The International Journal of Advanced Manufacturing Technology (2022) 120:4863–4887
https://doi.org/10.1007/s00170-022-09002-9

ORIGINAL ARTICLE

An ontology-based modelling and reasoning framework for assembly process selection

Shantanu Kumar Das[1] · **Abinash Kumar Swain**[1]

Received: 18 June 2021 / Accepted: 2 March 2022 / Published online: 22 March 2022

Abstract

Assembly joining process selection is a knowledge-intensive task that needs an efficient tool to capture, represent, reuse, and share knowledge related to various joint requirements. This paper presents an ontology-based knowledge framework for identifying the appropriate assembly joining process to support designers and process planners effectively. A joining process selection (JPS) ontology is developed to represent different core concepts like feature, material, product, joint requirement, and joining process. Semantic Web Rule Language (SWRL) is used for ontology mapping of joining process selection concepts to retrieve the required knowledge for process selection that integrates several instances and knowledge rules. Further, a five-step sequential procedure is established to select the joining process from the CAD model automatically. The proposed approach automatically infers the possible, probable, and most probable joining processes through rule-based reasoning. Based on the evaluation of the ontology, the precision, recall, and F-measure obtained are 89.4%, 85.7%, and 87.5%, respectively. Finally, the efficacy of the ontology is evaluated using industrial case studies from the automotive and aerospace industry.

Keywords Ontology · Assembly · Joining process selection · Knowledge representation · CAD

1 Introduction

Assembly process planning is an essential stage in manufacturing products. According to relevant statistics, the assembly process consumes 50% of total production time, 20% of the unit production cost, and roughly about one third of the manufacturing company's labor [1]. Assembly planning has a paramount impact on product delivery time, cost, quality, durability, and maintenance [2]. Therefore, any endeavor to automate assembly planning is essential [3]. Appropriate selection of the joining process is one of the most critical part of assembly process planning. The choice of the joining process requires knowledge about joint design, material, the thickness of components, different joint requirements, etc. The vast number of materials, joining processes, joint design types, and functional requirements and the complex interrelationships between these attributes often make the selection of joining process for a given joint a difficult task. Moreover, industries are now using cost-effective multi-material joining methods [4]. The uses of multi-material components in the industry and several combinations of joint types, and its service condition make the selection process much complicated. Unfortunately, research in this area has not progressed much. Generally, the knowledge on assembly process selection are available in handbooks, research papers, and most often in the mind of experienced process planner. Moreover, this knowledge has been found to be implicit and unstructured. Hence, there is a need for formal specification of assembly process domain knowledge to select the suitable joining process. Therefore in this research, an ontology-based modelling and reasoning framework has been proposed for identifying the appropriate assembly joining process. Such a framework will provide the necessary support to the designers and process planners.

In literature, there are already many methods proposed for process selection, such as multi-criteria decision [5], fuzzy logic algorithm [6], ASTEK tool [7], and rule-based expert system [8]. Further, a knowledge-based advisory system [4] has been proposed to select suitable joining methods

---

✉ Abinash Kumar Swain
abinash.swain@me.iitr.ac.in

Shantanu Kumar Das
sdas1@me.iitr.ac.in

[1] Department of Mechanical and Industrial Engineering, Indian Institute of Technology, Roorkee 247667, India

for multi-material joining. A database using a general tree structure was then created to be fed into the advisory system. The data used in this framework are supplied interactively by the designer and cannot be accessed from CAD models. Most of the approaches listed above used database representation and data used in this framework are supplied interactively by the designer. So, it is a very complex process for a designer to interactively supply a large amount of extracted joint information (liaison) from the CAD model [9] to select the assembly joining process. Although these methods offer some advantages in process selection, they are generally limited in modeling the complex mappings between joining process selection concepts. Those pieces of knowledge can be inferred in a semantic model using deduction algorithms. Database representation does not have such kind of inferential capabilities. Therefore, it is very difficult to directly integrate the liaison database and retrieve the process selection knowledge using the above approaches.

In the last few decades, various ontologies have been developed for the definition and validation of assembly processes [10], assembly sequence planning [11], product variant design [12], and semantic inconsistencies in the welding process [13]. Mereotopological formal ontology and standard ontology technologies have been used to represent and differentiate assembly joint information in collaborative product design environments [14]. In this research, the authors have addressed the selection of assembly processes considering geometrical information in assembly joints only. However, in actual practice, geometric and non-geometric information is required to be considered for the process selection. For example, the selected joining process should satisfy the joint requirements like lightweight, low cost, high strength, etc., for a specific liaison. Additionally, it is necessary to check whether the selected joining process is applicable to liaison having a dissimilar combination of material. Further, the ontology-based knowledge representation model has been proposed for unit manufacturing processes. The ontology defines Joining class to represent the processes in which multiple parts or sub-assemblies are joined together. The joining class taxonomies only provide classification and hierarchical relationships for unit process terms in accordance with their shape change characteristics [15]. Core domain ontology has been developed [13] for the welding process to eliminate the involved inconsistencies among various standards. However, the proposed ontology is only limited to the welding process. In another study, an ontology-based framework [12] has been proposed for decision support in assembly variant design. The developed core concept in domain ontology layer in assembly process planning are found to be immensely generic and require further development. Thus, it can be stated that the available ontologies in the literature are not directly applicable for assembly joining process selection.

In this research, an attempt has been made to address the above needs using an ontology-based approach. The core of this research is the development of joining process selection ontology (JPS). This paper analyzes various joint requirements and discusses the different geometric and non-geometric relationships needed to select the joining process. The main contributions of this paper include (i) developed a JPS ontology for representing joining process selection knowledge, (ii) incorporation of extracted liaison knowledge [9] from CAD model into this JPS ontology, (iii) ontology mapping of joining process selection concepts through SWRL rules, (iv) joining process selection based on liaison information using rule-based reasoning method and SQWRL query engine, and (v) data-driven ontology evaluation method to test the completeness and conciseness of the JPS ontology.

The rest of the paper is structured as follows. Various literature related to the proposed work is reviewed in Sect. 2. The overall framework for joining process selection is described in Sect. 3. In Sect. 4, a joining process selection ontology is constructed. The proposed ontology-based framework for joining process selection is explained using SWRL rules and query engine in Sect. 5. In Sect. 6, two case studies are used to demonstrate the effectiveness of the proposed approach, and the developed ontology is evaluated in Sect. 7. Finally, discussions are carried out, and conclusions are drawn in Sects. 8 and 9, respectively.

2 Literature review

Our proposed work has been inspired by recent progress in several different areas, such as assembly joining process selection, ontology-based approach for joining process selection, which we review below.

2.1 Assembly process selection

Though some authors used liaison for assembly process planning, very few have used it for assembly joining process selection using database representation [4–8]. Kim et al. [4] developed a knowledge-based advisory system to select suitable joining methods for multi-material joining. This method used a concept map to represent the data and understand the semantic relationship between them. The data used in this system are mainly two types, i.e., process and performance attributes, which describe the designers' requirements and constraints on the joints. The data used in this framework are supplied interactively by the designer and cannot be accessed from CAD models. L'Eglise et al. [5] used a multi-criteria decision aid method for joining process selection at the early design stage. The designer chooses the right joining process design based on various knowledge related to the process like joint geometry, joint properties, materials, production, process, etc. LeBacq et al. [6] developed software that asks various questions to the designer to select the joining process. Different assembly joining processes are ranked according to multi-criteria evaluation using the fuzzy logic algorithm. Lae et al. [7] used ASTEK for the process selection that used the combination of free search, questionnaire-based, and analogy-based approaches. In this software, the user answers four questions like the geometry of the joint, materials to be joined, required functions from the joint, and joining production conditions which determine the requirements of joining processes. After filling these questionnaires, screening and ranking the different choices are done to select a suitable joining process. Darwish et al. [8] developed a knowledge-based system for recognizing the most appropriate welding processes suitable for specific joint requirements. This knowledge-based system used the knowledge about product type and some of the process capabilities, namely material type, material thickness, method of use, quality level, joint type, and welding position for the selection of suitable welding processes. In this study, the proposed framework is written using EXSYS Professional, where only 30 industrial important welding processes are used for the selection. Swain [16] used liaisons to specify the assembly process at various levels of detail using different attributes of the liaison like root gap, the thickness of the component, relative overlap, relative orientation, and relative location. This method used geometric information for process selection. Esawi and Ashby [17] used a relational database containing data tables for the creation of software, which helps in joining process selection. Mesa et al. [18] proposed a framework to select the suitable joining methods to fulfill the functionality capabilities of open-architecture products. The joining method is selected based on two alternatives, i.e., joint complexity index and functional characteristics like task complexity index, tool complexity, alignment complexity, and fixture requirement. These two alternative methods help in making decisions to select the most suitable joining methods taking into account the need for assembly and disassembly cycles of the product and only applicable to open-architecture products. Bond et al. [19] developed a joining process selector for sheet metal joining based on the quality function deployment (QFD) principle where correlation matrix and selection matrix are used to map the product's functional requirements into a joining process list.

The database representation is used in most of the above frameworks [4–8] for the assembly joining process. Data used in these frameworks are supplied interactively by the designer, which is very time-consuming for automating process selection from the CAD model. Database representation does not have inferential capabilities, making it difficult to retrieve the joining process selection knowledge by integrating the liaison database.

2.2 Ontology-based approach for assembly process selection

Ontology-based knowledge-driven approaches have been proposed for intelligent decision-making in various domains [10, 13–15, 20–22]. Lohse et al. [10] developed an assembly process ontology using liaison to configure the assembly process specification for both existing and new systems. This developed ontology is used to define the assembly process using task, operation, and action levels. Kim et al. [14] used the mereotopoloy to represent and distinguish the joining processes from other topologically and geometrically similar joining processes. This mereotopological representation of joining processes carries joining entities like glue, weld nugget, or mating boundary. Moreover, these joining entities are not available in the CAD model and are supplied interactively by the designer. Imran and Young [20] developed a knowledge-sharing framework based on assembly reference ontology (ARO) using a set of reference concepts related to hole and shaft assembly. This assembly reference ontology represents the tolerances and fits knowledge for selecting the assembly process restricted to hole and shaft assembly. Zhang et al. [15] developed an ontology-based knowledge representation method for a unit process, where the joining process is represented based on the concept of material flows. Gruhier et al. [21] developed a framework for assembly-oriented design (AOD), which describes the relationships between product parts and their evolution to time and space. This framework used mereotopological theory to represent the assembly joining process with the help of spatial, temporal, and spatiotemporal primitives to facilitate the understanding of assembly and design changes. The spatiotemporal primitives used in this representation are theoretical and thus cannot be accessed from CAD models. Solano [22] developed an ontology for semantic definition and classification of welding processes which facilitates interoperability. Most of the framework and ontology-based approaches used in the literature [20, 21] are not using the joint information for the selection of joining process, and the data used in these frameworks is supplied interactively by the designer [10, 14]. Saha et al. [13] developed an ontology for joining process representation for welding, and it is applicable mainly for checking the equivalence between the ISO welding standard and American Welding Society (AWS) standard to resolve the inconsistencies among the standards. The developed ontology is only applicable to the welding process, and the focus was to resolve the inconsistencies among the standards. Das and Swain [12] developed an ontology-based framework for decision support in assembly variant design. The developed ontology-based framework is helpful in guiding and assisting designers by providing suitable design decisions. The design suggestions needed for designing the variant product, the type variant design needed to create the variant product, and its effect on joint information, etc., are inferred by this developed ontology. The developed core concept in the domain ontology layer in the assembly process planning domain was immensely generic and cannot be used in the present state for assembly process selection.

3 Overview of the framework

An ontology-based framework for automatic joining process selection through liaison is proposed based on the analysis of joint requirements and its suitability to the existing liaison in a product. The proposed approach aims to select suitable joining processes at the design stage, i.e., directly from the CAD model, with the help of liaisons.

A joining process selection (JPS) ontology is developed to achieve the proposed goal, which carries different layers of information represented systematically. All joining process selection knowledge is collected from diverse sources for developing a generalized ontology [23–25]. The extracted liaison information from the CAD model of a product is integrated into this ontology for joining process selection. The knowledge required for process selection represented using SWRL rules, and an SQWRL query engine is employed to query the necessary joining processes which fulfill the joint requirements. The systematic overview of the proposed ontology-based framework is described in Fig. 1. The essence of the proposed approach is the integration of automatically extracted liaison knowledge in the ontology for automatic selection of the joining process through a rule-based reasoning system. The pipeline of our approach mainly consists of the following steps.

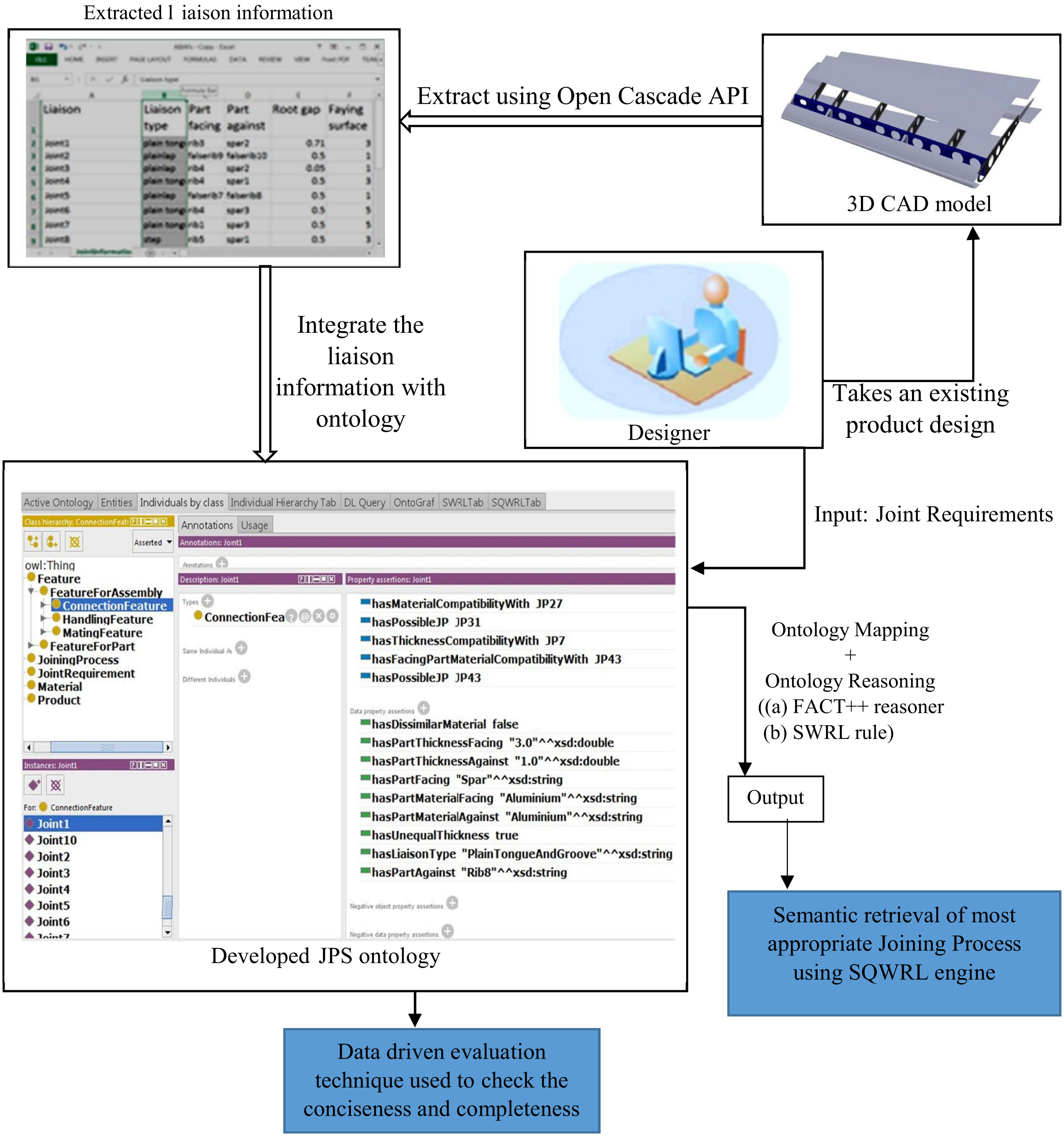
- **Extraction of liaison information:** The JPS ontology represents knowledge related to the both geometric and process requirements of various joints. The geometric knowledge of joint (liaison information) is automatically extracted from the CAD model using Open cascade API by taking a 3D CAD model of an existing product. This liaison information is extracted based on the algorithm adopted from the literature [9, 26].
- **Construct the JPS ontology:** This ontology includes three knowledge layers such as knowledge base layer, reasoning layer, and user interface layer, as shown in Fig. 2. The knowledge base layer is created based on OWL DLs, including a terminology box (TBox), an assertion box (ABox), and a rule base. To define a knowledge base, it's considered necessary to formally delineate (or box) statements that are generally understood to be true within the domain of discourse. This is the Tbox portion of knowledge representation for defining the various concept of an ontology using data and object properties. Statements in the TBox tend to remain static over time (due to the nature of truth), whereas the Abox can keep changing as more assertions are made, or existing assertions are rendered invalid. The ABox defines the instances of different concepts; for example, JR1 and JP1 are the corresponding instances of the *JointRequirement* class and the *JoiningProcess* class in TBox. These populated instances of *JointProcess* class and *JointRequirement* class are represented as

$$JP = [JP1, JP2\ldots, JPi\ldots, JPn],\ JR = [JR1, JR2\ldots, JRi\ldots, JRn]$$

