce is applied to do parallel process among computer clusters. In the cluster, one computer is considered as master node and the others are worker nodes. Master node can distribute jobs to worker nodes and supervise the progress of job. So, users can concentrate on the realization of job without considering data split and scheduling.

In this paper, each query and reasoning task can be seen as a combination of several jobs and go through MapReduce phase. Actually, the application of MapReduce programming model is the procedure of overriding map and reduce function through some algorithms.

In map phase, map function takes a (key, value) pair as input. The input source is usually files from HDFS or table from HBase. According to different sources, key will be file name or table name and value will be line of text or one row data. After filtering useless data, a intermediate (ikey, ivalue) pair will be generated as output of map phase. Then different ivalues are integrated with the same ikey. Another (ikey, list <ivalue>) pair is taken as input of reduce function. The final output of reduce phase is a (okey, ovalue) pair after merging the redundant ivalues and doing some simple operation like addition or connection.

These two engines play different roles. Reasoning engine is responsible for the efficient machining processes reasoning and undertakes more work in the early stage of knowledge base construction. While query engine is rarely used in the early stage of knowledge base because there are no sufficient individuals existing. With continuous collection of knowledge, task burden would transfer from reasoning engine to query engine.

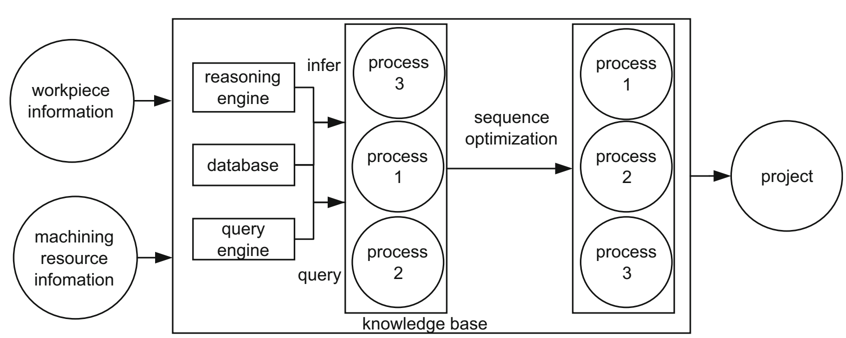


Fig. 7 Internal collaboration of knowledge base

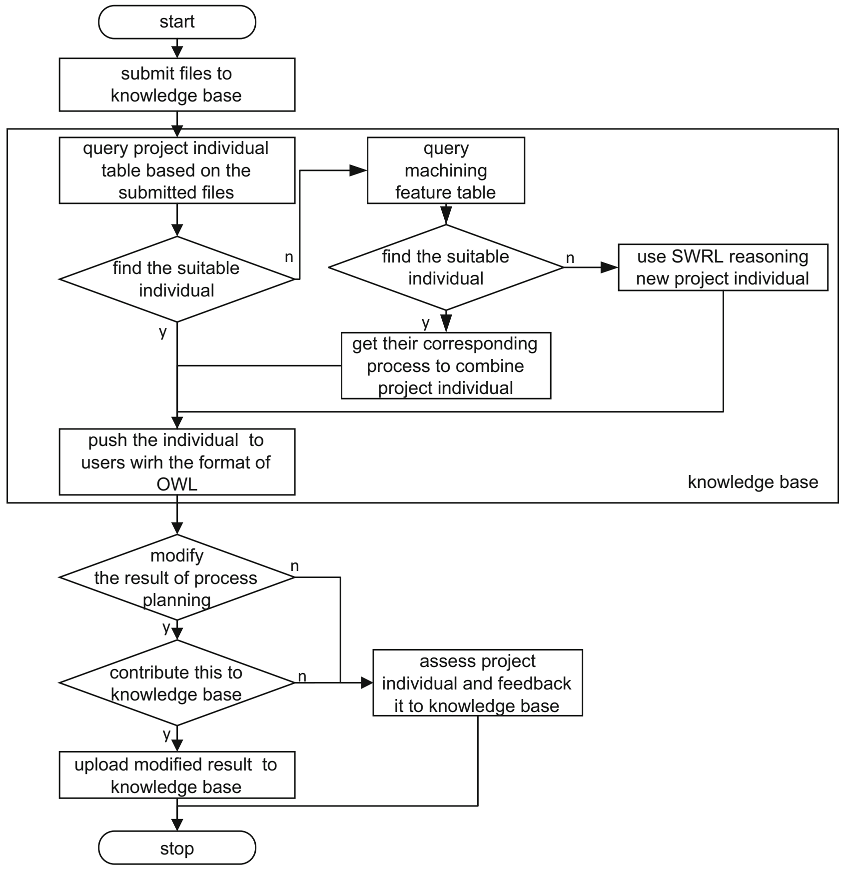


Fig. 8 Flow of knowledge acquisition and feedback

Inputs of these two engine are the same, including information of workpiece and available machining resource. Concrete features as well as machining requirements of workpiece should be combined to generate generalized features.

Reasoning procedure is achieved through combination of MapReduce and collected logic knowledge. As machining processes reasoning for each generalized feature is relatively independent, they can be done simultaneously. This job could be splitted and more detailed by reasoning technology, cutting tool, machine function, and machining strategy at the same time. At last, machining processes reasoned by this engine are not only pushed for users but also stored in database for knowledge expansion.

Query procedure is achieved through combination of MapReduce and HBase API. First of all, project individuals in database is inquired whether there is one similar to the project to be executed with same features and machining requirement. If there is no eligible individual, inquiring whether there is a generalized feature in knoweldge base is similar to the generalized feature to be executed. If such feature existed, its corresponding processes can be retrieved.

As machining process planning also involves process sequence optimization, logic knowledge is also utilized to optimize process sequence after determining machining process for each feature. Figure 7 internal collaboration of knowledge base shows the internal collaboration of knowledge base.

4.2 Knowledge acquisition workflow

To get machining process from knowledge base, query is the first step. If there is no directly used project individual, reasoning would be done subsequently. Figure 8 shows the flow.

Table 4 Comparison between knowledge bases

	Traditional machining process database	Machining process knowledge base designed in this paper
Concurrent processing ability	low quality	high quality
Knowledge storage capacity	usually limited by single computer	almost no storage limit (achieved by continually adding nodes into cluster)
Knowledge base scalability	limited	good

Two files are submitted to knowledge base to get machining process planning results. One is task description file which includes all machining features of the workpiece and their machining requirements. The other is resource description file including tools, fixtures and functions which a CNC machine tool has.

Because STEP-NC file carries information about machining features and parts of machining requirement, contents of task description file could be extracted from STEP-NC file using ST-Developer toolkit. While contents of resource description file could be structured based on the information defined by machining resource ontology.

As knowledge base is built in the cloud, these two files must transfer through network. So, both of them are represented in the schema of XML because of its extensible, transportable, and understandable feature. Actually, information contained in these submitted files can be considered as simplified instance of descriptive knowledge.

Each machine tools in the workshop owns its resource description file. Traditional process planning is usually made several days ahead of actual machining and thus cannot respond to the change of machining resource rapidly. But this resource description file can be modified based on the change of machine tools to enable process planning be done at the shopfloor level.

With knowledge increasing, more project individuals would be pushed to users when users ask knowledge base for machining process planning. Effective evaluation and feedback can contribute to the filter of low-quality knoweldge. After applying machining process planning, users can evaluate it according to the actual effect. Besides the overall assessment, other assessments of the three aspects are also made including machining efficiency, energy consumption, and economical efficiency.

Both files submission and feedback should be performed by the CNC controller. Information interaction between CNC controller and knowledge base in the cloud is realized through files transformation on the network. This network connection is only necessary when users ask for new machining process planning, commonly used process planning would be stored in the cache of CNC controller and called from the cache locally.

4.3 Discussion

4.3.1 Performance aspect discussion

In performance aspect, knowledge base designed in this paper has some differences from traditional machining process database due to the application of cloud technology. Table 4 lists comparisons from three aspects. It shows that the concurrent processing ability, knowledge storage capability and knowledge base scalability of the machining process knowledge base designed in this paper have better performances than traditional machining process database.

4.3.2 Design aspect discussion

In design aspect, compared with traditional machining process database, the advantages of the CNC machining process knowledge base built in this paper are listed as below:

- Knowledge representing method. In this paper, ontology is used to represent machining process knowledge. Semantics in the application of knowledge expression makes machining process more readable and gives a rigorous way for knowledge reasoning.
- Query and reasoning method. MapReduce programming model is adopted to do query and reasoning in this design. This model gives a parallel processing of large data set, which is adapted to large-scale data. This view has been verified by Fadhli et al. [32]. They did experiment to assess performances by comparing the execution time, CPU usage, and RAM usage of each approach and it shows that MapReduce performs better than the single host computer.
- Database selection. The NoSQL database HBase is selected to store massive knowledge rather than using traditional relational database. HBase has lots of advantages to face to today's machining process innovation and related knowledge continuous accumulation. Its storage mode is based on columns and it supports dynamic column adding. These two features show its scalability is better than RDBMS. HBase also has feature of splitting table automatically which is suitable for distributed storage. Though HBase does not support condition query, it provides API to scan table based on MapReduce programming model, so condition query can be realized to some extent by overriding this API.