The "individuals by class" tab is used to define the ABox in the developed ontology for both object and data property assertions related to a specific concept, as shown in Fig. 3. For example, the instances of *JoiningProcess* class are represented as shown in Fig. 3 using only the direct object and data properities as defined in the Table 1. The properties through which a core concept is represented, the same properties are applicable to define each instance of that core concept. Similarly, the instances of the other core concpts are defined in this proposed ontology.

Due to the limited reasoning capability of OWL, SWRL rule-based language [27] is defined in the knowledge base to represent the process selection knowledge. A Semantic Query enhanced Web Rule Language (SQWRL) is used in the reasoning layer to query the required joining process selection knowledge by the designer or the process planner. This layer generates specifications applied to the knowledge bases and returns the querying or inference results to the user interface layer. The user interface layer helps the designer or process planner assign the desired querying or reasoning tasks, based on which different joining processes are retrieved. The consistency of the proposed JPS ontology is checked by running FACT++ reasoner.

- **Integration of liaison knowledge into JPS ontology:** The automatically extracted liaison knowledge from the CAD model is first stored in an excel file, as shown in Fig. 1. Various liaison information like liaison type, the components between which the liaison exists, root gap, number of faying surfaces, the thickness of the part, etc., are stored in different columns in an excel file. Then, this Excel file data is uploaded using Cellfie plugin in Protégé [28] where the tabular data is converted into OWL axiom structures like class, individual, data, and object property using transformation rules. Then, these axioms are imported into the JPS ontology using the "import generated axioms" action. Cellfie is helping to insert all the generated axioms to the developed ontology and stored



Liaison	Liaison type	Part facing	Part against	Root gap	Faying surface
Joint1	plain tong	rib3	spar2	0.71	3
Joint2	plainlap	falserib9	falserib10	0.5	1
Joint3	plainlap	rib4	spar2	0.05	1
Joint4	plain tong	rib4	spar1	0.5	3
Joint5	plainlap	falserib7	falserib8	0.5	1
Joint6	plain tong	rib4	spar3	0.5	5
Joint7	plain tong	rib1	spar3	0.5	5
Joint8	step	rib5	spar1	0.5	3

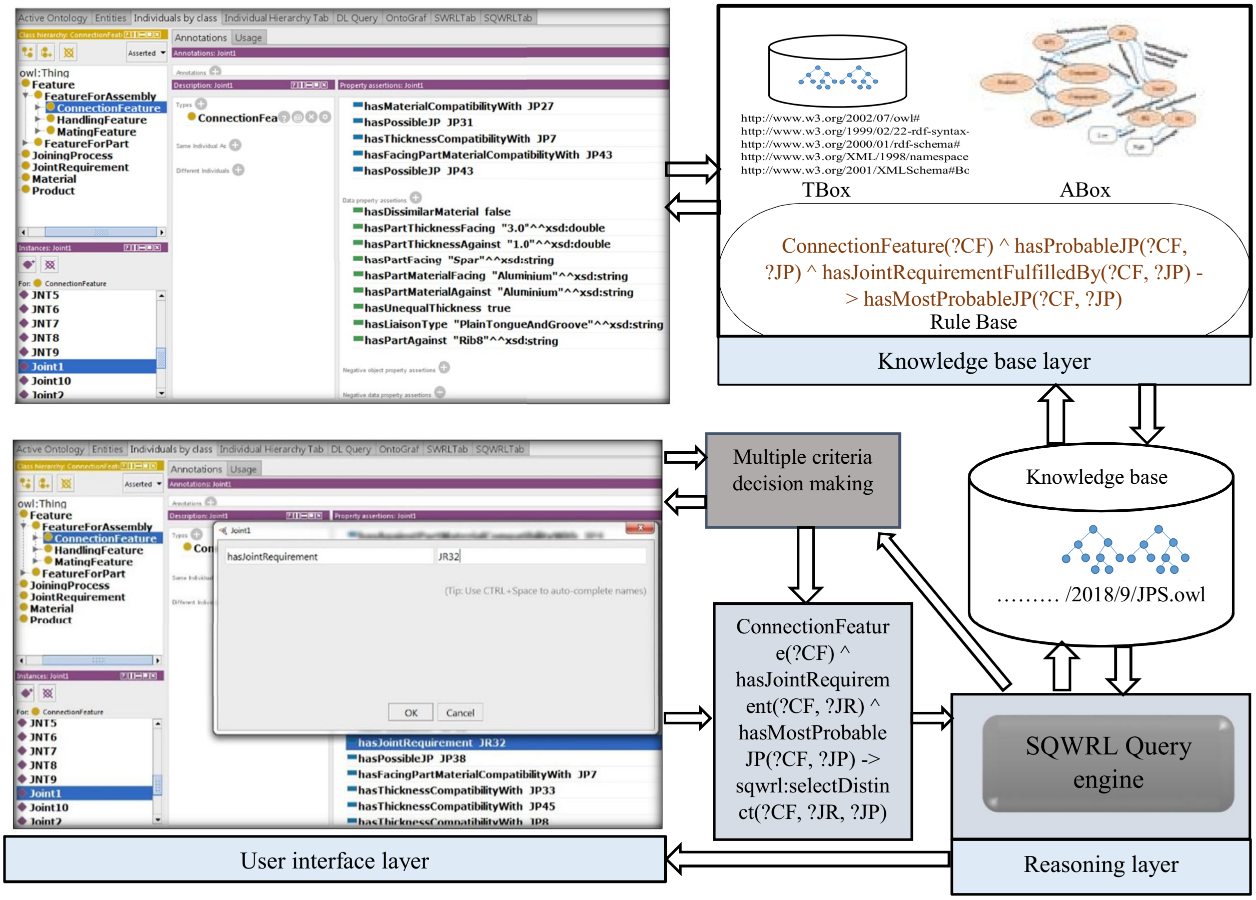
hasMaterialCompatibilityWith JP27
hasPossibleJP JP31
hasThicknessCompatibilityWith JP7
hasFacingPartMaterialCompatibilityWith JP43
hasPossibleJP JP43

hasDissimilarMaterial false
hasPartThicknessFacing "3.0"^^xsd:double
hasPartThicknessAgainst "1.0"^^xsd:double
hasPartFacing "Spar"^^xsd:string
hasPartMaterialFacing "Aluminium"^^xsd:string
hasPartMaterialAgainst "Aluminium"^^xsd:string
hasUnequalThickness true
hasLiaisonType "PlainTongueAndGroove"^^xsd:string
hasPartAgainst "Rib8"^^xsd:string

Fig. 1 The overall framework of the proposed approach

as various instances of corresponding concepts of JPS ontology. This liaison knowledge is stored as instances of *ConnectionFeature* class and further used to select the joining process by using a rule-based reasoning system.

- **Ontology mapping of JPS concepts:** The JPS ontology also represents knowledge related to the process requirements of various joint design using object and data properties. The knowledge required for assembly joining process selection is extracted from the expert reviews defined in different sources [1, 3, 4, 8, 9, 14, 16, 23 24, 25], and stored as instances in the ABox of the proposed ontology. The object and data properties related to process requirements are represented using SWRL rules for generating new knowledge required for joining process selection. This



hasMaterialCompatibilityWith JP27
hasPossibleJP JP31
hasThicknessCompatibilityWith JP7
hasFacingPartMaterialCompatibilityWith JP43
hasPossibleJP JP43

hasDissimilarMaterial false
hasPartThicknessFacing "3.0"^^xsd:double
hasPartThicknessAgainst "1.0"^^xsd:double
hasPartFacing "Spar"^^xsd:string
hasPartMaterialFacing "Aluminium"^^xsd:string
hasPartMaterialAgainst "Aluminium"^^xsd:string
hasUnequalThickness true
hasLiaisonType "PlainTongueAndGroove"^^xsd:string
hasPartAgainst "Rib8"^^xsd:string

http://www.w3.org/2002/07/owl#
http://www.w3.org/1999/02/22-rdf-syntax-
http://www.w3.org/2000/01/rdf-schema#
http://www.w3.org/XML/1998/namespace
http://www.w3.org/2001/XMLSchema#Bc

ConnectionFeature(?CF) ^ hasProbableJP(?CF, ?JP) ^ hasJointRequirementFulfilledBy(?CF, ?JP) -> hasMostProbableJP(?CF, ?JP)

Joint1

hasJointRequirement JR32

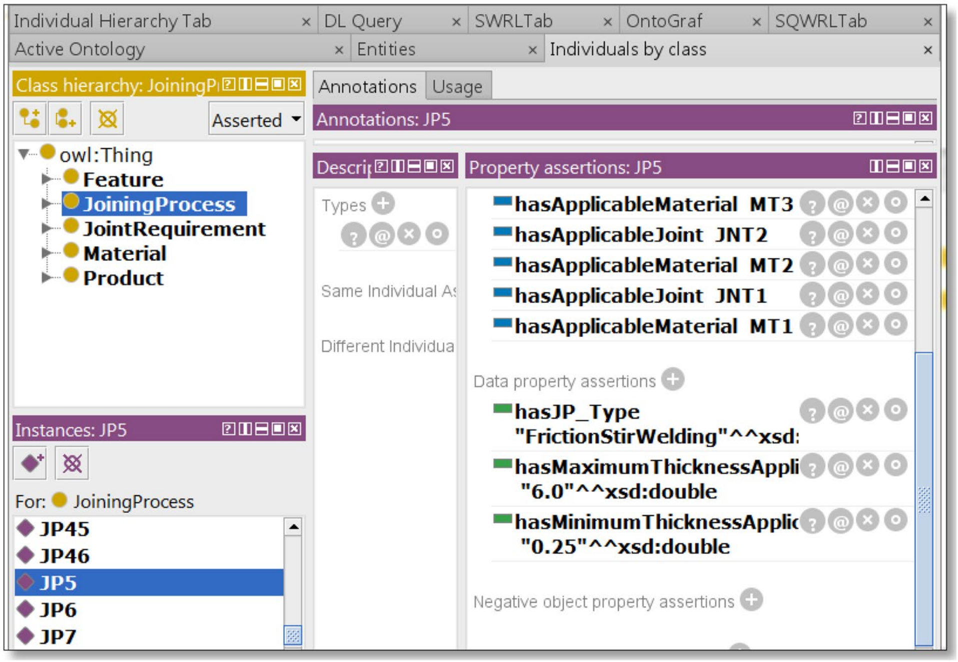
(Tip: Use CTRL+Space to auto-complete names)

hasJointRequirement JR32
hasPossibleJP JP38
hasFacingPartMaterialCompatibilityWith JP7
hasThicknessCompatibilityWith JP33
hasThicknessCompatibilityWith JP45
hasThicknessCompatibilityWith JP8

ConnectionFeature(?CF) ^ hasJointRequirement(?CF, ?JR) ^ hasMostProbableJP(?CF, ?JP) -> sqwrl:selectDistinct(?CF, ?JR, ?JP)

Fig. 2 The knowledge structure of JPS ontology

process requirements knowledge base, and the integrated liaison knowledge is further used for inferring the possible, probable, and most probable joining process selection through ontology mapping. Thus, the relationships between different concepts injoining process selection knowledge are mapped by utilizing the SWRL rules.



hasApplicableMaterial MT3
hasApplicableJoint JNT2
hasApplicableMaterial MT2
hasApplicableJoint JNT1
hasApplicableMaterial MT1

hasJP_Type "FrictionStirWelding"^^xsd:
hasMaximumThicknessAppli "6.0"^^xsd:double
hasMinimumThicknessApplic "0.25"^^xsd:double

Fig. 3 instances(ex. JP5) of the joining process class defined using data and object properties

Table 1 The relational mapping among classes in the JPS ontology

	Product	Joint requirement	Feature	Material	Joining process
Product	**hasPart** **hasAssembly** **hasSubassembly**		**hasLiaison**	**hasMaterial**	
JointRequirement			IsJointRequirementOf		
Feature	IsLiaisonOf	**hasJointRequirement**			IsApplicableJointOf **hasJointCompatibilityWith** **hasThicknessCompatibilityWith** **hasMaterialCompatibilityWith** **hasAgainstPartMaterialCompatibilityWith** **hasFacingPartMaterialCompatibilityWith** **hasPossibleJP** **hasProbableJP hasJointRequirementFulfilledBy** **hasMostProbableJP**
Material	IsMaterialOf				IsApplicableMaterialOf
JoiningProcess			**hasApplicableJoint** IsJointCompatibleWith IsThicknessCompatibleWith IsMaterialCompatibleWith IsAgainstPartMaterialCompatibleWith IsFacingPartMaterialCompatibleWith IsPossibleJP_Of IsProbableJP_Of IsFulfilledJointRequirementOf IsMostProbableJP_Of	**hasApplicableMaterial**	

- **Joining process selection through ontology reasoning:** The selected joining processes are inferred based on the joint requirements of a product by using a rule-based reasoning method. Finally, the SQWRL query engine is loaded to retrieve the required results by the designer or process planner.

4 A knowledge representation model for joining process selection

4.1 Conceptual analysis of joining process selection knowledge

The main objective of the joining process selection framework is to select the suitable joining process for a specific liaison from the set of available joining methods. The selected joining process should be compatible with the joint design type, the material type of the component, and its thickness. Also, it should satisfy all the joint requirements of a specific liaison. So, the joining process selection is a complicated task, which depends upon the knowledge associated with a particular liaison and its joint requirements. On the other hand, this joining process selection knowledge usually exists in an unstructured format, and the relation between different knowledge is also implicit. Hence, there is necessary to represent this joining process selection knowledge in a structured and formal manner. In this paper, an ontology-based approach is used to resolve these issues. In this approach, different concepts are used to represent many entities and their relationships, which is the foundation for knowledge representation. The joining process selection knowledge involves many concepts, which are defined below.

(i) Joining process

The joining process concept represents the essential characteristic of an assembly process, which is defined by two attributes, i.e., nature of joining process and the relationship of each process to a specific joint design type, material type, and thickness of the component. The nature of the joining process can be thermal spraying, high-speed impact nailing, friction stir welding, etc. Each joining process applies to some material, specific joint design type, and it is also compatible within a particular range of thickness of the component. So, the joining process concept can be defined as.

$$\left\{J = jp_1^n,\ jp_2^n,\ jp_3^n \ldots \ldots jp_k^n,\ jp_1^r,\ jp_2^r,\ jp_3^r \ldots \ldots jp_l^r\right\}$$

where k is the number of joining processes, and l is the number of relationships associated with the joining process.

(ii) Product

For an assembled product, the joining processes depend upon the thickness of each part associated with the liaison. The thickness of the parts related to a particular liaison may or may not be equal. Some joining processes are only applicable to liaison associated with parts having equal thickness. So, the relations between joining processes and parts of a product are tightly integrated.

(iii) Feature

The feature can be a region of a part or a connection between components (connection feature) to form a joint. This connection feature carries several joint information (liaison), which are helpful for the joining process selection.

(iv) Material

For an assembled product, the joining processes depend upon the material type of each component associated with a particular liaison. The material type of the components related to a particular liaison may or may not be similar. Some joining processes are only applicable to liaison associated with components having similar material. So, material types and their combinations (i.e., whether similar or dissimilar material) are the important factors for selecting joining processes.

(v) Joint requirement

An important factor in joining process selection is that it should satisfy all the joint requirements for a specific liaison. These joint requirements can be high strength, high aesthetic, lightweight, low cost, etc. Each joining process has unique design, economic, and quality characteristics that should match these joint requirements.

4.2 An ontology-based concept model for joining process selection

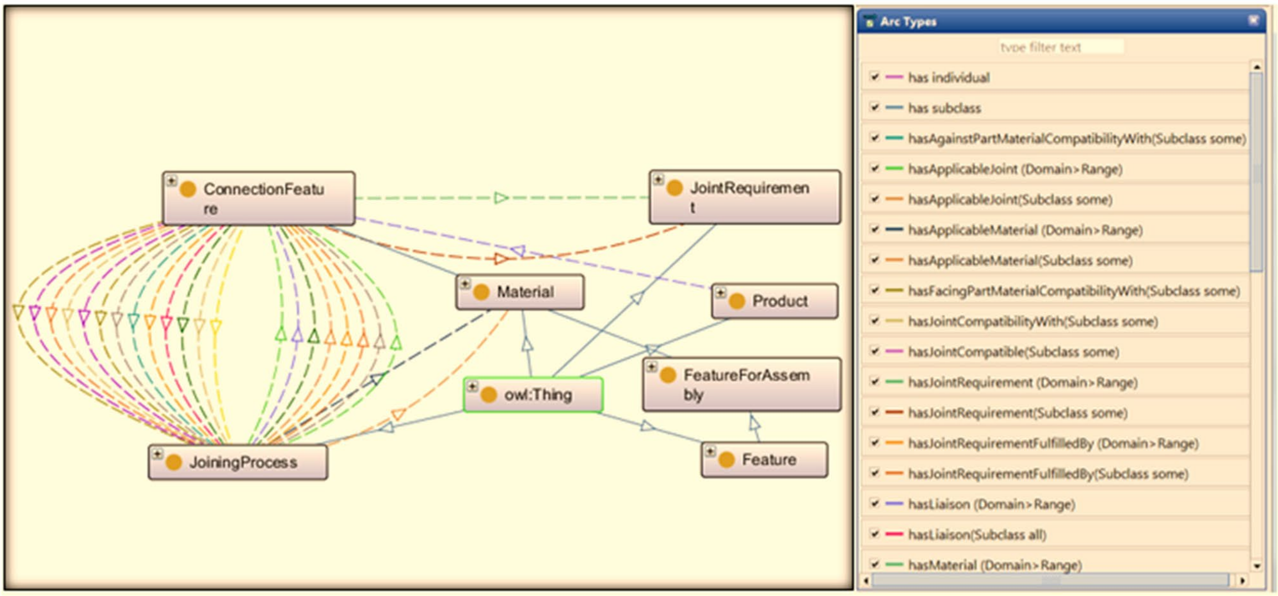
Based on the conceptual analysis of joining process selection knowledge, this research proposed a JPS ontology model. This ontology-based model is constructed by extending the terminology of Assembly Design Ontology [14] to represent various core concepts of JPS ontology using OWL ontology classes. The object properties are used to describe the relation between classes and between instances. The relation between classes/instances and data values are defined by the data properties in the ontology. The ontology represents the knowledge in the form of "subject-predicate-object". The subject and object are class or instance, while properties are a predicate. For example, "*JoiningProcess* hasApplicableJoint is a *Feature*." The other direction of object property can be reversely linked with the inverse of property, i.e., "*Feature* IsApplicableJointOf a *JoiningProcess*." Table 1 shows the object properties used in the JPS ontology for relational mapping among classes. The bold font is used to define the direct (one direction) relation between the pair of entities with domain and range in Protégé, and the other direction is linked to the reverse with the inverse of property.

The ontology carries nineteen object properties, which established the relational mapping between different classes, as shown in Fig. 4. The JPS ontology carries five basic top-level classes, namely, *JoiningProcess* class, *Product* class, *Feature* class, *Material* class, and *JointRequirement* class. The details of JPS ontology classes are defined below.

(a) *JoiningProcess* class

The wide variety and the vast number of joining processes make the taxonomy of joining processes an important knowledge activity. In this research, *JoiningProcess* class is used to represent the joining process concept. The joining process is classified based on the fundamental forces involved in joining processes like mechanical forces, chemical forces, physical forces, or a combination of them to produce a joint [25].

The taxonomy of *JoiningProcess* class is shown in Fig. 5a, where we classify it into five subclasses, namely *MechanicalJoining*, *Welding*, *AdhesiveJoining*, *HybridJoining*, and *VariantJoining*. These five top-level subclasses are further classified into several subclasses, as shown in Fig. 5a, based on the nature of the processes and the type of element involved between joining components. For example, hybrid joining is classified into weld bonding, rivet bonding, and weld brazing process. The selection of a more detailed process helps the process planner to decide the necessary preparatory and the supporting process to do the actual



- ☑ has individual
- ☑ has subclass
- ☑ hasAgainstPartMaterialCompatibilityWith(Subclass some)
- ☑ hasApplicableJoint (Domain>Range)
- ☑ hasApplicableJoint(Subclass some)
- ☑ hasApplicableMaterial (Domain>Range)
- ☑ hasApplicableMaterial(Subclass some)
- ☑ hasFacingPartMaterialCompatibilityWith(Subclass some)
- ☑ hasJointCompatibilityWith(Subclass some)
- ☑ hasJointCompatible(Subclass some)
- ☑ hasJointRequirement (Domain>Range)
- ☑ hasJointRequirement(Subclass some)
- ☑ hasJointRequirementFulfilledBy (Domain>Range)
- ☑ hasJointRequirementFulfilledBy(Subclass some)
- ☑ hasLiaison (Domain>Range)
- ☑ hasLiaison(Subclass all)
- ☑ hasMaterial (Domain>Range)

Fig. 4 The relational mapping graph among core concepts of JPS ontology

assembly operation. Each joining process applies to some specific joint, material, and range of thickness of the component. These properties of *JoiningProcess* class are represented by data properties in Protégé, as shown in Fig. 6. The selected joining process should be compatible with the material type and thickness of the component involved in an assembly. The joining process should fulfill a connection feature's joint requirements and compatible with its joint design type. So, these are the knowledge bases needed to consider for the automatic selection of joining processes.

(b) *Product* class

The *Product* class represents an assembled product having many parts, and it captures the knowledge about the product's structure. The *Product* class is classified into *Assem-*

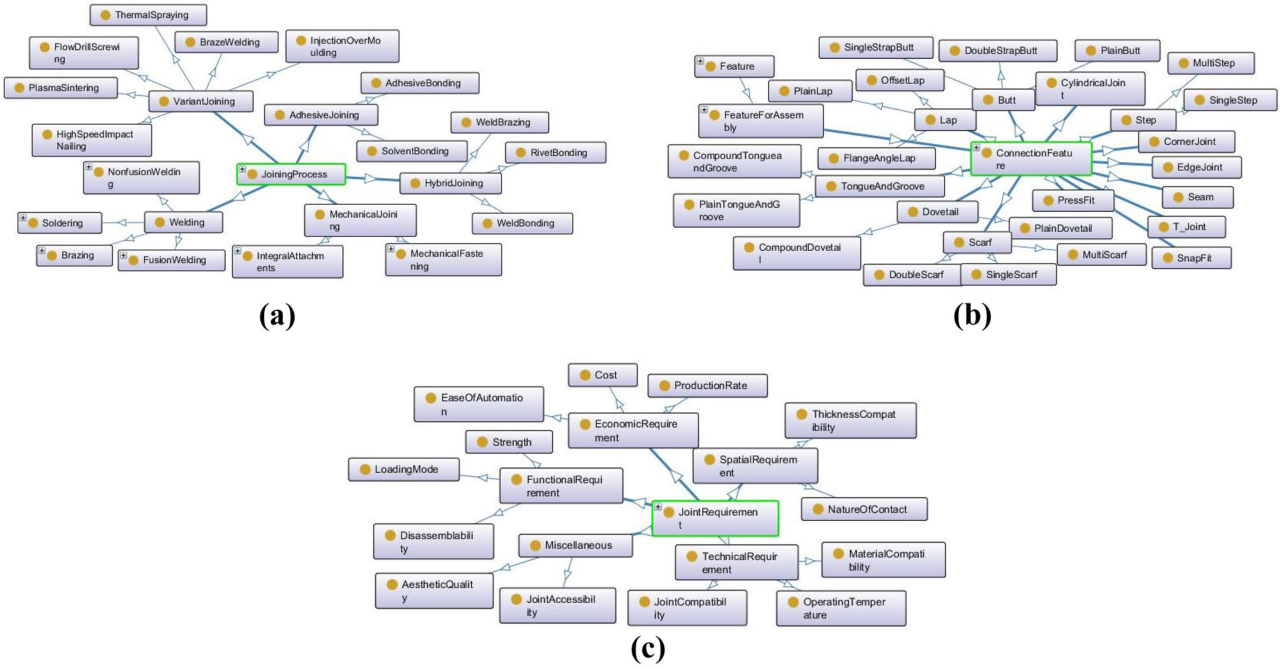
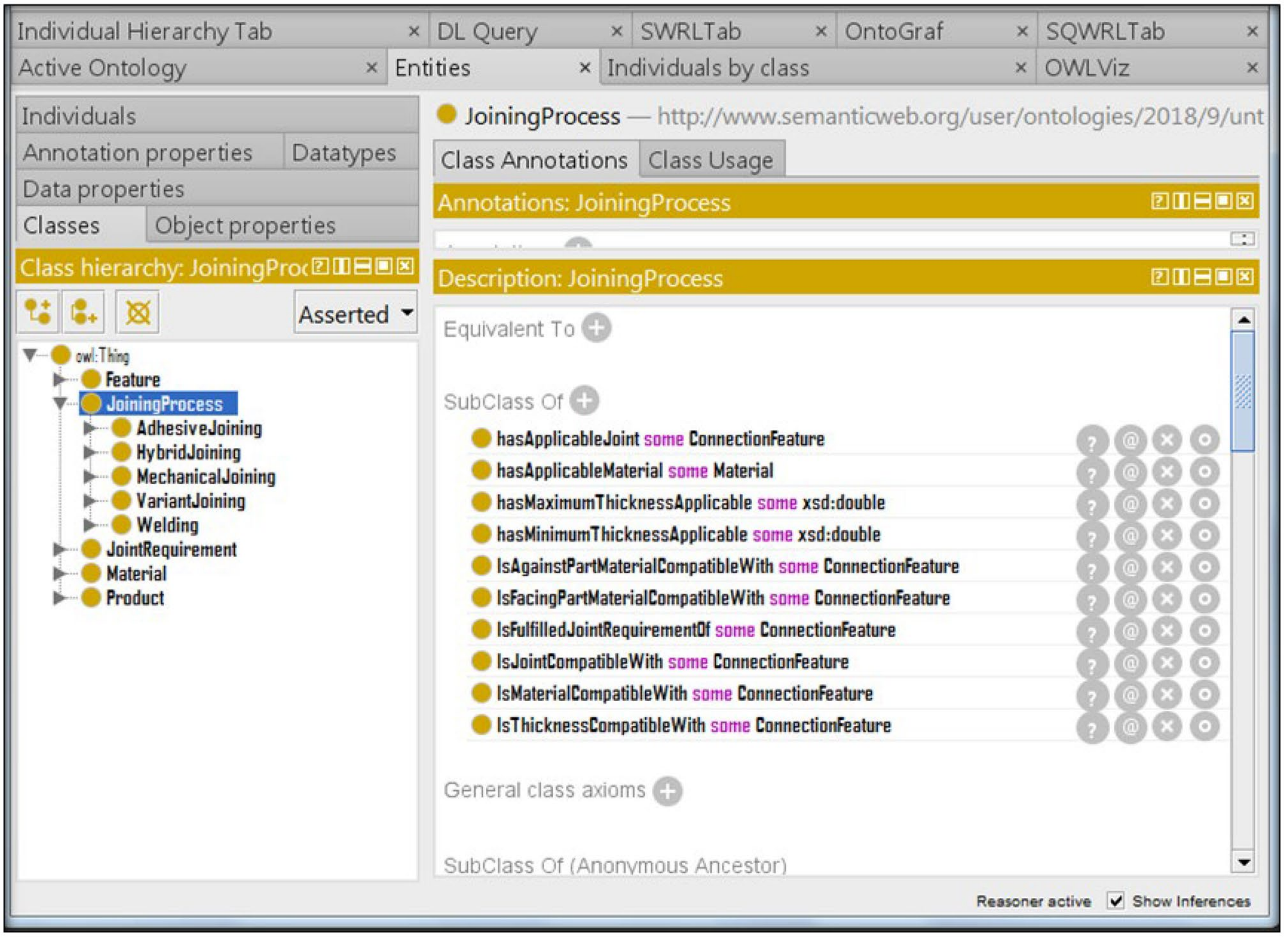


Fig. 5 Classification of core concepts of JPS ontology. (**a**) *JoiningProcess* (**b**) *ConnectionFeature* (**c**) *JointRequirement*



JoiningProcess — http://www.semanticweb.org/user/ontologies/2018/9/unt

- owl:Thing
  - Feature
  - JoiningProcess
    - AdhesiveJoining
    - HybridJoining
    - MechanicalJoining
    - VariantJoining
    - Welding
  - JointRequirement
  - Material
  - Product

SubClass Of

- hasApplicableJoint some ConnectionFeature
- hasApplicableMaterial some Material
- hasMaximumThicknessApplicable some xsd:double
- hasMinimumThicknessApplicable some xsd:double
- IsAgainstPartMaterialCompatibleWith some ConnectionFeature
- IsFacingPartMaterialCompatibleWith some ConnectionFeature
- IsFulfilledJointRequirementOf some ConnectionFeature
- IsJointCompatibleWith some ConnectionFeature
- IsMaterialCompatibleWith some ConnectionFeature
- IsThicknessCompatibleWith some ConnectionFeature

Fig. 6 Representation of *JoiningProcess* with object and data properties

bly class, *Subassembly* class, and *Part* class, as shown in Fig. 7. Each assembly and subassembly has at least one liaison. Thus, the *Assembly* class and *Subassembly* class is associated with the *ConnectionFeature* class using "hasLiaison" object property. For an assembled product, the joining processes depend upon the thickness of each component involved with a specific liaison. Thus, the knowledge about *Product* class and its relation with the *JoiningProcess* class is an essential criterion for the joining process selection.