5 Conclusion and future work

In this paper, an approach of design and development of a CNC machining process knowledge base using cloud technology is presented. Knowledge collection method as well as the implementation of knowledge are introduced.

For knowledge collection, knowledge is divided into descriptive knowledge and logical knowledge. To collect knowledge more effectively, a knowledge model expressed with OWL is built according to the description of STEP-NC, making knowledge more understandable and universal. A specific mapping relation between STEP-NC and OWL is proposed which can covert STEP-NC format to OWL and make knowledge storage applicable in HBase.

For knowledge base development and use, HBase is designed for knowledge storage. Query engine and reasoning engine are designed for knowledge process. Six tables are designed in HBase to store concepts and individuals separately which makes descriptive knowledge update and expansion convenient. Query and reasoning engines take advantage of MapReduce programming model to achieve a distributed and parallel process, making query and reasoning more effective. Knowledge use method is also presented in this paper. Task description file and resource description file are submitted by users to knowledge base and then knowledge base will push several project individuals to users as results of query or reasoning. Users can evaluate these process planning solutions, and thus, unuseful knowledge would be gradually filtered out with the help of user evaluation.

The future work will focus on two parts. One is further research on machining process reasoning rules. A reasoning rule editor will be designed and the effectiveness of the reasoning engine will be improved and verified. The other is the development of CNC controller which can interpret and execute the result which was pushed by knowledge base with OWL format, making CNC control system have the ability of intelligence.

Acknowledgments The work is supported by National Natural Science Foundation of China (Grant No. 51405270).