(c) *Feature* class

In this research, the *Feature* class represents the features associated with a single component or an assembly. The single-component features are described in *FeatureForPart* class, and the assembly features are represented in *FeatureForAssembly* class. The *FeatureForAssembly* class is further classified into *ConnectionFeature* class, *HandlingFeature* class, and *MatingFeature* class. As this research is focused on the joining process selection, the *ConnectionFeature* class is considered for further study. The selection of joining processes depends upon the liaison, which is automatically extracted from the CAD model and stored as instances of *ConnectionFeature* class represented with several data and object properties, as shown in Fig. 8. The taxonomy of *ConnectionFeature* class is shown in Fig. 5b, where the connection feature is subdivided into the lap, butt, step, tongue and groove, dovetail feature, etc. These connection feature and their data properties are essential attributes for joining process selection. Each joining process is involved with some specific joints. So, these relational mappings between the *ConnectionFeature* class and *JoiningProcess* class are required to form an important knowledge base for joining process selection. The relational mapping of knowledge between *JoiningProcess* and *ConnectionFeature* class with the help of object properties is shown in Fig. 7.

(d) *Material* class

For an assembled product, the material type of each component associated with a particular liaison is represented as *Material* class, and its taxonomy is shown in Fig. 9. Each joining process applies to some material. Thus, the relational mapping between *JoiningProcess* class and *Material* class cre-

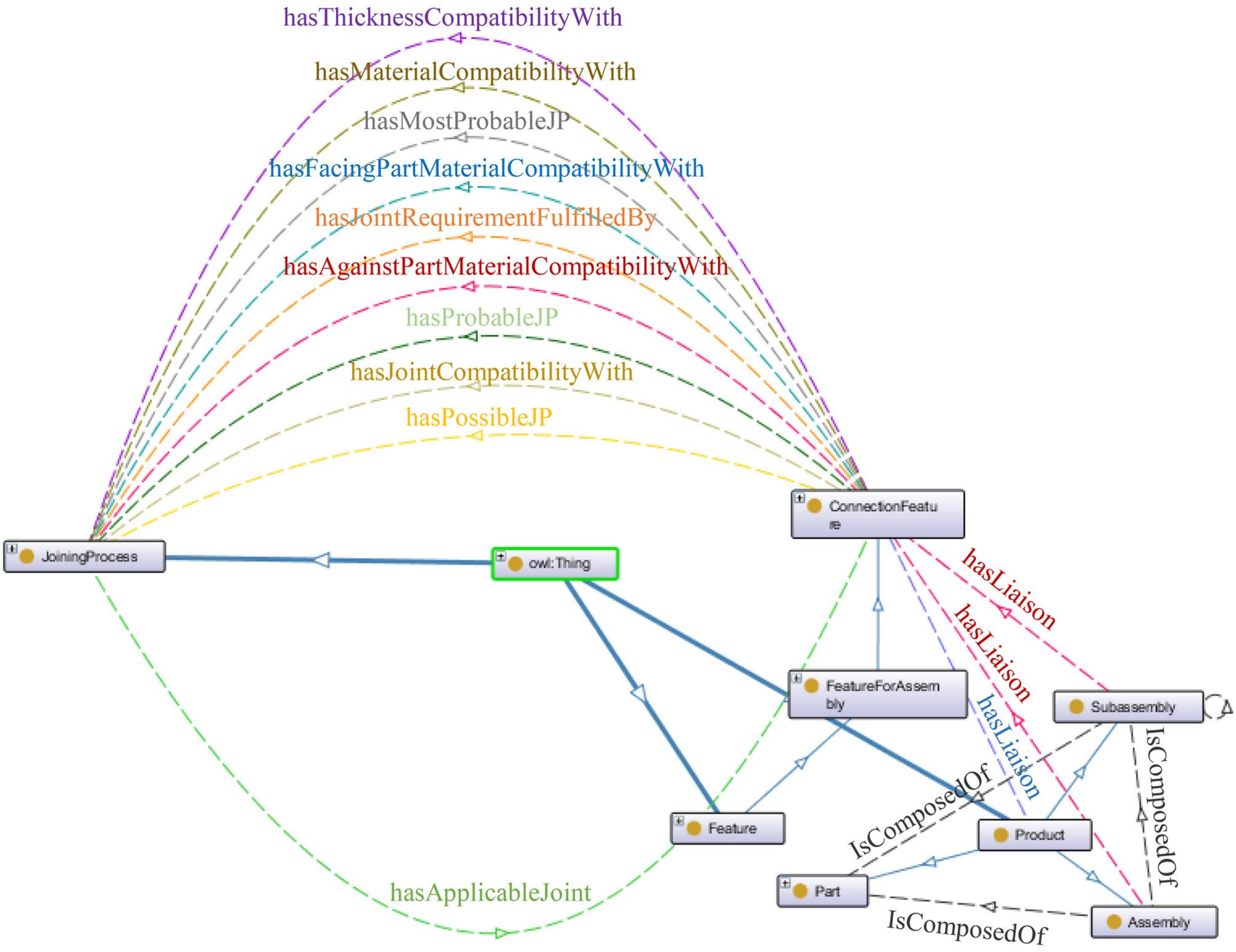


Fig. 7 Relational mapping between *JoiningProcess* and *ConnectionFeature* class

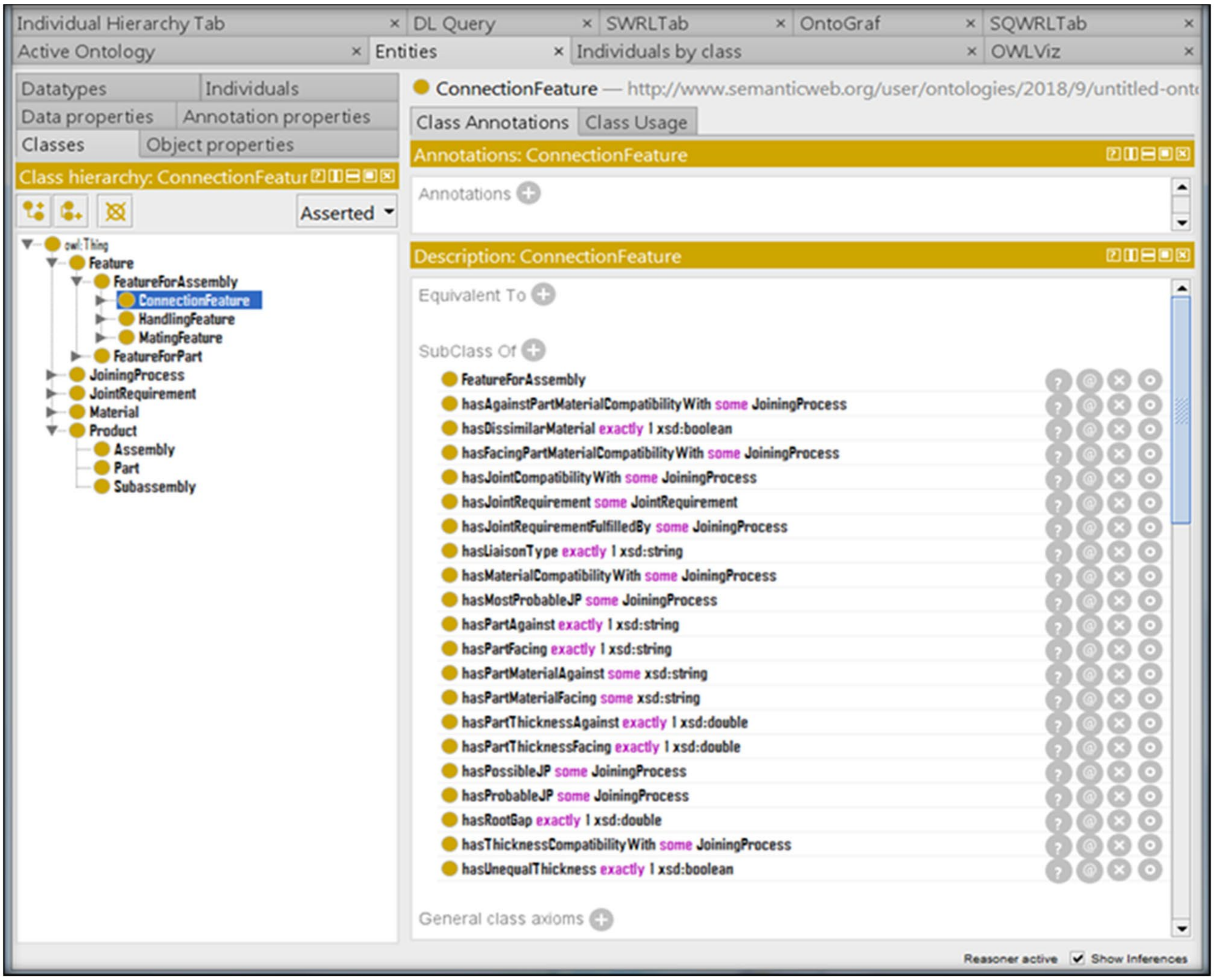
ates an important knowledge base for joining process selection. The joining process selection depends upon whether the material type of components involved with a specific liaison is compatible with the material applicable for a specific joining process and whether the joining process is suitable to both the similar and dissimilar nature of the material of the components. Thus, these relational mappings between *JoiningProcess* and *Material* classes are required, shown in Fig. 9.

(e) *JointRequirement* class

For an assembled product, there are certain joint requirements of the customer for a specific liaison, which is represented as *JointRequirement* class in this research, and its taxonomy is shown in Fig. 5c. Each joining process has specific design, economic, and quality characteristics that should be compatible with these joint requirements of a specific liaison. These relational mappings of knowledge between *JointRequirement* and *ConnectionFeature* class; and *ConnectionFeature* and *JoiningProcess* class creates an important knowledge base for the joining process selection, which is shown with the help of object properties in Fig. 10.

4.3 The relational mapping between JPS concepts

The above ontology model only provides a semantic representation for joining process selection knowledge at the conceptual level. The joining process selection knowledge is implied in the specific individuals of related core concepts using the presented concept model. In general, the relationships in OWL ontologies can be represented as a set of description logic (DL) predicates, which gives the relations precise semantics, as shown in Table 2. In DL, concepts are mapped to unary and relations to binary predicates. However, many semantic relations cannot be defined by the OWL DL ontology explicitly, such as causal relationships. Therefore, it is necessary to define new knowledge rules that can combine the ontological concepts and instances to construct



FeatureForAssembly
hasAgainstPartMaterialCompatibilityWith some JoiningProcess
hasDissimilarMaterial exactly 1 xsd:boolean
hasFacingPartMaterialCompatibilityWith some JoiningProcess
hasJointCompatibilityWith some JoiningProcess
hasJointRequirement some JointRequirement
hasJointRequirementFulfilledBy some JoiningProcess
hasLiaisonType exactly 1 xsd:string
hasMaterialCompatibilityWith some JoiningProcess
hasMostProbableJP some JoiningProcess
hasPartAgainst exactly 1 xsd:string
hasPartFacing exactly 1 xsd:string
hasPartMaterialAgainst some xsd:string
hasPartMaterialFacing some xsd:string
hasPartThicknessAgainst exactly 1 xsd:double
hasPartThicknessFacing exactly 1 xsd:double
hasPossibleJP some JoiningProcess
hasProbableJP some JoiningProcess
hasRootGap exactly 1 xsd:double
hasThicknessCompatibilityWith some JoiningProcess
hasUnequalThickness exactly 1 xsd:boolean

Fig. 8 Representation of *ConnectionFeature* with object and data properties

new rules. These knowledge rules can be constructed by using the SWRL rule, which is based on the combination of the OWL DL and RuleML [27]. Hence, in this paper, SWRL is chosen to define and infer new knowledge or reason the stored knowledge against the information provided by the assessors. The following will introduce how to construct the knowledge rules by mapping among the core concepts.

4.3.1 Knowledge rules for mapping the concepts between *JoiningProcess* and *Product* class

The knowledge about *Product* class and its relational mapping with the *JoiningProcess* class is an essential criterion for the joining process selection. Each assembled product has a number of liaisons produced due to the combination of the number of components. The thickness and material type of these components involved with a liaison are essential attributes for joining process selection. The selection of joining process varies according to the combination of material type (i.e., whether similar or dissimilar material) and the combination of the thickness (i.e., whether equal or unequal thickness) of the components involved with a liaison, which is evaluated by two important data properties, i.e., “hasDissimilarMaterial” and “hasUnequalThickness.” These essential data properties are evaluated using SWRL rules, as shown in Table 3.

4.3.2 Knowledge rules for mapping the concepts between *JoiningProcess* and *Material* class

The relational mapping between *JoiningProcess* class and *Material* class is required to form an important knowledge base for joining process selection. Each joining process is applicable to some specific material type, which is represented by object property “hasApplicableMaterial.” Each joining process applies to components associated with a liaison having

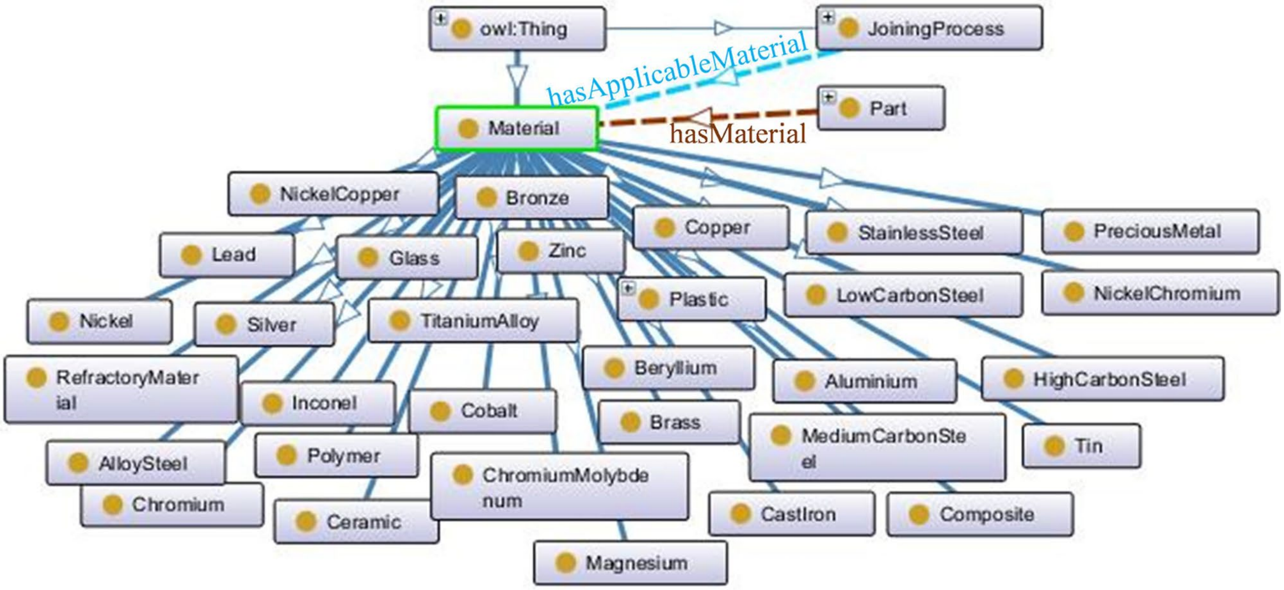


Fig. 9 Relational mapping between *JoiningProcess* and *Material* class

a minimum and maximum thickness. This range of thickness of the component also depends upon the material type. These relational mappings are done by using data property like “hasMinimumThicknessApplicable” and “hasMaximumThicknessApplicable.” These JPS properties are represented using SWRL rules, and some are shown in Table 4.

4.3.3 Knowledge rules for mapping the concepts between *JoiningProcess* and *Feature* class

The connection feature and their data properties are important attributes for joining process selection. Each joining process is applicable to some specific joint type, which is represented by object property “hasApplicableJoint.” The existing liaison in the CAD model should be compatible with the appropriate joint of a joining process, evaluated by using object property “hasJointCompatibilityWith.” Also, the thickness of the component associated with a liaison should be compatible with the selected joining process, which is evaluated by using object property “hasThicknessCompatibilityWith.” Also, the joining process selection depends upon whether the material type of component involved with a specific liaison is compatible with the material applicable for a particular joining process, which is evaluated by Object

Fig. 10 The relational mapping between the joining process and joint requirement

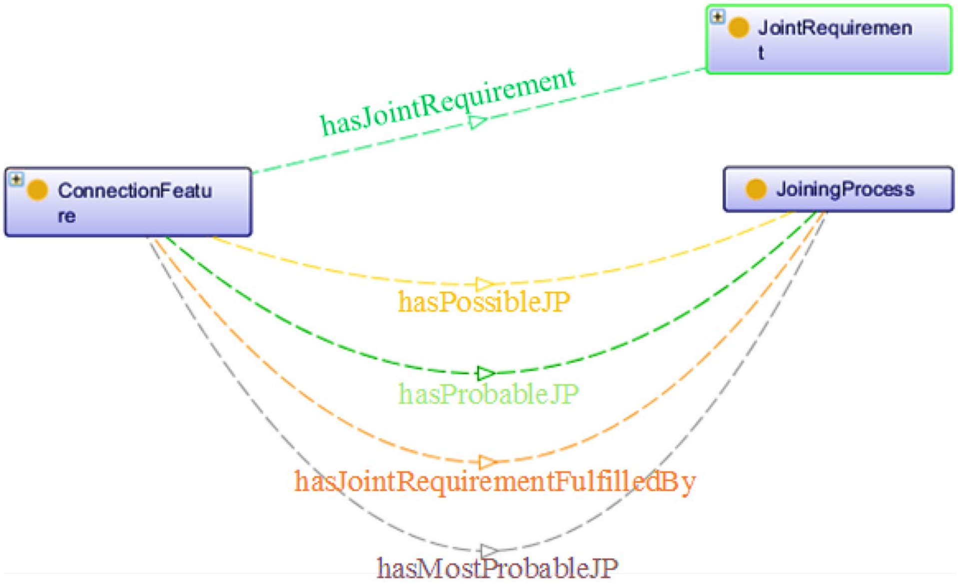


Table 2 The OWL DL representation of JPS ontology concept

Class name	Rule type	DL rule
Product	Cardinality	Product≡ Assembly ⊔ Subassembly⊔ Part Product≡ Assembly
Assembly	Min Cardinality	Assembly≡(IsComposedOf ≥ 1 Subassembly)⊔ (IsComposedOf ≥ 2Part)
Subassembly	Min Cardinality	Subassembly≡ (IsComposedOf ≥ 2Part)
Part	Cardinality	Part ⊑ Product
FeatureForAssembly	Cardinality	FeatureForAssembly ≡ ConnectionFeature ⊔ HandlingFeature ⊔ MatingFeature
ConnectionFeature	ComplementOf	ConnectionFeature ≡ FeatureForAssembly ¬ ⊓ HandlingFeature ¬ ⊓ MatingFeature
Process	Cardinality	Process ≡ JoiningProcess ⊔ ManufacturingProcess
JoiningProcess	ComplementOf	JoiningProcess ≡ Process ¬ ⊓ ManufacturingProcess
Material	Cardinality	Material ⊑ (∃ isMaterialTypeOf. Part)
Lap	Cardinality	Lap⊑(ConnectionFeature)⊓ (hasRelativeOrientation = Parallel and non-coplanar)
Butt	Cardinality	Butt ⊑(ConnectionFeature)⊓ (hasRelativeOrientation = Coplanar)⊓ (hasRelativeOverlap = BothComplete)
CornerJoint	Cardinality	CornerJoint ⊑ (ConnectionFeature)⊓ (hasRelativeOrientation = Perpendicular)⊓ (hasRelativeLocation = Corner)
T_Joint	Cardinality	T_Joint ⊑(ConnectionFeature)⊓ (hasRelativeOrientation = Perpendicular)⊓ (hasRelativeLocation = Middle)
Dovetail	Cardinality	Dovetail ⊑ (ConnectionFeature)Π(hasSlot = True)⊓ (hasJointFeatureType = Dovetail)
TongueAndGroove	Cardinality	TongueAndGroove ⊑ (ConnectionFeature) Π (hasSlot = True) Π (hasJointFeatureType = TongueAndGroove)
Scarf	Cardinality	Scarf ⊑ (ConnectionFeature) Π (has Scarf = True)
Step	Cardinality	Scarf ⊑ (ConnectionFeature) Π (has Step = True)

properties "AgainstPartMaterialCompatibility," "FacingPartMaterialCompatibility," and "MaterialCompatibility." So, these relational mapping between the *ConnectionFeature* class and *JoiningProcess* class is required to form an important knowledge base for joining process selection. These JPS properties are represented using SWRL rules, and some are shown in Table 5.

4.3.4 Knowledge rules for mapping the concepts between *JoiningProcess* and *JointRequirement* class

Each liaison has specific joint requirements, the selected joining process should fulfill. So, the relational mapping of knowledge between *JointRequirement* and *ConnectionFeature* class, and *ConnectionFeature* and *JoiningProcess* class is creating an important knowledge base for the joining process selection. Each joining process should be compatible with the joint requirements of a specific liaison, which is evaluated by using object property "hasJointRequirementFulfilledBy" and one of the SWRL rules is shown in Table 6.

5 Rule-based reasoning system for joining process selection

In this section, the rule-based reasoning system for automatic joining process selection is defined as shown in the flowchart in Fig. 11. This flowchart includes five main steps, which take a 3D CAD model as input and generate suitable joining processes as output. Steps 2, 3, and 4 are the main

Table 3 SWRL rules for mapping between *JoiningProcess* class and *Product* class

No.	JSP concepts	SWRL rules	Explanation
1	DissimilarMaterial	ConnectionFeature(?CF) ^ hasPartMaterialFacing(?CF, ?MA) ^ hasPartMaterialAgainst(?CF, ?MF) ^ swrlb:notEqual(?MA, ?MF)—> hasDissimilarMaterial(?CF, true)	If the material type of facing part and against part involved with a connection feature is not equal, then the connection feature has dissimilar material part
2	UnequalThickness	ConnectionFeature(?CF) ^ hasPartThicknessFacing(?CF, ?PF) ^ hasPartThicknessAgainst(?CF, ?PA) ^ swrlb:notEqual(?PA, ?PF)—> hasUnequalThickness(?CF, true)	If the thickness of facing part and against part involved with a connection feature is not equal, then the connection feature has unequal thickness part

Table 4 SWRL rules for mapping between *JoiningProcess* class and *Material* class

No.	JSP concepts	SWRL rules	Explanation
1	ApplicableMaterial	JoiningProcess(?JP) ^ hasJP_Type(?JP, "MIG_Welding") ^ Material(?M) ^ hasMaterialType(?M, ?MTE) ^ swrlb:equal(?MTE, "LowCarbon-Steel")—> hasApplicableMaterial(?JP, ?M)	If the type of joining process is equal to MIG welding, then its applicable material is low carbon steel
3	MinimumThicknessApplicable	JoiningProcess(?JP) ^ hasJP_Type(?JP, "MIG_Welding") ^ hasApplicableMaterial(?JP, ?AM) ^ hasMaterialType(?AM, ?MT) ^ swrlb:equal(?MT, "Aluminum")—> hasMinimumThicknessApplicable(?JP, 0.5)	If the type of joining process is equal to MIG welding and its applicable material is aluminium, then the joining process has minimum thickness applicable is 0.5 mm
4	MaximumThicknessApplicable	JoiningProcess(?JP) ^ hasJP_Type(?JP, "TIG_Welding") ^ hasApplicableMaterial(?JP, ?AM) ^ hasMaterialType(?AM, ?MT) ^ swrlb:equal(?MT, "Aluminum")—> hasMaximumThicknessApplicable(?JP, 15)	If the type of joining process is equal to MIG welding, and its applicable material is aluminium, then the joining process has maximum thickness applicable is 15 mm

steps that perform automatic reasoning for generating proper joining processes for a specific liaison. This is supported by the knowledge base containing JPS ontology instances and SWRL rules. The detailed information about the five main steps is defined below as follows.

Step 1: Connection feature and its joint requirement instance population

The liaison information is automatically extracted from the CAD model and stored as instances of *ConnectionFeature* class in the JPS ontology using the Cellfie tool. The joint requirements of liaison associated with a product are stored as instances of *JointRequirement* class in the proposed ontology as defined in the Sect. 3.

Step 2: Inferred the possible joining processes

In this step, the possible joining processes for a specific liaison are inferred based on two attributes, i.e., unequal thickness and dissimilar material. These are the two attributes that preliminarily eliminate the joining processes that do not apply to a specific type of combination of material and thickness. These two attributes are evaluated by using SWRL rules, as shown in Table 7. There is a set of rules covering different combinations of material, thickness, and applicable joining processes. The execution of the reasoning process is facilitated by the DROOLS rule engine, which is integrated into the protégé tool as the "SWRLTab" [27]. The inferred possible joining processes are represented as JPps = [JPps1, JPps2…, JPpsi…, JPpsn], which is the output of step 2. This is the first stage of elimination to select the possible joining processes. These possible joining processes are further processed to select the probable joining processes applicable to a specific liaison.

Step 3: Inferred the probable joining processes

This is the second stage of elimination to select the probable joining processes. The probable joining processes are evaluated by checking the compatibility of material, thickness, and joint design type involved with an existing liaison in the CAD model with the applicable material, thickness, and joint design type suitable to a specific joining process, respectively. In this step, the probable joining processes for a particular liaison are inferred based on three attributes, i.e., material compatibility, thickness compatibility, and joint compatibility. These three attributes are used to eliminate the selected possible joining processes for selecting probable joining processes. These attributes are reasoned by using SWRL rules, and some of them are shown in Table 6. The output of step 3 is the different probable joining processes, which is represented as JPpb = [JPpb1, JPpb2…, JPpbi…, JPpbn]. These probable joining processes are further processed to select the most probable joining processes that satisfy a specific liaison's joint requirements.

Step 4: Inferred the most probable joining processes

In order to select the most probable joining processes, a further elimination process is used to eliminate the selected

Table 5 SWRL rules for mapping between *JoiningProcess* class and *Feature* class

No.	JSP concepts	SWRL rules	Explanation
1	ApplicableJoint	JoiningProcess(?JP) ^ hasJP_Type(?JP, “MIG_Welding”) ^ ConnectionFeature(?CF) ^ hasLiaisonType(?CF, “Plain-Lap”)—> hasApplicableJoint(?JP, ?CF)	If the type of joining process is equal to MIG welding, then its applicable joint is plain lap
2	JointCompatibility	ConnectionFeature(?CF) ^ hasLiaisonType(?CF, ?LT1) ^ hasPossibleJP(?CF, ?JP) ^ hasApplicableJoint(?JP, ?JPNT) ^ hasLiaisonType(?JPNT, ?LT2) ^ swrlb:equal(?LT1, ?LT2)—> hasJointCompatibilityWith(?CF, ?JP)	If the joint design type of a connection feature is equal with the applicable joint of a possible joining process, then the connection feature has joint compatibility with the joining process
3	ThicknessCompatibility	ConnectionFeature(?CF) ^ hasPartThicknessFacing(?CF, ?PF) ^ hasPartThicknessAgainst(?CF, ?PA) ^ hasPossibleJP(?CF, ?JP) ^ hasApplicableMaterial(?JP, ?AM) ^ hasMinimumThicknessApplicable(?JP, ?MIT) ^ hasMaximumThicknessApplicable(?JP, ?MXT) ^ swrlb:greaterThanOrEqual(?PA, ?MIT) ^ swrlb:lessThanOrEqual(?PA, ?MXT) ^ swrlb:greaterThanOrEqual(?PF, ?MIT) ^ swrlb:lessThanOrEqual(?PF, ?MXT)—> hasThicknessCompatibilityWith(?CF, ?JP)	Both the facing part and against part involved with a connection feature has thickness is within range of minimum and maximum thickness applicable for a possible joining process and for a specific material, then the connection feature has thickness compatibility with the joining process
4	AgainstPartMaterialCompatibility	ConnectionFeature(?CF) ^ hasPartMaterialAgainst(?CF, ?MA) ^ hasPossibleJP(?CF, ?JP) ^ hasApplicableMaterial(?JP, ?JPAM) ^ hasMaterialType(?JPAM, ?MTEA) ^ swrlb:equal(?MA, ?MTEA)—> hasAgainstPartMaterialCompatibilityWith(?CF, ?JP)	If the material type of against part involved with a connection feature is equal with the applicable material of a possible joining process, then the connection feature has material compatibility with the joining process
5	FacingPartMaterialCompatibility	ConnectionFeature(?CF) ^ hasPartMaterialFacing(?CF, ?MA) ^ hasPossibleJP(?CF, ?JP) ^ hasApplicableMaterial(?JP, ?JPAM) ^ hasMaterialType(?JPAM, ?MTEA) ^ swrlb:equal(?MA, ?MTEA)—> hasFacingPartMaterialCompatibilityWith(?CF, ?JP)	If the material type of facing part involved with a connection feature is equal with the applicable material of a possible joining process, then the connection feature has material compatibility with the joining process
6	MaterialCompatibility	ConnectionFeature(?CF) ^ hasFacingPartMaterialCompatibilityWith(?CF, ?JP) ^ hasAgainstPartMaterialCompatibilityWith(?CF, ?JP)—> hasMaterialCompatibilityWith(?CF, ?JP)	Both the facing part and against part involved with a connection feature has material compatibility with the possible joining process, then the connection feature has complete material compatibility with the joining process

Table 6 SWRL rules for mapping between *JoiningProcess* class and *JointRequirement* class

No.	JSP concepts	SWRL rules	Explanation
1	hasJointRequirementFulfilledBy	ConnectionFeature(?CF) ^ hasProbableJP(?CF, ?JP) ^ hasJP_Type(?JP, ?JT) ^ swrlb:equal(?JT, "MIG_Welding") ^ hasJointRequirement(?CF, ?FRQ) ^ hasStrength(?FRQ, "High") ^ hasCost(?FRQ, "Medium") ^ hasProductionRate(?FRQ, "Slow") ^ hasAestheticQuality(?FRQ, "Medium") ^ hasAutomatedAssembly(?FRQ, true) ^ hasDesignForDisassembly(?FRQ, false) ^ hasAccessRequiredForJoining(?FRQ, "Twosides")—> hasJointRequirementFulfilledBy(?CF, ?JP)	If the connection feature has probable joining process is equal to MIG welding and its joint requirements are high strength, medium cost, slow production rate, medium aesthetic quality, automated assembly, two sides access required and not designed for disassembly, then the joint requirements of connection feature has fulfilled by the probable joining process

probable joining processes based on using one object property, i.e., "hasJointRequirementFulfilledBy." In this process, it is finally checked that the probable joining processes should fulfill all the joint requirements of a liaison. If it is satisfied all the joint requirements, then that joining process is the most suitable joining process to a specific liaison. The reasoning process is supported by rules covering the different combinations of joint requirements and joining processes. The output of step 4 is the most probable joining process, which is represented as JPmpb = [JPmpb1, JP mpb2…, JPmpbi…, JPmpbn].