References

1. Yeo SH, Wong YS, Rahman M (1991) Knowledge-based systems in the machining domain. Int J Adv Manuf Technol 6(1):35–44
2. Nagasaka Y, Wibisono H, Ohtaki H, Ishikawa Y (1994) Management system of knowledge base in intelligent machining. Jsme International Journalserc Dynamics Control Robotics Design & Manufacturing 37(4):836–843
3. Lee SW, Song JY, Lee HK (2008) Construction and operation of a knowledge base on intelligent machine tools. Wseas Transactions on Systems 7(3):148–155
4. Nassehi A, Newman ST, Allen RD (2006) STEP-NC compliant process planning as an enabler for adaptive global manufacturing. Robot Comput Integr Manuf 22(5–6):456–467
5. Zhang X, Liu R, Nassehi A, Newman ST (2011) A STEP-compliant process planning system for CNC turning operations. Robot Comput Integr Manuf 27(2):349–356
6. Allen RD, Harding JA, Newman ST (2005) The application of STEP-NC using agent-based process planning. Int J Prod Res 43(4):655–670
7. Zhang X, Nassehi A, Newman ST (2014) Feature recognition from CNC part programs for milling operations. Int J Adv Manuf Technol 70(1):397–412. doi:10.1007/s00170-013-5275-4
8. Zhang X, Nassehi A, Newman ST (2015) A meta-model of computer numerical controlled part programming languages. Proceedings of the Institution of Mechanical Engineers Part B Journal of Engineering Manufacture 229(7)
9. Zhang X, Nassehi A, Safaieh M, Newman ST (2013) Process comprehension for shopfloor manufacturing knowledge reuse. Int J Prod Res 51(23–24):7405–7419. doi:10.1080/00207543.2012.757669
10. Borst WN (1997) Construction of engineering ontologies for knowledge sharing and reuse. Universiteit Twente 18(1):44–57
11. Lemaignan S, Siadat A, Dantan JY, Semenenko A. (2006) MASON: a proposal for an ontology of manufacturing domain. In: Distributed intelligent systems: collective intelligence and its applications, 2006. DIS 2006. IEEE Workshop on, pp 195–200
12. Chi YL (2010) Rule-based ontological knowledge base for monitoring partners across supply networks. Expert Syst Appl 37(2): 1400–1407
13. Barbau R, Krima S, Rachuri S, Narayanan A, Fiorentini X, Foufou S, Sriram RD (2012) OntoSTEP: enriching product model data using ontologies. Comput Aided Des 44(6):575–590
14. Mell P, Grance T (2011) The NIST definition of cloud computing. Commun ACM 53(6):50–50
15. Armbrust M, Fox A, Griffith R, Joseph AD, Katz RH, Konwinski A, Lee G, Patterson DA, Rabkin A, Stoica I, Zaharia M (2015) Above the clouds: a Berkeley view of cloud computing. EECS Department University of California Berkeley, vol 53
16. Khalid A (2010) Cloud computing: applying issues in small business. In: International Conference on Signal Acquisition and Processing. pp 278–281
17. Pallis G (2010) Cloud computing: the new frontier of internet computing. IEEE Internet Comput 14(5):70–73
18. Tao F, Zhang L, Venkatesh VC, Luo Y, Cheng Y (2011) Cloud manufacturing: a computing and service-oriented manufacturing model. ProcIMechE Part B:Journal of Engineering Manufacture 225:1969–1976
19. Tao F, Zhang L, Lu K, Zhao D (2012) Research on manufacturing grid resource service optimal-selection and composition framework. Enterprise Information Systems 6(2):237–264. doi:10.1080/17517575.2010.540677
20. Tao F, Zhang L, Nee AYC (2011) A review of the application of grid technology in manufacturing. Int J Prod Res 49(13):4119–4155. doi:10.1080/00207541003801234
21. Tao F, Guo H, Zhang L, Cheng Y (2012) Modelling of combinable relationship-based composition service network and the theoretical proof of its scale-free characteristics. Enterprise Information Systems 6(4):373–404. doi:10.1080/17517575.2011.621981
22. Tao F, Cheng Y, Xu LD, Zhang L, Li BH (2014) CCIoT-CMfg: cloud computing and Internet of things-based cloud manufacturing service system. IEEE Transactions on Industrial Informatics 10(2): 1435–1442. doi:10.1109/TII.2014.2306383
23. Tao F, Zhang L, Liu Y, Cheng Y, Wang L, Xu X (2015) Manufacturing service management in cloud manufacturing: overview and future research directions. J Manuf Sci Eng 137(4): 040912–040912. doi:10.1115/1.4030510
24. Tao F, LaiLi Y, Xu L, Zhang L (2013) FC-PACO-RM: a parallel method for service composition optimal-selection in cloud manufacturing system. IEEE Transactions on Industrial Informatics 9(4):2023–2033. doi:10.1109/TII.2012.2232936
25. Tao F, Li C, Liao T, Laili Y (2015) BGM-BLA: a new algorithm for dynamic migration of virtual machines in cloud computing. IEEE Transactions on Services Computing PP 99:1–1. doi:10.1109/TSC.2015.2416928
26. Tao F, Zuo Y, Xu LD, Zhang L (2014) IoT-based intelligent perception and access of manufacturing resource toward cloud manufacturing. IEEE Transactions on Industrial Informatics 10(2): 1547–1557. doi:10.1109/TII.2014.2306397
27. Tao F, Cheng Y, Zhang L, Nee AYC (2015) Advanced manufacturing systems: socialization characteristics and trends. J Intell Manuf 1–16. doi:10.1007/s10845-015-1042-8
28. Li J, Tao F, Cheng Y, Zhao L (2015) Big data in product lifecycle management. Int J Adv Manuf Technol 81(1):667–684. doi:10.1007/s00170-015-7151-x

29. Tao F, Cheng J, Cheng Y, Gu S, Zheng T, Yang H SDMSim: A manufacturing service supply–demand matching simulator under cloud environment. Robotics and Computer-Integrated Manufacturing. doi:http://dx.doi.org/10.1016/j.rcim.2016.07.001
30. Ren L, Zhang L, Tao F, Zhao C, Chai X, Zhao X (2013) Cloud manufacturing: from concept to practice. Enterprise Information Systems 9(2):186–209
31. ter Horst HJ (2005) Completeness, decidability and complexity of entailment for RDF schema and a semantic extension involving the OWL vocabulary. Web Semant Sci Serv Agents World Wide Web 3(2–3):79–115. doi:10.1016/j.websem.2005.06.001
32. Fadhli M, Gani TA, Melinda AY (2012) Comparison on efficiency of computational efforts between cluster computation (MapReduce) and single host computation. In: Cloud Computing and Social Networking (ICCCSN), 2012 International Conference on, pp 1–5