Step 5: Semantic retrieval of joining processes using SQWRL

The main aim of this research is the automatic selection of joining processes for a specific liaison of a product, which fulfills all the joint requirements. Different stages of process selection like possible joining processes, probable joining processes, and the most probable joining processes are retrieved based on the various attributes, joint requirements, and the liaison information that exists in a product. So, for a particular set of joint requirements and liaisons, these different most probable joining processes are retrieved and stored in the ontology. Thus, all the decisions are stored in the instance repository, and there might be duplication in the result. So, to deduplication of result, an SQWRL language (the "selectDistinct" statement specifically) is used in ontological context to retrieve distinct joining processes based on various combinations of joint requirements and liaisons. The SQWRL retrieval statements are shown in Table 8.

6 Case studies

In this section, two case studies are used to demonstrate the effectiveness of the proposed ontology-based joining process selection system. The primary purpose of the study is to show an application of the extracted liaison information from the CAD model for selecting suitable joining processes for a particular liaison. For this, an existing CAD model is used to extract the liaison information for further processing. The liaison information is stored in the excel file and incorporated into the developed JPS ontology as instances of *ConnectionFeature* class to be used as input for testing the ontology-based decision system. In this research, liaison information from two CAD models, namely the wing of an aircraft and automotive instrumental panel, as shown in Figs. 12 and 13, respectively, are used to validate the proposed approach.

6.1 Instance population

To infer the possible, probable, and most probable joining processes for a particular joint, the liaison information

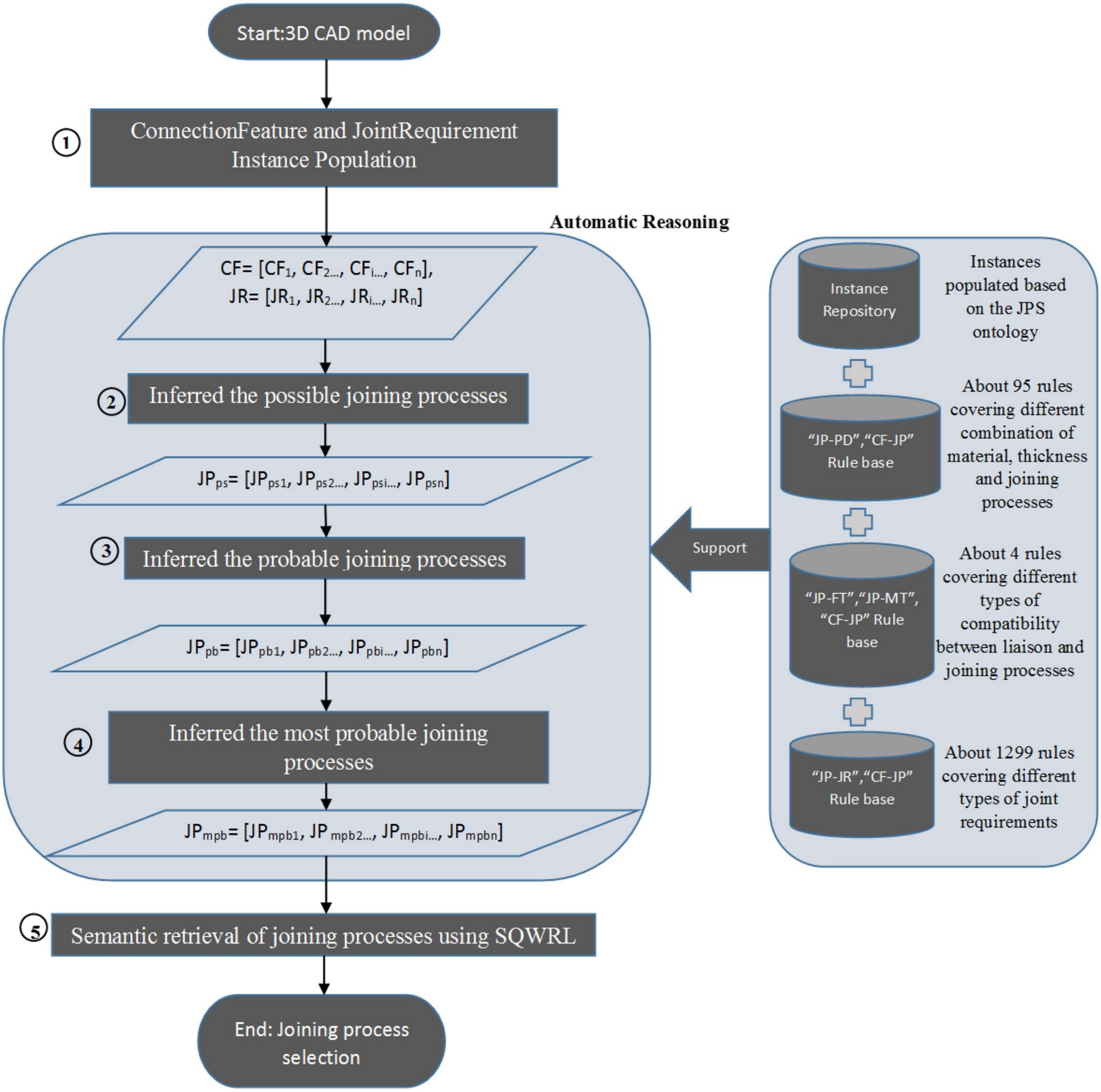


Fig. 11 Flowchart of rule-based reasoning system for automatic joining process selection

should be first populated in the knowledge base. The liaison information is populated as instances of *ConnectionFeature* class using the Protégé tool. For example, the wing of an aircraft carries six instances of liaison information as "Joint1", "Joint2", "Joint3", "Joint4", "Joint5", and "Joint6" as shown in Fig. 12, and automotive instrumental panel carries four instances of liaison information as "Joint7", "Joint8", "Joint9", and "Joint10" as shown in Fig. 13. The data properties of instances of *ConnectionFeature* are specified automatically using the Cellfie tool in Protégé according to their values extracted from the CAD model. For example, the data properties of "Joint6" and "Joint10" are shown in Figs. 12 and 13, respectively. Each liaison has specific joint requirements represented by an object property "hasJointRequirement" as shown in Fig. 12 and defined in Table 1, which has an inverse property "IsJointRequirementOf". In this case study, the instance "Joint6" has certain joint requirements, which is represented as "hasJointRequirement" whose value is an instance "JR34." The instance "JR34" is described the joint requirements by specific data properties, as shown in Fig. 12. Based on the liaison's data properties and joint requirements, the possible, probable, and most

Table 7 SWRL rules for joining process selection

No.	JSP concepts	SWRL rules	Explanation
I	PossibleJP	ConnectionFeature(?CF) ^ hasUnequalThickness(?CF, false) ^ hasDissimilarMaterial(?CF, false) ^ JoiningProcess(?JP) ^ hasJP_Type(?JP, ?JT) ^ swrlb:equal(?JT, "ElectronBeamWelding")—> hasPossibleJP(?CF, ?JP)	If the connection feature has no dissimilar and unequal thickness part and the joining process is electron beam welding type, then the joining process is the possible joining process of the connection feature
2	ProbableJP	ConnectionFeature(?CF) ^ hasPossibleJP(?CF, ?JP) ^ hasThicknessCompatibilityWith(?CF, ?JP) ^ hasMaterialCompatibilityWith(?CF, ?JP) ^ hasJointCompatibilityWith(?CF, ?JP)—> hasProbableJP(?CF, ?JP)	If the connection feature has thickness compatibility, material compatibility and joint compatibility with its possible joining process, then the possible joining process is the probable joining process of the connection feature
3	MostProbableJP	ConnectionFeature(?CF) ^ hasProbableJP(?CF, ?JP) ^ hasJointRequirementFulfilledBy(?CF, ?JP)—> hasMostProbableJP(?CF, ?JP)	If the joint requirement of connection feature has fulfilled by its probable joining process, then the probable joining process is the most probable joining process of the connection feature

probable joining processes are inferred. In this manner, all the instances of *ConnectionFeature* class are populated.

6.2 Inferred the possible, probable, and most probable joining processes

A knowledge base is formed based on the instance population, which is further used for reasoning the possible, probable, and most probable joining processes using the rule-based reasoning engine. The "SWRLTab" is used to implement this rule-based reasoning system in Protégé with the help of the DROOLS inference engine. The rule bases "JP-PD" and "CF-JP" are used for inferring the possible joining processes based on the liaison information stored in the instance of *ConnectionFeature* class. In this case study, the probable joining processes for "Joint6" and "Joint10" are inferred, as shown in Figs. 12 and 13, respectively. The rule bases "JP-FT," "JP-MT," and "CF-JP" are used for inferring the probable joining processes for a particular liaison. The rule bases "JP-JR" and "CF-JP" are used to infer the most probable joining processes for a particular liaison. The most probable joining processes for a specific liaison, i.e., "Joint6" and "Joint10" is inferred, as shown in Figs. 12 and 13, respectively. So, in this paper, a three-stage screening process is used to find the most suitable joining processes for a particular liaison.

6.3 Querying the required joining processes using SQWRL

The required possible, probable, and most probable joining processes are inferred for each liaison based on the liaison information and their joint requirements. There may be duplication in the result due to the overlapping of joining processes in the instance repository. An "SQWRLTab" is used to query the possible, probable, and most probable joining processes to avoid duplication of results, as shown in Fig. 14. Based on the different joint requirements, the most suitable joining processes for a particular liaison are obtained, as shown in Fig. 14.

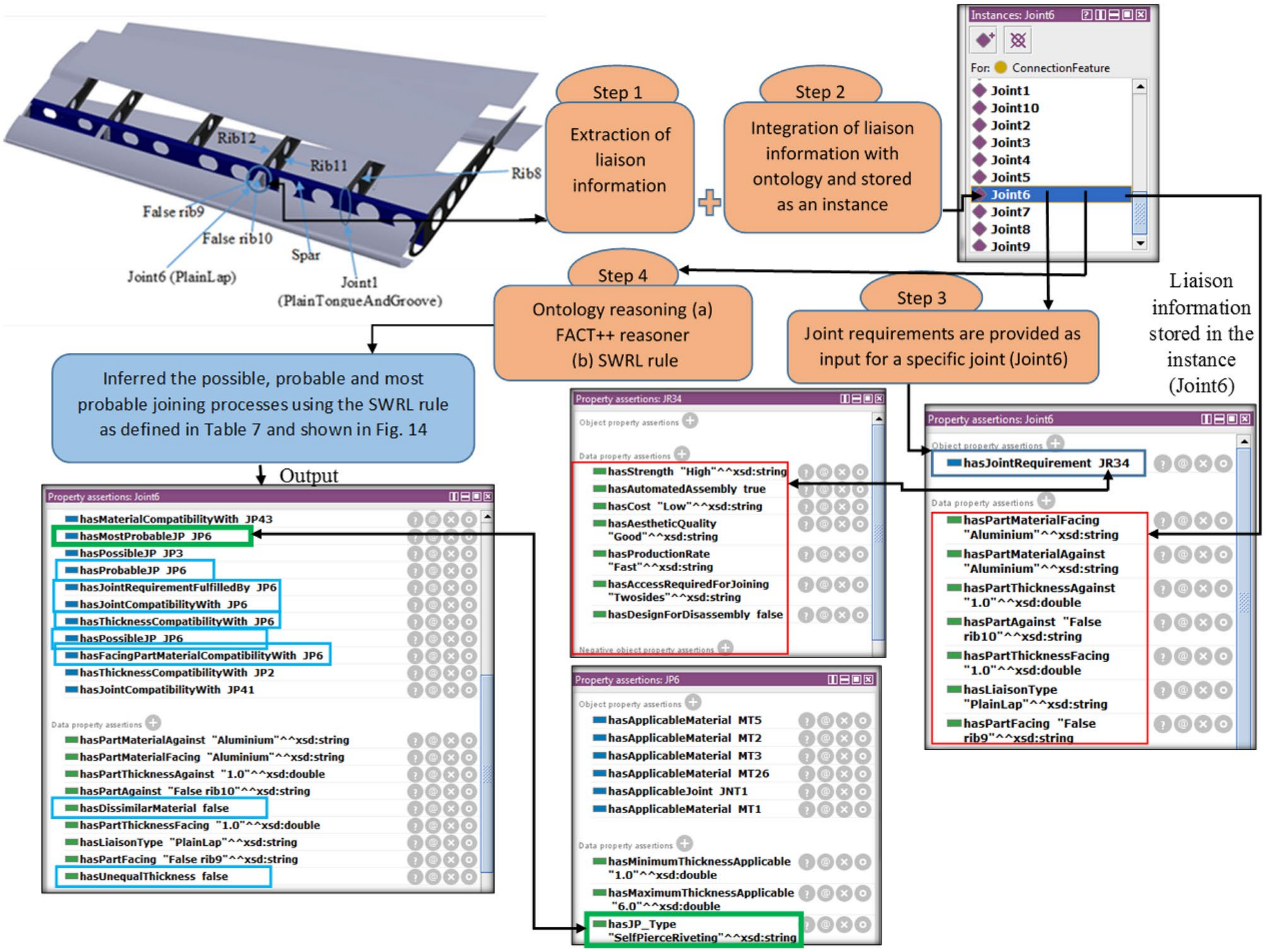
7 Ontology evaluation

A data-driven ontology evaluation method, i.e., the precision and recall technique, is used to test the conciseness and completeness of the JPS ontology. This method does a comparative analysis of the developed ontology against a predefined set of knowledge items by counting the related terms that appear between a collection of knowledge items

Table 8 Query the joining processes using SQWRL

No.	JPS concepts	SQWRL statement
1	QueryJointRequirement	ConnectionFeature(?CF) ^ hasJointRequirement(?CF, ?JRQ) ^ hasMostProbableJP(?CF, ?JP) ^ hasJP_Type(?JP, ?JT)—> sqwrl:selectDistinct(?CF, ?JRQ, ?JT) ^ sqwrl:orderBy(?CF) ^ sqwrl:columnNames("Liaison," "Joint Requirement," "Most Probable Joining Process")
2	QueryPossibleJoiningProcess	ConnectionFeature(?CF) ^ hasLiaisonType(?CF, ?L) ^ hasPartFacing(?CF, ?PF) ^ hasPartAgainst(?CF, ?PA) ^ hasPossibleJP(?CF, ?JP) ^ hasJP_Type(?JP, ?JT)—> sqwrl:selectDistinct(?CF, ?L, ?PF, ?PA, ?JT) ^ sqwrl:orderBy(?CF) ^ sqwrl:columnNames("Liaison," "LiaisonType," "Part Facing," "Part Against," "Possible Joining Process")
3	QueryProbableJoiningProcess	ConnectionFeature(?CF) ^ hasLiaisonType(?CF, ?L) ^ hasPartFacing(?CF, ?PF) ^ hasPartAgainst(?CF, ?PA) ^ hasProbableJP(?CF, ?JP) ^ hasJP_Type(?JP, ?JT)—> sqwrl:selectDistinct(?CF, ?L, ?PF, ?PA, ?JT) ^ sqwrl:orderBy(?CF) ^ sqwrl:columnNames("Liaison," "LiaisonType," "Part Facing," "Part Against," "Probable Joining Process")
4	QueryMostProbableJoiningProcess	ConnectionFeature(?CF) ^ hasLiaisonType(?CF, ?L) ^ hasPartFacing(?CF, ?PF) ^ hasPartAgainst(?CF, ?PA) ^ hasMostProbableJP(?CF, ?JP) ^ hasJP_Type(?JP, ?JT)—> sqwrl:selectDistinct(?CF, ?L, ?PF, ?PA, ?JT) ^ sqwrl:orderBy(?CF) ^ sqwrl:columnNames("Liaison," "LiaisonType," "Part Facing," "Part Against," "Most Probable Joining Process")

and the ontology. These predefined data sets are extracted from the expert reviews defined in different sources [1, 3, 4, 8, 9, 14, 16, 23–25]. Several questions are obtained from the expert reviews for the ontology evaluation covering necessary concepts related to assembly joining process selection. A list of relevant and retrieved entities are



Instances: Joint6 — For: ConnectionFeature
Joint1
Joint10
Joint2
Joint3
Joint4
Joint5
Joint6
Joint7
Joint8
Joint9

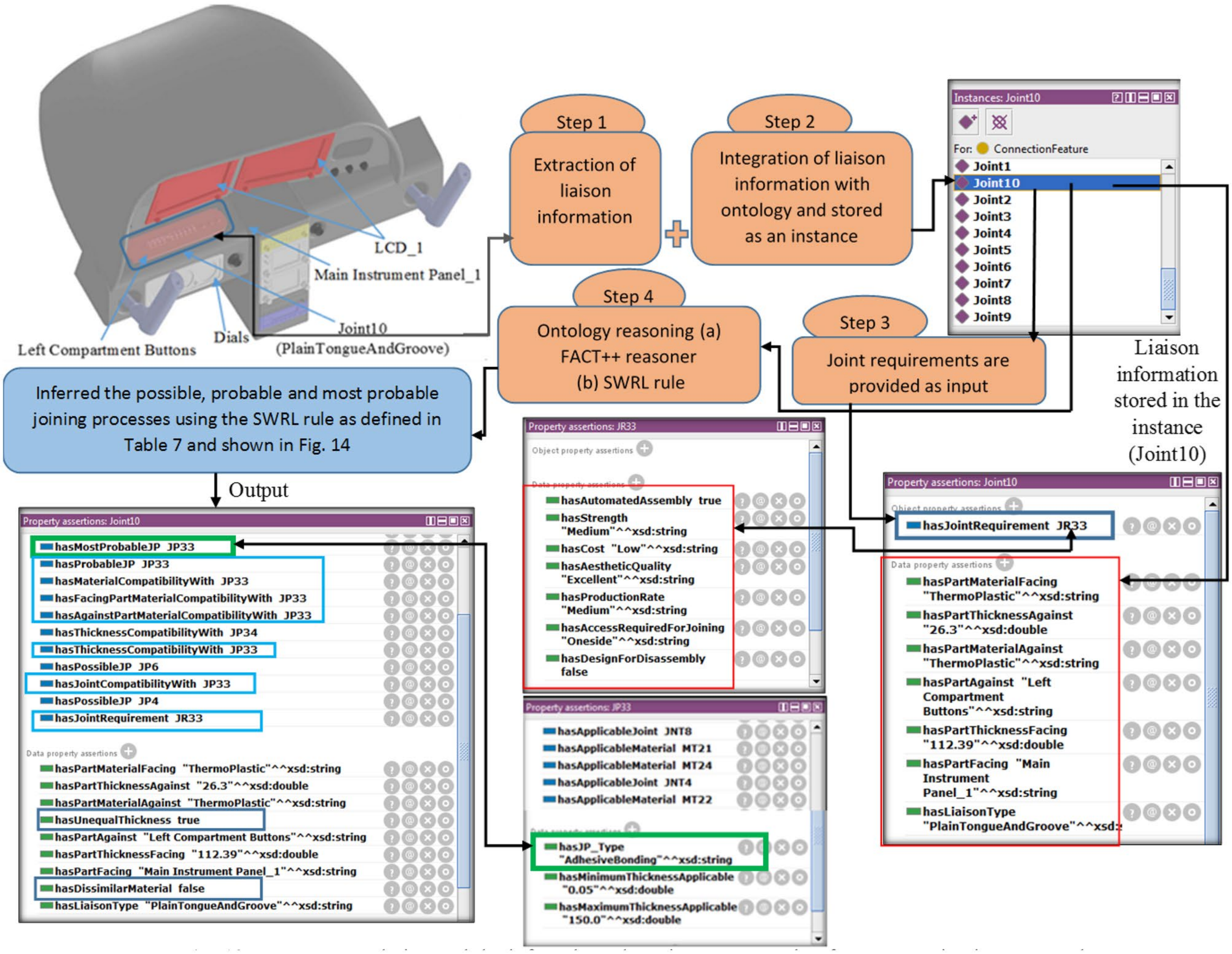
Property assertions: Joint6
hasJointRequirement JR34
hasPartMaterialFacing "Aluminium"^^xsd:string
hasPartMaterialAgainst "Aluminium"^^xsd:string
hasPartThicknessAgainst "1.0"^^xsd:double
hasPartAgainst "False rib10"^^xsd:string
hasPartThicknessFacing "1.0"^^xsd:double
hasLiaisonType "PlainLap"^^xsd:string
hasPartFacing "False rib9"^^xsd:string

Property assertions: JR34
hasStrength "High"^^xsd:string
hasAutomatedAssembly true
hasCost "Low"^^xsd:string
hasAestheticQuality "Good"^^xsd:string
hasProductionRate "Fast"^^xsd:string
hasAccessRequiredForJoining "Twosides"^^xsd:string
hasDesignForDisassembly false

Property assertions: JP6
hasApplicableMaterial MT5
hasApplicableMaterial MT2
hasApplicableMaterial MT3
hasApplicableMaterial MT26
hasApplicableJoint JNT1
hasApplicableMaterial MT1
hasMinimumThicknessApplicable "1.0"^^xsd:double
hasMaximumThicknessApplicable "6.0"^^xsd:double
hasJP_Type "SelfPierceRiveting"^^xsd:string

Property assertions: Joint6
hasMaterialCompatibilityWith JP43
hasMostProbableJP JP6
hasPossibleJP JP3
hasProbableJP JP6
hasJointRequirementFulfilledBy JP6
hasJointCompatibilityWith JP6
hasThicknessCompatibilityWith JP6
hasPossibleJP JP6
hasFacingPartMaterialCompatibilityWith JP6
hasThicknessCompatibilityWith JP2
hasJointCompatibilityWith JP41
hasPartMaterialAgainst "Aluminium"^^xsd:string
hasPartMaterialFacing "Aluminium"^^xsd:string
hasPartThicknessAgainst "1.0"^^xsd:double
hasPartAgainst "False rib10"^^xsd:string
hasDissimilarMaterial false
hasPartThicknessFacing "1.0"^^xsd:double
hasLiaisonType "PlainLap"^^xsd:string
hasPartFacing "False rib9"^^xsd:string
hasUnequalThickness false

Fig. 12 Instance population and the inferred results using a case study of a wing of an aircraft



Property assertions: Joint10

hasMostProbableJP JP33
hasProbableJP JP33
hasMaterialCompatibilityWith JP33
hasFacingPartMaterialCompatibilityWith JP33
hasAgainstPartMaterialCompatibilityWith JP33
hasThicknessCompatibilityWith JP34
hasThicknessCompatibilityWith JP33
hasPossibleJP JP6
hasJointCompatibilityWith JP33
hasPossibleJP JP4
hasJointRequirement JR33

hasPartMaterialFacing "ThermoPlastic"^^xsd:string
hasPartThicknessAgainst "26.3"^^xsd:double
hasPartMaterialAgainst "ThermoPlastic"^^xsd:string
hasUnequalThickness true
hasPartAgainst "Left Compartment Buttons"^^xsd:string
hasPartThicknessFacing "112.39"^^xsd:double
hasPartFacing "Main Instrument Panel_1"^^xsd:string
hasDissimilarMaterial false
hasLiaisonType "PlainTongueAndGroove"^^xsd:string

Property assertions: JR33

hasAutomatedAssembly true
hasStrength "Medium"^^xsd:string
hasCost "Low"^^xsd:string
hasAestheticQuality "Excellent"^^xsd:string
hasProductionRate "Medium"^^xsd:string
hasAccessRequiredForJoining "Oneside"^^xsd:string
hasDesignForDisassembly false

Property assertions: JP33

hasApplicableJoint JNT8
hasApplicableMaterial MT21
hasApplicableMaterial MT24
hasApplicableJoint JNT4
hasApplicableMaterial MT22

hasJP_Type "AdhesiveBonding"^^xsd:string
hasMinimumThicknessApplicable "0.05"^^xsd:double
hasMaximumThicknessApplicable "150.0"^^xsd:double

Property assertions: Joint10

hasJointRequirement JR33

hasPartMaterialFacing "ThermoPlastic"^^xsd:string
hasPartThicknessAgainst "26.3"^^xsd:double
hasPartMaterialAgainst "ThermoPlastic"^^xsd:string
hasPartAgainst "Left Compartment Buttons"^^xsd:string
hasPartThicknessFacing "112.39"^^xsd:double
hasPartFacing "Main Instrument Panel_1"^^xsd:string
hasLiaisonType "PlainTongueAndGroove"^^xsd:

Fig. 13 Instance population and the inferred results using a case study of an automotive instrumental

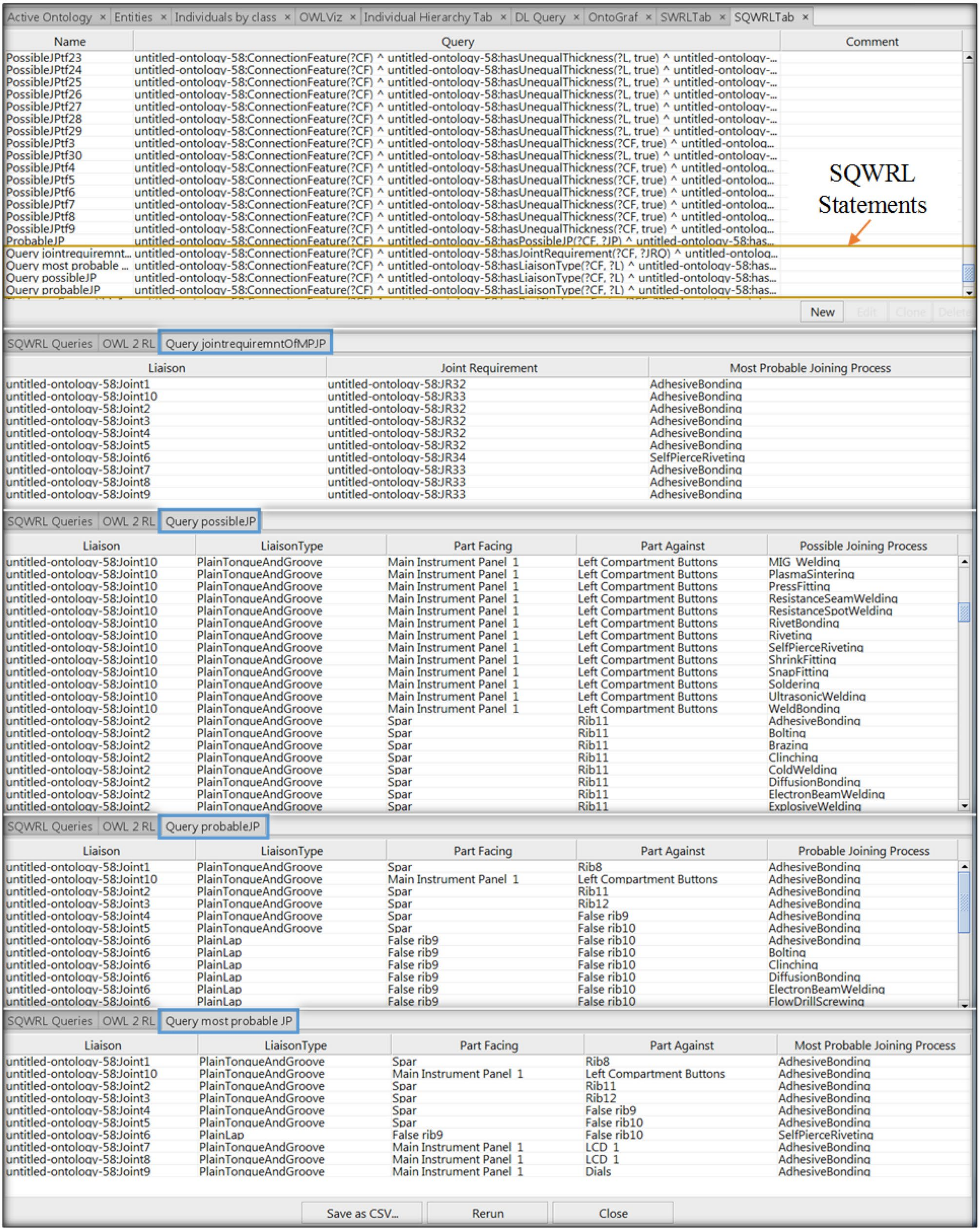
identified from a sample question by manually annotation to extract the main concepts needed to answer them as defined in Fig. 15. From this figure, it can be observed that while there are eleven relevant entities needed to answer the sample question, only nine entities are retrieved. Specifically, while the “Joint Accessibility and Loading Mode” information items are relevant to answer the question, they could not be retrieved from the developed ontology. The same procedure is applied to count the total number relevant information retrived from the total relevant information for the calculation of precision, recall and F-measure using the below formula.

$$\text{Precision} = \frac{\mathit{Number\ of\ relevant\ information\ retrieved}\ [TP]}{\mathit{Number\ of\ information\ retrieved}\ [TP + FP]}$$

$$\text{Recall} = \frac{\mathit{Number\ of\ relevant\ information\ retrieved}\ [TP]}{\mathit{Number\ of\ relevant\ information}\ [TP + FN]}$$

$$\text{F} - \text{measure} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

The high performance of JPS ontology demonstrates that it contains a high percentage of the relevant entities (Recall = 85.7%) for supporting the assembly joining process selection, as shown in Table 9. This reinforces the completeness of the ontology. The precision rate (89.4%) of JPS ontology indicates the percentage of retrieved information is relevant for joining process selection. This supports the conciseness of ontology.



Name	Query	Comment
PossibleJPtf23	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf24	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf25	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf26	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf27	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf28	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf29	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf3	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
PossibleJPtf30	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?L, true) ^ untitled-ontology-...	
PossibleJPtf4	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
PossibleJPtf5	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
PossibleJPtf6	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
PossibleJPtf7	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
PossibleJPtf8	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
PossibleJPtf9	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasUnequalThickness(?CF, true) ^ untitled-ontolog...	
ProbableJP	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasPossibleJP(?CF, ?JP) ^ untitled-ontology-58:has...	
Query jointrequiremnt...	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasJointRequirement(?CF, ?JRQ) ^ untitled-ontolog...	
Query most probable ...	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasLiaisonType(?CF, ?L) ^ untitled-ontology-58:has...	
Query possibleJP	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasLiaisonType(?CF, ?L) ^ untitled-ontology-58:has...	
Query probableJP	untitled-ontology-58:ConnectionFeature(?CF) ^ untitled-ontology-58:hasLiaisonType(?CF, ?L) ^ untitled-ontology-58:has...	

Query jointrequiremntOfMPJP

Liaison	Joint Requirement	Most Probable Joining Process
untitled-ontology-58:Joint1	untitled-ontology-58:JR32	AdhesiveBonding
untitled-ontology-58:Joint10	untitled-ontology-58:JR33	AdhesiveBonding
untitled-ontology-58:Joint2	untitled-ontology-58:JR32	AdhesiveBonding
untitled-ontology-58:Joint3	untitled-ontology-58:JR32	AdhesiveBonding
untitled-ontology-58:Joint4	untitled-ontology-58:JR32	AdhesiveBonding
untitled-ontology-58:Joint5	untitled-ontology-58:JR32	AdhesiveBonding
untitled-ontology-58:Joint6	untitled-ontology-58:JR34	SelfPierceRiveting
untitled-ontology-58:Joint7	untitled-ontology-58:JR33	AdhesiveBonding
untitled-ontology-58:Joint8	untitled-ontology-58:JR33	AdhesiveBonding
untitled-ontology-58:Joint9	untitled-ontology-58:JR33	AdhesiveBonding

Query possibleJP

Liaison	LiaisonType	Part Facing	Part Against	Possible Joining Process
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	MIG Welding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	PlasmaSintering
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	PressFitting
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	ResistanceSeamWelding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	ResistanceSpotWelding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	RivetBonding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	Riveting
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	SelfPierceRiveting
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	ShrinkFitting
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	SnapFitting
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	Soldering
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	UltrasonicWelding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	WeldBonding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	AdhesiveBonding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	Bolting
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	Brazing
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	Clinching
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	ColdWelding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	DiffusionBonding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	ElectronBeamWelding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	ExplosiveWelding

Query probableJP

Liaison	LiaisonType	Part Facing	Part Against	Probable Joining Process
untitled-ontology-58:Joint1	PlainTongueAndGroove	Spar	Rib8	AdhesiveBonding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	AdhesiveBonding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	AdhesiveBonding
untitled-ontology-58:Joint3	PlainTongueAndGroove	Spar	Rib12	AdhesiveBonding
untitled-ontology-58:Joint4	PlainTongueAndGroove	Spar	False rib9	AdhesiveBonding
untitled-ontology-58:Joint5	PlainTongueAndGroove	Spar	False rib10	AdhesiveBonding
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	AdhesiveBonding
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	Bolting
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	Clinching
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	DiffusionBonding
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	ElectronBeamWelding
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	FlowDrillScrewing

Query most probable JP

Liaison	LiaisonType	Part Facing	Part Against	Most Probable Joining Process
untitled-ontology-58:Joint1	PlainTongueAndGroove	Spar	Rib8	AdhesiveBonding
untitled-ontology-58:Joint10	PlainTongueAndGroove	Main Instrument Panel 1	Left Compartment Buttons	AdhesiveBonding
untitled-ontology-58:Joint2	PlainTongueAndGroove	Spar	Rib11	AdhesiveBonding
untitled-ontology-58:Joint3	PlainTongueAndGroove	Spar	Rib12	AdhesiveBonding
untitled-ontology-58:Joint4	PlainTongueAndGroove	Spar	False rib9	AdhesiveBonding
untitled-ontology-58:Joint5	PlainTongueAndGroove	Spar	False rib10	AdhesiveBonding
untitled-ontology-58:Joint6	PlainLap	False rib9	False rib10	SelfPierceRiveting
untitled-ontology-58:Joint7	PlainTongueAndGroove	Main Instrument Panel 1	LCD 1	AdhesiveBonding
untitled-ontology-58:Joint8	PlainTongueAndGroove	Main Instrument Panel 1	LCD 1	AdhesiveBonding
untitled-ontology-58:Joint9	PlainTongueAndGroove	Main Instrument Panel 1	Dials	AdhesiveBonding

Fig. 14 Querying the possible, probable, and most probable joining processes in Protégé

Table 9 Precision and recall rate of an ontology

Ontology	Precision	Recall	F-measure
JPS	89.4%	85.7%	$2\times\frac{Precision\times Recall}{Precision+Recall}=87.5\%$

8 Discussion

The JPS ontology can support designers and process planners for automatic joining process selection from the CAD model. The developed ontology takes the extracted liaison information as the input and can suggest the possible, probable, and the most probable joining processes through rule-based reasoning. To select the most suitable joining processes for a particular liaison, the designers need to know the applicability of a specific joining process to a specific combination of material type and the thickness type that exist at the joint location. Also, the compatibility (i.e., joint, material, and thickness) of a specific liaison with a particular joining process should be scrutinized.

From the result, it is concluded that the joint requirements may be the same for a whole product or distinct for different liaisons in a product. For example, the joint requirements of an automotive instrumental panel are defined by an instance JR33, and it is the same for all liaisons like “Joint7”, “Joint8”, “Joint9”, and “Joint10.” But, in the case of the wing of an aircraft, the joint requirements of liaison “Joint6” (i.e., JR34) is different from all other liaison (“Joint1”, “Joint2”, “Joint3”, “Joint4”, and “Joint5”) as shown in Fig. 14. There may be more than one most probable joining process for a set of joint requirements. In this case, the designer or the process planner selects the most suitable joining process based on the availability of the equipment and its operating condition. The product having kinematic joints where there is no need for any of these joining processes, this proposed framework should be avoided.

Earlier several authors used liaisons for joining process selection [10, 14, 16] was at abstract level. In the present approach, the liaison has been used for assembly process selection considering the allied processes. Moreover, the data used in their frameworks are supplied interactively by the designer. In the present work, the extracted liaison knowledge has been integrated automatically in the JPS ontology. In the past, some of the authors have used ontology-based approach for joining process representation [14, 15], considering only geometric information. But in actual practice, geometric and non-geometric information is required for process selection, that is considered in this research. From the data-driven evaluation of the JPS ontology, the precision, recall, and F-measure obtained are 89.4%, 85.7%, and 87.5%, respectively. This evaluation demonstrates satisfactory performance for conciseness and completeness of the ontology.

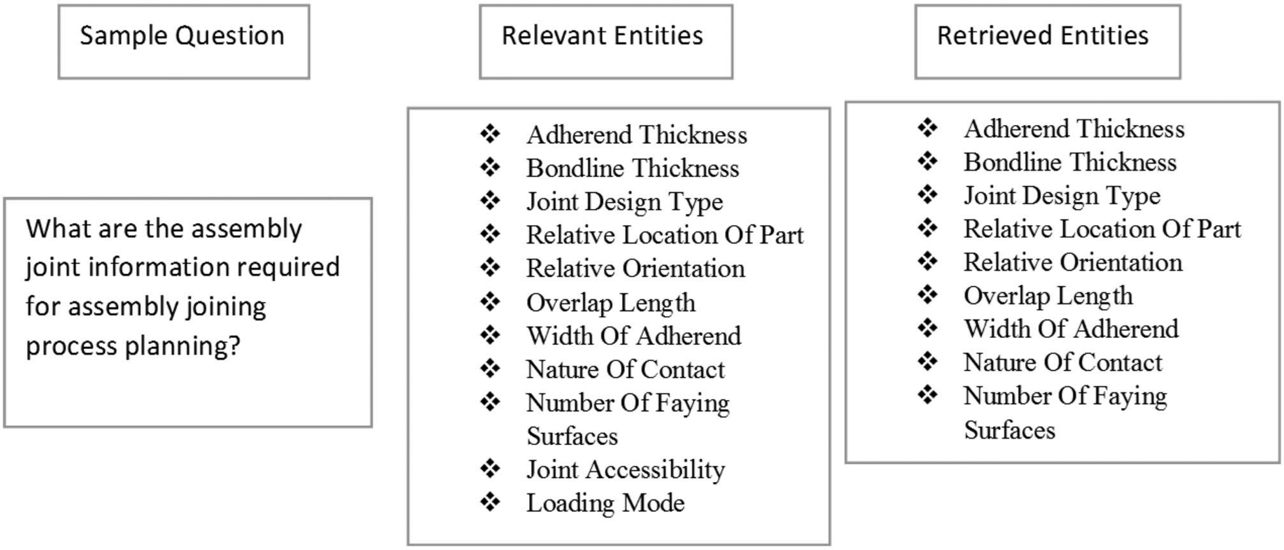


Fig. 15 Manually annotated question for relevant and retrieved entities

9 Conclusion and future works

Appropriate selection of the assembly joining process is a knowledge-intensive process, where it is necessary to develop a knowledge-based framework to capture the necessary knowledge required to select the suitable joining process. In this paper, an ontology-based knowledge framework is developed for the selection of the assembly joining process. The proposed framework is supported by the knowledge-based system, where the extracted liaison information from the CAD model is incorporated into the developed JPS ontology. In this ontology, the basic different core concepts and relationships involved in assembly joining process selection are analyzed and represented in detail. An SWRL rule-based query engine is employed for retrieving the possible, probable, and most probable joining processes for each liaison based on the liaison information and their joint requirements. A SQWRL query engine is employed to query the required joining process selection knowledge for each liaison by the designer or the process planner. Case studies are performed using industrial CAD models for validating the proposed approach. Data-driven evaluation technique is used to measure the conciseness and completeness of the ontology.

The developed JPS ontology can be extended in a number of ways. Firstly, the ontology and its rule set could be extended for identifying the suitable post-processing operations, their sequencing, as well as appropriate resources needed to create a product. The existing ontology is limited to the selection of fixed joints and can be extended to select kinematic joints like gear, ball joint, magnetic track, belt and pulley, bearing, etc. The JPS ontology can be extended for the assembly process planning for the product variant design.

Acknowledgements This research is financially supported by the Ministry of Human Resources and Development (MHRD), Government of India.

Declarations

Ethics approval We declare that accepted principles of ethical and professional conduct have been followed in our research.

Consent to participate Not applicable.

Consent for publication We declare our consent for publication.

Research involving human and animal Not applicable.

Conflict of interest The authors declare no competing interests.

References

1. Nof SY, Wilhelm WE, Warnecke H (2012) Industrial assembly. Springer Science & Business Media
2. Xu LD, Wang C, Bi Z et al (2012) AutoAssem: an automated assembly planning system for complex products. IEEE Trans Ind Informat 8(3):669–678. https://doi.org/10.1109/TII.2012.2188901
3. Rafibakhsh N (2017) Automated assembly planning: from CAD model to virtual assembly process. Doctoral dissertation, Oregon State University
4. Kim JH, Wang LS, Putta K, Haghighi P, Shah JJ, Edwards P (2019) Knowledge based design advisory system for multi-material joining. J Manuf Syst 52:253–263. https://doi.org/10.1016/j.jmsy.2019.03.003
5. L'Eglise T, De Lit P, Fouda P (2001) A multicriteria decision-aid system for joining process selection. In Proceedings of the 2001 IEEE International Symposium on Assembly and Task Planning (ISATP2001). Assembly and Disassembly in the Twenty-First Century (Cat. No. 01TH8560) 324–329. https://doi.org/10.1109/ISATP.2001.929043
6. LeBacq C, Brechet Y, Shercliff HR, Jeggy T, Salvo L (2002) Selection of joining methods in mechanical design. Mater Des 23(4):405–416. https://doi.org/10.1016/S0261-3069(01)00093-0
7. Lae L, Brechet Y, LeBacq C, Jeggy T, Salvo L (2002) Knowledge-based systems for selecting joining processes. Adv Eng Mater 4(6):403–407. https://doi.org/10.1002/1527-2648(20020605)4:6<403::AID-ADEM403>3.0.CO;2-6
8. Darwish SM, Tamimi AA, Habdan SA (1997) A knowledge base for metal welding process selection. Int J Mach Tools Manuf 37(7):1007–1023. https://doi.org/10.1016/S0890-6955(96),00073-9
9. Swain AK, Sen D, Gurumoorthy B (2014) Extended liaison as an interface between product and process model in assembly. Robot Comput Integr Manuf 30(5):527–545. https://doi.org/10.1016/j.rcim.2014.02.005
10. Lohse N, Hirani H, Ratchev S, Turitto M (2005) An ontology for the definition and validation of assembly processes for evolvable assembly systems. In (ISATP 2005) The 6th IEEE International Symposium on Assembly and Task Planning: From Nano to Macro Assembly and Manufacturing. pp 242–247. https://doi.org/10.1109/ISATP.2005.1511480
11. Qiao L, Qie Y, Zhu Z et al (2018) An ontology-based modelling and reasoning framework for assembly sequence planning. Int J Adv Manuf Technol 94:4187–4197. https://doi.org/10.1007/s00170-017-1077-4
12. Das SK, Swain AK (2020) An ontology-based framework for decision support in assembly variant design. ASME J Comput Inf Sci Eng 21(2):021007. https://doi.org/10.1115/1.4048127
13. Saha S, Usman Z, Li WD, Jones S, Shah N (2019) Core domain ontology for joining processes to consolidate welding standards. Robot Comput Integr Manuf 59:417–430. https://doi.org/10.1016/j.rcim.2019.05.010
14. Kim KY, Yang H, Kim DW (2008) Mereotopological assembly joint information representation for collaborative product design. Robot Comput Integr Manuf 24(6):744–754. https://doi.org/10.1016/j.rcim.2008.03.010
15. Zhang Y, Luo X, Zhang H, Sutherland JW (2014) A knowledge representation for unit manufacturing processes. Int J Adv Manuf Technol 73(5–8):1011–1031. https://doi.org/10.1007/s00170-014-5864-x
16. Swain AK (2012) Integrating product model with assembly process model using liaisons. PhD Thesis, IISc Bangalore, India
17. Esawi AMK, Ashby MF (2004) Computer-based selection of joining processes: methods, software and case studies. Mater Des 25(7):555–564. https://doi.org/10.1016/j.matdes.2004.03.002
18. Mesa JA, Illera D, Esparragoza I, Maury H, Gómez H (2018) Functional characterisation of mechanical joints to facilitate its selection during the design of open architecture products. Int J Prod Res 56(24):7390–7404. https://doi.org/10.1080/00207543.2017.1412530

19. Bond D, Suzuki FA, Scalice RK (2020) Sheet metal joining process selector. J Braz Soc Mech Sci Eng 42(5):1–15. https://doi.org/10.1007/s40430-020-02310-9
20. Imran M, Young B (2015) The application of common logic based formal ontologies to assembly knowledge sharing. J Intell Manuf 26(1):139–158. https://doi.org/10.1007/s10845-013-0768-4
21. Gruhier E, Demoly F, Kim KY, Abboudi S, Gomes S (2016) A theoretical framework for product relationships description over space and time in integrated design. J Eng Des 27(4–6):269–305. https://doi.org/10.1080/09544828.2016.1144049
22. Solano L (2021) Ontological modelling of welding processes. In IOP Conference Series. Mat Sci Eng 1193(1):012019. IOP Publishing. https://doi.org/10.1088/1757-899X/1193/1/012019
23. Houldcroft PT (1990) Which process?: a guide to the selection of welding and related processes. Elsevier
24. Swift KG, Booker JD (2003) Process selection: from design to manufacture. Elsevier
25. Messler RW (2004) Joining of materials and structures: from pragmatic process to enabling technology. Butterworth-Heinemann
26. Das SK, Swain AK (2019) Classification, representation and automatic extraction of adhesively bonded assembly features. Assembly Autom 39(4):607–623. https://doi.org/10.1108/aa-07-2018-095
27. O’Connor M (2018) SWRLTab: a development environment for working with SWRL rules in Protégé-OWL
28. Stanford University (2018) PROTÉGÉ 5.2. Available at: https://protege.stanford.edu/